Towards Robust AI Model for Cyber-Physical Intelligent Transportation Systems by Hash-Based Ensemble Learning

Kamrul Hasan[1],Varshini Guduru[1], Saleh Zein-Sabatto[1], Deo Chimba[1], Imtiaz Ahmed[2]
[1] Tennessee State University, Nashville, TN, USA
[2]Howard University, Washington, DC, USA
Email: {*mhasan1, vguduru, mzein, dchimba*}@tnstate.edu, {*imtiaz.ahmed*} @howard.edu

2023 IEEE Latin-American Conference on Communications (LATINCOM) | 979-8-3503-2687-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/LATINCOM59467.2023.10361868

***Abstract*—In the evolving landscape of vehicular technology, autonomous and connected features lead to heightened connectivity among vehicles, intelligent devices, and infrastructures, broadening the vulnerability to cyber-attacks within the Internet of Vehicles (IoV) systems. Intrusion Detection Systems (IDSs) leverage Machine Learning (ML) to detect these malicious intrusions, but they face challenges from data poisoning attacks, given the critical role of training data in threat modeling. The vulnerability of training data emerges as a significant threat vector during ML model training. This study introduces a hash function-enabled ensemble ML training framework tailored for IDS, mitigating data poisoning attacks during model training. We test and validate our framework using hash-enabled ensemble ML algorithms—including random forests, support vector machines, and decision trees—on the benchmark CICIDS 2017 dataset. Results demonstrate that the hash-enabled random forests model guarantees a misclassification rate below 0.5%.**

I. Introduction

Amidst the swift evolution of autonomous and connected technologies, vehicles are progressively assimilating into an expansive ecosystem comprising other vehicles, smart devices, and infrastructure. This amalgamation proffers myriad advantages, including augmented road safety, streamlined traffic regulation, and refined driving experiences. However, this pervasive connectivity also uncovers novel vulnerabilities, potentially exposing the Internet of Vehicles (IoV) systems to cyber threats [1]. IoV systems, anchored in flawless communication and coordination amongst vehicles and their environment, are susceptible to nefarious cyber incursions. Such intrusions can harness the interconnectivity of vehicles to subvert operational coherence, imperil safety, and infringe upon the confidentiality of vehicle occupants. Hence, there is an unequivocal necessity to architect stalwart defense stratagems to insulate IoV systems from cyber adversities [2]. Conventional IDSs predominantly lean on pre-established rules or signatures, circumscribing their prowess in pinpointing nascent or mutating threat paradigms. In contravention of this restraint, Machine Learning (ML) methodologies have emerged as frontrunners in sculpting IDSs adept at navigating capricious threat landscapes. This manuscript delineates the formulation of an IDS framework predicated on a hash-augmented ensemble model deploying Random Forest Machine Learning. The advocated framework endeavors to discern many attack vectors in IoV networks by harnessing the synergy of ensemble learning and hashing modalities. Ensemble learning amalgamates multiple ML prototypes to elevate detection precision while hashing modalities intensify the proficiency of pattern correlation within voluminous datasets. A paramount impediment in ML-driven IDSs is the security appraisal of such paradigms, especially in mission-critical domains like IoV systems. The adulteration of training datasets, termed data poisoning, emerges as a profound menace, potentially jeopardizing the efficacy and trustworthiness of ML architectures. Such malevolent data manipulations have been evidenced across diverse ML algorithms. In retaliation to these challenges, the mooted hash ensemble IDS paradigm not solely accentuates detection acuity but also contends with the susceptibilities of ML frameworks to data poisoning onslaughts. With the integration of rigorous hashing mechanisms and ensemble education, the architecture seeks to proffer a robust and tenacious bulwark against cyber adversaries in IoV matrices. The cardinal ambition of this investigation is to cultivate a sophisticated IDS blueprint adept at pinpointing and neutralizing cyber incursions within IoV architectures. The propounded hash ensemble construct emerges as a promising conduit to fortify vehicular cyber defenses and amplify the inherent robustness of IoV frameworks. This treatise aspires to manifest the efficacy and applicability of the proffered paradigm in tangible operational environments through exhaustive appraisals and elucidations.

