Navigating Network Traffic: An Exploration of Maze Algorithm Applications in Machine Learning

Shayma Ismail Ali[1], Yuvaraj Duraisamy[2], Bilal Hikmat Rasheed[3], Shakir Mahoomed Abas[4], Toreen Dilshad Masood[5]
[1,2,3,4]Department of Computer Science, Cihan University-Duhok, Duhok, Iraq.
[5]Department of Computer Science, University of Duhok, Duhok, Iraq.
[1,5]PG Student, Akre University for Applied Sciences, Akre, KRG, Iraq.

Article Info

Article history:

Received Oct 9, 2023
Revised Dec 20, 2023
Accepted Feb 12, 2024

Keywords:

Network traffic
Maze algorithm
Machine learning
Smart devices
Real-Time Analysis

ABSTRACT

To Determine network performance and highlight the need for integrated and intelligent solutions in response to the growing amount of data generated by smart devices. It launches a link between web browsing and queuing, emphasizing the importance of factors such as time and availability. The proposed method is evaluating some studies on machine learning techniques used for network traffic forecasting, Internet tools, and techniques for analyzing traffic flows. Network traffic is the data flow across a computer network, emphasizing the need for scalable and intelligent solutions due to the rise in data generation from smart devices. Additionally, it draws an analogy between navigating network paths and solving mazes in everyday life based on factors like time, effort, and convenience. Navigating paths resembles solving mazes, as individuals choose routes from origin to destination based on factors like time, effort, and convenience, akin to maze-solving techniques. Algorithms use real-time analytics, machine learning, and historical data to improve network efficiency. Finally, the maze algorithm is measured the network traffic optimization. It is very useful in machine learning, real-time analysis, and historical data to improve network performance.

Corresponding Author:

Shayma Ismail Ali,
Department of Computer Science, Cihan University-Duhok, Duhok, Iraq.
PG Student, Akre University for Applied Sciences, Akre, KRG, Iraq.
Email: shayma.ali@duhokcihan.edu.krd

1. INTRODUCTION

Network traffic is the movement of data over a computer network at any given moment. It denotes the volume of data being transmitted among devices that are linked to the network. The data can manifest in diverse formats such as files, emails, web pages, multimedia material, and other forms, which can be analyzed by network-based traffic analysis. To effectively monitor and understand traffic, there is a need for innovative networking solutions in response to the recent surge in data generation and the diverse development of smart devices. For these solutions to effectively handle large amounts of data, they must possess the ability to scale and exhibit intelligent behavior.

Humans frequently face mazes in their daily lives as they navigate from one location to another using designated routes, whether by walking or by transportation. There are numerous potential paths we can select from our starting point to a certain endpoint, which are influenced by various factors such as time, exertion, and convenience. Selecting the optimal path to travel from our starting point to our desired location might be compared to the strategies employed to navigate through mazes [1].

The time complexity and spatial complexity of each algorithm, including Prim's algorithm, Kruskal's algorithm, DFS (Depth First Search), Eller's method, Wilson's Algorithm, Hunt and Kill algorithm, and Wilson's Algorithm, will be utilized to evaluate their performance in maze generation. Three maze-solving agents have been developed to assess the efficacy of the algorithms. The examination seeks to comprehend the technical and engineering facets of each algorithm, juxtapose the time and space complexity of each method, and evaluate their effectiveness when employed in demanding maze-generating scenarios. The plan entails implementing and evaluating each algorithm on mazes of different dimensions and complexities while measuring the time it takes for each method to construct a maze [2].

Machine learning methods can be categorized into three distinct classes: supervised learning, unsupervised learning, and semi-supervised learning. These approaches are employed for tasks like as classification, prediction, and clustering. The paper showcases instances of classification and clustering methods employed in the development of network attack detection. Machine learning techniques are used to identify probable anomalies or intrusions in network computer systems. Machine learning is crucial for identifying probable incursion patterns and extracting meaningful and relevant patterns from current datasets to make intelligent decisions or predictions. The text outlines the several phases of machine learning algorithms, including data cleansing, classification, entity recognition, subjectivity classification, and feature selection. These processes are employed to identify network security issues and prevent network assaults [3, 4].

Researchers have thoroughly investigated instances of traffic breakdown, developing probabilistic models that indicate a correlation between traffic flow and the likelihood of breakdown incidents. The conventional emphasis on a solitary variable has moved due to the availability of more extensive traffic data, uncovering that breakdown incidents are influenced by several aspects beyond only the velocity of traffic flow. Machine learning has made significant progress in accurately predicting outcomes by utilizing neural networks that take into account multiple variables, such as temporal and geographical connections. Significant study gaps now exist in comprehending the influence of traffic flow from ramps on breakdown incidents through the use of neural networks [5].

