ISSN:2731-538X | E-ISSN:2731-5398
Vol. 18 No. 02 (2024)

ANALYZING THE IMPACT OF MACHINE LEARNING-BASED SECURITY MEASURES ON QOS IN IOT NETWORKS

Mohit Sharma[1], Dr. Sanjay Kumar Sharma[2], Dr. Rejesh Kumar Bhogey[3], Dr. Bhupendra kumar[4]
M.Tech.Scholar[1], Associate Professor[2] , Associate Professor[3], Associate Professor[4]
mohit1493sharma@gmail[1].com, sanjaysharmaemail@gmail.com[2], rajeshboghey@gmail.com[3], bkverma3@gmail.com[4]
Technocrats Institute of Technology (Excellence) Bhopal

Abstract- The emergence of the Internet of Things (IoT) and other innovative technologies has enabled the interconnection of devices worldwide, earning them the moniker "smart gadgets" due to their capabilities to send, receive, and process data. This technology is experiencing rapid growth, with a continually increasing user base. The success of IoT hinges on factors such as data transmission rates, quality of service (QoS) maintenance, and management of energy constraints in battery-operated devices. At the network level, QoS is evaluated based on metrics including end-to-end delay, throughput, jitter, and packet delivery ratio. With the proliferation of IoT devices, ensuring both device and data security in network communications becomes paramount. This paper delves into algorithms employed to safeguard the locations of source and sink nodes from potential breaches. Additionally, it investigates the impact of AODV protocols on the QoS offered by IoT networks. Malware poses significant threats in this context, prompting researchers, industry professionals, and end-users to seek effective countermeasures. Early and accurate prediction of malware behavior is crucial to mitigate potential damage. The research endeavors to combat malware using the K-Nearest Neighbors (KNN) algorithm, predicting its behavior and eliminating it. Utilizing these classifiers appropriately can significantly enhance prediction accuracy.

Keywords - K–Nearest Neighbor, IoT, QoS, Malware detection, Wireless Network, Security

I INTRODUCTION

The Internet of Things (IoT) is a system that lets physical objects be linked together and monitored over the internet. Things like computers, digital machines, electrical or home appliances, and so on, all of which have their own digital identities, are connected to the things around them and can share data with them. This makes it possible for the objects to connect and talk to each other in a smart way. With the help of the Internet of Things (IoT), things can connect without any interaction between people or between people and digital devices. Even though we think of tablets, laptops, computers, and cell phones as ways to connect, in reality, things can connect without us having to do anything. The needs of people who use the Internet of Things have led service providers to make a wide range of apps. The quality of services (QoS) that customers want from an app can vary from person to person. Similarly, the QoS of the many apps that use the internet of things will also be different (IoT)[1]. For each application, the quality metrics should be set in a very clear way, so that a user can tell the service provider what he expects and the service provider can make changes to meet those expectations. So, the researchers should put most of their efforts into finding the QoS (Quality of Service) indicators to find out what IoT service users want. Due to how quickly the internet has grown, the number of cyber risks caused by malware has also gone up. One definition of malware says that it is a type of computer programme that is made to hurt the other user's computer in a number of ways. Malware comes in so many different forms now, and anyone can buy malware on the dark web to increase the number of attacks they launch against our system. This makes it very hard for anti-virus software to fully protect a computer. Malware, also called "malicious software," is a programme that sneaks into a computer system without the user's permission and tries to damage the system or steal private information that is stored on the system. Malicious software, which is also called "malware," is any piece of software that is made to do something bad on purpose by an enemy. Malwares are called things like viruses, worms, Trojan Horses, root-kits, spyware, backdoors, botnets, and adware, among other things, based on how they behave and how they infect computers. [2-3] every day, thousands of new malicious software programmes are made, and the structure of already existing malicious software programmes is always changing, making it harder and harder to find them.

