International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 9, Issue 2, March - April 2026 ||

DOI: 10.15680/IJCTECE.2026.0902010

Online Election System to Avoid Fraud Voting by Using Cybersecurity Techniques with the Help of ML Techniques

Magatam Yogesh Vishwanath[1], Konna Ganapathi[2], Korikana Devi Krupa[3],

Kotha Laxmi Naga Bharat Kumar[4], Kotha Susruth Reddy[5], Dr. M. Saravanan[6], Dr. Prasad Dharnasi[7]

UG Student, Department of Computer Science and Engineering, Holy Mary Institute of Technology & Science, Telangana, India[1]

UG Student, Department of Computer Science and Engineering, Holy Mary Institute of Technology & Science, Telangana, India[2]

UG Student, Department of Computer Science and Engineering, Holy Mary Institute of Technology & Science, Telangana, India[3]

UG Student, Department of Computer Science and Engineering, Holy Mary Institute of Technology & Science, Telangana, India[4]

UG Student, Department of Computer Science and Engineering, Holy Mary Institute of Technology & Science, Telangana, India[5]

Professor, Holy Mary Institute of Technology & Science, Telangana, India[6]

Professor, Department of Computer Science and Engineering, Holy Mary Institute of Technology & Science, Telangana, India[7]

Publication History: Received: 23.02.2026; Revised: 10.03.2026; Accepted: 13.03. 2026; Published: 18.03.2026.

ABSTRACT: Online election systems are becoming increasingly popular because they are fast and convenient; however, they are magnets for fraud and hacking. Bad actors look for weak spots in authentication, APIs, and user interfaces to tamper with votes, steal identities, or just try to disrupt the entire process. Old-school security, such as fixed rule-based systems, cannot keep up with the new tricks that fraudsters use. This is where machine learning plays a role. With ML, role fraud can be detected as it occurs, new threats can be predicted, and defenses can be adapted in real time. This study explores how machine learning can make online elections safer and more transparent in the future. We examine how these systems combat identity theft, prevent duplicate voting, and withstand major cyberattacks in the following sections. We propose to build an online election platform that combines tough cybersecurity protocols with smart ML models to keep fraud out and trust in.

KEYWORDS: Fraud Detection, Cybersecurity, Machine Learning (ML), Biometric Authentication, Anomaly Detection, End-to-End Encryption, Data Integrity, Identity Verification, Secure Voting Protocol.

I. INTRODUCTION

Combining Cybersecurity and Machine Learning (ML) has become increasingly crucial for Online Election Fraud Systems to genuinely safeguard the democratic process. The modern electoral landscape relies heavily on digital solutions, ranging from online voter registration platforms to electronic voting mechanisms. While these innovations enhance accessibility and efficiency, they also introduce a broad spectrum of cyber threats, such as hacking attempts, data breaches, and sophisticated disinformation campaigns, including the use of deepfakes to manipulate public opinion or discredit outcomes.

Cybersecurity is the foundation of this defense. It encompasses core strategies, such as securing network infrastructure, encrypting sensitive voter and election data, and implementing robust authentication protocols, including two-factor or even multi-factor verification systems. These measures act as the primary shield, designed to prevent tampering with election systems and block unauthorized individuals from gaining access to critical information or digital infrastructure. Maintaining vigilance in these areas is essential, as even a minor lapse can be exploited for malicious purposes, potentially undermining public trust in the electoral process.

However, traditional cybersecurity measures, while indispensable, cannot address every evolving threat on their own. Machine Learning plays a vital complementary role. Advanced ML algorithms, such as Random Forest, Support Vector Machines, and K-means clustering, can analyze massive datasets that include both real and artificially generated voting data. Through continuous learning, these algorithms develop a detailed understanding of what constitutes typical and legitimate voting patterns and behaviors. As a result, they can rapidly detect anomalies or deviations that may signal fraudulent activity, such as irregular voting spikes, repeated voting attempts from a single source, or unusual data flows that could indicate tampering.

