Trust Management and Bad Data Reduction in Internet of Vehicles Using Blockchain and AI

Conference Paper · June 2023

DOI: 10.1109/VTC2023-Spring57618.2023.10200207

Rashmi Erandika Ratnayake
University College Dublin

Madhusanka Liyanage
University College Dublin

Trust Management and Bad Data Reduction in Internet of Vehicles Using Blockchain and AI

Rashmi Ratnayake, Madhusanka Liyanage, Liam Murphy
School of Computer Science, University College Dublin, Ireland
Emails: rashmi.ratnayake@ucdconnect.ie, madhusanka@ucd.ie, liam.murphy@ucd.ie

***Abstract*—Blockchain offers cryptographically secure storage for recording transactions. However, one issue with blockchains is the problem of bad data and data reliability, where bad data refers to inaccurate, incomplete, or irrelevant data. This paper investigates how machine learning (ML) can be used to identify inaccurate sensor data added to a blockchain in Internet of Vehicles (IoV) applications. A solution for reducing the inclusion of incorrect data using a reputation-based method is proposed. We suggest that if an accurate ML model can be built for a task that can be completed using the input sensor data, it is possible to use the same model to assess the accuracy of new input data samples for which the actual task outcome is known. A road surface-type classification task is performed using Convolutional Neural Network models on the Passive Vehicular Sensors Datasets, and a pre-trained model is used in a novel solution approach involving edge servers and validators on a blockchain network. Our research shows that ML can be used to identify bad data on the blockchain and to reduce the addition of unreliable data to the blockchain in an IoV context. The proposed solution is generalizable and can be applied to any scenario where an accurate ML model can be devised for a task that can be accomplished using some blockchain input data.**

***Index Terms*—Internet of Vehicles, Trust Management, Bad Data Reduction, Blockchain, Machine Learning, Smart Contracts**

I. Introduction

Vehicle sensor data is crucial for gaining a better knowledge of the world around a vehicle in autonomous driving. Vehicles and infrastructures share sensor data in the Internet of Vehicles (IoV) to maximize the field of view and make more accurate decisions based on sensor data. However, there are many challenges with vehicle data sharing. A key problem is the existence of dishonest and malicious peers in the system [1]. Also, the messages shared between vehicles are not always reliable due to the complicated network structure and rapid mobility [2]. In several recent studies [1]–[4], using blockchain has been proposed as a potential solution for these issues. An issue with blockchain, however, is that while it can ensure the originality of data added to it, it cannot ensure the accuracy of the data itself. If the sensor data provided by the vehicles are inaccurate, the decisions made based on the shared data may be erroneous.

Many efforts are currently being made to address the bad data problem in blockchains. Some of them are data validation techniques and consensus mechanisms for validating transactions, smart contracts for verifying the authenticity of a data source, data auditing features that allow third-party auditors to monitor and verify data stored in the blockchain, and reputation systems to encourage users to add high-quality data. Using oracles, which are trusted third-party services, and Application Programming Interfaces (APIs) to feed data to the blockchain from well-known, reliable sources is the main focus of the majority of studies on blockchain bad data. However, only a few studies have focused on data that cannot be fed in through oracles or verified through above methods, such as Internet of Things (IoT) sensor data. Such data measurements can have flaws due to various factors like sensor defects, sensor installation errors, human errors in measurements, and intentional manipulations. More research on methods to validate such data on blockchains is required.

To solve the problem of adding inaccurate sensor data to the blockchain, we suggest a machine learning (ML) based method in this paper. We test the feasibility of our suggested approach through a prototype implementation, and present the findings from its analysis.

The rest of this paper is organized as follows: Section II describes related work. Our proposed solution approach is presented in Section III. In Section IV, the experimental results are discussed, and Section V discusses our approach compared to other solutions. Section VI concludes the paper and provides future directions.