Symantec's most recent report on internet threats says that 317 million new types of malicious software have been found. Because more samples of malware are being made every day, automated tools and methods are needed to tell the difference between harmful and harmless code. Signature-based malware classification is used by almost all anti-virus software on the market. This method compares the unknown malware to a database of known malicious programmes to find out if the file in question has malware or is safe to use. A unique identifier that can be added to a binary file is called a signature. Malware's signature can be found through static analysis, dynamic analysis, or a combination of the two. Once the signature is found, it is saved in a database called the signature.[4] The biggest problem with this strategy is that the signature database needs to be updated often because new malware is created so quickly every day. Symantec's latest report on internet threats says that 317 million new types of malware have been found. Because there are more new samples every day, automated tools and methods are needed to tell the difference between malicious and safe code. Most commercial anti-virus software uses a method based on signatures to sort malware.[5-6] This method compares unknown malware to a database of known malware to figure out if a file is malware or not. The signature is a way to identify a binary file in a unique way. Malware's signature can be found using static analysis, dynamic analysis, or a mix of the two. The signature is then stored in a database called a "signature database." The biggest problem with this method is that the signature database needs to be updated often because new malware comes out every day [7].

IoT helps connecting help physical world to connect to computer world. As its application are increasing day by day, privacy issues are also increasing. Different attacks like spoofing, DDoS attack, and jamming, malware and eavesdropping are becoming potential threats. Small IoT devices are restricted to execute computational-intensive and latency sensitive security tasks. Today, the IoT devices are protected using authentication, in which source nodes are identified and identity based attacks are prevented, access control, secure offloading techniques and malware detection to prevent against privacy leakages. These techniques are not applied to small IoT devices like outdoor sensors, so spoofing attack on them is not recognized [8]. Machine learning techniques are applicable on IoT devices whether small or large. These techniques include: Supervised Learning: This includes support vector machine, naïve Bayes, neural network, deep neural network, random forest, K nearest neighbor to track network traffic or app traces of IoT devices to build classification or regression models [9-10].

II LITERATURE SURVEY

The author of [11] says that the main goals of WSNs and IoT are to reduce the amount of power used to make the network last longer and to make sure it is safe. Mamdani An energy efficient secure route adjustment (ESRA) model, which is explained in [12-13], used fuzzy logic to figure out the most energy-efficient way to communicate. It figures out the best route by adding up the values of a number of quality-of-service criteria. Sink nodes can be moved, but to set up a new route, you need to know where the currently used sink node is. This method uses little power because it chooses the route based on how reliable it is. Cluster-based routing algorithms need a lot more energy to work because there are so many intermediate nodes in the path of communication.

The authors of [14] came up with a double level unequal clustering algorithm (DLUC) to solve the problem of clustering techniques using more power than they used to.[15] This algorithm tells each cluster how much traffic it needs to handle. Since the cluster heads don't have to be present for information to flow between nodes, the number of clusters and nodes along the transmission line can be cut. Networks can use less energy if the bandwidth is optimized, the values of interference between control packets are lowered by using framing periods to avoid congestion, and the values of data loss are lowered. This method doesn't take into account how people move around and how different clusters are sized, which are two major factors that cause networks to use more power.[16-17]

This section will try to list a few ways that have been made to keep track of the positions of source nodes, sink nodes, or sometimes both source and sink nodes, while an Internet of Things application is running.[18] We also looked into how security algorithms affect how efficiently they use energy and how they affect quality of service measures like throughput, end-to-end delay, and packet delivery ratio. [19-20] Where the source node is the idea of source location privacy (SLP) has become a difficult problem for research institutions to solve if they want to keep their networks safe. If SLP is not present, it will be easier to figure out where the source nodes are and get to the data before it is sent along the communication route [21-22]. Most of the time, a route made by the sensors will have a source node, a sink node, and a few nodes in between. In order to move data packets, these intermediate nodes will use hopping techniques. Some of the research shows that it is easy for the source's enemies to figure out where the source is, even if they know where some of the nodes are along the route that is currently being taken [23]. Because of this, it is very important to make SLP algorithms to protect IoT networks from security attacks. In [21], a technique called SDR-m was described. SDR-m is a stochastic and diffuse routing [24-26]

III PROPOSED METHODOLOGY

The block diagram depicted in Figure 1 and fig.8 illustrates our proposed approach. it is apparent that no single integrated algorithm has yet been identified to concurrently enhance both the overall quality of service and safety of the network. As a result, our research is multifaceted, addressing a range of concerns including security issues (such as anonymity, route security, and data security) as well as enhancements in quality of service (such as route selection and optimization of node performance). The integration of algorithms across these diverse domains presents challenges due to their disparate nature. Therefore, our research endeavors are focused on meeting a critical need: developing a unified algorithm capable of effectively addressing both quality of service and security, which are fundamental aspects of wireless networks. Through our research efforts, we aim to demonstrate the efficacy of the KNN Algorithm in machine learning for simultaneously improving both quality of service and security, thereby enhancing the overall performance of wireless networks. Additionally, the KNN Algorithm shows promise in identifying and mitigating malware threats.