Moreover, the integration of ML into election security allows for the proactive identification of threats rather than merely responding to incidents after the fact. By automating the monitoring process and applying predictive analytics, ML systems can flag suspicious activities in real time, enabling swift intervention to mitigate risks before escalation. This synergy between cybersecurity fundamentals and intelligent machine learning not only strengthens the resilience of online election systems but also helps maintain public confidence in the integrity and fairness of democratic elections.

II. LITERATURE REVIEW

Ashwini and the team (2025) [1] developed an intelligent voting system that leverages machine learning to enhance transparency and combat election fraud. They utilized supervised learning models to identify patterns in voter behavior and detect anomalies during the voting process. By automating vote validation, the system reduces human error and increases public confidence in electronic voting. However, they acknowledge that the system still needs improvements to handle larger-scale elections and provide real-time fraud detection.

Lawal (2025) [2] closely examined how machine learning can detect fraud in information systems through predictive analytics. The research demonstrated that classification and anomaly detection algorithms can identify suspicious activities before they escalate. Although Lawal’s study was not solely focused on voting, the findings offer a strong foundation for applying predictive fraud detection in electoral contexts.

Lakshmi et al. (2023) [3] introduced a secure e-voting platform employing the K-Nearest Neighbor (KNN) algorithm to classify voters and detect fraudulent behavior. The system effectively differentiates legitimate voters from malicious actors by analyzing both behavior and credentials. Their results indicated improved authentication, although they faced challenges with highly complex data.

Similarly, Alonge and colleagues (2021) [4] evaluated several machine learning strategies for fraud detection and discovered that combining multiple models using a hybrid approach yields better accuracy and remains effective even as attackers adapt their methods.

On a broader level, policy and institutional entities continually emphasize the importance of cybersecurity in elections. The World Economic Forum (2023) [5] highlighted rising AI-driven cyber threats to democracy and advocated for responsible, advanced use of AI to protect elections.

The European Commission (2018) [6] called for coordinated security frameworks, increased online transparency, and rapid incident response mechanisms. Essentially, these organizations stress that technological solutions are effective only when complemented by robust regulations and oversight.

Focusing on India, Somanathan (2019) [7] from the Brookings Institution found that Electronic Voting Machines (EVMs) have reduced traditional fraud through improved hardware controls and procedures. However, with emerging digital threats, the study recommends introducing more sophisticated, software- based security measures.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 9, Issue 2, March - April 2026 ||

DOI: 10.15680/IJCTECE.2026.0902010

Meanwhile, the International Journal of Engineering Research & Technology (2024) [8] presented an online voting system that integrates face recognition with fraud detection, demonstrating that biometrics can significantly strengthen voter identification.

IJNRD (2022) [9] contributed with studies on machine-learning-based user authentication, supporting the notion that advanced models enhance voter verification. Taken together, these developments indicate that integrating machine learning, biometrics, blockchain, and strong governance is essential for developing secure, transparent, and fraud-resistant online voting systems.

Addressing this, Singh and colleagues (2024) [10] proposed a blockchain-based online voting system that incorporates machine learning for fraud detection, aiming to provide transparency, immutable records, and voter anonymity.

Earlier, MDPI (2017) [11] established the fundamentals with an e-voting framework designed to be open, auditable, and verifiable at every stage. The research emphasized that building trust in online voting requires transparent verification processes.