The rest of the paper is structured as follows. Section 2 Literature review and Section 3 is the Background: network traffic, ML, and Maze algorithm. The Evolution of Each Paper in Result and Discussion in Section 4. Conclusion in Section 5, and References in Section 6.

2. LITERATURE REVIEW

2.1. Network Traffic

[6] Employed machine learning algorithms to forecast traffic in the LTE network and conducted a comparative analysis of three distinct algorithms - Bagging, Random Forest, and SVM. Utilized a publicly available cellular traffic dataset to evaluate the performance of each technique and presented a comprehensive assessment of their respective quality metrics, highlighting the strengths and weaknesses of each approach. Comparing these strategies can aid in picking the optimal algorithm for a certain task.

[7] Emphasized that machine learning and deep learning algorithms are utilized in network traffic categorization to effectively monitor and identify the underlying applications, protocols, or services that are responsible for QoS, content filtering, lawful interception, and other specific objectives.

[8] Used online machine-learning methods to analyze network traffic streams in the field of computer networking. It discusses the significance of traffic data analytics, specifically focusing on OL-based network traffic stream analysis and the effectiveness of OL techniques in network traffic classification. It also touches upon potential future developments in the field of analyzing traffic data streams.

[9] Demonstrated the functionality of the NetFlow analyzer in assessing different components of a network, such as filters, mixers, frequency-sensitive networks, and other devices that operate on radio frequency. The study also addresses the design model of NetFlow Analyzer and its operational frequencies, highlighting its capability to analyze stability in measurements of open loops, audio components, and ultrasonics.

[10] Emphasized the meticulous curation of information for training wireless intrusion detection systems and the distinct obstacles presented by IoT networks. The transmission of data across wireless networks carries important information, with the most significant characteristics mostly found in the physical and data link layers. Furthermore, the obstacles and tactics involved in the implementation of intrusion detection systems (IDS) in wireless networks. The primary goal of the Internet of Things (IoT) is to facilitate the seamless transmission of data between two systems or devices without the need for human intervention.

[11] Described the technique called Network Traffic Monitoring and Analysis (NTMA). The NTMA includes a crucial subject known as Network Traffic Prediction (NTP), which focuses primarily on

forecasting network load and its associated behavior. There are generally two approaches to implementing NTP techniques: statistical-based and Machine Learning (ML)-based procedures.

[12] Found that improving the effectiveness of traffic signal control (TSC) leads to enhanced urban mobility and an elevated level of living. Due to the recent notable increase in the use of reinforcement learning (RL) in different network-level time series classification (TSC) domains, we conducted a comprehensive and replicable literature review to examine all previous studies that employed RL in the network-level TSC domain, referred to as RL in NTSC or RL-NTSC for brevity.

[13] Demonstrated the deployment of a Cyborg Intelligence-driven ensemble intrusion detection system to enhance the security of IoT-enabled networks utilized for network traffic in smart cities. The ensemble intrusion method, based on the Cyborg Intelligence architecture, utilizes various intrusion detection algorithms to effectively identify possible security risks and attacks on IoT-enabled networks. The researchers noted that the proposed framework was successfully implemented. High precision, a high rate of detection, and a low rate of false positives.

2.2. Maze Algorithm

[14] Explored the issue of the "Micro Mouse," a robot with the ability to navigate and solve a maze. They addressed a crucial aspect of the robot, known as the "Decision-making Algorithm" or, in simpler terms, "Robot Intelligence". The authors initially employed rudimentary wall follower logic to navigate the maze, then subsequently refined the algorithm to efficiently and intelligently solve the maze in the least possible duration. The method has been developed to a highly advanced degree, comparable to the Flood-Fill algorithm.

[15] Utilized the Maze algorithm to address the task of converting ladder diagrams into instruction lists. Employed the Maze algorithm to scan the ladder diagram, resulting in the creation of a doubly linked list with a solitary node. Next, determine whether the connections between pairs of nodes are serial or parallel based on the information provided by a single node. The binary trees were used to hold the serial or parallel connections of all nodes. Ultimately, they conducted a thorough examination of the binary trees to produce lists of instructions. It is worth mentioning that the maze algorithm exhibited a low level of complexity and was able to accurately determine the related relationship between nodes in the ladder diagram.