II. Related Work

Several previous studies addressing the blockchain bad data problem can be found in the literature. The work in [5] focuses on what can be done when incorrect or malicious data is added to the blockchain. Their research describes three solutions for removing bad data identified on the blockchain: ‘rollback’, ‘do nothing’, and ‘overturn’. Their study, however, does not discuss how bad data can be identified in the first place. The issue of bad data on blockchains used in supply chain contexts has been highlighted in a number of studies [6]–[8]. The study [6] discusses that the application of blockchain in food supply chains does not solve the issues with IoT data quality, and the data that is very securely recorded on the blockchain may simply be ‘immutable garbage’. They propose that the concept of ‘common knowledge’ among agents can be used to validate historical data and frame desired future possibilities. [7] points out that, despite using blockchains to store the data, accurate tracking and monitoring of the fish supply chain cannot be achieved without using peripheral sensors due to the bad data problem.

Using oracles to feed in data to the blockchain from verified sources through smart contracts is proposed in multiple studies [9]–[12]. These employ voting-based and reputation-based methods for enhancing data reliability. However, such oracles or APIs can provide only data that can be obtained through another source which is verifiable, and are not applicable to more dynamic data like IoT sensor data, of which the measurements depend largely on the specific sensor environment at a particular time. Blockchain-based solutions involving the introduction of novel consensus mechanisms have been proposed to resolve critical message dissemination issues in VANETs (Vehicular Ad Hoc Networks) in several studies [13], [14]. Proof of Event (PoE) is introduced in [13] as a two-pass validation on an event which is achieved by Road Side Units (RSUs) and vehicles using two different threshold-based validation algorithms. The work [14] proposes Proof of Location (PoL) where RSUs provide location certificates to vehicles within their communication ranges in order to verify that the event data shared by the vehicles are true. However, even though location can be a good factor in verifying some event data, it is not appropriate for vehicle sensor data verification. In the IoV context, [15] has proposed a trust management system based on blockchain technology and deep learning. It uses a deep learning model using Fully Connected Networks to calculate the trustworthiness of messages shared between vehicles. The model bases the truthfulness of a message on external factors like location, speed, time, vehicle type, and vehicle familiarity, and not the shared message itself. Although such factors can help identify the reliability of the source of a message, such a model cannot guarantee the accuracy of shared sensor data.

In our study, we propose that if an accurate ML model can be created for a task that can be performed using the input sensor data, it should be possible to use the same model to evaluate the accuracy of fresh samples of input data where the real task outcome is known. The novelty of this work is the use of a ML model run by edge servers and validators to perform a task that should produce a common output when run with a specific set of blockchain input sensor data. This allows us to identify incorrect sensor data added to the blockchain and reduce the addition of bad data in the future by identifying the sources from which the data was added. Given the possibility of malicious edge servers, data added by the edge servers is subjected to a second round of validation by validators. The capacity to identify the correct outcome based on the majority output is further increased by the fact that the task outcomes are not required to have a binary value. We validate our proposal by comparing the results with [15], which is the closest implementation to ours in the existing literature.

III. Solution Approach

A. Solution Overview

The novel solution proposed for the blockchain bad data problem in this work uses ML in a blockchain network to identify the accuracy of the data, and proposes a method to reduce the addition of false data using a reputation-based method.

In order to describe the general solution we propose, we incorporate it in an IoV scenario. The proposed solution design for the IoV context is presented in Fig. 1. A pre-trained ML model capable of performing a road surface-type classification using vehicle sensor data is stored on the InterPlanetary File System (IPFS) and installed in all edge servers. It should be noted that multiple models could be stored which could be specific to a geographical area. Initially, the vehicles and edge servers are assigned a moderate reputation score. The process that follows is briefly described below:

1) Vehicles upload data to the edge servers. Each edge server is assumed to collect sensor data from close proximity vehicles within the same road surface-type.
2) The edge server runs the ML model on the uploaded sensor data and, based on the majority classification, labels the data as ‘true’ if the output corresponds with the majority or as ‘false’ otherwise. It also appends the majority classification obtained to each data record in the sample collected. Then it uploads the labeled data to the IPFS and adds the respective hashes from the IPFS to the blockchain. The acceptance of the data to the blockchain is based on the reputation scores of the vehicles and edge servers, as described in step 8.
3) A smart contract on the blockchain activates validation tasks on each data entry that is added to the blockchain.
4) Validators can view pending validation tasks through their application. They can access the data corresponding to a task and the pre-trained ML model from the IPFS and run the model on the data. Validators are a separate group of users contributing to data verification and reputation score calculation.
5) Validators submit votes on whether the label assigned to the data entry is valid based on the ML model outcomes.
6) When a pre-defined threshold vote count is reached, a smart contract determines if the label applied to the data record by the edge server is accurate based on a majority vote from the validators; if not, a new data entry is recorded on the IPFS with the correct label, and the corresponding hash is added to the blockchain.
7) Another smart contract updates the reputation scores of the respective edge server and the vehicle depending on whether the edge server’s initial classifications were accurate and whether the vehicle’s uploaded sensor data contributed to the majority classification, respectively.
8) At the point of data entry to the blockchain, the reputation scores of the vehicles (R_{V_i}) and the edge server (R_{E_x}) that contributed the data are combined to calculate the total reputation score (R_T) as in equation (1). This total reputation score is used to determine if the data should be accepted into the blockchain based on whether it is above a pre-defined threshold reputation score.

$$R_T = R_{E_x} + \sum_{i=1}^{n} R_{V_i} \tag{1}$$

Hence, as time passes, the data from highly reputed vehicles uploaded to highly reputed edge servers are more likely to be added to the blockchain. This decreases the amount of inaccurate sensor data that will be added to the blockchain.

B. Additional Integrations

To incentivize validators to run the ML models and vote based on the results accordingly rather than consistently casting the same vote:

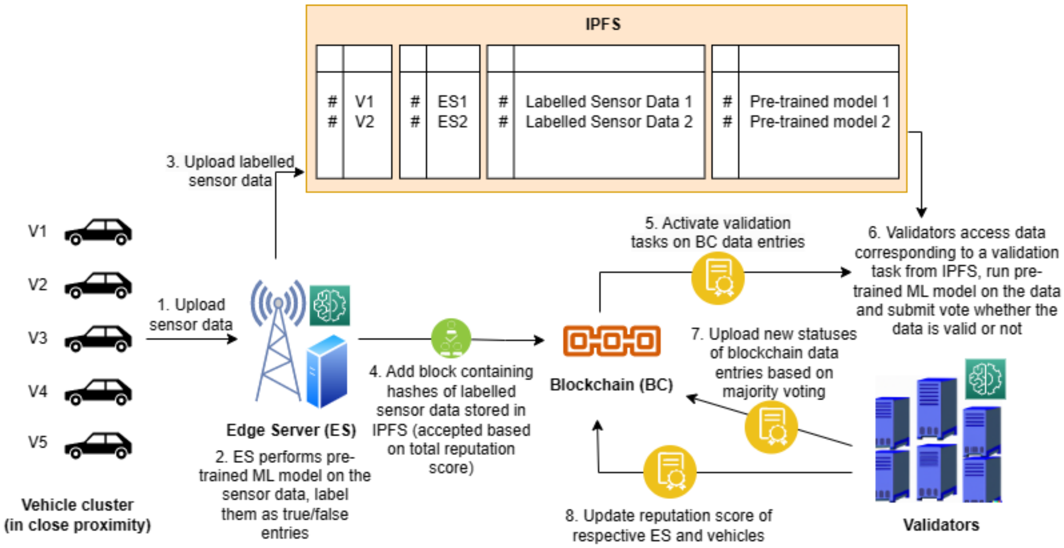
IPFS
# V1
# V2
# ES1
# ES2
# Labelled Sensor Data 1
# Labelled Sensor Data 2
# Pre-trained model 1
# Pre-trained model 2
3. Upload labelled sensor data
V1
V2
V3
V4
V5
Vehicle cluster (in close proximity)
1. Upload sensor data
Edge Server (ES)
2. ES performs pre-trained ML model on the sensor data, label them as true/false entries
4. Add block containing hashes of labelled sensor data stored in IPFS (accepted based on total reputation score)
Blockchain (BC)
5. Activate validation tasks on BC data entries
6. Validators access data corresponding to a validation task from IPFS, run pre-trained ML model on the data and submit vote whether the data is valid or not
7. Upload new statuses of blockchain data entries based on majority voting
8. Update reputation score of respective ES and vehicles
Validators

