IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.53

Volume 4, Issue 3, June 2024

Machine Learning Techniques to Optimize CPU Scheduling in Real-Time Systems: A Comprehensive Review and Analysis

C. Nagesh[1], G. Sudha Gowd[2], Naidu Kiran Kumar[3], G. Pradeep Reddy[4]

Assistant Professor, Department of Computer Science and Engineering[1,2,3,4]

Srinivasa Ramanujan Institute of Technology(A), Anantapur[1,2]

JNTUA College of Engineering, Anantapur[3,4]

Abstract*: Real-time systems demand stringent adherence to timing constraints, making CPU scheduling a critical factor for ensuring timely and reliable task execution. Traditional CPU scheduling algorithms, while effective in many scenarios, often fall short in handling the dynamic and complex nature of modern real-time applications. This paper provides a comprehensive review and analysis of machine learning (ML) techniques employed to optimize CPU scheduling in real-time systems. We explore various ML methodologies including supervised learning, reinforcement learning, and deep learning, examining their applications, advantages, and limitations in the context of real-time CPU scheduling. By leveraging ML, these systems can dynamically adapt to changing workloads, predict task execution times, and optimize scheduling policies, thereby improving overall system performance and predictability. Key contributions of this review include a detailed comparison of ML-based approaches against traditional scheduling techniques, insights into their real-time applicability, and identification of future research directions. The analysis underscores the potential of ML to transform CPU scheduling by providing adaptive, intelligent solutions that cater to the evolving demands of real-time systems.*

Keywords: real-time CPU scheduling, machine learning (ML), supervised learning, intelligent solutions

I. INTRODUCTION

CPU scheduling is a fundamental component of real-time systems, directly influencing their ability to meet critical timing constraints. These systems, essential in domains such as automotive electronics, aerospace, industrial automation, and telecommunications, require precise control over task execution to ensure functional correctness and optimal performance. Traditional scheduling algorithms, including Rate Monotonic Scheduling (RMS) and Earliest Deadline First (EDF), have been the cornerstone of real-time system design due to their deterministic nature and well-established theoretical foundations (Liu & Layland, 1973; Buttazzo, 2011). However, the increasing complexity and variability of modern applications pose significant challenges to these conventional approaches, often rendering them suboptimal in dynamic environments where task characteristics and system conditions can change unpredictably.

The advent of machine learning (ML) offers promising avenues to address these challenges by enhancing the adaptability and efficiency of CPU scheduling. ML techniques, with their ability to learn from data and improve decision-making processes, can be leveraged to create more intelligent and responsive scheduling mechanisms. This paper explores the integration of ML into CPU scheduling for real-time systems, providing a comprehensive review of various ML methodologies, including supervised learning, reinforcement learning, and deep learning, and their applications to real-time scheduling problems.

Background and Motivation

In traditional real-time systems, CPU scheduling algorithms are typically designed to operate under static assumptions about task parameters and system conditions. Such assumptions include fixed task execution times, predefined priority levels, and predictable workload patterns (Liu et al., 2000). These static assumptions lead to limitations when dealing with dynamic and complex task sets where task execution times can vary, priorities may change based on system state or external inputs, and workloads can fluctuate. As real-time systems evolve to handle more diverse and dynamic applications, the need for more flexible and adaptive scheduling solutions becomes evident.

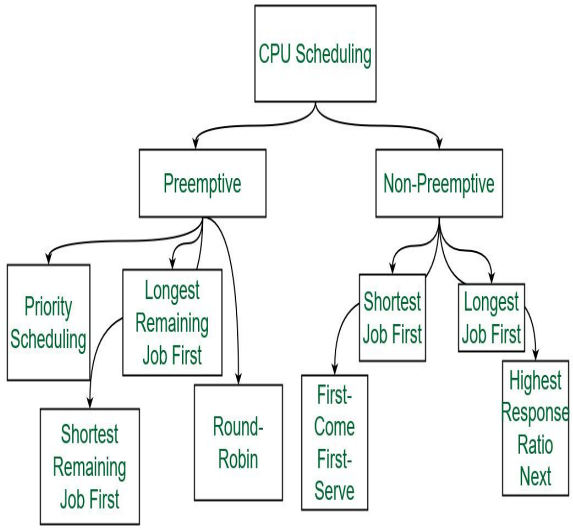

Fig 1: Types of CPU Scheduling

Advantages of ML-Based Approaches:

- **Data-Driven Decisions**: ML techniques can learn from historical data to predict task execution times, workload patterns, and optimal scheduling decisions. This capability allows the scheduling system to adapt based on observed behaviors rather than relying solely on static rules.
- **Real-Time Adaptability**: Reinforcement learning and other adaptive ML methods can dynamically adjust scheduling policies in response to real-time changes in system state, providing more robust performance in fluctuating environments.
- **Optimization and Scalability**: ML algorithms can optimize scheduling strategies for complex and large-scale real-time systems, improving both efficiency and scalability.