[16] Applied several techniques to create mazes, including graph-based approaches, recursive division techniques, and alternative algorithms like Eller's algorithm and Wilson's algorithm that do not rely on graph theory. Furthermore, an alternative algorithm is employed for the automated and stochastic creation of mazes, although it does not explicitly indicate the particular technique utilized.

[17] Explored the significance of independent motion in robotics and the use of maze-solving algorithms in the process of making decisions. Examined various maze-solving algorithms and the latest advancements in autonomous maze-solving robotic systems. Explored several maze-solving algorithms and classified them into two categories based on the specific maze environment they are designed for. Furthermore, they examine the recent implementation of maze-solving autonomous systems and provide a meticulous evaluation to aid researchers and developers in choosing the most appropriate maze-solving algorithm, taking into account elements such as algorithm complexity, memory requirements, deployment expenses, and intricacy, among other considerations.

[18] Outlined the design and implementation of a robot capable of navigating mazes by following a predetermined path. Additionally, I developed an algorithm to navigate, acquire knowledge, and solve the maze by employing infrared sensors. Furthermore, I utilized a Bluetooth module to transmit real-time data and visualize it in MATLAB while the robot was in action. Enhancing the robot's performance and speed was proposed through the incorporation of superior hardware. Additionally, it is recommended to include an optimization technique to enhance the solving of looping mazes. The suggestion was to enhance the storage technique for the robot's judgments, optimize the path during operation to improve memory consumption, and employ rubber wheels with a greater grip coefficient to minimize sliding on specific surfaces that may affect maze mapping. The user's text is empty.

[19] Studied an issue of maze generation by utilizing a specific set of parameters that directly impact the level of difficulty in the resulting mazes. The many maze creation techniques, including Prim's algorithm, recursive backtracking algorithm, and Kruskal's algorithm, and their potential modifications to produce mazes of varying levels of difficulty. Additionally, it examines current techniques for evaluating labyrinth complexity and suggests a methodology that considers the number of inconsequential barriers as well as the intricacy of the solution path and branches inside the maze.

[20] Created a maze framework for a Python-based 2D role-playing game named "Maze Adventure." Explores the process of building the game, starting with the maze-generating algorithm to the implementation of sprites and the technique of integrating the created maps with character collisions. In addition, the study presents specific information regarding gaming aspects such as experience and level

computation, atmospheric fog, and character animation. The authors' conclusion is that completing Maze Adventure demonstrates the Python language's ability to utilize one of the latest forms of media: Games.

[21] Used the Depth-First Search (DFS), Breadth-First Search (BFS), and A* search algorithms to successfully solve the Maze Transversal issue. Utilizing unsuitable methods can modify the duration of the computer's computation for identifying the most efficient route, resulting in increased waiting time for the agent/player during the execution process. The Maze transversal challenge involved comparing pathfinding algorithms such as DFS, BFS, and A*. The comparative analysis was conducted by executing several algorithms in three mazes of identical dimensions but with distinct obstacles, while closely monitoring the duration of execution and the length of the path. The findings indicate that the A* search algorithm is the optimal choice for solving the Maze Transversal problem, as it efficiently determines the shortest path to the goal in the least amount of time and distance.

3. BACKGROUND

3.1. Network Traffic Analysis

Network traffic refers to the volume of data being transmitted via a computer network at a specific moment. Network traffic, known as data traffic, is fragmented into data packets and transmitted across a network before being reconstructed by the recipient device or computer.

The significance of network traffic analysis is growing in contemporary society as it enables the monitoring of network traffic. Previously, administrators were only required to oversee a small number of network devices or a restricted quantity of PCs. The network capacity was comparatively limited, frequently below 100 Mbps. Presently, administrators are required to oversee high-speed cable networks that surpass 1 Gbps, in addition to intricate networks such as ATM and wireless networks. To proficiently oversee the network, promptly address network problems, prevent network outages, and guarantee network safety, administrators necessitate contemporary network traffic analysis technologies. These tools facilitate the analysis and comprehension of network traffic patterns, the detection of anomalies, and the maintenance of optimal network performance [22].