Fig. 1: Overview of the proposed solution for the IoV context

- Reputation scores are maintained for validators depending on whether their votes contribute towards the majority vote or not
- Two reward pools are maintained to compensate validators - one for situations in which the majority votes for valid data entry and the other for situations in which the majority votes for invalid data entry.

Validators are required to constantly validate data entries to maintain their status as a validator, or else their reputation score is decreased. This helps ensure that the validators actively participate in the validation process.

C. *Blockchain Deployment*

The blockchain network used in the proposed solution is a consortium blockchain where the peer nodes interact with authorized edge servers. Edge servers with properly installed pre-trained ML models are positioned at specific locations along the sides of the road, from where the vehicle data would be collected. Edge servers must be registered on the network by a central administrative body to be able to contribute data to the blockchain. However, the data stored on the blockchain is permanently retained and available for public access.

D. *ML-based Bad Data Selection*

The pre-trained ML model is stored in the IPFS and installed on all edge servers. Within the solution, for each sample of data, the ML model is run first by the edge servers before being added to the blockchain and second by the validators after being added to the blockchain. When data from a vehicle cluster in close proximity within a particular road surface-type is collected by an edge server, the edge server runs the ML model on the set of data and labels each record as ‘true’ or ‘false’ based on the majority classification. In addition, the edge server also adds information on what the majority classification was into each data record. This labeled set of data is stored in the IPFS, and the hash values of the stored data are added to the blockchain. Then, for each data entry added, a validation task is published, which can be accessed by the validators. The validators run the ML model on each data sample corresponding to a validation task and vote on whether the majority classification and the label assigned for a record are valid or not. Based on majority voting from the validators, a smart contract determines whether each data entry contains true or false sensor data and whether the majority classification and labels assigned by the edge servers are correct. At a time of conflict, a new data record is added to the IPFS with the corrected classification and label and a reference to the older entry. The hash of the new data entry is added to the blockchain. Since the ML model is stored in the IPFS, the likelihood of a majority of validators submitting the correct vote is increased as the model output is the same for a given data sample.

E. *Reputation Scheme*

A point-based reputation system is used which assigns a numerical score to vehicles, edge servers, and validators based on their actions. The reputation score is adjusted for vehicles based on the quality of the data they provide, for edge servers according to the accuracy of the labels and classifications they assign, and for validators based on whether the votes they provide align with the majority vote. All updates to reputation scores are carried out using smart contracts. It is assumed that the rewarding mechanism for vehicles and validators based on reputation scores is carried out by the administrative authority.

IV. Experimental Results

To evaluate the performance of the proposed solution, we built a prototype using Hyperledger Fabric 2.0 and Java. Fig. 2 shows the structure of the prototype model implemented. The functionalities of the vehicles, the edge servers, and the validators were simulated using fabric client Java applications. Utilizing the Passive Vehicular Sensors (PVS) Datasets [16] for a road surface-type classification task using Convolutional Neural Network (CNN) models [17], we assess the performance of the proposed solution approach by altering the number of malicious vehicles in vehicle clusters of 200, 100,