The primary contributions of this paper are as follows:

- Introduction of an innovative framework fusing ensemble learning and hashing techniques for enhanced IoV attack detection.
- Assurance of ML model reliability in IoV systems by safeguarding training datasets against malicious modifications.
- Amplification of detection accuracy for diverse cyber-attacks through the adoption of ensemble learning.
- Validation of the framework's effectiveness and resilience via real-world IoV applications and examination on the CICIDS2017 dataset [3].

The rest of the paper is organized as follows: Section II reviews pertinent related works on the robustness of IDS in IoV. Section III introduces the systems architecture. Section IV elucidates data preparation, the implementation, and the result analysis of the hash-based ensemble random forest models. Finally, Section V offers concluding remarks based on our findings.

II. Related Work

The escalating connectivity in IoV systems has accentuated concerns related to their security, driving significant research focus. Among the prevalent methods, ML techniques for intrusion detection in vehicular networks have been a mainstay. Notwithstanding their value, existing studies present certain gaps. Islam et al. [4] showcased an anomaly detection framework for vehicular ad-hoc networks utilizing a fusion of unsupervised and supervised learning. Though this framework registered commendable accuracy, it was contingent upon the caliber of training data, potentially compromising adaptability to emergent threats. In contrast, Ma et al. [5] presented a deep learning-powered real-time intrusion detection for IoV. Their system highlighted formidable detection rates with minimal false positives, signifying deep learning's prowess in this domain. Yet, vulnerabilities to data poisoning attacks were unaddressed.

Addressing data poisoning, Li et al. [6] unveiled a Deep Reinforcement Learning (DRL) framework tailored for crowd-sensing systems. They circumvented particular poisoning challenges by employing a biphasic DRL algorithm for data selection during training. Nonetheless, the method posed computational challenges and made potentially compromising assumptions regarding adversarial knowledge. Several other researchers [7]–[11] suggested weighted ensemble strategies, yet these remain susceptible to manipulative attacks targeting training data weights. Sun et al. [12] introduced an adaptive mechanism countering data poisoning in federated ML systems. Preliminary evaluations on synthetic datasets registered superiority over peers in accuracy and robustness metrics. However, certain caveats like untested real-world performance and potential computational intensiveness linger. Chen et al. [13] presented a rigorous review of data poisoning attacks in IoV contexts. Despite its comprehensiveness, this study narrowed its scope to IoV-specific poisoning attacks and omitted contemporary advancements. Yerlikaya and Bahtiyar [14] delivered a holistic literature review centered on ML algorithmic vulnerabilities to data poisoning, encapsulating attack techniques, and ramifications. The analysis, though insightful, was restricted to data poisoning, sidelining other attack modalities.

As a defense against poisoning, Anisetti et al. [15]. advanced a hashing mechanism to condense training data. However, this compression possibly compromised the integrity of the data and the consequent IDS accuracy.

The hash ensemble IDS framework proposed herein aspires to remediate these gaps, enhancing intrusion detection accuracy in IoV systems. Recognizing the imperative of thoroughly assessing ML models' security, especially in mission-critical applications, this work hopes to augment this evaluative paradigm significantly.

III. Architecture

A. The network and threat model

The Internet of Vehicles (IoV) architecture in Fig. 1 has transformed the automotive industry, enabling vehicles to communicate internally and externally. Internally, cars use the Controller Area Network (CAN) bus, a robust communication protocol specifically designed for automotive environments. The CAN bus oversees the transmission of signals between various electronic control units (ECUs) within the vehicle. A gateway is essential in facilitating external communications with infrastructures and other vehicles, serving as the convergence point between internal and external networks. This becomes especially vital as we move towards a more connected world with Vehicle-to-Everything (V2X) communications, where vehicles continuously interact with their surrounding environment. Given the sensitive nature of this data exchange, security becomes paramount. Hence, integrating an intrusion detection system (IDS) within the gateway is crucial. This IDS can detect and mitigate real-time data modifications or poisoning attacks. Such attacks, whether from internal or external sources, can compromise the vehicle's functionality and safety. By leveraging a hash-based ensemble random forest model, the IDS provides an added layer of security. This model allows for quick and accurate detection of any discrepancies in data transmission, ensuring a safer and more secure IoV ecosystem.

