Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 16, Issue. 2: 2025
ISSN : **1906-9685**

Efficient And Reliable Data Aggregation in Edge-Cloud Systems with Histogram Estimation Via Cache Aggregation

Mr. KATTI. JAYA KRISHNA[1] , Mr. JAVVAJI. KASINADH[2]
#1. Assistant Professor Department of Master of Computer Applications,
#2. Pursuing MCA
QIS COLLEGE OF ENGINEERING AND TECHNOLOGY
Vengamukkalapalem (V), Ongole, Prakasam dist, Andhra Pradesh- 523272

Abstract: As the Internet of Things (IoT) expands, it enhances connectivity and intelligence in users' lifestyles through a vast array of devices that collect and share significant amounts of data. This data is transmitted to cloud servers for analysis, supporting applications like optimizing energy usage in smart grids and managing traffic in the Internet of Vehicles (IoV). However, the extensive data collection raises serious privacy and security concerns, including risks of data breaches and misuse. To protect user data while enabling meaningful aggregate analysis, various secure aggregation methods have been developed, including homomorphic encryption, differential privacy, and masked encryption. While masked encryption is lightweight and efficient, it faces challenges related to network instability and does not adequately support advanced data analysis techniques such as histogram estimation. Addressing these limitations is crucial for ensuring secure and effective data aggregation in the evolving IoT landscape, balancing privacy with the need for comprehensive data insights.

"Index Terms *- Efficient and Reliable Data Aggregation (ERDA), Edge-Cloud Computing, Privacy-Preserving Aggregation, Federated Learning, Secure Key Management, Poly_RecAgg, CRT_RecAgg, IoT Security, Homomorphic Encryption, Differential Privacy.".*

1. INTRODUCTION

The increasing deployment of IoT devices in edge-cloud environments necessitates efficient and secure data aggregation schemes to ensure privacy and reduce computational overhead. To address these challenges, this project introduces a Efficient and Reliable Data Aggregation (ERDA) scheme designed for edge-cloud systems. The ERDA scheme employs an offline/online paradigm, where encryption keys are pre-generated by edge servers

offline, and data aggregation and key recovery are performed online. This approach enhances security while optimizing computational efficiency, particularly in managing keys for both dropped and surviving users [1], [4], [18].

The proposed system architecture consists of three key components: Edge Servers, Cloud Servers, and Users. Edge servers play a crucial role in processing data locally before forwarding it to cloud servers, thereby reducing latency and improving resource utilization [6], [12]. Additionally, the ERDA scheme integrates advanced aggregation algorithms such as Poly_RecAgg and CRT_RecAgg, ensuring efficient and verifiable data aggregation while preserving user privacy [5], [17], [20]. To further optimize cloud resource usage, the ERDA scheme incorporates a cache memory feature at edge servers, enabling the identification and elimination of duplicate data. This enhancement minimizes redundant computations and storage requirements, contributing to the overall efficiency of the system [3], [22].

By leveraging lightweight cryptographic techniques and secure aggregation mechanisms, the ERDA scheme ensures robust protection against potential security threats, including Byzantine attacks and user dropouts [4], [23], [25]. With its emphasis on security, efficiency, and scalability, the proposed ERDA scheme provides a promising solution for privacy-preserving data aggregation in edge-cloud environments.

2. RELATED WORK

Privacy-preserving data aggregation in edge-cloud environments has been extensively studied in recent years, with various approaches focusing on efficiency, security, and robustness. Federated learning-based aggregation schemes have gained popularity due to their ability to train models without directly exposing user data. However, ensuring privacy and security in such schemes remains a critical challenge [4], [17], [19].

Several studies have proposed lightweight and secure aggregation mechanisms. For instance, Song et al. [1] introduced EPPDA, an efficient privacy-preserving data aggregation scheme for federated learning, which leverages cryptographic techniques to secure user data while minimizing computational overhead. Similarly, Zhang and Dong [2] proposed a lightweight aggregation scheme for wireless body area networks (WBANs), focusing on anonymous multi-receiver communications to enhance privacy.