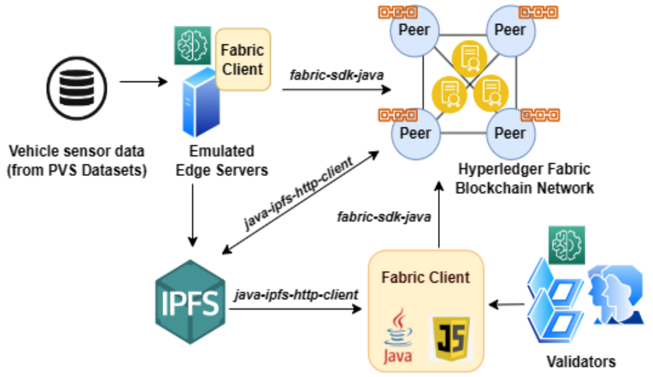

Fig. 2: Prototype implementation

and 50 vehicles. We use the best model achieved in the study [17] as the pre-trained model for our experiments. The best model uses six out of the nine datasets available in PVS datasets for training the model and the remaining datasets as the validation set. The PVS datasets contain sensor data collected from three different vehicles driven by three distinct drivers in three scenarios. The data for testing our experiments were extracted from the datasets set aside as the validation set when building the best model, and in order to create a simulation of vehicles of different counts, the data records used in the validation set were considered as coming from multiple vehicles. To increase the randomization in selecting the test data, data from all three datasets in the validation set were used and were selected randomly in each iteration. The data preprocessing steps carried out were: extracting subsets of fields; normalizing values; and reshaping the data to suit the model. The model architecture consists of three convolutional layers, each with 128 filters and a kernel size of 5, followed by a global average pooling layer, two fully connected layers, each with 128 neurons and a Rectified Linear Unit (ReLU) activation function, and a softmax activation function in the output layer.

Smart contracts were implemented to carry out the following tasks:

- Activating validation tasks on data entries added to the blockchain
- Receiving votes corresponding to the tasks from validators, calculating the majority vote, and adding new corrected entries to the blockchain when mismatches are found
- Updating reputation scores of vehicles and edge servers

Multiple experiments were carried out, and some numerical results obtained are described below.

We define the metric 'data accuracy' as the proportion of all instances that are non-malicious in the data sample. As seen in Table I, when the percentage of malicious vehicles increases, the data accuracy proportionately decreases as more incorrect data are added to the sample, and it shows that the ML model is capable of predicting the data accuracy to a level very close to that of the real data accuracy in all three scenarios. This suggests that our proposal that a precise ML model can be used to determine the correctness of data when the task outcome is known is valid.

TABLE I: Variation of real and predicted data accuracy with malicious vehicle percentage

Malicious Vehicle Percentage	**Real Data Accuracy**	**Predicted Data Accuracy**		
		200	***100***	***50 - vehicles***
0%	1.0	0.975	0.970	0.960
10%	0.9	0.895	0.890	0.893
20%	0.8	0.793	0.785	0.786
30%	0.7	0.694	0.681	0.679
40%	0.6	0.598	0.591	0.584
50%	0.5	0.497	0.487	0.494
60%	0.4	0.395	0.386	0.386
70%	0.3	0.297	0.295	0.294
80%	0.2	0.197	0.196	0.194
90%	0.1	0.100	0.096	0.096

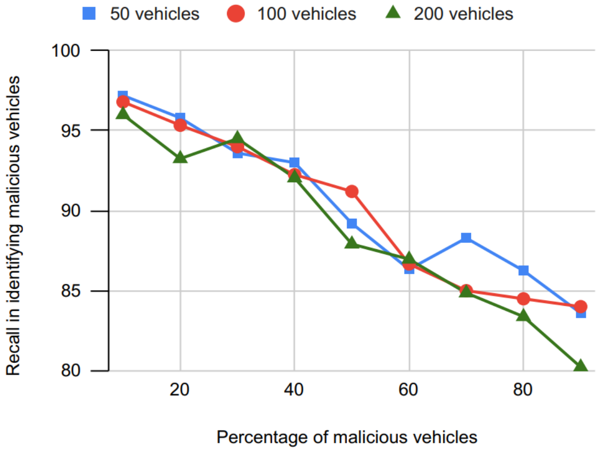

Fig. 3: Variation of recall rate with number of malicious vehicles

As observed in Fig. 3, the recall rate of malicious data identified by the model in all three vehicle samples is above 90% up to a malicious vehicle concentration of 50% and generally over 80% throughout the malicious vehicle concentrations from 10% to 90%. Fig. 4 shows that the precision rate of identifying malicious vehicles is above 90% and keeps increasing with the increase in malicious vehicle percentage.

