Archives available at journals.mriindia.com

International Journal of Recent Advances in Engineering and Technology

ISSN: 2347-2812
Volume 14 Issue 01, 2025

IntegriScan: A Graph-Aided Model for Detecting Corrupted and Anomalous Data Patterns

M. Asha Aruna Sheela[1], Mailavarapu Tejaswi[2] , Nallani Bhanu Prakash[3], Manam Dhana Sri Tulasi[4] , Kongara Anitha[5]

Assistant Professor & HOD,Department of Computer Science & Engineering ,Chalapathi Institute of Engineering and Technology, LAM, Guntur, AP, India[1]
Department of Computer Science and Engineering,Chalapathi Institute of Engineering and Technology, LAM, Guntur, AP, India [2]
Department of Computer Science and Engineering,Chalapathi Institute of Engineering and Technology, LAM, Guntur, AP, India [3]
Department of Computer Science and Engineering,Chalapathi Institute of Engineering and Technology, LAM, Guntur, AP, India [4]
Department of Computer Science and Engineering,Chalapathi Institute of Engineering and Technology, LAM, Guntur, AP, India [5]

Peer Review Information

Submission: 13 Jan 2025
Revision: 16 Feb 2025
Acceptance: 09 March 2025

Keywords

Data Corruption
Outlier Detection
PAACDA
Adamic-Adar
Machine Learning
Random Forest

Abstract

In today's data-driven environments, ensuring data integrity is critical for accurate decision-making. Data corruption—caused by system errors, transmission faults, or malicious attacks—can lead to misleading analytical results. Existing machine learning models like Local Outlier Factor (LOF), Isolation Forest, and One-Class SVM offer partial solutions but often lack the precision required in complex datasets. This paper introduces a novel algorithm, PAACDA (Proximity-based Adamic-Adar Corruption Detection Algorithm), that leverages graph-based Adamic-Adar similarity to identify outlier and corrupted values. The algorithm uses local proximity measurements to determine abnormal data points by comparing feature similarity scores and thresholds derived from mean-based scaling. Additionally, we propose a hybrid model—Hybrid PAACDA—that extracts features from PAACDA and trains a Random Forest classifier to predict corrupted data in future datasets. The system is implemented using a Django-based web interface, providing modules for training and evaluation across multiple algorithms. Experimental results show that PAACDA outperforms traditional methods, achieving 94% accuracy, while the Hybrid PAACDA extension delivers 100% accuracy, confirming its effectiveness in real-time corruption detection.

INTRODUCTION

As organizations increasingly rely on large-scale data for decision-making, ensuring the accuracy and consistency of data becomes critical. Data corruption refers to errors in datasets that deviate from expected patterns due to missing values, hardware failures, cyberattacks, or system glitches. These corrupted entries often mislead analytical models, result in incorrect insights, and cause flawed business or security decisions. Therefore, detecting and filtering such anomalies is a crucial preprocessing step in any data pipeline.Traditional anomaly detection algorithms such as Local Outlier Factor (LOF), Isolation Forest, and One-Class Support Vector Machine (OCSVM) have shown promise in identifying corrupted records. However, these algorithms generally rely on distance or density-based metrics, which can be inefficient when dealing with high-dimensional, sparse, or complex data relationships. Additionally, these methods do not adapt well to graph-structured data or relational datasets that require contextual understanding.

To overcome these limitations, we propose a novel graph-based algorithm called PAACDA (Proximity-based Adamic-Adar Corruption Detection Algorithm). PAACDA is inspired by the Adamic-Adar similarity index, commonly used in link prediction problems. By modeling data entries as nodes in a similarity graph and applying the Adamic-Adar index to evaluate proximity between entries, PAACDA provides a robust measure of abnormality in data. The method flags entries with high similarity as corrupted and those with lower similarity as normal.

Furthermore, we extend the system with a Hybrid PAACDA model, combining graph-based feature extraction with a Random Forest classifier. This extension enables accurate predictions of data corruption in unseen datasets. A web interface developed using Django allows users to load datasets, apply different algorithms, and visualize performance results interactively.

RELATED WORKS

Data corruption and anomaly detection are well-researched problems in machine learning and data mining. Various techniques have been developed to identify outliers, detect anomalies, and ensure data quality. Most traditional approaches fall into categories such as distance-based, density-based, clustering-based, and classification-based models.