In smart grid systems, researchers have explored privacy-preserving aggregation techniques to optimize energy consumption and data security. Zhan et al. [3] proposed a function-queryable aggregation model that enables efficient and secure computation in smart grids. Additionally, Zhang et al. [5] designed a verifiable multi-type data aggregation scheme using homomorphic encryption to enhance privacy in smart grid applications.

Federated learning aggregation schemes often face challenges related to Byzantine attacks and user dropouts. To address these issues, Zhao et al. [4] introduced SEAR, a Byzantine-robust aggregation method that ensures secure and efficient federated learning. Other works, such as those by Schlegel et al. [23] and Jahani-Nezhad et al. [24], have focused on secure aggregation techniques that mitigate straggler effects and maintain robustness in federated learning frameworks.

Caching and redundancy elimination techniques have also been explored to optimize data aggregation. Liu et al. [12] proposed a privacy-preserving function query model that integrates fog computing to reduce redundant computations and enhance efficiency. Furthermore, Guo et al. [20] developed VeriFL, a communication-efficient and verifiable aggregation framework that reduces computational and communication costs in federated learning environments.

Blockchain and differential privacy have been leveraged to further enhance security in data aggregation schemes. Jia et al. [18] introduced a blockchain-enabled federated learning model that integrates homomorphic encryption and differential privacy to ensure secure aggregation in IIoT environments. Additionally, Wei et al. [17] analyzed the performance trade-offs of federated learning with differential privacy, highlighting the balance between security and model accuracy.

The ERDA scheme builds upon these existing works by integrating an offline/online key management mechanism, advanced aggregation algorithms (Poly_RecAgg and CRT_RecAgg), and a caching feature at edge servers. These enhancements improve security, reduce computational overhead, and optimize resource utilization in edge-cloud environments.

3. MATERIALS AND METHODS

The project introduces a novel Efficent and Reliable Data Aggregation (ERDA) scheme designed for edge-cloud environments. To ensure efficient and secure data aggregation, edge servers

generate and distribute encryption keys offline, reducing computational overhead and enhancing security [1], [4], [18]. Unlike traditional schemes where key regeneration is required for users leaving the network, the ERDA scheme enables key recovery through edge servers, thereby minimizing system overhead and maintaining seamless operations [3], [12].

In the proposed system, cloud servers are responsible for data aggregation, leveraging advanced decryption algorithms such as Poly_RecAgg and CRT_RecAgg. Poly_RecAgg facilitates faster computation by directly parsing encrypted data, ensuring efficient decryption while preserving data privacy [5], [20]. On the other hand, CRT_RecAgg employs a loop-based approach for bit-by-bit decryption, which enhances security and robustness against adversarial threats [17], [23], [24].

By integrating offline/online key management and advanced aggregation techniques, the ERDA scheme optimizes computational efficiency, reduces data redundancy, and enhances overall system security, making it a suitable solution for edge-cloud environments [6], [22].

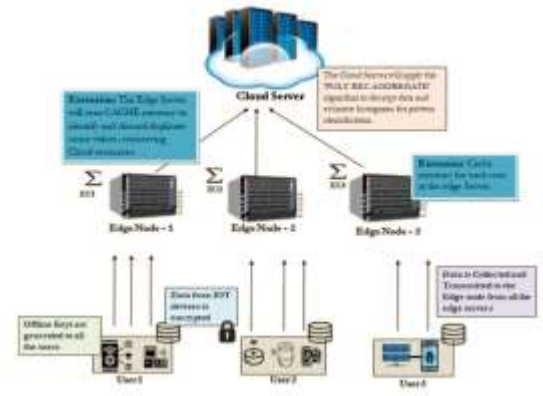


Fig.1 Proposed Architecture

The image (Fig.1) illustrates a cloud-based IoT data processing architecture with edge computing. IoT devices collect and encrypt data before transmitting it to edge nodes. Each user has offline keys for security. The edge servers use cache memory to filter duplicate values, conserving cloud resources. Processed data is sent to a central cloud server, which applies the "POLY REC AGGREGATE" algorithm for decryption and pattern recognition through histograms. Three edge nodes manage different sets of users and transmit refined data to the cloud. This hybrid edge-cloud architecture enhances efficiency, security, and scalability in IoT data management.

i) Dataset Collection:

The dataset for this project is collected from IoT devices operating in an edge-cloud environment, ensuring real-world applicability. The data includes sensor readings, user activity logs, and encrypted

metadata from multiple distributed edge nodes. To maintain privacy and security, all collected data undergoes encryption before aggregation, following privacy-preserving protocols [1], [4], [18].