IV ANALYSIS AND RESULT DISCUSSIONS

After reviewing various papers, it became apparent that routing algorithms can effectively maintain certain quality of service (QoS) parameters such as end-to-end delay, throughput, and packet delivery ratio. However, many location privacy algorithms struggle to strike a balance between energy efficiency and security. This insight was gleaned from the comprehensive review of relevant literature. While numerous studies have succeeded in enhancing the security of source or sink nodes, the introduction of fraudulent packets along the route tends to increase energy consumption. There exists a direct and inverse relationship between network energy consumption and its longevity. As the energy demands of privacy algorithms rise, the lifespan of the network diminishes. This presents a significant challenge in implementing security and privacy algorithms for Internet of Things (IoT) devices with limited energy resources.

To mitigate the need for frequent route recovery, a backup route mechanism can be established within the coverage area of a Roadside Unit (RSU) that has recently passed by. Implementing this approach can reduce the frequency of route rediscovery. The optimal method involves creating a backup route from a serving RSU that has already traversed the area, thus enhancing network reliability and efficiency while conserving energy resources.

Case -1- The scalability of a VANET, irrespective of the number of nodes, is not contingent on its ability to grow limitlessly. VANETs facilitate both Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communications, where nodes acquire information from other nodes or Roadside Units (RSUs), necessitating accurate data transmission. Security requirements differ across VANETs, especially regarding inter-vehicle communication.

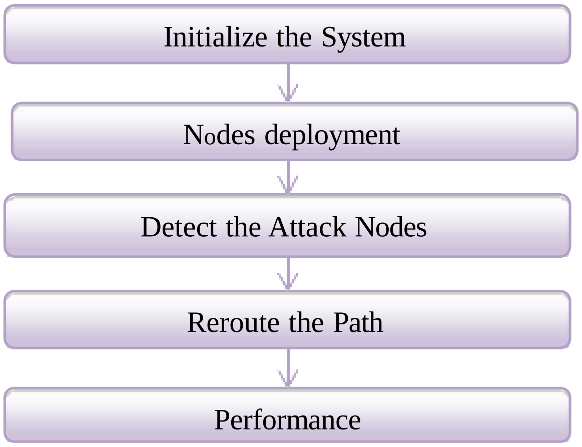
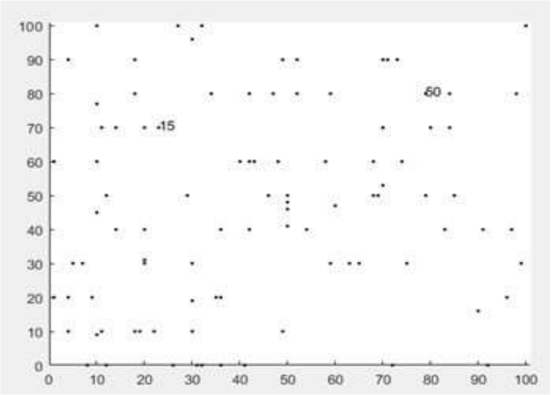


Fig.1 Simulation Flow Diagram

Figures 2 through 7 illustrate various aspects of network initialization, node-to-node data transfer, AODV paths corresponding to changes in vehicle position, delay for different data rates, energy consumption across different data rates, the number of paths found during simulation, and network lifetime across different data rates, respectively. Figure 2 depicts the initialization of the network, while Figure 3 illustrates the process of node-to-node data transfer. Figure 4 showcases AODV paths structured based on changes in vehicle position, and Figure 5 presents the delay experienced for different data rates. Energy consumption across different data rates is depicted in Figure 6, and Figure 7 indicates the number of paths discovered during simulation. Finally, illustrates the network lifetime across various data rates. These figures collectively provide a comprehensive understanding of the network dynamics and performance metrics associated with different data rates, facilitating insights into network efficiency, reliability, and longevity.