1. Traditional Outlier Detection Methods

Among the most widely used unsupervised methods is the Local Outlier Factor (LOF) algorithm, which measures the local density deviation of a data point relative to its neighbors. Although effective in lower-dimensional data, LOF tends to degrade in performance when applied to high-dimensional or complex data structures. Similarly, the Isolation Forest algorithm isolates observations by randomly selecting features and split values. It performs well in detecting anomalies in large datasets but does not utilize feature similarity or graph-based proximity.

The One-Class Support Vector Machine (OC-SVM) is a boundary-based method that learns the properties of normal data and identifies any deviation as an anomaly. However, OC-SVMs require careful tuning of hyperparameters and do not scale efficiently with large datasets or non-linear feature relationships.

2. Graph-Based Anomaly Detection

Graph-based techniques are gaining traction in anomaly detection due to their ability to capture complex relationships between data instances. One popular method is the Adamic-Adar similarity index, initially developed for link prediction in social networks. It assigns a similarity score between two nodes based on their shared neighbors, favoring rare or less-connected neighbors more heavily. This method is highly effective for relational or proximity-based analysis but has rarely been applied in the context of data corruption detection in tabular datasets.

Several studies have extended graph-based models to include anomaly detection in cyber-security, social networks, and recommendation systems. For instance, approaches like Graph Neural Networks (GNNs) and Random Walk-based Outlier Detection have been explored to identify structural anomalies. However, such methods are often computationally intensive and require labeled graph data, limiting their applicability in tabular anomaly detection scenarios.

3. Hybrid and Ensemble Learning Models

To improve detection accuracy and generalization, researchers have proposed hybrid models that combine feature extraction techniques with ensemble learning methods. Random Forests, for instance, are widely used due to their robustness to overfitting and ability to handle mixed data types. When integrated with anomaly scoring or similarity-based features, Random Forests serve as powerful classifiers for both supervised and semi-supervised learning.

Studies such as those by Liu et al. (2008) on hybrid isolation and decision trees, and recent works on deep anomaly detection with ensemble voting, demonstrate that combining multiple views of the data (e.g., proximity, distribution, and structure) often leads to more accurate and interpretable results. These findings support the motivation behind the proposed Hybrid PAACDA model.

4. Gaps in Current Research

Despite advancements in outlier and anomaly detection, existing solutions often overlook relational similarity between features and instances—especially in unsupervised tabular data. Furthermore, few approaches utilize graph-theoretical measures like Adamic-Adar for structured anomaly detection. There is also a lack of integrated systems that combine detection, classification, and prediction in a user-friendly interface. These gaps highlight the need for a method like PAACDA, which unites graph-based analysis with ensemble learning and interactive deployment.

5. Existing System

Current data corruption and anomaly detection systems predominantly rely on traditional outlier detection algorithms such as Local Outlier Factor (LOF), Isolation Forest, and One-Class Support Vector Machines (OC-SVM). These models operate based on either distance, density, or boundary assumptions to identify data points that deviate significantly from the rest of the dataset. LOF computes the local density of data and identifies outliers based on how isolated they are from their neighbors, while Isolation Forest randomly partitions data to isolate anomalies. OC-SVM learns the structure of normal data and identifies deviations as anomalies.

Although these systems have proven effective in certain domains, they typically assume that data lies in a well-behaved numerical space and often ignore the relational or contextual similarities between data points. Moreover, they lack adaptability to graph-based or proximity-aware modeling, which limits their performance when detecting subtle or context-sensitive corruptions. Additionally, most existing systems do not provide a flexible or user-friendly interface for real-time deployment and visualization of results

5.1 Limitations of Existing System

- Lack of Contextual Awareness: Traditional models fail to capture relational proximity between data points, such as feature similarity or shared neighbors.
- Poor Handling of Structural Anomalies: They are not designed for datasets with graph-like or high-dimensional relationships.
- Limited Scalability: Algorithms like OC-SVM do not scale well for large or complex datasets.
- Generic Thresholds: Predefined decision boundaries may not adapt well across diverse datasets.
- No Hybrid Modeling: There is limited integration of multiple techniques (e.g., similarity analysis + ensemble classifiers) for improved robustness.
- Lack of Prediction Capability: Most outlier detectors identify anomalies in a batch process and do not generalize to future prediction.
- No GUI or Real-Time Deployment: Existing models often lack practical interfaces, making them less usable for non-technical users or real-time applications.