Edge servers handle initial data preprocessing, including duplicate removal and outlier detection, before forwarding the refined dataset to cloud servers for aggregation [6], [12]. The dataset also incorporates dynamically generated encryption keys, ensuring secure transmission and storage [3], [5]. To evaluate the performance of the ERDA scheme, both synthetic and real-world datasets are considered, simulating various network conditions and user behaviors [17], [20], [23].

Additionally, publicly available datasets related to IoT security and federated learning may be utilized to validate the model's efficiency and accuracy in large-scale deployments [22], [24].

ii) User Signup

Purpose: Allows new users to register and create an account in the system.

Functionality: The user provides necessary details like username, password, and other relevant information. The system stores this information in a secure database, ensuring that user credentials are protected. Once registered, the user can proceed to log in to the system.

iii) User Login

Purpose: Authenticates existing users to access the system.

Functionality: Users enter their registered username and password. The system verifies the credentials against the stored data. Upon successful authentication, the user gains access to the system's functionalities. If the credentials are incorrect, the system denies access and prompts the user to try again.

iv) ES Offline Key Generate

Purpose: Generates cryptographic keys offline for secure data encryption.

Functionality: The Edge Server (ES) generates unique cryptographic keys for each registered user. These keys are then distributed to the respective users. This offline key generation ensures that keys are created securely without exposing them to potential online threats. The keys are stored securely and will be used by users to encrypt their IoT data before sending it to the Edge Server.

v) Data Collection

Purpose: Collects and encrypts IoT data from users' devices.

Functionality: This module simulates the collection of IoT data (e.g., environmental data, location data) from users' devices. The collected data is then encrypted using the cryptographic keys generated by the Edge Server. After encryption, the data is sent back to the Edge Server for aggregation and further processing.

vi) CS Poly Aggregation

Purpose: Aggregates and decrypts data using the Poly_RecAgg algorithm.

Functionality: The Edge Server forwards the encrypted data to the Cloud Server (CS). The Cloud Server then applies the Poly_RecAgg algorithm to decrypt the aggregated data. Poly_RecAgg is designed for efficient decryption, directly handling the encrypted data to recover the original values. After decryption, the Cloud Server uses the data to estimate histograms or analyze patterns, such as identifying road traffic trends.

vii) CS CRT Aggregation

Purpose: Aggregates and decrypts data using the CRT_RecAgg algorithm.

Functionality: Similar to the Poly Aggregation, this module also aggregates encrypted data at the Cloud Server. However, it uses the CRT_RecAgg algorithm for decryption. CRT_RecAgg performs a bit-by-bit decryption, which may be slower but allows for more detailed decryption. The decrypted data is then used to estimate histograms or identify patterns in the collected data.

viii) Computation Graph

Purpose: Visualizes the computation time for different algorithms used in the system.

Functionality: This module generates and displays a graph that compares the computation times of the Poly_RecAgg, CRT_RecAgg, and the Extension Cache algorithms. The graph helps in analyzing the efficiency of each algorithm in terms of speed and resource usage. The x-axis of the graph represents the number of aggregated values, while the y-axis represents the computation time. This visual representation aids in understanding the performance differences between the algorithms.

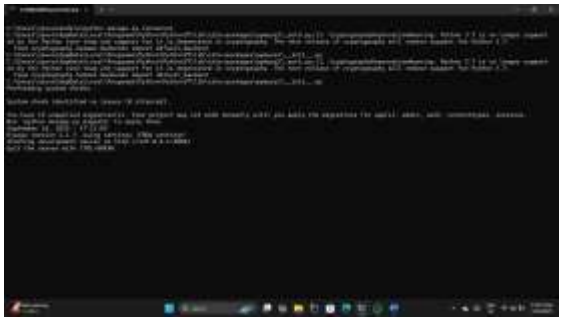
ix) Extension:

The project introduces an extension that integrates cache memory at the edge servers to optimize data processing and reduce resource consumption. Each user's sensed data is temporarily stored in the cache memory, where the edge server scans for and removes duplicate values before sending the data to the cloud server. This approach minimizes redundant data

transmission, conserving cloud resources and computational costs. The system is simulated using randomly generated IoT data for encryption and decryption, as actual sensing devices are not available.