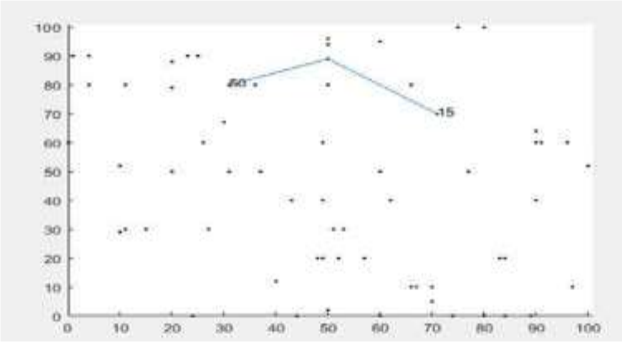


Fig.2 Initialization Network

Fig.3 Node To Node Data Transfer

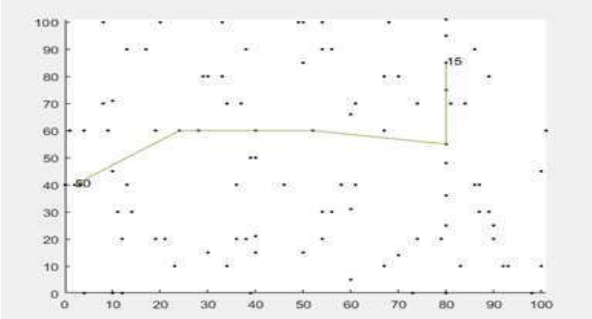


Fig.4 AODV Paths for Each Change in Vehicle Position Into a Structure

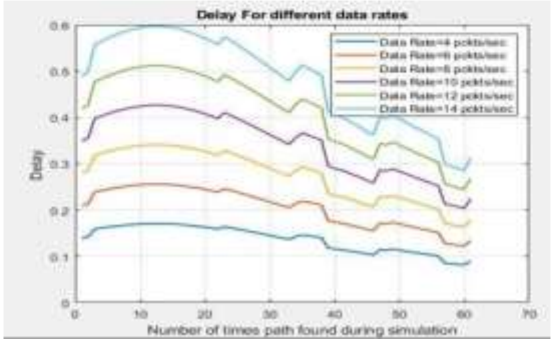


Fig.5 Delay for Different Data Rates

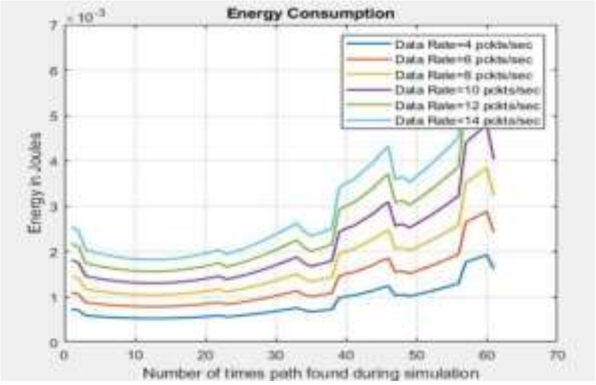


Fig.6 Energy Consumption for Different Data Rates

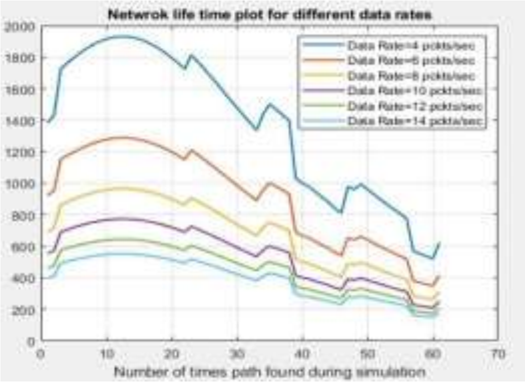


Fig.7 Network Life Time for Different Data Rates

Case-2

The proposed system aims to enhance the quality of service (QoS) and security of wireless networks by employing adaptive measures triggered when performance evaluation metrics fall below predefined thresholds. A key aspect of this enhancement involves utilizing the K Nearest Neighbors (KNN) algorithm for malware detection.