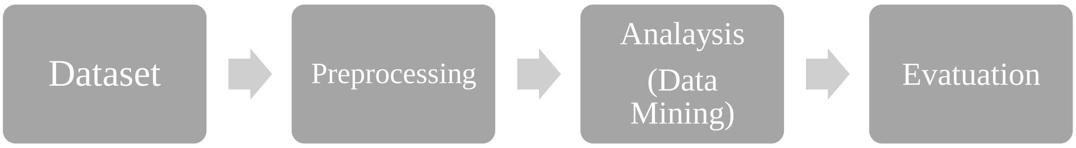

Figure 1. Generic structure of network traffic analysis [20]

Data sets

Testing and assessing are essential components in the examination of network traffic. To evaluate the effectiveness of different research efforts, it is recommended to use a consistent dataset. In recent years, a multitude of standardized data sets have been employed for this objective. Here, we offer a collection of prominent data sets that researchers utilize for the analysis of network traffic [23].

Preprocessing

Preprocessing is an essential stage employed to transform real-world data into a coherent and understandable format. Real-world data is often incomplete and contains noise in certain patterns. Essentially, the data we intend to analyze using data mining techniques is mostly incomplete and inconsistent, as it contains errors and outlier values. Preprocessing strategies are essential to improve the data quality before implementing data mining algorithms. Consequently, this aids to enhancing the precision and effectiveness in the subsequent data mining task. The need of preprocessing techniques is especially apparent in network traffic analysis, considering the many formats and dimensions of network traffic patterns. In the next sections, we provide comprehensive explanations of the methods utilized in network traffic analysis [24].

Analysis (Data Mining)

Data mining is a technique that reveals valuable patterns from extensive amounts of data. The article explores various data mining approaches, algorithms, and specific organizations that have adopted data mining technologies to improve their businesses and achieve outstanding results [25].

Data Mining Algorithms and Techniques

Knowledge discovery from databases involves the utilization of a range of methods and techniques, such as Classification, Clustering, Regression, Artificial Intelligence, Neural Networks, Association Rules, Decision Trees, Genetic methods, and the Nearest Neighbor method [26].

Data Mining Applications

Data mining is an emerging technology that is still in the process of reaching its full potential. Nevertheless, numerous sectors are already employing it on a regular basis. These corporations encompass several sectors such as retail, healthcare, finance, and insurance. Several firms are integrating data mining with tools such as statistics, pattern recognition, and other essential techniques. Data mining enables the discovery of intricate patterns and correlations that would otherwise be arduous to uncover. This technology is widely embraced by numerous firms due to its capacity to provide valuable insights into client behavior and facilitate informed marketing strategies. This document provides a comprehensive analysis of business challenges and the corresponding remedies identified through the utilization of data mining technology [25] [26].

Evaluation metrics

[27] Various metrics are employed in data mining techniques to examine and analyze the data. The performance of the classifier for various data sets is evaluated using measures such as detection rate, false positive rate, accuracy, and time cost. Multiple metrics are available to quantify the accuracy of predictions. The measurements employed involve the utilization of a confusion matrix. Each metric is defined as follows:

Table 1. Metric Table

A	True Negatives (TN)	Total numbers of normal packets correctly classified.
B	True Positives (TP)	Total numbers of malicious packets correctly classified.
C	False Negatives (FN)	False Negatives are total numbers of malicious packets incorrectly classified as normal packets.
D	False Positives (FP)	False positive is the Total number of normal packets incorrectly classified as malicious packets.

A matrix containing correct and incorrect predictions in the form of TPs, FPs, FNs, and TNs is known as a confusion matrix.

Accuracy of Model

For any classification model, model accuracy is given by the total number of correct classifications:

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

Sensitivity of a model

The sensitivity of a model measures the proportion of TP examples or positive cases that were correctly classified. It is measured as:

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

Specificity

Specificity is also another good measure to indicate a good balance of a model being excessively conservative or excessively aggressive. The specificity of a model measures the proportion of negative examples that have been correctly classified.

$$\text{Specificity} = \frac{TN}{TN+FP}$$

Precision and Recall

Here are two other performance measures of a supervised learning model that are similar to sensitivity and specificity. These are precision and recall. While precision gives the proportion of positive predictions that are truly positive, recall gives the proportion of TP cases overall actually positive cases.

$$\text{Precision} = \frac{TP}{TP+FP} \qquad \text{Recall} = \frac{TP}{TP+FN}$$

Network traffic prediction

Network traffic prediction is a significant concern that has garnered considerable attention in the computer network field. Network traffic prediction is a common problem that is valuable for network monitoring, network security, congestion avoidance, and network performance enhancement. Researchers employ many methodologies to forecast network traffic. The methodologies have been classified into four main categories: linear time series model, nonlinear time series model, hybrid model, and decomposition model. Figure 2 provides a comprehensive overview of the four categories that classify different methodologies used for network traffic prediction [28].