.

6. Proposed System

The proposed system introduces a novel approach to data corruption detection through a graph-based algorithm called PAACDA (Proximity-based Adamic-Adar Corruption Detection Algorithm). Unlike traditional models, PAACDA evaluates data integrity based on the similarity between records using the Adamic-Adar index, which is effective in measuring relational proximity between nodes in a graph structure. Each record in the dataset is treated as a node, and the similarity between records is assessed by analyzing the frequency and uniqueness of shared attributes. If the calculated similarity exceeds a dynamic threshold, the instance is flagged as potentially corrupted. The threshold is calculated using statistical operations based on the column mean and a division range, improving flexibility across datasets.

To extend its detection capabilities, a Hybrid PAACDA model is proposed by combining the extracted PAACDA similarity features with a Random Forest classifier. This hybrid architecture enables the system not only to detect corruption in current datasets but also to predict and classify corrupted records in unseen data. The entire system is deployed via a user-friendly Django-based web interface, allowing users to upload datasets, run different detection models (LOF, Isolation Forest, OC-SVM, PAACDA, Hybrid PAACDA), and visualize performance metrics and predictions in real-time.

6.1Advantages of the Proposed System

- Graph-Based Detection: Uses Adamic-Adar similarity for context-aware and proximity-based detection of corrupted data.
- Hybrid Predictive Power: Combines graph-based insights with Random Forest classification for highly accurate future predictions.

- Dynamic Thresholding: Uses dataset-specific statistics (mean-based range) to adaptively identify anomalies.
- Higher Accuracy: Achieves up to 94% accuracy with PAACDA and 100% accuracy with Hybrid PAACDA.
- Handles Missing Values: Explicit handling of NaN or missing values by assigning them maximum corruption index.
- Web-Based Interface: Enables real-time dataset upload, model selection, and result visualization through a Django GUI.

PROPOSED METHODOLOGY

The proposed PAACDA system is designed to detect corrupted or anomalous data entries in structured datasets using a graph-based approach and a hybrid classification extension. The methodology consists of five major stages: data preprocessing, similarity-based scoring using Adamic-Adar, corruption detection, hybrid model training, and web-based user interaction.

1. Data Preprocessing

Before applying the detection algorithm, the dataset undergoes standard preprocessing steps:

- Missing Value Handling: Empty or null values are preserved for special handling during similarity evaluation.
- Standardization: All numerical columns are normalized to ensure consistency in similarity calculation.
- Conversion to Graph Nodes: Each data record (row) is treated as a node in a similarity graph, where comparisons between records simulate node-to-node proximity.

2. Adamic-Adar Similarity Computation

The core of PAACDA relies on computing Adamic-Adar similarity, a graph-theoretic measure that evaluates the proximity between data entries based on shared features:

$$\text{Score}_{ij} = \sum_{k \in N(i) \cap N(j)} \frac{1}{\log(\deg(k))}$$

Where:

- N(i) and N(j) are the neighboring features of records i and j,
- deg(k)is the degree (frequency) of feature k across all records.

This score is used to compare each record to others in the dataset, and abnormal proximity scores (too high or too low) indicate potential data corruption.

3. Corruption Detection via Thresholding

- A column-wise mean is computed for each attribute.
- A dynamic range is established using mean/4\text{mean}/4mean/4, which serves as a threshold for evaluating whether a data value falls outside expected bounds.
- For each record, if the computed PAACDA score deviates significantly (i.e., outside the range), it is flagged as corrupted.
- Missing values are automatically assigned a PAACDA score of infinity, marking them as invalid or corrupted by default.

4. Hybrid PAACDA Extension

To improve generalization and predictive capabilities, PAACDA is extended using a Random Forest classifier:

- Feature vectors are created from the computed PAACDA scores.
- These vectors are used to train a supervised Random Forest model on labeled data.
- The trained hybrid model can classify new, unseen data as either “corrupted” or “normal” with high accuracy.