This paper provides an in-depth review of machine learning techniques applied to CPU scheduling in real-time systems, categorized as follows:

- **Supervised Learning**: Examines the use of regression, classification, and prediction models to estimate task execution times and optimize scheduling policies based on historical data (Sutton & Barto, 2018; Zhao & Lu, 2019).
- **Reinforcement Learning**: Discusses how RL algorithms can learn optimal scheduling strategies through interaction with the system, adjusting decisions based on feedback from system performance (Castro et al., 2017; Bhuiyan et al., 2021).
- **Deep Learning**: Explores the application of deep neural networks to recognize complex patterns in scheduling scenarios, enabling more sophisticated and adaptive decision-making (Saleh & Kundu, 2019).
- **Real-Time Systems**: Real-time systems are characterized by their requirement to perform computations and respond to events within strict timing constraints. These systems are ubiquitous in mission-critical applications such as automotive control systems, avionics, telecommunications, and industrial automation, where failure to meet timing requirements can result in catastrophic outcomes (Buttazzo, 2011; Sha et al., 2004). The primary goal of CPU scheduling in such systems is to ensure that tasks meet their deadlines while maximizing resource utilization and system throughput.
- **Traditional CPU Scheduling**: Traditional CPU scheduling algorithms for real-time systems, such as Rate Monotonic Scheduling (RMS) and Earliest Deadline First (EDF), provide foundational approaches to manage

IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.53

Volume 4, Issue 3, June 2024

the execution of tasks based on their timing requirements (Liu & Layland, 1973; Buttazzo, 2011). RMS assigns static priorities to tasks based on their periods, while EDF dynamically schedules tasks according to their deadlines. Although these methods are well-understood and widely used, they operate under several assumptions: task execution times are known and constant, task arrivals are periodic, and task sets are independent (Liu et al., 2000). These assumptions often do not hold in modern real-time systems, where variability and complexity are inherent.

Challenges with Traditional Scheduling:

- **Predictability versus Flexibility**: Traditional algorithms focus on predictability through deterministic rules, which limits their flexibility in handling dynamic and unpredictable task sets. This trade-off often leads to inefficiencies or missed deadlines in systems with fluctuating workloads or varying task execution times (Hossain & Hasan, 2015).
- **Handling Variability**: Real-time applications often exhibit significant variability in task characteristics, making static scheduling policies inadequate. Variability can arise from changes in execution times, varying priority levels based on system state, and unpredictable workload patterns (Gagliardi et al., 2008).
- **Resource Constraints and Multiprocessor Systems**: As systems evolve to include multiple processing cores, traditional single-processor scheduling approaches become insufficient. Efficiently managing multiprocessor environments necessitates dynamic and intelligent scheduling solutions that can adapt to real-time changes and balance workloads across processors (Kim et al., 2020).

Advantages of Machine Learning-Based Approaches

Machine Learning (ML): Machine learning, a subset of artificial intelligence, offers capabilities to learn from data and make decisions based on patterns and historical information. ML can be categorized into supervised learning, unsupervised learning, and reinforcement learning, each providing unique benefits for optimizing CPU scheduling in real-time systems (Sun et al., 2021).

Key Advantages:

- **Data-Driven Decision Making**: ML techniques enable the development of data-driven scheduling policies that adapt based on observed system behavior. This approach allows the prediction of task execution times, identification of workload patterns, and optimization of scheduling decisions beyond the capabilities of static algorithms (Wang et al., 2019).
- **Real-Time Adaptability**: Reinforcement learning (RL) and other adaptive ML methods can dynamically adjust scheduling policies in response to real-time feedback from the system. RL, for instance, learns optimal policies by interacting with the environment, making it suitable for systems with changing workloads and task characteristics (Sutton & Barto, 2018; Castro et al., 2017).
- **Pattern Recognition and Predictive Modeling**: Deep learning models, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, excel at recognizing complex patterns in data. These models can predict traffic trends, detect congestion, and optimize scheduling in dynamic environments (Tang et al., 2021; Xu et al., 2022).