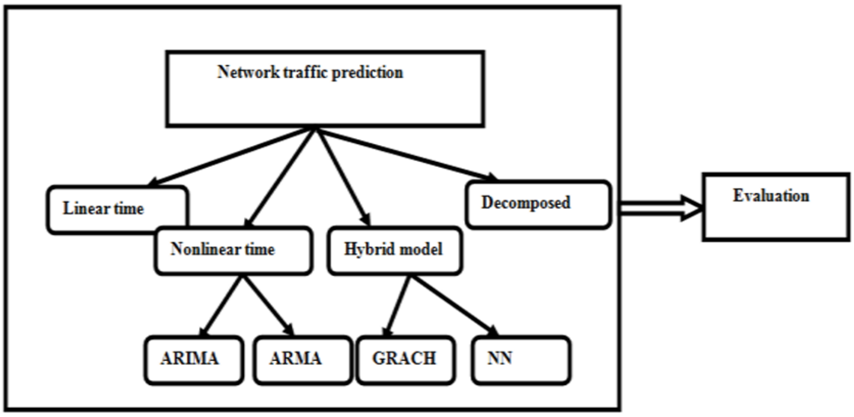

Figure 2. Network traffic prediction technique [28]

Linear time Series technique

Linear Time Series approaches refer to the covariance structures used in analyzing time series data. Two widely recognized sub-categories of linear time series models are Auto-Regressive (AR) models and Moving Average (MA) models. These two models can be merged to create auto-regressive moving average models. Linear time series is a conventional method used for predicting network traffic. We introduce a range of linear time series methodologies that academics are employing to forecast network traffic [22].

- Autoregressive Moving Average (ARMA)
- Autoregressive Integrated Moving Average Model (ARIMA)

Nonlinear time series techniques

Nonlinear time series are produced by non-linear dynamic equations. They display characteristics that cannot be replicated by linear processes, including as variations in time, cycles that are not symmetrical, and structures involving higher moments, thresholds, and discontinuities. This model employs many techniques, including neural networks and fuzzy logic, to forecast network traffic. We provide a range of nonlinear time series methods utilized by researchers for the prediction of network traffic [29]

- GARCH technique
- Neural Network techniques

3.2. Maze Generation

[30] Mazes have strong connections to ancient labyrinths, have historically carried spiritual importance, and subsequently transformed into forms of entertainment. In contemporary society, scientists, specifically mathematicians, are captivated by the enigmatic nature of mazes. In addition to providing pleasure, mazes are utilized in psychology studies to evaluate spatial awareness and intelligence in both people and animals. Moreover, they have practical uses in the field of physics, particularly in the examination of crystal structures. Although there are many algorithms available for producing mazes, only a small number of them have been assessed in terms of their difficulty and complexity levels.

The data structure employed to depict the maze. Initially, we consider the grid graph consisting of squares. At first, all the connections between edges are represented as walls. The algorithms transform designated walls into corridors. Once the algorithms have completed their task, the subgraph consisting of tunnels forms a maze with a tree-like topology. Typically, mazes may have loops, overpasses, and other elements, but our focus was on straightforward designs.

Creating labyrinthine structures allow us to elucidate the data structure employed to depict the maze. Initially, we consider the square grid graph. At first, all the edges symbolize barriers. The algorithms transform designated walls into passageways. Once the algorithms have completed their task, the subgraph composed of passages forms a maze with a tree-like topology. Typically, mazes may include loops, overpasses, and other elements, however, our focus was on simpler variations. Each labyrinth possesses distinct characteristics that we employ for study, including its dimensions, quantity of intersections, number of branches, average length of branches, and length of dead ends (branches that do not further divide). Recursive Backtracking (RB)

- Aldous-Broder Algorithm (AB)
- Wilson's Algorithm (W)

- Prim's Algorithm (P)
- Hunt and Kill (HK)
- Kruskal's Algorithm (K)

4. RESULT AND DISCUSSION

Various aspects of some authors in different mensurate for network traffic are displayed in

Table 2. Summaries of Review Papers in Network Traffic applications of machine learning and data analytics in network traffic analysis. Researchers explored algorithms such as Bagging, Random Forest, SVM, and deep learning to predict and classify traffic in LTE networks, ensuring the selection of suitable models for specific tasks.

Online machine-learning techniques were proposed for real-time traffic stream analysis, emphasizing the significance of analytics in understanding network behaviors.