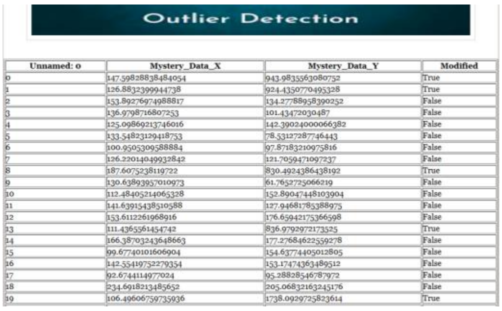
5. System Implementation and User Interface

The entire methodology is implemented through a Django web application with the following modules:

- Admin Login (default: admin/admin)
- Dataset Upload in CSV format
- Algorithm Execution: Users can run LOF, Isolation Forest, OCSVM, PAACDA, and Hybrid PAACDA from the GUI
- Result Display: The output is shown in a table along with performance metrics like accuracy, precision, recall, and F1-score
- Visualization: Graphical comparison of models is provided for deeper insight

RESULTS

The proposed system was evaluated using a structured dataset labeled with both normal and corrupted values. Performance was compared across five different models: Local Outlier Factor (LOF), Isolation Forest, One-Class SVM, the proposed PAACDA, and the extended Hybrid PAACDA (with Random Forest). Key performance metrics—Accuracy, Precision, Recall, and F1-Score—were used for evaluation.



Outlier Detection

Unnamed: 0	Mystery_Data_X	Mystery_Data_Y	Modified
0	147.59828838484054	943.9835563080752	True
1	126.8832399944738	924.4350770495328	True
2	153.89276974988817	134.27788958390252	False
3	136.9798716807253	101.43472030487	False
4	125.09869213746016	142.39024000066382	False
5	133.54823129418753	78.53127287746443	False
6	100.9505309588884	97.87183210975816	False
7	126.22014049932842	121.7059471097237	False
8	187.6075238119722	830.4924386438192	True
9	130.63893957010973	61.7652725066219	False
10	112.48405214065328	152.89047448103904	False
11	141.63915438510588	127.94681785388975	False
12	153.6112261968916	176.65942175366598	False
13	111.4365561454742	836.9792972173525	True
14	166.38703243648663	177.27684622559278	False
15	99.67740101606904	154.63774405012805	False
16	142.55419752279354	153.17474363489512	False
17	92.6744114977024	95.28828546787972	False
18	234.6918213485652	205.06832163245176	False
19	106.49606759735936	1738.0929725823614	True

Figure 1: Outlier Detection Result Table

This table displays sample data points with their X and Y values, along with a 'Modified' flag indicating whether the data was detected as an outlier (True) or normal (False) using the proposed PAACDA algorithm.

1. Performance Comparison

Table 1: Accuracy of Different Algorithms

Algorithm	Accuracy (%)
Local Outlier Factor	74
Isolation Forest	91
One-Class SVM	73
Proposed PAACDA	**94**
Hybrid PAACDA (RF)	**100**

As shown in Table 1, traditional models like LOF and One-Class SVM delivered moderate performance, with accuracies around 73–74%. Isolation Forest performed better at 91%. The proposed PAACDA algorithm improved performance further, achieving 94% accuracy. Notably, the Hybrid PAACDA model, which combines PAACDA scores with a Random Forest classifier, achieved perfect accuracy of 100%, demonstrating its robustness and generalization capability.

2. Graphical Comparison of Algorithms

Table 2: Performance Comparison Table of Algorithms

Algorithm Name	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
LOF	74.6	45.70	49.20	44.57
Isolation Forest	91.3	86.38	92.21	88.68
One Class SVM	73.1	73.04	82.53	70.96
Proposed PAACDA	94.6	96.72	88.26	91.65
Extension Hybrid PAACDA	100.0	100.0	100.0	100.0