B. Hash-based ensemble random forest enabled IDS

The working process of the hash-based ensemble random forest enabled IDS is depicted in Fig. 2, and step by step evolution is given below:

Dataset Splitting: The initial dataset is bifurcated into training (80%) and testing datasets (20%). The training dataset instructs the machine learning model, while the testing dataset assesses its efficiency.

Training Set Creation: Three data types comprise the training set:

- **Benign Data:** Represents standard, uncompromised system operations.
- **Attack Data:** Originates from compromised or attacked systems.
- **Mixed Data:** An amalgamation of benign and attack data, this is employed to gauge model proficiency.

Model Generation: A machine learning model incorporating the Hash-based Ensemble framework is constructed. This algorithm combines hash functions with random forests for effective data categorization. Here, hash functions process the data into a condensed format, facilitating the random forest's role in data classification. The hashing equation ensures that features are randomly distributed into buckets,

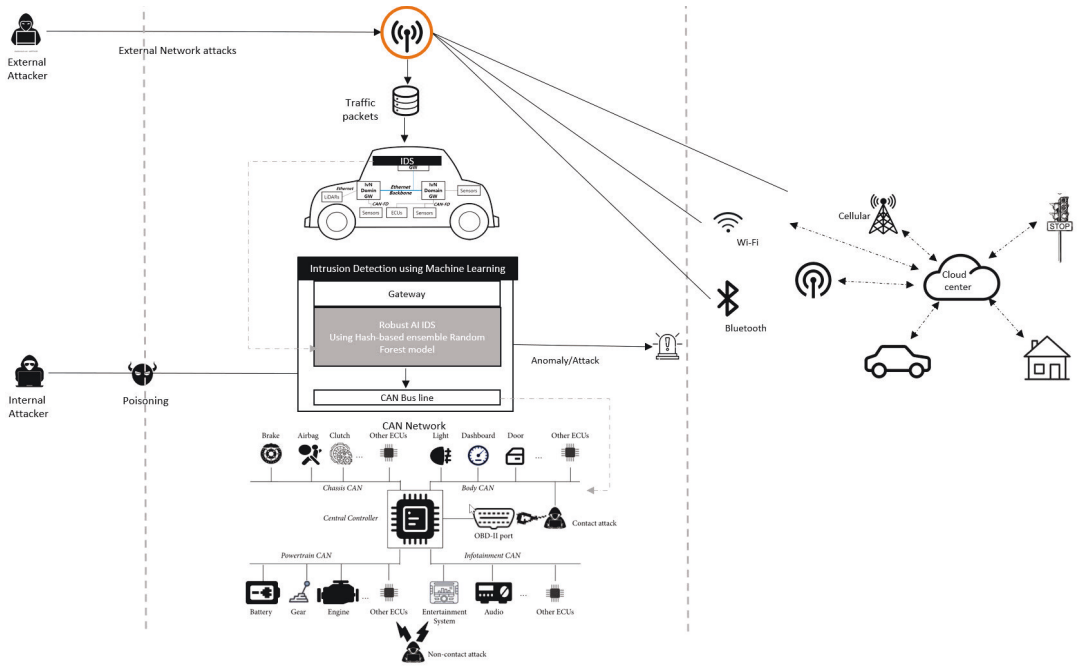

Fig. 1. Cyber-physical system of the intelligent transportation system.

and while there may be collisions (different features ending up in the same bucket), this randomness can be beneficial for training diverse trees in the ensemble.

Model Training N distinct Hash-based random forest models, where N might be a modest number like 10 or 20, are instructed using the (20%) training datasets. A supervised learning approach directs this training.

Model Evaluation: Models cultivated in the training step undergo testing, subject to varied attack percentages. By manipulating the attack ratio from 0% to 100%, the model's precision is deduced from the quotient of accurate predictions and total predictions.

Optimal Model Selection Employing the majority voting technique, the best-performing model is identified. This algorithm selects the label with the highest frequency for a fresh data point. Due to its straightforwardness and efficacy, majority voting ensures that the most apt model for precise predictions is chosen.

C. Mathematical representation of hash-based ensemble random forests models:

The majority voting equation for a hash-based ensemble of random forests is articulated as follows: Let S be an input string and M represent the ensemble model comprised of several hash-based random forest models. Each model within the ensemble is symbolized by M_i, where i iterates from 1 through N, with N signifying the cumulative count of models in the ensemble. For each model, M_i in M, apply the corresponding hash function $H_i()$ to the instance X, which produces a hash value $H_i(S)$. The hash value is calculated from the eqn.1.

$$H(S) = \sum_{i=1}^{L-1} f(S[i]) mod N \tag{1}$$

Where $H(S)$ is the hash value of the string S, $f(s[i])$ is the hash value of the character $S[i]$, L is the length of the string S, and N is the number of models.