The use of NetFlow analyzers showcased capabilities in measuring various network aspects. Additionally, studies addressed challenges in selecting datasets for wireless intrusion detection systems, reinforcement learning's impact on traffic signal control for urban transportation, and the implementation of Cyborg Intelligence-based ensemble intrusion detection systems to enhance the security of IoT-enabled networks in smart cities.

These investigations collectively contribute to advancing the understanding and practical applications of network traffic analysis.

Table 2. Summaries of Review Papers in Network Traffic

Author(s)	Year	Main Focus	Algorithm(s) Used
Stepanov et al	2020	Predicting traffic in LTE network using ML algorithms	Bagging, Random Forest, SVM
Arzo et al	2022	Network traffic classification using ML and DL algorithms	Machine Learning, Deep Learning
Shahraki et al	2022	Online machine-learning techniques for network traffic analysis	Online machine-learning techniques for traffic classification
Islam et al	2023	NetFlow analyzer for measuring various parts of a network	NetFlow Analyzer
Ponnusamy et al	2022	Selecting datasets for training wireless intrusion detection systems	IoT networks, Wireless intrusion detection systems
Lohrasbinasab et al	2021	Network Traffic Monitoring and Analysis (NTMA)	Statistical-based and ML-based Network Traffic Prediction
Noaeen et al	2022	Review of literature on reinforcement learning in traffic signal control	Reinforcement Learning in Network-Level Traffic Signal Control
Onyema et al	2022	Cyborg Intelligence-based ensemble intrusion detection system for IoT-enabled networks in smart cities	Cyborg Intelligence framework
Lohrasbinasab et al	2021	Network Traffic Monitoring and Analysis (NTMA)	Statistical-based and ML-based Network Traffic Prediction

Various aspects of some authors used maze algorithms for different mensurates are displayed in

Table 3. Summaries of Review Papers in Maze Algorithm. Beginning with the evolution of maze-solving algorithms for robots, ranging from basic wall following to sophisticated techniques like the Flood-Fill algorithm, the review explores the diverse applications of maze algorithms. These applications include translating ladder diagrams into instruction lists with low complexity, generating mazes through graph-based and non-graph theory-based methods, analyzing the role of maze-solving algorithms in autonomous robotic movement, constructing line-follower robots with optimization strategies, parameterizing maze generation algorithms for difficulty levels, developing maze structures for a 2D role-playing game, and comparing pathfinding algorithms for the Maze Transversal problem. The collective findings emphasize algorithmic

evolution, versatility, and practical implementations across robotics, game development, and problem-solving scenarios.

Table 3. Summaries of Review Papers in Maze Algorithm

Author(s)	Year	Problem Addressed	Algorithm(s) Used	Key Points
Mishra and Bande	2008	Maze-solving robot decision-making algorithm	Wall follower logic to Flood-Fill algorithm	Evolution of the Micro Mouse robot algorithm from basic wall following to advanced Flood-Fill algorithm for efficient maze-solving.
Tana and Ju	2011	Translating ladder diagrams into instruction lists using the Maze algorithm	Maze algorithm with doubly linked lists	Application of Maze algorithm to convert ladder diagrams into instruction lists, emphasizing low complexity and correct identification of connections.
Dubey and Sarita	2016	Maze generation methods	Graph-based, recursive division, Eller's, Wilson's, etc.	Exploration of various maze generation methods, including graph-based and non-graph theory-based algorithms, with a focus on algorithm diversity.
Alamri	2020	Role of maze-solving algorithms in autonomous movement decision-making	Various maze-solving algorithms	Categorization of maze-solving algorithms based on environmental factors, aiding researchers in algorithm selection for robotic systems.
Harja and Nascu	2020	Line-follower maze-solving robot algorithm	Infrared sensors, Bluetooth module, optimization	Construction, programming, and optimization of a line-follower robot for maze-solving, addressing hardware improvement and path optimization.
Peachey	2022	Parameterized Maze Generation Algorithm	Prim’s, recursive backtracking, Kruskal's, etc.	Examination of maze generation algorithms, modification for difficulty levels, and proposal of an approach considering non-significant walls and path complexity.
Yang and He	2022	Maze structure generation for a 2D role-playing game in Python (Maze Adventure)	Not specified	Overview of constructing a game from maze generation to character elements using Python, demonstrating the language's capability in game development.
Princess Chinemerem	2022	Comparison of DFS, BFS, and A* search algorithms for Maze Transversal problem	DFS, BFS, A*	Evaluation of pathfinding algorithms in the Maze Transversal problem, recommending A* for shortest path determination based on execution time and length.