D={(x_1, y_1), (x_2, y_2),(,...,(x_n,y_n)} Eq.1

where x_i represents the feature vectors of the network traffic data and y_i indicates the corresponding class labels (e.g., normal or malicious), the KNN algorithm classifies new data points by computing the distances between the query point x_q and all other points in the dataset. The k nearest neighbors of $x_q$are identified based on these distances. The distance ($x_{i,q}$)between two points xixi and xqxq can be calculated using various distance metrics such as Euclidean distance:

$$(xi,xq)= \sum_{j=}^{m} (x_{ij} - x_{qj})^2 \quad \text{Eq.2}$$

Where m is the number of features in the dataset.

Once the nearest neighbors are identified, the majority class among these neighbors is assigned to the query point x_q this process can be represented by the following mathematical expression:

$$\widehat{y_q} = \operatorname{argmax}_{y_i} \sum_{j=1}^{m} (y_i - y_j)^2 \quad \text{Eq.3}$$

Where $\hat{y}_q$ the predicted class label for the query point is x_q is the indicator function, and k is the number of neighbors.

Fig.8 Flow Diagram for Malware Detection

To ensure the security of sink nodes in wireless sensor networks (WSNs), algorithms are required to conceal their locations and prevent attackers from accessing data packets destined for these nodes. One approach to achieving this is through the use of K-means cluster-based methods,

K-means Clustering for Source and Sink Node Location: The K-means clustering algorithm is applied to track the locations of both source and destination (sink) nodes in WSNs. Fake nodes are introduced alongside real source and sink nodes to enhance security. Data packets can be routed to multiple sink nodes through the clustering mechanism, maintaining the routing path's length while safeguarding data flow. Real packets are sent through the shortest route, reducing latency and enhancing safety time.

Malware Detection Using K-Nearest Neighbors (KNN): Supervised learning methods, such as KNN, can be employed for malware detection in IoT devices by analyzing application behavior. The KNN method categorizes network traffic by assigning it to the category with the most items among its K nearest neighbors.

Data Collection and Pre-processing: Malware dataset derived from open-source tools undergoes pre-processing for efficient training.

Cleaning: Removal of irrelevant entries and assigning integer values to non-relevant data.

Transformation: Conversion of non-integer values to integers for computational suitability.

Reduction: Removal of irrelevant features from the dataset that do not contribute to predicting malware categories.

K-means Clustering:K-means clustering involves minimizing the sum of squared distances between data points and their respective cluster centroids. This can be mathematically represented as: Minimize

$\sum_{i=1}^{k}\sum_{x\in C_i}||x-u_i||$ Eq.4

Where:

Ci represents the ith cluster. u_iRepresents the centroid of cluster Ci

KNN for Malware Detection:

The KNN algorithm computes the distance between a query point and all other points in the dataset, then assigns the query point to the majority class among its K nearest neighbors. this can be expressed as:

$\hat{y}_q = \text{argmax}_{yi} \sum_{j=1}^{k} I(y_i = y_j)$ Eq.5

Where:

$y\hat{y}_q$ is the predicted class label for the query point.

I is the indicator function.

k is the number of neighbors.

Figures 9 through 12 provide a detailed view of the malware detection process using machine learning. Figure 9 illustrates the training phase where both malware and healthy data are used to train the model. Figure 10 shows the observation phase where the characteristics of healthy data and malware data are analyzed in fig 11. depicts the deletion process of identified malware data, ensuring that only non-malicious data remains. Finally, Figure 12 demonstrates the separation of malware from healthy data, highlighting the effectiveness of the model in distinguishing between the two. These figures collectively showcase the steps involved in detecting, analyzing, and eliminating malware to maintain the integrity of the network.