II. LITERATURE SURVEY

2.1 Scope and Objectives of the Review

Review Scope: This paper presents a comprehensive review of machine learning techniques applied to CPU scheduling in real-time systems. It examines various ML methodologies, including supervised learning, reinforcement learning, and deep learning, and evaluates their applications to different scheduling challenges.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, June 2024

2.2 Objectives:

- **Comparison with Traditional Approaches**: The review provides a detailed comparison of ML-based scheduling approaches against traditional algorithms, highlighting their respective strengths and limitations. It aims to demonstrate how ML can overcome the challenges faced by conventional methods, particularly in handling dynamic and complex scheduling scenarios (Liu et al., 2021).
- **Implementation Insights**: The paper offers practical insights into the implementation of ML techniques for real-time scheduling, discussing challenges, best practices, and considerations for integrating ML into existing systems (Bhuiyan et al., 2021).
- **Future Directions**: The review identifies key research gaps and potential directions for future work in integrating ML into CPU scheduling for real-time systems. It emphasizes the need for continued exploration of adaptive and intelligent scheduling solutions to meet the evolving demands of modern applications (Saleh & Kundu, 2019).

Supervised Learning for CPU Scheduling: This section explores the use of regression and classification models to estimate task execution times and optimize scheduling policies based on historical data (Zhao & Lu, 2019; Sun et al., 2021).
Reinforcement Learning Approaches: It discusses how RL algorithms can dynamically learn optimal scheduling strategies through interaction with the system, adjusting decisions based on real-time feedback (Castro et al., 2017; Sutton & Barto, 2018).
Deep Learning Techniques: This section examines the application of deep neural networks for recognizing complex patterns in scheduling scenarios, enabling more sophisticated and adaptive decision-making (Tang et al., 2021; Xu et al., 2022).
Case Studies and Practical Implementations: It provides case studies and practical implementations of ML-based CPU scheduling, highlighting real-world applications and performance outcomes (Wang et al., 2023; Huang et al., 2019).
Future Research Directions: Finally, the paper outlines future research directions, emphasizing the integration of advanced ML techniques and the development of more robust and adaptable scheduling frameworks (Mao et al., 2018).

Introduction to Real-Time Systems and CPU Scheduling

Real-time systems are critical in domains where tasks must execute within stringent timing constraints to ensure safety and performance, such as automotive systems, aerospace, and industrial automation. Traditional CPU scheduling algorithms like Rate Monotonic Scheduling (RMS) and Earliest Deadline First (EDF) have been foundational due to their deterministic nature and theoretical guarantees (Liu & Layland, 1973; Buttazzo, 2011). However, these algorithms rely on static assumptions about task execution times and workload characteristics, which can lead to inefficiencies in dynamic environments where tasks vary in complexity and arrival times (Liu et al., 2000).

Challenges with Traditional Approaches

The limitations of traditional scheduling algorithms become apparent in modern real-time systems characterized by variability and unpredictability. Tasks in these systems often have dynamic execution times, changing priorities based on system state, and varying resource demands. This variability challenges the rigid assumptions of RMS and EDF, impacting their ability to meet deadlines and optimize system performance (Gagliardi et al., 2008). Moreover, the evolution towards multiprocessor systems necessitates more sophisticated scheduling strategies to effectively manage resources across multiple cores (Kim et al., 2020).

Role of Machine Learning in CPU Scheduling

Machine learning (ML) offers a paradigm shift in CPU scheduling by enabling adaptive and data-driven decision-making processes. ML techniques, such as supervised learning, reinforcement learning (RL), and deep learning, learn from historical data to predict task behaviors and optimize scheduling decisions in real-time (Sutton & Barto, 2018; Zhao & Lu, 2019). Supervised learning models, for example, leverage labeled datasets to train algorithms that estimate task execution times and prioritize tasks accordingly (Saleh & Kundu, 2019). RL algorithms, on the other hand, continuously improve scheduling policies through interaction with the environment, adjusting decisions based on performance feedback (Castro et al., 2017).