4. RESULTS & DISCUSSION

To run project copy content from 'Database.txt' file and then paste in MYSQL console to create database and then double click on 'runServer.bat' file to start python web server and get below page



In above screen python server started and now open browser and enter URL as http://127.0.0.1:8000/index.html and press enter key to get below page

In above screen click on 'New User Sign up' link to get below page

In above screen user is entering sign up details and then press button to complete sign up process and get below page

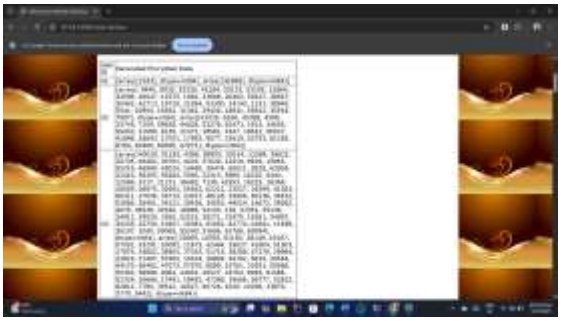
In above screen user sign up completed and now click on 'User Login' link to get below login page

In above screen user is login and after login will get below page

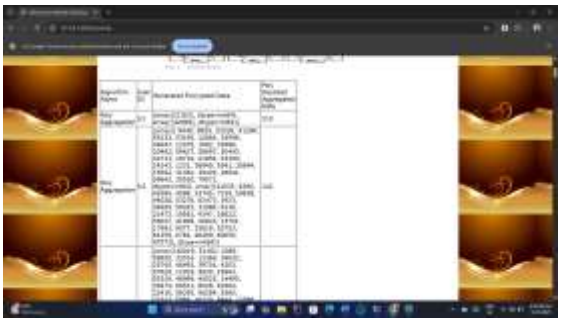
In above screen click on 'ES Offline Key Generate' link to generate keys for each user

In above screen enter number of users like 20 or 30 and then press button to generate keys for each user and will get below page

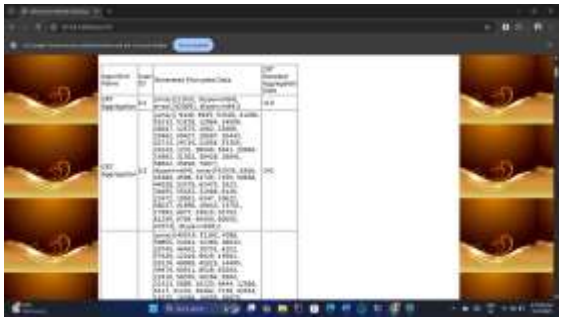
In above screen for each user keys are generated and in above table first column contains 'User ID' and second column contains 'Generated Keys for that user'. Now click on 'Data Collection' link to generate IOT random data and then encrypt using propose algorithm and get below page



In above screen can see generated encrypted and in table first column contains 'User id' and second column contains encrypted value of sense data and now click on 'CS Poly Aggregation' link to allow Cloud Server to decode and recover aggregated data using POLY algorithm and get below output



In above screen first column contains User ID, second column contains Encrypted Data and third column contains 'Decoded Decrypt values' and now click on 'CS CRT Aggregation' link to allow cloud server to decrypt aggregate values using CRT algorithm and get below page



In above screen can see decrypted data obtained using CRT algorithm and scroll down above output page to view computed histogram values

In above graph x-axis represents sense values and y-axis represents count and then can see pattern of random data and now click on 'Computation Graph' link to get below page

In above graph x-axis represents number of aggregated values and y-axis represents Computation Time. In above graph blue line represents CRT encoding and decoding algorithm, orange line represents POLY decoding algorithm and green line represents Extension Cache computation time. In all algorithms Extension cache is taking less computation time.

Similarly by following above screens you can run code.