Using the model M_i to predict the Class Label (CL) for the hashed instance $H_i(S)$. The CL can be defined as eqn.2

$$CL = M_i.predict(H_i(S)) \tag{2}$$

Now, majority voting is typically performed by selecting the class label that receives the highest number of votes from the ensemble models. If there is a tie, additional tie-breaking strategies can be employed, such as selecting the class label with the highest confidence or considering the class label predicted by the leader model. Mathematically, the majority voting can be represented as eqn.3:

$$MV(S) = \underset{CL}{\operatorname{argmax}} \sum_{i=1}^{N} \mathbf{Indicator}(M_i.\mathrm{pred}(H_i(S)) = CL) \tag{3}$$

where $MV(S)$ represents the final predicted class label for the string S, $\underset{CL}{\operatorname{argmax}}$ returns the CL that maximizes the expression $Indicator(M_i.predict(H_i(S)) = CL$ is an indicator function that equals 1 if the prediction of model M_i for the hashed instance $H_i(S)$ is equal to class label CL, and 0 otherwise, and $\sum_{i=1}^{N}$ indicates the summing of indicator functions over all models in the ensemble.

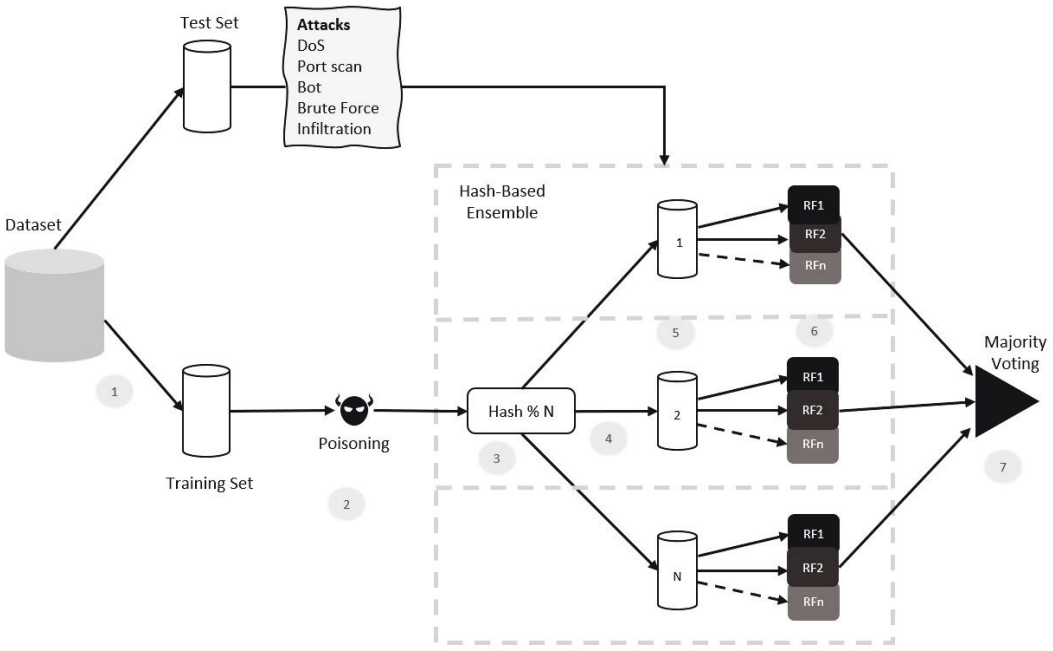

Fig. 2. Process flow of the proposed hash-based ensemble mechanism using random forest models.

IV. Data preparation, Experimental Settings, and Result Analysis:

A. Data Preparation:

We utilized the CICIDS2017 dataset [3] to emulate a car's internal CAN bus and external gateway communications, where our hash-ensembled, random forest-based IDS was integrated. The dataset contains 2,257,797 network traffic entries, with 1,499,522 standard and 758,275 attack records. These records are divided into a training set of 1,599,265 and a testing set of 658,532.