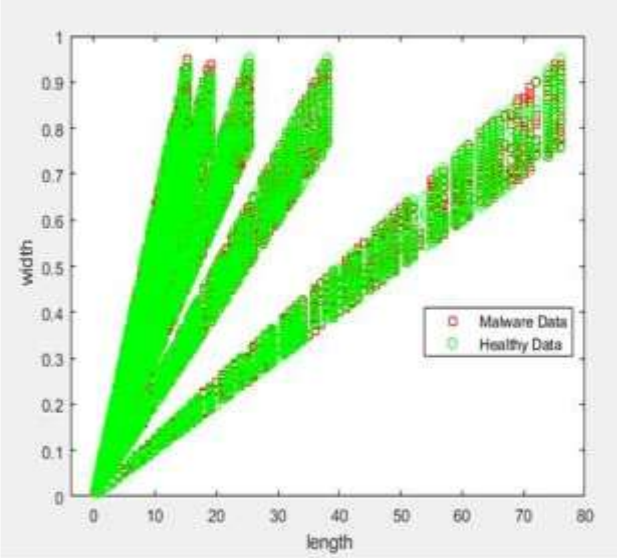


Fig.9 traning of malware and heathy data

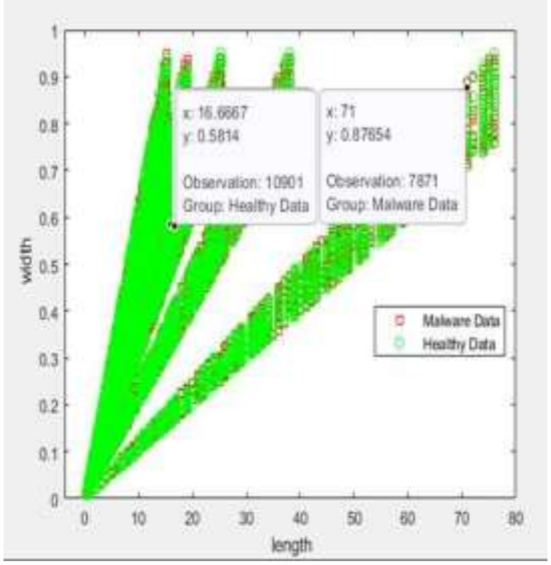


Fig.10 Obesrvation Of Healthy Data And Malware Data

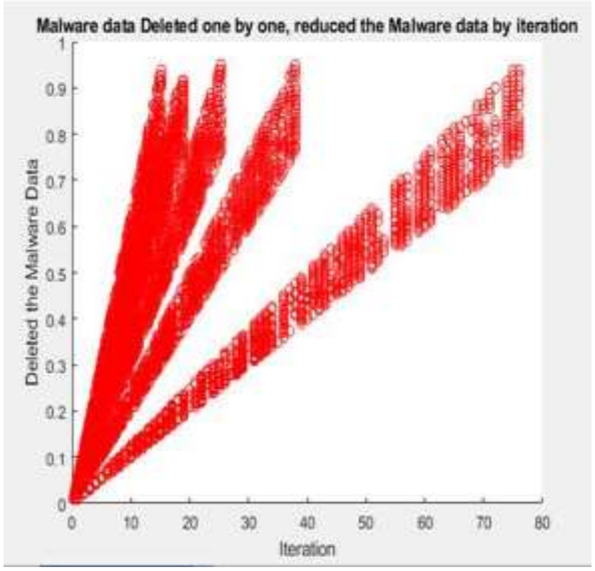


Fig.11 malware data delation

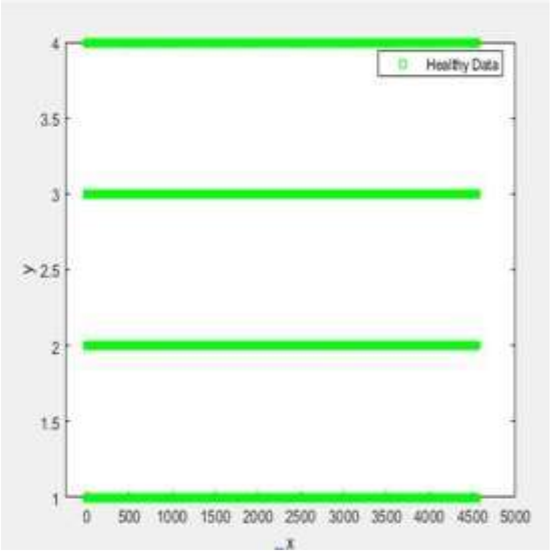


Fig.12 seprate the malware and healthy data

Table 1 IoTQoS Result

Parameters values	**Parameters values**
Avg Delay	206.118855
Avg Energy	84.594327
Avg PDR	274.825140
Avg Through	206.118855

Table 1 presents the results of key IoT quality of service (QoS) parameters. The average delay recorded is 206.118855 milliseconds, indicating the time taken for data packets to travel from the source to the destination. The average energy consumption is 84.594327 units, reflecting the energy efficiency of the network. The average packet delivery ratio (PDR) is 274.825140, demonstrating the reliability of packet transmissions within the network. Lastly, the average throughput is also 206.118855 units, which represents the rate at which data is successfully delivered over the network. These values collectively provide insights into the performance and efficiency of the IoT network under study.