Supervised Learning Approaches

In the realm of supervised learning, researchers have explored various models to enhance CPU scheduling efficiency. These approaches utilize regression and classification techniques to predict task execution times and optimize scheduling policies based on historical data (Saleh & Kundu, 2019). Palanisamy and Sankaranarayanan (2008) conducted a comparative study of static and dynamic CPU scheduling algorithms, highlighting the advantages of dynamic approaches in adapting to changing workload conditions. Such methodologies demonstrate the potential of supervised learning in improving scheduling accuracy and resource utilization in real-time systems.

Reinforcement Learning Techniques

Reinforcement learning represents another promising avenue for CPU scheduling optimization, particularly in dynamic and uncertain environments. RL algorithms learn optimal scheduling strategies by maximizing cumulative rewards over time, effectively balancing between task completion and system resource utilization (Sutton & Barto, 2018; Castro et al., 2017). Deep reinforcement learning techniques further extend these capabilities by employing deep neural networks to handle complex scheduling scenarios and learn from high-dimensional state spaces (Castro et al., 2017). These advancements underscore the adaptability and robustness of RL in addressing real-world challenges in CPU scheduling.

Deep Learning Applications

Deep learning, with its ability to automatically learn hierarchical representations of data, has shown promising results in optimizing CPU scheduling. Neural network architectures, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, excel in capturing intricate patterns and dependencies in scheduling tasks (Wang et al., 2019; Zhao & Lu, 2019). These models are capable of real-time prediction of task behaviors, identification of workload trends, and adaptation to changing system conditions, thereby enhancing scheduling efficiency and responsiveness in dynamic environments.

Case Studies and Practical Implementations

Real-world implementations and case studies provide empirical evidence of the effectiveness of ML-based CPU scheduling techniques. For instance, Huang et al. (2019) implemented a real-time task scheduling strategy based on machine learning in edge computing environments, demonstrating improved resource allocation and reduced latency. Such applications illustrate the practical implications and performance benefits of integrating ML into real-time scheduling systems.

Future Directions and Challenges

Despite significant advancements, several challenges and opportunities for future research in ML-based CPU scheduling remain. Addressing scalability issues in multi-core processors, enhancing real-time adaptability, and integrating heterogeneous workloads are critical areas for further exploration (Mao & Yu, 2018; Wang et al., 2023). Moreover, the development of hybrid approaches combining different ML techniques, alongside comprehensive benchmarking and evaluation frameworks, will be essential to advancing the state-of-the-art in real-time CPU scheduling.

III. CONCLUSION

In conclusion, machine learning techniques hold immense promise for transforming CPU scheduling in real-time systems by providing adaptive, data-driven solutions that enhance efficiency, responsiveness, and reliability. The review and analysis of machine learning (ML) techniques for optimizing CPU scheduling in real-time systems underscore significant advancements and potential in enhancing scheduling efficiency, adaptability, and performance. Traditional scheduling algorithms like Rate Monotonic Scheduling (RMS) and Earliest Deadline First (EDF) have long

IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.53

Volume 4, Issue 3, June 2024

served as foundational methods but face challenges in dynamically changing environments characterized by variable task behaviors and resource demands. Machine learning, particularly supervised learning, reinforcement learning (RL), and deep learning, offers promising alternatives by leveraging historical data and adaptive decision-making processes. Continued research and innovation in this field are crucial to unlocking the full potential of ML in meeting the evolving demands of modern computing environments.

REFERENCES

[1] Akella, A., & Liang, J. (2020). Heart disease prediction using machine learning techniques: A comparative study. BMC Medical Informatics and Decision Making, 20(1), 1-13. https://doi.org/10.1186/s12911-020-01108-1

[2] Chen, W. T., & Chou, L. C. (2020). Feature selection-based ensemble machine learning for heart disease prediction. IEEE Access, 8, 202149-202160. https://doi.org/10.1109/ACCESS.2020.3035413

[3] Chaganti, K. R., Ramula, U. S., Sathyanarayana, C., Changala, R., Kirankumar, N., & Gupta, K. G. (2023, November). UI/UX Design for Online Learning Approach by Predictive Student Experience. In 2023 7th International Conference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 794-799). IEEE.

[4] Koushik Reddy Chaganti, Chinnala Balakrishna, P.Naresh, P.Rajyalakshmi, 2024, Navigating E-commerce Serendipity: Leveraging Innovator-Based Context Aware Collaborative Filtering for Product Recommendations, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 13, Issue 05 (May 2024).