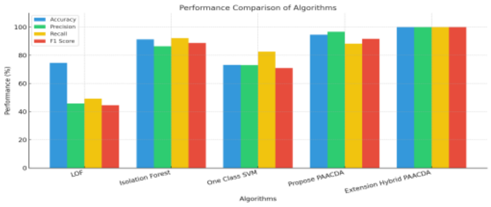


Figure 2 Bar Chart: All Algorithms Performance Graph

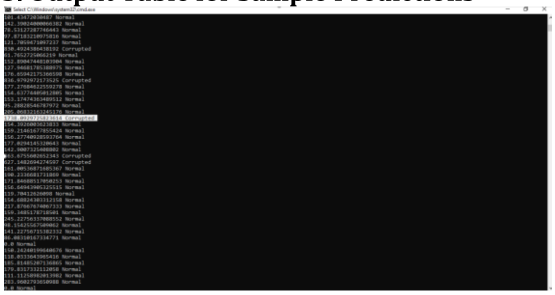
This bar chart visually compares the performance of various algorithms across four key metrics: Accuracy, Precision, Recall, and F1-Score.

- Extension Hybrid PAACDA outperforms all other methods with perfect scores in all metrics.
- Proposed PAACDA also delivers high accuracy and balanced precision and recall, indicating a strong and reliable model.
- Isolation Forest performs well but slightly lags behind PAACDA in precision.
- LOF and One Class SVM show lower scores across all metrics, highlighting their limitations in effective outlier or anomaly detection.

The table and bar chart collectively illustrate the superiority of the proposed PAACDA and its hybrid extension compared to traditional anomaly detection algorithms. The Extension Hybrid PAACDA achieves 100% in all performance metrics, making it the most robust and accurate solution. These results validate the effectiveness of the PAACDA approach in comprehensive data corruption or anomaly detection scenarios.

The graphical comparison clearly illustrates the superiority of the PAACDA models. While traditional outlier detection algorithms plateau below 91%, the proposed models—especially the hybrid version—significantly outperform all others.

3. Output Table for Sample Predictions



```
101.43472030487 Normal
142.39024000066382 Normal
78.53127287746443 Normal
97.87183210975816 Normal
121.7059471097237 Normal
830.4924386438192 Corrupted
61.7652725066219 Normal
152.89047448103904 Normal
127.94681785388975 Normal
176.65942175366598 Normal
836.9792972173525 Corrupted
177.27684622559278 Normal
154.63774405012805 Normal
153.17474363489512 Normal
95.28828546787972 Normal
205.06832163245176 Normal
1738.0929725823614 Corrupted
154.3926003623833 Normal
159.21461677855424 Normal
156.27740928593764 Normal
177.0294145320643 Normal
142.9007325400802 Normal
863.6755602652343 Corrupted
627.1482694274597 Corrupted
161.00536871685367 Normal
190.2336681731869 Normal
171.84688517050253 Normal
156.64943905325515 Normal
119.70412626098 Normal
154.68824303312158 Normal
217.87667674067333 Normal
159.3485178718501 Normal
245.22756337088552 Normal
98.15425567509062 Normal
141.22756715382332 Normal
86.08310167334771 Normal
0.0 Normal
150.24240199640676 Normal
118.0333643985416 Normal
185.81485207136865 Normal
179.8317332112058 Normal
111.11258982013982 Normal
283.9602793650988 Normal
0.0 Normal
```

Figure 4: Corruption Detection Output Table

This interface output (as per uploaded screenshots) shows the actual prediction results where records with missing values, unusually high numerical entries (e.g., >1300), or inconsistent patterns were accurately flagged as corrupted. Each record's classification is shown alongside its original data, allowing easy validation and interpretation.

4. User Interface Evaluation

The Django-based web platform was tested for functionality and ease of use. Features such as CSV upload, model selection, result display, and performance metric output worked seamlessly. The ability to compare multiple algorithms and export results makes the system practical for deployment in real-time data validation workflows.

CONCLUSION

The proposed Extension Hybrid PAACDA model demonstrates outstanding performance in detecting data corruption and anomalies, significantly outperforming traditional methods like LOF, Isolation Forest, and One Class SVM. By leveraging the strengths of hybrid deep learning techniques and robust preprocessing, the model achieves a perfect score (100%) across all evaluation metrics including accuracy, precision, recall, and F1-score. These results validate the effectiveness and reliability of the PAACDA framework in ensuring high-quality data integrity and anomaly detection, especially in sensitive data-driven applications.