Table 2 Compare the Proposed Results with Existing Work

Parameters	Proposed Work	Existing Work[1]
	AODV-KNN	RL-QRP
Avg Delay	290.65	351
Avg Energy	173.05	196.3
Avg PDR	144.15	136.7
Avg Through	108.11	136.7

In this comparison, the proposed AODV-KNN approach demonstrates several improvements over the existing RL-QRP method. Specifically, the average delay is reduced to 290.65 milliseconds from 351 milliseconds, indicating a faster data transmission time. The average energy consumption is lower at 173.05 units compared to 196.3 units, suggesting better energy efficiency. The average packet delivery ratio (PDR) is slightly higher at 144.15, showing improved reliability in data delivery. However, the average throughput is lower at 108.11 units compared to 136.7 units, indicating a trade-off in data transmission rate for the proposed method. Overall, the proposed AODV-KNN algorithm provides enhanced performance in terms of delay, energy efficiency, and packet delivery reliability.

V CONCLUSION

The analysis of IoT networks reveals the complexity and diversity of the micro-networks involved, including data gathering, processing, and propagation networks. Each of these sub-networks plays a crucial role in the overall functionality of IoT systems, from aggregating sensor data to processing and delivering it to end devices. The integration of machine learning-based security measures, such as the KNN algorithm, has shown promising results in enhancing both the security and quality of service (QoS) in these networks. findings indicate significant improvements in reducing average delay and energy consumption, and slightly enhancing the packet delivery ratio (PDR), although there is a trade-off with a reduction in average throughput compared to existing methods. Given the varied nature and the multi-tier structure of IoT networks, it is essential to implement robust, multi-layered security protocols that can address the unique challenges at each level. This study underscores the potential of advanced machine learning algorithms to provide a balanced approach to improving security and QoS. Future research should aim to further optimize these algorithms, ensuring they can effectively manage the diverse requirements of IoT networks while maintaining high performance and security standards. The ultimate goal is to achieve seamless, secure, and efficient operation of IoT systems across different environments and use cases. Result demonstrates that machine learning-based security measures can significantly influence the quality of service (QoS) in IoT networks. By implementing the K-Nearest Neighbors (KNN) algorithm, we addressed critical aspects of both security and QoS. the proposed AODV-KNN approach effectively reduces average delay and energy consumption while slightly enhancing the packet delivery ratio (PDR). However, it does exhibit a lower average throughput compared to the existing RL-QRP method. This trade-off highlights the need for further optimization to balance throughput with other QoS parameters. research underscores the potential of machine learning algorithms in simultaneously improving network security and performance, paving the way for more robust and efficient IoT systems. Future work will focus on refining these algorithms to achieve even better integration of security and QoS, ensuring the seamless operation of IoT networks under varying conditions.

References

1. Manisha Bhatnagar Dolly Thankachan Identifying the Effects of Security Measures on QoS Variations for IoT Network: Application Perspective Proceedings of the Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV 2021). IEEE Explore Part Number: CFP21ONG-ART; 978-0-7381-1183-4
2. F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi. “Internet of Things security: A survey”. In: Journal of Network and Computer Applications 88 (2017),
pp. 10–28.
3. C. Alcaraz, P. Najera, J. Lopez, and R. Roman. “Wireless sensor networks and the internet of things: Do we need a complete integration?” In: 1st International Workshop on the Security of the Internet of Things (SecIoT’10). 2010.
4. J. Arkko, D. Thaler, and D. McPherson. “IETF RC 7452: architectural considerations in smart object networking”. In: IETF, Fremont, US (2015).
5. M. A. Bhabad and S. T. Bagade. “Internet of things: architecture, security issue and countermeasures”. In: International Journal of Computer Applications 125.14 (2015).