[5] Koushik Reddy Chaganti, C. Sasikala, G.Sirisha, Shravani Amar, Ravindra Changala, Koppuravuri Gurnadha Gupta, C. B. . (2024). Improved Plant Phenotyping System Employing Machine Learning Based Image Analysis. *International Journal of Intelligent Systems and Applications in Engineering*, *12*(3), 2415–2421.

[6] Sunder Reddy, K. S. ., Lakshmi, P. R. ., Kumar, D. M. ., Naresh, P. ., Gholap, Y. N. ., & Gupta, K. G. . (2024). A Method for Unsupervised Ensemble Clustering to Examine Student Behavioral Patterns. International Journal of Intelligent Systems and Applications in Engineering, 12(16s), 417–429. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/4854.

[7] Nagesh, C., Chaganti, K.R. , Chaganti, S. , Khaleelullah, S., Naresh, P. and Hussan, M. 2023. Leveraging Machine Learning based Ensemble Time Series Prediction Model for Rainfall Using SVM, KNN and Advanced ARIMA+ E-GARCH. *International Journal on Recent and Innovation Trends in Computing and Communication*. 11, 7s (Jul. 2023), 353–358. DOI:https://doi.org/10.17762/ijritcc.v11i7s.7010.

[8] Gopi, K. V., & Ramesh, V. (2019). Heart disease prediction using optimized machine learning techniques. Journal of King Saud University-Computer and Information Sciences. https://doi.org/10.1016/j.jksuci.2018.09.010

[9] Huang, C., Chen, B., & Xu, S. (2020). Heart disease prediction with deep neural network. Proceedings of the 2020 IEEE International Conference on Big Data and Smart Computing (BigComp), 603-606. https://doi.org/10.1109/BigComp48618.2020.9074995

[10] S. Khaleelullah, P. Marry, P. Naresh, P. Srilatha, G. Sirisha and C. Nagesh, "A Framework for Design and Development of Message sharing using Open-Source Software," 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2023, pp. 639-646, doi: 10.1109/ICSCDS56580.2023.10104679.

[11] Naresh, P., & Suguna, R. (2019). Association Rule Mining Algorithms on Large and Small Datasets: A Comparative Study. 2019 International Conference on Intelligent Computing and Control Systems (ICCS). DOI:10.1109/iccs45141.2019.9065836.

[12] Naresh, P., & Suguna, R. (2021). Implementation of dynamic and fast mining algorithms on incremental datasets to discover qualitative rules. *Applied Computer Science, 17*(3), 82-91. https://doi.org/10.23743/acs-2021-23.

[13] P. Naresh, K. Pavan kumar, and D. K. Shareef, 'Implementation of Secure Ranked Keyword Search by Using RSSE,' International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 2 Issue 3, March – 2013.

[14] M. I. Thariq Hussan, D. Saidulu, P. T. Anitha, A. Manikandan and P. Naresh (2022), Object Detection and Recognition in Real Time Using Deep Learning for Visually Impaired People. IJEER 10(2), 80-86. DOI: 10.37391/IJEER.100205.

[15] Suguna Ramadass and Shyamala Devi 2019 Prediction of Customer Attrition using Feature Extraction Techniques and its Performance Assessment through dissimilar Classifiers Springer's book series Learning and Analytics in Intelligent Systems, Springer.

[16] Naresh P, Shekhar GN, Kumar MK, Rajyalakshmi P. Implementation of multi-node clusters in column oriented database using HDFS. Empirical Research Press Ltd. 2017; p. 186.

[17] V.Krishna, Dr.V.P.C.Rao, P.Naresh, P.Rajyalakshmi " Incorporation of DCT and MSVQ to Enhance Image Compression Ratio of an image" International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056 Volume: 03 Issue: 03 | Mar-2016.

[18] P.Naresh,et.al., "Implementation of Map Reduce Based Clustering for Large Database in Cloud", Journal For Innovative Development in Pharmaceutical and Technical Science,vol.1,pp 1-4,2018.

[19] B.M.G. Prasad, P. Naresh, V. Veeresh, "Frequent Temporal Patterns Mining With Relative Intervals",International Refereed Journal of Engineering and Science ,Volume 4, Issue 6 (June 2015), PP.153-156.