The internal car dataset was derived from genuine network traffic captured in a controlled setting using the CICFlowMeter tool, transforming packet-level details into flow-level data. In contrast, the external set simulated diverse network activities, encompassing regular traffic and attack patterns. Each network flow in the dataset is detailed with features like statistical measures, protocol specifics, flow duration, and service-related attributes, offering a comprehensive view of network behaviors.

The dataset catalogs various traffic types, each labeled appropriately, from benign to different attack modalities. Notably, the training set was modified to mimic data poisoning attack patterns. The dataset also mirrors real-world conditions, showcasing a class imbalance with benign traffic predominating over malicious actions.

B. Experimental Setup:

Our hash-ensembled random forest implementation utilizes ensemble models ranging from 10 to 20 individual models. This intricate design was executed using select Python libraries, ensuring optimal compatibility and efficiency. Importantly, to guarantee robust computational performance and seamless execution of our models, we deployed the algorithm on our dedicated cluster server. This server is equipped with state-of-the-art hardware, featuring an 11th Generation Intel® Core™ i9-11900KF processor that operates at a clock speed of 3.50GHz across 16 cores. Complementing this powerful CPU is a substantial Random Access Memory (RAM) of 62.5GB, which ensures efficient data handling and concurrent processing capabilities. Such a robust computational environment accelerates our model's training and inference times and guarantees precision and consistency in performance.

C. Result Analysis:

The performance of the framework was assessed using the following metrics:

- **Accuracy:** Percentage of records correctly classified.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precision:** Percentage of records classified as positive that are truly positive.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall:** Percentage of positive records that were correctly classified.

$$\text{Recall} = \frac{TP}{TP + FN}$$

TABLE I
EVALUATION RESULTS

Algorithm	Accuracy%	Precision%	Recall%	F1-score%
Hash-based ensemble framework using random forest Models	99.60	99.70	99.50	99.60
Hash-based ensemble framework using Support Vector Machines	94.50	93.60	93.40	93.50
Hash-based ensemble framework using Decision Trees	95.40	96.50	93.30	94.87

- **F1-score:** Weighted average of precision and recall.

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

Where:

- TP (True Positives): Number of records correctly classified as positive.
- TN (True Negatives): Number of records correctly classified as negative.
- FP (False Positives): Number of records incorrectly classified as positive.
- FN (False Negatives): Number of records incorrectly classified as negative.

Now, Table I presents the evaluation results of our proposed hash-based ensemble framework utilizing Random Forest models. Our findings indicate that this framework outperforms other methods regarding accuracy, precision, recall, and F1 score. Even when subjected to a poisoned dataset, the system retains a superior level of accuracy, a testament to the efficacy of the integrated hash function and ensemble approach. The ensemble strategy, which encompasses multiple random forest models, aggregates predictions. By relying on a majority voting system, this diminishes the potential impact of individual poisoned models or data instances. Furthermore, hash-based indexing and partitioning ensure the dataset's even distribution among the ensemble models. This strategic partitioning hinders direct manipulation of the comprehensive dataset and acts as a defensive measure against expansive poisoning attacks. Continuous performance monitoring and validation are integral components of our methodology to bolster the system's resilience. This vigilant oversight facilitates the early detection of anomalies or inconsistencies potentially caused by data poisoning. Furthermore, periodic retraining and model updates fortify the system's defenses, making it adaptable to emergent attack paradigms.

Table II presents the attack percentage in each dataset alongside the accuracy of five random forest models (RF1 through RF5) on benign samples only. Notably, all models achieved 100% accuracy on the dataset, which lacks attacks. As the attack percentage rises in other datasets, model accuracy decreases, underscoring the impact of attacks on classification performance.