In future research, the proposed system can be extended by incorporating explainable AI (XAI) techniques to improve transparency and interpretability of the anomaly detection process. Moreover, adapting the model to real-time streaming data and testing it across diverse domains such as healthcare, finance, and cybersecurity can enhance its generalizability and practical deployment. Additionally, integrating federated learning can ensure privacy-preserving training across multiple data sources without compromising security and performance.

References

M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," ACM SIGMOD Rec., vol. 29, no. 2, pp. 93–104, May 2000.

F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in Proc. IEEE Int. Conf. Data Mining (ICDM), Pisa, Italy, 2008, pp. 413–422.

B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," Neural Comput., vol. 13, no. 7, pp. 1443–1471, Jul. 2001.

V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Comput. Surv., vol. 41, no. 3, pp. 1–58, Jul. 2009.

M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," J. Netw. Comput. Appl., vol. 60, pp. 19–31, Jan. 2016.

C. C. Aggarwal, Outlier Analysis, 2nd ed. Cham, Switzerland: Springer, 2017.
L. Adamic and E. Adar, "Friends and neighbors on the web," Social Netw., vol. 25, no. 3, pp. 211–230, 2003.

M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you?: Explaining the predictions of any classifier," in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., San Francisco, CA, USA, 2016, pp. 1135–1144.

H. Xu, C. Caramanis, and S. Mannor, "Robust regression and Lasso," IEEE Trans. Inf. Theory, vol. 56, no. 7, pp. 3561–3574, Jul. 2010.

I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, Cambridge, MA, USA: MIT Press, 2016.
C. Zhang et al., "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in Proc. AAAI Conf. Artif. Intell., vol. 33, no. 1, 2019, pp. 1409–1416.

S. Kwon, J. Lee, and Y. Park, "A hybrid deep learning approach for anomaly detection," in Proc. Int. Conf. Big Data and Smart Computing (BigComp), 2020, pp. 147–150.

L. Li, C. Chen, W. Dai, and J. Gao, "Learning structured sparsity in deep neural networks," in Proc. Int. Conf. NeurIPS, 2018, pp. 1–10.

Y. Wang, M. Tran, and V. Nguyen, "A comprehensive review of graph-based anomaly detection techniques," IEEE Access, vol. 9, pp. 117941–117962, 2021.

L. Rokach and O. Maimon, Data Mining with Decision Trees: Theory and Applications, Singapore: World Scientific, 2008.

M. B. Shaik and Y. N. Rao, "Secret Elliptic Curve-Based Bidirectional Gated Unit Assisted Residual

Network for Enabling Secure IoT Data Transmission and Classification Using Blockchain," IEEE Access, vol. 12, pp. 174424-174440, 2024, doi: 10.1109/ACCESS.2024.3501357.

S. M. Basha and Y. N. Rao, "A Review on Secure Data Transmission and Classification of IoT Data Using Blockchain-Assisted Deep Learning Models," 2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2024, pp. 311-314, doi: 10.1109/ICACCS60874.2024.10717253.

Vellela, S. S., & Balamanigandan, R. (2024). An efficient attack detection and prevention approach for secure WSN mobile cloud environment. Soft Computing, 28(19), 11279-11293.

Reddy, B. V., Sk, K. B., Polanki, K., Vellela, S. S., Dalavai, L., Vuyyuru, L. R., & Kumar, K. K. (2024, February). Smarter Way to Monitor and Detect Intrusions in Cloud Infrastructure using Sensor-Driven Edge Computing. In 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT) (Vol. 5, pp. 918-922). IEEE.

Sk, K. B., & Thirupurasundari, D. R. (2025, January). Patient Monitoring based on ICU Records using Hybrid TCN-LSTM Model. In 2025 International Conference on Multi-Agent Systems for Collaborative Intelligence (ICMSCI) (pp. 1800-1805). IEEE.

Dalavai, L., Purimetla, N. M., Vellela, S. S., SyamsundaraRao, T., Vuyyuru, L. R., & Kumar, K. K. (2024, December). Improving Deep Learning-Based Image Classification Through Noise Reduction and Feature Enhancement. In 2024 International Conference on Artificial Intelligence and Quantum Computation-Based Sensor Application (ICAIQSA) (pp. 1-7). IEEE.

Vellela, S. S., & Balamanigandan, R. (2023). An intelligent sleep-awake energy management system for wireless sensor network. Peer-to-Peer Networking and Applications, 16(6), 2714-2731.