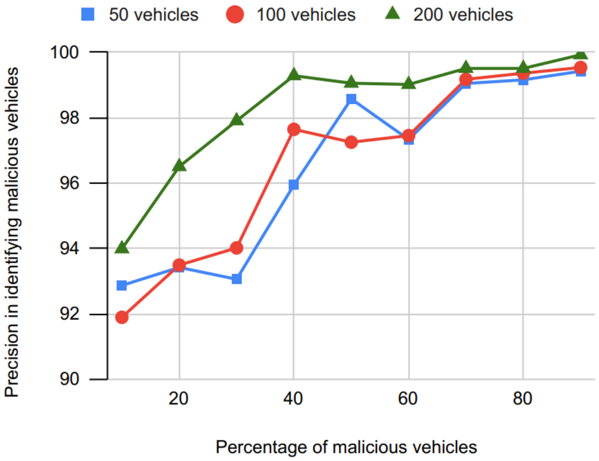

Fig. 4: Variation of precision rate with number of malicious vehicles

This shows that the model is capable of accurately identifying truly malicious vehicles, even in higher proportions of malicious vehicles.

Since the ML model run by edge servers and validators are the same, and the reputation score updates are done based on the validators' majority vote, both the vehicles that provide inaccurate sensor data and the edge servers that upload mislabeled data records could be accurately identified. Hence, the total reputation score of vehicles and the edge server could be used as an ideal metric to determine which data records to be accepted to the blockchain in order to reduce the addition of bad data over time.

V. Discussion

A. Comparison

We compare our findings with those of the study in [15]. The recall and precision values obtained in their work begin at a very high level for smaller concentrations of malicious nodes but show a drastic reduction with the increase in the proportion of malicious nodes. Our findings indicate that even with increased proportions of malicious vehicles, it is still possible to detect them with higher recall and precision rates when utilizing the approach proposed in this research.

Further, we compare the contribution of our paper with related works in Table II.

TABLE II: Comparison of proposed solution with related work

Characteristic	Ref. **[9]–[11]**	Ref. **[13]**	Ref. **[14]**	Ref. **[15]**	**Our Work**
Includes bad data identification	–	✓	–	✓	✓
Involves data source credibility assessment	✓	✓	✓	✓	✓
Focuses on accuracy of data itself	–	–	–	–	✓
Includes bad data reduction method	✓	✓	✓	✓	✓
Solution applicable to IoT sensor data verification	–	–	–	–	✓

B. Limitations

The results highly depend on the specific ML model and how well it is generalized for performing the particular task on fresh data samples. Also, the solution relies on the majority output; hence, it is assumed that most data samples would contribute towards the correct output. However, on a positive note, the majority does not necessarily equate to more than 50% for a task that involves multiple outcomes. Defining threshold values of malicious percentages and reputation scores by conducting more experiments with different data distributions is an important future direction.

VI. Conclusions and Future Works

This paper developed an approach for resolving the issue of bad data in blockchains and described a solution to the problem of faulty sensor data being shared in IoV. The feasibility and performance of our solution approach were demonstrated experimentally by developing a prototype. It could be seen that the approach performs well with both high and low percentages of malicious vehicles in the vehicle cluster. The proposed solution is generalizable and can be applied to any use case where it is possible to create an appropriate ML model for a task that can be carried out utilizing the blockchain input data.

In future works, we plan to incorporate AI-enabled consensus mechanisms for bad data reduction on the blockchain and further analyze the results.