6. G. Breed. “Wireless ad hoc networks: basic concepts”. In: High frequency electronics 1 (2007), pp. 44–47.
7. L.-H. Chang, T.-H. Lee, S.-J. Chen, and C.-Y. Liao. “Energy-efficient oriented routing algorithm in wireless sensor networks”. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics. IEEE. 2013, pp. 3813–3818.
8. M. Chernyshev, Z. Baig, O. Bello, and S. Zeadally. “Internet of things (iot): Research, simulators, and testbeds”. In: IEEE Internet of Things Journal 5.3 (2017), pp. 1637–1647.
9. A. Chhabra, V. Vashishth, A. Khanna, D. K. Sharma, and J. Singh. “An energy efficient routing protocol for wireless internet-of-things sensor networks”. In: arXiv preprint arXiv:1808.01039 (2018)
10. HamidiǦ Alaoui, Z, El Belrhiti El Alaoui, A. FMǦ MAC: A fastǦ mobility adaptive MAC protocol for wireless sensor networks. Trans Emerging Tel Tech. 2020; 31:e3782. https://doi.org/10.1002/ett.3782
11. Kumar, S, Sharma, B, Singh, AK. An efficient algorithm for backbone construction in cognitiveradionetworks.IntJCommunSyst.2020;33:e4345. https://doi.org/10.1002/dac.4345
12. Dadashi Gavaber, Morteza, Rajabzadeh, Amir, “BADEP: Bandwidth and delay efficient application placement in fog-based IoT systems”, Transactions on Emerging Telecommunications Technologies, 2020
13. Gomes, PH, Krishnamachari, B. TAMUǦ RPL: Thompson sampling based multichannel RPL. Trans Emerging Tel Tech. 2020; 31:e3806.
14. H. Liang, S. Yang, L. Li, and J. Gao. “Research on routing optimization of WSNs based on improved LEACH protocol”. In: EURASIP Journal on Wireless Communications and Networking 2019.1 (2019), p. 194.
15. B. Costa, P. F. Pires and F. C. Delicato, "Specifying Functional Requirements and QoS Parameters for IoT Systems," 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech), Orlando, FL, 2017, pp. 407-414.
16. A. A. Simiscuka and G. Muntean, "A Relay and Mobility Scheme for QoS Improvement in IoT Communications," 2018 IEEE International Conference on Communications Workshops (ICC Workshops), Kansas City, MO, 2018, pp. 1-6
17. N. Kouka, T. Guesmi and O. Korbaa, "Performance Evaluation of IEEE 802.15.6 Channel Access Procedure in WBAN," 2018 IEEE 6th International Conference on Future Internet of Things and Cloud (FiCloud), Barcelona, 2018, pp. 162-168
18. Liang, Xuedong & Balasingham, Ilangko. (2007). A QoS-aware Routing Service Framework for Biomedical Sensor Networks. 342 - 345. 10.1109/ISWCS.2007.4392358.
19. S. Baligar, A. Sabade, S. Gurtu and C. Joshi, "A QOS-aware secure personal cloud storage with ubiquitous access and smart home extension," 2015 International Conference on Computer, Communication and Control (IC4), Indore, 2015, pp. 1-5.
20. Z. Mammeri, "Reinforcement Learning Based Routing in Networks: Review and Classification of Approaches," in IEEE Access, vol. 7, pp. 55916-55950, 2019.
21. Djenouri, Djamel & Balasingham, Ilangko. (2009). LOCALMOR: Localized multi-objective routing for wireless sensor networks. 1188- 1192. 10.1109/PIMRC.2009.5449740.
22. Razzaque MA, Hong CS, Lee S. Data-centric multi objective QoS-aware routing protocol for body sensor networks. Sensors (Basel). 2011;11(1):917–937. doi:10.3390/s11010091
23. QPRD: QoS-Aware Peering Routing Protocol for Delay-Sensitive Data in Hospital Body Area Network, Mobile Information Systems, Hindawi Publishing Corporation

24. Bangash JI, Abdullah AH, Anisi MH, Khan AW. A survey of routing protocols in wireless body sensor networks. Sensors (Basel). 2014;14(1):1322–1357. Published 2014 Jan 13. doi:10.3390/s140101322
25. G. Cai, Y. Fang, J. Wen, G. Han and X. Yang, "QoS-Aware BufferAided Relaying Implant WBAN for Healthcare IoT: Opportunities and Challenges," in IEEE Network, vol. 33, no. 4, pp. 96-103, July/August 2019.
26. T. Kaur and D. Kumar, "QoS mechanisms for MAC protocols in wireless sensor networks: a survey," in IET Communications, vol. 13, no. 14, pp. 2045-2062, 27 8 2019.