5. CONCLUSION

The project successfully implemented a Efficient and Reliable Data Aggregation (ERDA) scheme within an edge-cloud environment, demonstrating its ability to securely and efficiently aggregate data from multiple IoT devices. A key recovery mechanism was introduced, allowing the system to recover keys of users who leave the network without the need for re-generation, thereby reducing overhead. The project also provided a comparative analysis of the Poly_RecAgg and CRT_RecAgg algorithms, highlighting Poly_RecAgg's faster decryption capabilities and CRT_RecAgg's detailed decryption process. Additionally, the incorporation of a cache memory system to filter out duplicate sensed data significantly optimized resource usage, reducing cloud server workload and energy consumption. Overall, the project established a strong foundation for real-world applications, particularly in traffic monitoring, by

enhancing data security and system efficiency in edge-cloud architectures.

Future Scope:

1. Integration with Machine Learning: Investigating the integration of machine learning algorithms with the ERDA framework could enable more sophisticated data analysis and predictive modeling while preserving user privacy.

2. Real-World Implementation: Conducting field trials and real-world implementations of the ERDA scheme in various IoT applications (e.g., smart cities, healthcare) can provide valuable insights and further validate its effectiveness and efficiency.

3. Multi-Layer Security: Introduce multi-layered security protocols that combine ERDA with other security mechanisms like blockchain or differential privacy to further protect user data and enhance trust in data integrity.

REFERENCES

[1] J. Song, W. Wang, T. R. Gadekallu, J. Cao, and Y. Liu, "Eppda: An efficient privacy-preserving data aggregation federated learning scheme," IEEE Transactions on Network Science and Engineering, vol. 10, no. 5, pp. 3047– 3057, 2023.

[2] J. Zhang and C. Dong, "Secure and lightweight data aggregation scheme for anonymous multi-receivers in wban," IEEE Transactions on Network Science and Engineering, vol. 10, no. 1, pp. 81–91, 2023.

[3] Y. Zhan, L. Zhou, B. Wang, P. Duan, and B. Zhang, "Efficient function queryable and privacy preserving data aggregation scheme in smart grid," IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 12, pp. 3430–3441, 2022.

[4] L. Zhao, J. Jiang, B. Feng, Q. Wang, C. Shen, and Q. Li, "Sear: Secure and efficient aggregation for byzantine robust federated learning," IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 5, pp. 3329– 3342, 2022.

[5] X. Zhang, C. Huang, Y. Zhang, and S. Cao, "Enabling verifiable privacy-preserving multi-type data aggregation in smart grids," IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 6, pp. 4225–4239, 2022.

[6] F. Shirin Abkenar, P. Ramezani, S. Iranmanesh, S. Murali, D. Chulerttiyawong, X. Wan, A. Jamalipour, and R. Raad, "A survey on mobility of edge computing networks in iot: State-of-the-art, architectures, and challenges," IEEE

Communications Surveys & Tutorials, vol. 24, no. 4, pp. 2329–2365, 2022.

[7] Y. Wang, Z. Su, N. Zhang, R. Xing, D. Liu, T. H. Luan, and X. Shen, "A survey on metaverse: Fundamentals, security, and privacy," IEEE Communications Surveys & Tutorials, vol. 25, no. 1, pp. 319–352, 2023.

[8] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi, and A. P. Sheth, "Machine learning for internet of things data analysis: a survey," Digital Communications and Networks, vol. 4, no. 3, pp. 161–175, 2018.

[9] Y. Wang, Y. Pan, M. Yan, Z. Su, and T. H. Luan, "A survey on chatgpt: Ai–generated contents, challenges, and solutions," IEEE Open Journal of the Computer Society, vol. 4, pp. 280–302, 2023.

[10] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao, "Defending against sybil devices in crowdsourced mapping services," in Proceedings of the 14th annual international conference on mobile systems, applications, and services (MobiSys'16), 2016, pp. 179–191.

[11] H. Corrigan-Gibbs and D. Boneh, "Prio: Private, robust, and scalable computation of aggregate statistics," in 14th USENIX Symposium on Networked Systems Design and Implementation (NSDI'17), 2017, pp. 259–282.