Acknowledgment

This work is partly supported by Science Foundation Ireland under CONNECT phase 2 (Grant no. 13/RC/2077_P2) project.

References

[1] P. K. Singh, R. Singh, S. K. Nandi, K. Z. Ghafoor, D. B. Rawat, and S. Nandi, "Blockchain-based adaptive trust management in internet of vehicles using smart contract," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3616–3630, 2020.

[2] H. Zhang, J. Liu, H. Zhao, P. Wang, and N. Kato, "Blockchain-based trust management for internet of vehicles," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1397–1409, 2020.

[3] U. Javaid, M. N. Aman, and B. Sikdar, "A scalable protocol for driving trust management in internet of vehicles with blockchain," *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11 815–11 829, 2020.

[4] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE internet of things journal*, vol. 6, no. 2, pp. 1495–1505, 2018.

[5] A. Carvalho, J. W. Merhout, Y. Kadiyala, and J. Bentley II, "When good blocks go bad: Managing unwanted blockchain data," *International Journal of Information Management*, vol. 57, p. 102263, 2021.

[6] W. Powell, M. Foth, S. Cao, and V. Natanelov, "Garbage in garbage out: The precarious link between iot and blockchain in food supply chains," *Journal of Industrial Information Integration*, vol. 25, p. 100261, 2022.

[7] P. Howson, "Building trust and equity in marine conservation and fisheries supply chain management with blockchain," *Marine Policy*, vol. 115, p. 103873, 2020.

[8] W. Powell, S. Cao, T. Miller, M. Foth, X. Boyen, B. Earsman, S. del Valle, and C. Turner-Morris, "From premise to practice of social consensus: How to agree on common knowledge in blockchain-enabled supply chains," *Computer Networks*, vol. 200, p. 108536, 2021.

[9] J. Adler, R. Berryhill, A. Veneris, Z. Poulos, N. Veira, and A. Kastania, "Astraea: A decentralized blockchain oracle," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2018, pp. 1145–1152.

[10] R. Kamiya, "Shintaku: An end-to-end-decentralized general-purpose blockchain oracle system," *Online https://gitlab.com/shintakugroup/paper/blob/master/shintaku. pdf*, 2018.

[11] F. Zhang, E. Cecchetti, K. Croman, A. Juels, and E. Shi, "Town crier: An authenticated data feed for smart contracts," in *Proceedings of the 2016 aCM sIGSAC conference on computer and communications security*, 2016, pp. 270–282.

[12] J. Guarnizo and P. Szalachowski, "Pdfs: practical data feed service for smart contracts," in *Computer Security–ESORICS 2019: 24th European Symposium on Research in Computer Security, Luxembourg, September 23–27, 2019, Proceedings, Part I 24*. Springer, 2019, pp. 767–789.

[13] Y.-T. Yang, L.-D. Chou, C.-W. Tseng, F.-H. Tseng, and C.-C. Liu, "Blockchain-based traffic event validation and trust verification for vanets," *IEEE Access*, vol. 7, pp. 30 868–30 877, 2019.

[14] R. Shrestha, R. Bajracharya, and S. Y. Nam, "Blockchain-based message dissemination in vanet," in *2018 IEEE 3rd international conference on computing, communication and security (ICCCS)*. IEEE, 2018, pp. 161–166.

[15] S. Wang, Y. Hu, and G. Qi, "Blockchain and deep learning based trust management for internet of vehicles," *Simulation Modelling Practice and Theory*, vol. 120, p. 102627, 2022.

[16] J. Menegazzo, "Pvs - passive vehicular sensors datasets," 2021. [Online]. Available: https://www.kaggle.com/ds/1105310

[17] J. Menegazzo and A. von Wangenheim, "Road surface type classification based on inertial sensors and machine learning," *Computing*, Feb. 2021. [Online]. Available: https://doi.org/10.1007/s00607-021-00914-0