TABLE II
TRAINED MODEL: BENIGN PERFORMANCE

Attack%	RF1%	RF2%	RF3%	RF4%	RF5%
0	100.00	100.00	100.00	100	100.00
10	95.01	95.01	95.01	95.01	95.01
15	89.99	89.99	89.99	89.99	89.99
20	85.01	85.01	85.01	85.01	85.01
25	80.01	80.01	80.01	80.01	80.01
30	75.01	75.01	75.01	75.01	75.01
35	70.00	70.00	70.00	70.00	70.00
40	65.00	65.00	65.00	65.00	65.00
45	60.01	60.01	60.01	60.01	60.01
50	55.00	55.00	55.00	55.00	55.00

TABLE III
TRAINED MODEL: ATTACK PERFORMANCE

Attack%	RF1%	RF2%	RF3%	RF4%	RF5%
0	0.00	0.00	0.00	0.00	0.00
10	4.99	4.92	4.96	4.92	4.92
15	10.01	9.94	9.98	9.94	9.91
20	14.96	14.91	14.94	14.91	14.88
25	19.96	19.91	19.91	19.91	19.88
30	24.97	24.92	24.92	24.92	24.89
35	29.98	29.93	29.91	29.91	29.88
40	34.97	34.92	34.9	34.9	34.87
45	39.97	39.91	39.88	39.88	39.85
50	44.97	44.91	44.87	44.87	44.84

Table III evaluates the models' ability to detect attacks based on specific metrics and thresholds. The performance metrics indicate the detection rates of the random forest models on attack samples only. All models registered a 0% detection rate for data devoid of attacks. As attacks increased in other datasets, models exhibited enhanced detection capabilities, with no benign instances misclassified. While performance varied among models (RF1 to RF5), each demonstrated consistent improvements in detection rates correlating with the increasing percentage of attacks in datasets.

In Table IV, the models exhibit consistent accuracy rates between 98.79% and 99.79% across varying datasets. The accuracy remains stable even with increased attack percentages, highlighting the models' robustness. These results also underscore the effectiveness of the trained random forest models in discerning between benign and attack samples within mixed datasets.

In Table V, we present the results of our ensemble model,

TABLE IV
TRAINED MODEL: MIXED PERFORMANCE

Attack%	RF1%	RF2%	RF3%	RF4%	RF5%
0	99.66	99.79	99.59	99.69	99.49
10	99.61	99.71	99.48	99.41	99.38
15	99.63	99.6	99.44	99.26	99.29
20	99.65	99.56	99.39	99.04	99.24
25	99.62	99.51	99.34	98.71	99.15
30	99.64	99.43	99.36	98.58	99.12
35	99.59	99.23	99.3	98.39	99.04
40	99.57	99.11	99.24	98.19	98.96
45	99.53	98.98	99.13	97.93	98.84
50	99.54	98.89	99.08	97.74	98.79

TABLE V
MAJORITY VOTING PERFORMANCE RESULTS

RF1%	RF2%	RF3%	RF4%	RF5%
99.66	98.98	99.59	97.93	99.49

which was determined using a majority voting mechanism. The ensemble approach primarily utilized the Max voting method, exemplified by the random forest model outcomes. This methodology amalgamates predictions from all constituent models and identifies the one that consistently delivers the highest accuracy. The ensemble model harnesses their collective insights by synergistically combining these predictions, resulting in robust and reliable decisions.

Finally, in Fig.3, accuracy is employed as the principal metric for assessing the performance of each model. The model boasting the highest accuracy is discerned as the optimally trained one, signifying its proficiency in distinguishing instances as benign or malicious. Utilizing this criterion not only pinpoints the model exhibiting superior performance but also establishes a robust foundation for intrusion detection within the given dataset.