This paper suggests that the maze-solving algorithm is tailored for network-based traffic optimization. This algorithm intelligently utilizes a combination of Dijkstra's and genetic algorithms, dynamically analyzing network conditions to find the most efficient paths for data transmission. Real-time traffic monitoring allows for adaptive rerouting, reducing congestion and enhancing overall network efficiency. Leveraging machine learning, the algorithm continuously refines its predictions, improving decision-making in response to evolving traffic patterns. By prioritizing routes based on historical data, it ensures an agile and proactive approach to managing network traffic, resulting in reduced latency and optimized data flow.

5. CONCLUSION

Several studies have investigated for network traffic prediction and classification and compared methods such as binary, random forest, SVM, and deep learning. Furthermore, through the study of online machine learning methods for network traffic flow analysis, we highlight the importance of traffic data analysis and propose a strategy for wireless intrusion detection systems in IoT networks. Implementation of an integrated detection system based on NTMA (Monitoring and Analysis) and cyborg intelligence to improve IoT-enabled network security in smart cities. These alternatives offer opportunities to improve network efficiency, QoS, and security.

We have conducted an investigation using DFS, BFS, and A* search algorithms to solve the transversal maze problem. This proposed method shows that the A* search algorithm outperforms the DFS and BFS and provides the shortest path to the goal with the best and longest time, proving its accuracy for the cross-maze problem. This adaptive approach analyzes the network processes in real time to prioritize routes based on historical data, enabling efficient data transfer. The algorithm continuously improves predictions, reduce congestion, reduces occupancy, and optimizes network efficiency.

REFERENCES

[1] H. Patel, “Maze Navigation Algorithm Techniques,” 2016, doi: 10.13140/RG.2.2.34628.96648.

[2] D. Mane, R. Harne, T. Pol, R. Asthagi, S. Shine, and B. Zope, “International Journal of INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING An Extensive Comparative Analysis on Different Maze Generation Algorithms.” [Online]. Available: www.ijisae.org

[3] W. A. Ali, M. Bendechache, M. FadhelAljunaid, and P. Sandhya, “A Review of Current Machine Learning Approaches for Anomaly Detection in Network Traffic.”

[4] F. Pacheco et al., “Towards the Deployment of Machine Learning Solutions in Network Traffic Classification: A Systematic Survey,” Communications Surveys and Tutorials, vol. 21, no. 2, pp. 1988–2014, 2018, doi: 10.1109/COMST.2018.2883147ï.

[5] A. Azab, M. Khasawneh, S. Alrabaee, K. K. R. Choo, and M. Sarsour, “Network traffic classification: Techniques, datasets, and challenges,” Digital Communications and Networks. KeAi Communications Co., 2023. doi: 10.1016/j.dcan.2022.09.009.

[6] N. Stepanov, D. Alekseeva, A. Ometov, and E. S. Lohan, “Applying Machine Learning to LTE Traffic Prediction: Comparison of Bagging, Random Forest, and SVM,” in International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, IEEE Computer Society, Oct. 2020, pp. 119–123. doi: 10.1109/ICUMT51630.2020.9222418.

[7] S. T. Arzo, Z. Akhavan, M. Esmaeili, M. Devetsikiotis, and F. Granelli, “Multi-Agent-Based Traffic Prediction and Traffic Classification for Autonomic Network Management Systems for Future Networks,” Future Internet, vol. 14, no. 8, Aug. 2022, doi: 10.3390/fi14080230.

[8] A. Shahraki, M. Abbasi, A. Taherkordi, and A. D. Jurcut, “A comparative study on online machine learning techniques for network traffic streams analysis,” Computer Networks, vol. 207. Elsevier B.V., Apr. 22, 2022. doi: 10.1016/j.comnet.2022.108836.

[9] R. Islam et al., “Design and Analysis of a Network Traffic Analysis Tool: NetFlow Analyzer,” International Journal of Communications, Network and System Sciences, vol. 16, no. 02, pp. 21–29, 2023, doi: 10.4236/ijcns.2023.162002.

[10] V. Ponnusamy, A. Yichiet, N. Z. Jhanjhi, M. Humayun, and M. F. Almufareh, “IoT Wireless Intrusion Detection and Network Traffic Analysis,” Computer Systems Science and Engineering, vol. 40, no. 3, pp. 865–879, Sep. 2021, doi: 10.32604/CSSE.2022.018801.

[11] Lohrasbinasab, A. Shahraki, A. Taherkordi, and A. Delia Jurcut, “From statistical- to machine learning-based network traffic prediction,” Transactions on Emerging Telecommunications Technologies, vol. 33, no. 4, Apr. 2022, doi: 10.1002/ett.4394.