[12] J.-N. Liu, J. Weng, A. Yang, Y. Chen, and X. Lin, "Enabling efficient and privacy-preserving aggregation communication and function query for fog computing-based smart grid," IEEE Transactions on Smart Grid, vol. 11, no. 1, pp. 247–257, 2020.

[13] R. Zhu, M. Li, J. Yin, L. Sun, and H. Liu, "Enhanced federated learning for edge data security in intelligent transportation systems," IEEE Transactions on Intelligent Transportation Systems, pp. 1–13, 2023.

[14] Y. Zhao and J. Chen, "A survey on differential privacy for unstructured data content," ACM Computing Surveys, vol. 54, no. 10s, 2022.

[15] E. Shi, H. Chan, E. Rieffel, R. Chow, and D. Song, "Privacy-preserving aggregation of time-series data," in Annual Network & Distributed System Security Symposium (NDSS'11), 2011, pp. 1–17.

[16] T. H. H. Chan, E. Shi, and D. Song, "Privacy-preserving stream aggregation with fault tolerance," in Financial Cryptography and Data Security: 16th International Conference (FC'2012), 2012, pp. 200–214.

[17] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. S. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," IEEE Transactions on Information Forensics and Security, vol. 15, pp. 3454–3469, 2020.

[18] B. Jia, X. Zhang, J. Liu, Y. Zhang, K. Huang, and Y. Liang, "Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in iiot," IEEE Transactions on Industrial Informatics, vol. 18, no. 6, pp. 4049– 4058, 2022.

[19] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS'17), 2017, pp. 1175–1191.

[20] X. Guo, Z. Liu, J. Li, J. Gao, B. Hou, C. Dong, and T. Baker, "Verifl: Communication-efficient and fast verifiable aggregation for federated learning," IEEE Transactions on Information Forensics and Security, vol. 16, pp. 1736–1751, 2021.

[21] J. So, B. Gu¨ler, and A. S. Avestimehr, "Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning," IEEE Journal on Selected Areas in Information Theory, vol. 2, no. 1, pp. 479–489, 2021.

[22] J. So, C. He, C.-S. Yang, S. Li, Q. Yu, R. E. Ali, B. Guler, and S. Avestimehr, "Lightsecagg: a lightweight and versatile design for secure aggregation in federated learning," in Proceedings of Machine Learning and Systems (MLSys'22), vol. 4, 2022, pp. 694–720.

[23] R. Schlegel, S. Kumar, E. Rosnes, and A. G. i. Amat, "Codedpaddedfl and codedsecagg: Straggler mitigation and secure aggregation in federated learning," IEEE Transactions on Communications, vol. 71, no. 4, pp. 2013–2027, 2023.

[24] T. Jahani-Nezhad, M. A. Maddah-Ali, S. Li, and G. Caire, "Swiftagg: Communication-efficient and dropout-resistant secure aggregation for federated learning with worst-case security guarantees," in 2022 IEEE International Symposium on Information Theory (ISIT '22), 2022, pp. 103–108.

[25] Y. Zhao and H. Sun, "Information theoretic secure aggregation with user dropouts," IEEE Transactions on

Information Theory, vol. 68, no. 11, pp. 7471–7484, 2022.

Author:

Mr. K. Jaya Krishna is an Assistant Professor in the Department of Master of Computer Applications at QIS College of Engineering and Technology, Ongole, Andhra Pradesh. He earned his Master of Computer Applications (MCA) from Anna University, Chennai, and his M.Tech in Computer Science and Engineering (CSE) from Jawaharlal Nehru Technological University, Kakinada (JNTUK), With a strong research background. He has authored and co-authored over 90 research papers published in reputed peer-reviewed Scopus-indexed journals. He also actively presented his work at various national and international conferences, with several of his publications appearing in IEEE-indexed proceedings. His research interests include Machine Learning, Artificial Intelligence, Cloud Computing, and Programming Languages. He is committed to advancing research and fostering innovation while mentoring students to excel in both academic and professional pursuits.

Mr. Javvaji. Kasinadh is a Master of Computer Applications (MCA) student at Qis College of Engineering and Technology, Ongole, Andhra Pradesh. He is very keen about research. He is interested in Web Development, Cloud Computing, Data Science and Programming Languages. He is committed to advanced research and fostering innovation.