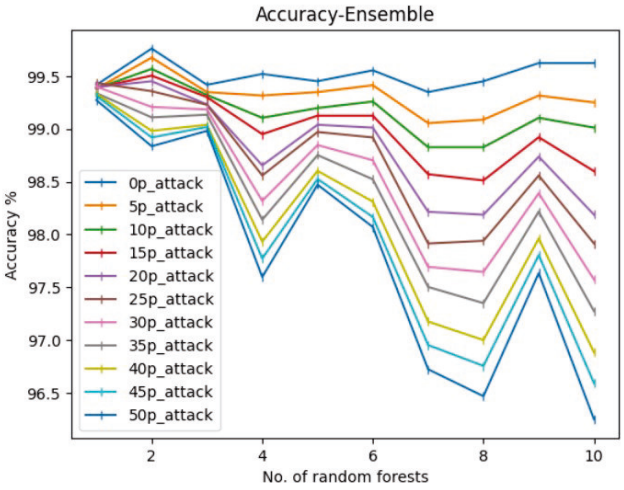

Fig. 3. Max voting results from random forest models

V. CONCLUSION

This research introduces a hash-based ensemble framework optimized for intrusion detection within the CICIDS 2017 dataset. By harnessing the diversity of hash functions and the prowess of random forest models, we have elevated intrusion detection's precision, robustness, and accuracy. A rigorous evaluation, through segregating the dataset into training and test sets, ensures a thorough validation of the framework's intrusion detection capabilities. Our results, marked by high accuracy, precision, and recall, are a testament to the efficacy of our approach. The majority voting mechanism in the ensemble model fortifies decision-making, while the iterative refinement of hash functions and random forest hyper-parameters underscores the adaptability and potency of the framework. Collectively, our hash-based ensemble framework is a novel contribution to intrusion detection, demonstrating enhanced detection accuracy and emphasizing its pertinence in addressing pressing network security challenges.

ACKNOWLEDGMENT

This material is based on work supported by the National Science Foundation Award Numbers 2205773 and 2219658.

REFERENCES

[1] M. Houmer, M. Ouaissa, M. Ouaissa, and S. Eddamiri, "Applying machine learning algorithms to improve intrusion detection system in iov," *Artificial Intelligence of Things in Smart Environments: Applications in Transportation and Logistics*, p. 35, 2022.

[2] Y. Sun, L. Wu, S. Wu, S. Li, T. Zhang, L. Zhang, J. Xu, and Y. Xiong, "Security and privacy in the internet of vehicles," in *2015 International Conference on Identification, Information, and Knowledge in the Internet of Things (IIKI)*, pp. 116–121, IEEE, 2015.

[3] R. Panigrahi and S. Borah, "A detailed analysis of cicids2017 dataset for designing intrusion detection systems," *International Journal of Engineering & Technology*, vol. 7, no. 3.24, pp. 479–482, 2018.

[4] S. H. Haji and S. Y. Ameen, "Attack and anomaly detection in iot networks using machine learning techniques: A review," *Asian journal of research in computer science*, vol. 9, no. 2, pp. 30–46, 2021.

[5] Y. Dong, R. Wang, and J. He, "Real-time network intrusion detection system based on deep learning," in *2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS)*, pp. 1–4, IEEE, 2019.

[6] M. Li, Y. Sun, H. Lu, S. Maharjan, and Z. Tian, "Deep reinforcement learning for partially observable data poisoning attack in crowdsensing systems," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6266–6278, 2019.

[7] M. F. Elrawy, A. I. Awad, and H. F. Hamed, "Intrusion detection systems for iot-based smart environments: a survey," *Journal of Cloud Computing*, vol. 7, no. 1, pp. 1–20, 2018.

[8] G. O. Anyanwu, C. I. Nwakanma, J.-M. Lee, and D.-S. Kim, "Real-time position falsification attack detection system for internet of vehicles," in *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–4, IEEE, 2021.

[9] W. Fu, X. Xin, P. Guo, and Z. Zhou, "A practical intrusion detection system for internet of vehicles," *China Communications*, vol. 13, no. 10, pp. 263–275, 2016.

[10] G. O. Anyanwu, C. I. Nwakanma, J.-M. Lee, and D.-S. Kim, "Falsification detection system for iov using randomized search optimization ensemble algorithm," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[11] K. Hasan, S. Shetty, and S. Ullah, "Artificial intelligence empowered cyber threat detection and protection for power utilities," in *2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC)*, pp. 354–359, IEEE, 2019.

[12] G. Sun, Y. Cong, J. Dong, Q. Wang, L. Lyu, and J. Liu, "Data poisoning attacks on federated machine learning," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 11365–11375, 2021.

[13] Y. Chen, X. Zhu, X. Gong, X. Yi, and S. Li, "Data poisoning attacks in internet-of-vehicle networks: Taxonomy, state-of-the-art, and future directions," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 20–28, 2022.

[14] F. A. Yerlikaya and Ş. Bahtiyar, "Data poisoning attacks against machine learning algorithms," *Expert Systems with Applications*, vol. 208, p. 118101, 2022.

[15] M. Anisetti, C. A. Ardagna, A. Balestrucci, N. Bena, E. Damiani, and C. Y. Yeun, "On the robustness of ensemble-based machine learning against data poisoning," *arXiv preprint arXiv:2209.14013*, 2022.