[12] M. Noaeen et al., “Reinforcement learning in urban network traffic signal control: A systematic literature review,” Expert Systems with Applications, vol. 199. Elsevier Ltd, Aug. 01, 2022. doi: 10.1016/j.eswa.2022.116830.

[13] E. M. Onyema, S. Dalal, C. A. T. Romero, B. Seth, P. Young, and M. A. Wajid, “Design of Intrusion Detection System based on Cyborg intelligence for security of Cloud Network Traffic of Smart Cities,” Journal of Cloud Computing, vol. 11, no. 1, Dec. 2022, doi: 10.1186/s13677-022-00305-6.

[14] S. Mishra and P. Bande, “Maze solving Algorithms for micro mouse,” in SITIS 2008 - Proceedings of the 4th International Conference on Signal Image Technology and Internet Based Systems, 2008, pp. 86–93. doi: 10.1109/SITIS.2008.104.

[15] A. Tan and C. Ju, “The application of maze algorithm in translating ladder diagram into instruction lists of programmable logical controller,” in Procedia Engineering, 2011, pp. 264–268. doi: 10.1016/j.proeng.2011.08.052.

[16] P. Dubey and K. Sarita, “Maze generation & solver,” International Journal of Scientific and Technical Advancements, vol. 2, no. 4, pp. 139–142, 2016, [Online]. Available: http://www.logicmazes.com

[17] S. Alamri, S. Alshehri, W. Alshehri, H. Alamri, A. Alaklabi, and T. Alhmiedat, "Autonomous Maze Solving Robotics: Algorithms and Systems," International Journal of Mechanical Engineering and Robotics Research, vol. 10, no. 12, pp. 668–675, Dec. 2021, doi: 10.18178/ijmerr.10.12.668-675.

[18] R. Covaci, G. Harja, and I. Nascu, "Autonomous Maze Solving Robot ∗," in 2020 22nd IEEE International Conference on Automation, Quality and Testing, Robotics - THETA, AQTR 2020 - Proceedings, Institute of Electrical and Electronics Engineers Inc., May 2020. doi: 10.1109/AQTR49680.2020.9129943.

[19] L. Peachey, "Parameterized Maze Generation Algorithm for Specific Difficulty Maze Generation," 2022.

[20] R. yang and C. He, "Maze Adventure: An Application of Maze Algorithm in Role-playing Game Development by Python," European Alliance for Innovation n.o., Oct. 2022. doi: 10.4108/eai.17-6-2022.2322876.

[21] P. Chinemerem, "A COMPREHENSIVE AND COMPARATIVE STUDY OF DFS, BFS, AND A* SEARCH ALGORITHMS IN A SOLVING THE MAZE TRANSVERSAL PROBLEM ☆," 2022.

[22] M. R. Joshi and T. H. Hadi, "A Review of Network Traffic Analysis and Prediction Techniques."

[23] M. Ghurab, G. Gaphari, F. Alshami, R. Alshamy, and S. Othman, "A Detailed Analysis of Benchmark Datasets for Network Intrusion Detection System," Asian Journal of Research in Computer Science, pp. 14–33, Apr. 2021, doi: 10.9734/ajrcos/2021/v7i430185.

[24] M. Andreoni Lopez, D. M. F Mattos, O. M. Carlos B Duarte, and G. Pujolle, "A Fast Unsupervised Preprocessing Method for Network Monitoring."

[25] B. M. Ramageri, "DATA MINING TECHNIQUES AND APPLICATIONS."

[26] V. S. Souza and D. A. Lima, "Identifying Risk Factors for Heart Failure: A Case Study Employing Data Mining Algorithms," Journal of Data Science and Intelligent Systems, Sep. 2023, doi: 10.47852/bonviewJDSIS32021386.

[27] H. Dalianis, "Evaluation Metrics and Evaluation," in Clinical Text Mining, Springer International Publishing, 2018, pp. 45–53. doi: 10.1007/978-3-319-78503-5_6.

[28] M. R. Joshi and T. H. Hadi, "A Review of Network Traffic Analysis and Prediction Techniques."

[29] A. Raza, S. M. A. Burney, and S. A. Raza, "Monte Carlo Simulation and Prediction of internet load using conditional mean and conditional variance model," 2007. [Online]. Available: https://www.researchgate.net/publication/228671976

[30] P. Gabrovšek, "Analysis of Maze Generating Algorithms."