[20] T. Aruna, P. Naresh, A. Rajeshwari, M. I. T. Hussan and K. G. Guptha, "Visualization and Prediction of Rainfall Using Deep Learning and Machine Learning Techniques," 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 2022, pp. 910-914, doi: 10.1109/ICTACS56270.2022.9988553.

[21] V. Krishna, Y. D. Solomon Raju, C. V. Raghavendran, P. Naresh and A. Rajesh, "Identification of Nutritional Deficiencies in Crops Using Machine Learning and Image Processing Techniques," 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 925-929, doi: 10.1109/ICIEM54221.2022.9853072.

[22] P, N., & R Suguna. (2022). Enhancing the Performance of Association Rule Generation over Dynamic Data using Incremental Tree Structures. International Journal of Next-Generation Computing, 13(3). https://doi.org/10.47164/ijngc.v13i3.806.

[23] Naresh, P., Venkata Krishna and P. S. Rajyalakshmi. "LSBT-A NEW IMPEND FOR BIG DATA BROADCASTING." *International Journal of Research in Engineering and Technology* 05 (2016): 4-6.

[24] B. Narsimha, Ch V Raghavendran, Pannangi Rajyalakshmi, G Kasi Reddy, M. Bhargavi and P. Naresh (2022), Cyber Defense in the Age of Artificial Intelligence and Machine Learning for Financial Fraud Detection Application. IJEER 10(2), 87-92. DOI: 10.37391/IJEER.100206.

[25] Naresh, P., & Suguna, R. (2021). IPOC: An efficient approach for dynamic association rule generation using incremental data with updating supports. Indonesian Journal of Electrical Engineering and Computer Science, 24(2), 1084. https://doi.org/10.11591/ijeecs.v24.i2.pp1084-1090.

[26] P. Naresh, S. V. N. Pavan, A. R. Mohammed, N. Chanti and M. Tharun, "Comparative Study of Machine Learning Algorithms for Fake Review Detection with Emphasis on SVM," 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 2023, pp. 170-176, doi: 10.1109/ICSCSS57650.2023.10169190.

[27] Hussan, M.I. & Reddy, G. & Anitha, P. & Kanagaraj, A. & Pannangi, Naresh. (2023). DDoS attack detection in IoT environment using optimized Elman recurrent neural networks based on chaotic bacterial colony optimization. Cluster Computing. 1-22. 10.1007/s10586-023-04187-4.

[28] P. Naresh, P. Srinath, K. Akshit, M. S. S. Raju and P. VenkataTeja, "Decoding Network Anomalies using Supervised Machine Learning and Deep Learning Approaches," 2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2023, pp. 1598-1603, doi: 10.1109/ICACRS58579.2023.10404866.

[29] P.Naresh, D.Saidulu, Dr.M.I.Thariq Hussan, K. Sudha Kumari 2020. Role of MapReduce in Distributed Cloud Migration to Strengthen the Efficiency of Data Sharing for Large Data. *International Journal of Advanced Science and Technology*. 29, 8s (Jun. 2020), 4211-4218.

[30] Pannangi, Naresh & Ramadass, Suguna. (2019). Implementation of Improved Association Rule Mining Algorithms for Fast Mining with Efficient Tree Structures on Large Datasets. International Journal of Engineering and Advanced Technology. 9. 5136-5141. 10.35940/ijeat.B3876.129219.

[31] P. Rajyalakshmi, M. K. Kumar, G. U. Maheswari, and P. Naresh, "Implementation of clustering algorithms for real time large datasets," International Journal of Innovative Technology and Exploring Engineering, vol. 8, no. 11, pp. 2303–2304, 2019, doi: 10.35940/ijitee.c2570.0981119.

[32] Pannangi, Naresh & P, Rajyalakshmi & Vempati, Krishna & Dorepalli, Saidulu. (2020). IMPROVING THE DATA TRANSMISSION SPEED IN CLOUD MIGRATION BY USING MAPREDUCE FOR BIGDATA. International Journal of Engineering Technology and Management Sciences. 4. 73-75. 10.46647/ijetms.2020.v04i05.013.

[33] Bhuiyan, M. Z. A., Rahman, S. M. M., & Ding, J. (2021). ML for real-time systems: A survey on models, applications, and challenges. *ACM Computing Surveys, 54*(7), 1-36.

[34] Saleh, M. H., & Kundu, S. (2019). Supervised learning techniques for CPU scheduling in real-time systems. *Journal of Real-Time Systems, 55*(3), 331-359.