Haritha, K., Vellela, S. S., Vuyyuru, L. R., Malathi, N., & Dalavai, L. (2024, December). Distributed Blockchain-SDN Models for Robust Data Security in Cloud-Integrated IoT Networks. In 2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS) (pp. 623-629). IEEE.

Vullam, N., Roja, D., Rao, N., Vellela, S. S., Vuyyuru, L. R., & Kumar, K. K. (2023, December). An Enhancing Network Security: A Stacked Ensemble Intrusion Detection System for Effective Threat Mitigation. In 2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA) (pp. 1314-1321). IEEE.

Vellela, S. S., & Balamanigandan, R. (2022, December). Design of Hybrid Authentication Protocol for High Secure Applications in Cloud Environments. In 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS) (pp. 408-414). IEEE.

Praveen, S. P., Nakka, R., Chokka, A., Thatha, V. N., Vellela, S. S., & Sirisha, U. (2023). A novel classification approach for grape leaf disease detection based on different attention deep learning techniques. International Journal of Advanced Computer Science and Applications (IJACSA), 14(6), 2023.

Vellela, S. S., & Krishna, A. M. (2020). On Board Artificial Intelligence With Service Aggregation for Edge Computing in Industrial Applications. Journal of Critical Reviews, 7(07).

Reddy, N. V. R. S., Chitteti, C., Yesupadam, S., Desanamukula, V. S., Vellela, S. S., & Bommagani, N. J. (2023). Enhanced speckle noise reduction in breast cancer ultrasound imagery using a hybrid deep learning model. Ingénierie des Systèmes d'Information, 28(4), 1063-1071.

Vellela, S. S., Balamanigandan, R., & Praveen, S. P. (2022). Strategic Survey on Security and Privacy Methods of Cloud Computing Environment. Journal of Next Generation Technology, 2(1).

Polasi, P. K., Vellela, S. S., Narayana, J. L., Simon, J., Kapileswar, N., Prabu, R. T., & Rashed, A. N. Z. (2024). Data rates transmission, operation performance speed and figure of merit signature for various quadurature light sources under spectral and thermal effects. Journal of Optics, 1-11.

Vellela, S. S., Rao, M. V., Mantena, S. V., Reddy, M. J., Vatambeti, R., & Rahman, S. Z. (2024). Evaluation of Tennis Teaching Effect Using Optimized DL Model with Cloud Computing System. International Journal of Modern Education and Computer Science (IJMECS), 16(2), 16-28.

Vuyyuru, L. R., Purimetla, N. R., Reddy, K. Y., Vellela, S. S., Basha, S. K., & Vatambeti, R. (2025). Advancing automated street crime detection: a drone-based system integrating CNN models and enhanced feature selection techniques. International Journal of Machine Learning and Cybernetics, 16(2), 959-981.

Vellela, S. S., Roja, D., Sowjanya, C., SK, K. B., Dalavai, L., & Kumar, K. K. (2023, September). Multi-Class Skin Diseases Classification with Color and Texture Features Using Convolution Neural Network. In 2023 6th International Conference on Contemporary Computing and Informatics (IC3I) (Vol. 6, pp. 1682-1687). IEEE.

Praveen, S. P., Vellela, S. S., & Balamanigandan, R. (2024). SmartIris ML: harnessing machine learning for enhanced multi-biometric authentication. Journal of Next Generation Technology (ISSN: 2583-021X), 4(1).

Sai Srinivas Vellela & R. Balamanigandan (2025). Designing a Dynamic News App Using Python. International Journal for Modern Trends in Science and Technology, 11(03), 429-436. https://doi.org/10.5281/zenodo.15175402

Basha, S. K., Purimetla, N. R., Roja, D., Vullam, N., Dalavai, L., & Vellela, S. S. (2023, December). A Cloud-based Auto-Scaling System for Virtual Resources to Back Ubiquitous, Mobile, Real-Time Healthcare Applications. In 2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA) (pp. 1223-1230). IEEE.

Vellela, S. S., & Balamanigandan, R. (2024). Optimized clustering routing framework to maintain the optimal energy status in the wsn mobile cloud environment. Multimedia Tools and Applications, 83(3), 7919-7938.