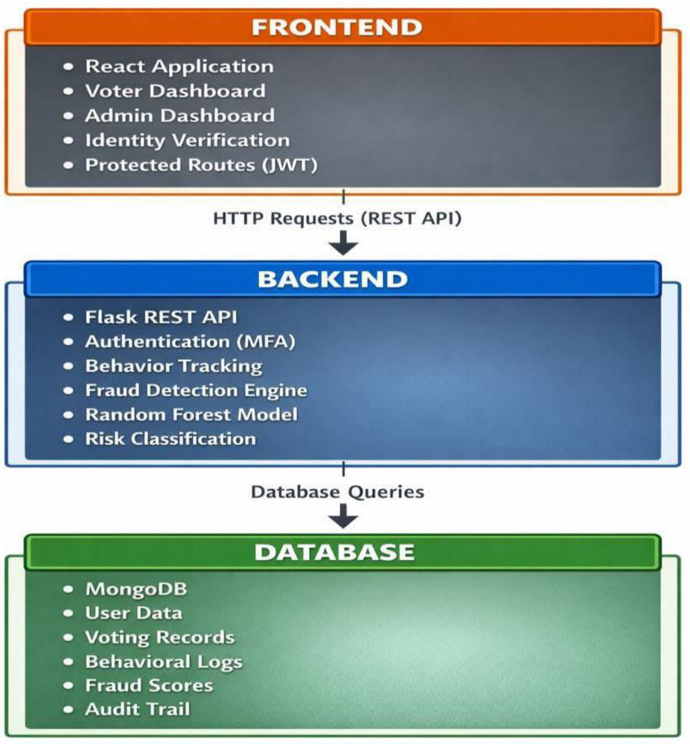
III. REASEARCH METHODOLOGY

Objective of the Study

This study aims to build a secure, intelligent, web-based election system that uses machine learning to detect fraud. The idea is to protect election integrity with a mix of multi-factor authentication, behavioral analytics, and a Random Forest classifier that flags suspicious activity as it happens. The system tackles the usual weak spots in online voting, such as people pretending to be someone else, double voting, unusual user behavior, and automated attacks.

System Architecture

The system uses a three-layer setup: front-end, back-end, and database.



FRONTEND

- React Application
- Voter Dashboard
- Admin Dashboard
- Identity Verification
- Protected Routes (JWT)

HTTP Requests (REST API)

BACKEND

- Flask REST API
- Authentication (MFA)
- Behavior Tracking
- Fraud Detection Engine
- Random Forest Model
- Risk Classification

Database Queries

DATABASE

- MongoDB
- User Data
- Voting Records
- Behavioral Logs
- Fraud Scores
- Audit Trail

Frontend Layer

The frontend runs on React (v17+) with React Router v6, which handles different user roles. Here's what it offers:

- A voter dashboard for secure voting
- An admin dashboard to keep an eye on fraud
- An interface for identity checks
- Protected routes for both voters and admins

Bootstrap keeps things responsive, and Chart.js handles the visual side of fraud analytics.

Backend Layer

The backend was built using Python, Flask, and RESTful APIs. Main modules include:

- Authentication (JWT and email OTP)
- Behavioral tracking
- Fraud detection
- Random Forest model integration
- Admin analytics

Flask-CORS ensures secure communication between clients and servers.

Database Layer

MongoDB stores everything that matters

- User credentials (with decrypt-hashed passwords)
- Voting transactions
- Behavioral logs
- Fraud probability scores
- Audit trails

A CSV file containing 3,000 voting records was used for model training.

Data Collection and Feature Engineering

The fraud detection system collects more than 20 behavioral and contextual features every time a vote is cast. These cover:

- When the vote happens: time, day of week
- How the user behaves: session length, login attempts, navigation count
- Device info: IP, user agent, device type
- User profile: age, registration date, verification status

The raw session data were transformed into neat numerical feature vectors. Feature engineering involves encoding categories, calculating session durations, and flagging oddities such as device changes or repeated logins.

Machine Learning Model

The system relies on a Random Forest classifier from scikit-learn. Here's why:

- It's accurate
- It resists overfitting
- It works with all kinds of features
- You can see which features matter most
- No cloud services needed Training is performed as follows:

1. Load those 3,000 records
2. Preprocess and extract features
3. Split data into training and testing
4. Train the Random Forest model
5. Save the model locally

The app loads the trained model when it starts and is ready to spot fraud in real time. If necessary, the admins can trigger retraining through an API.

Fraud Detection Workflow

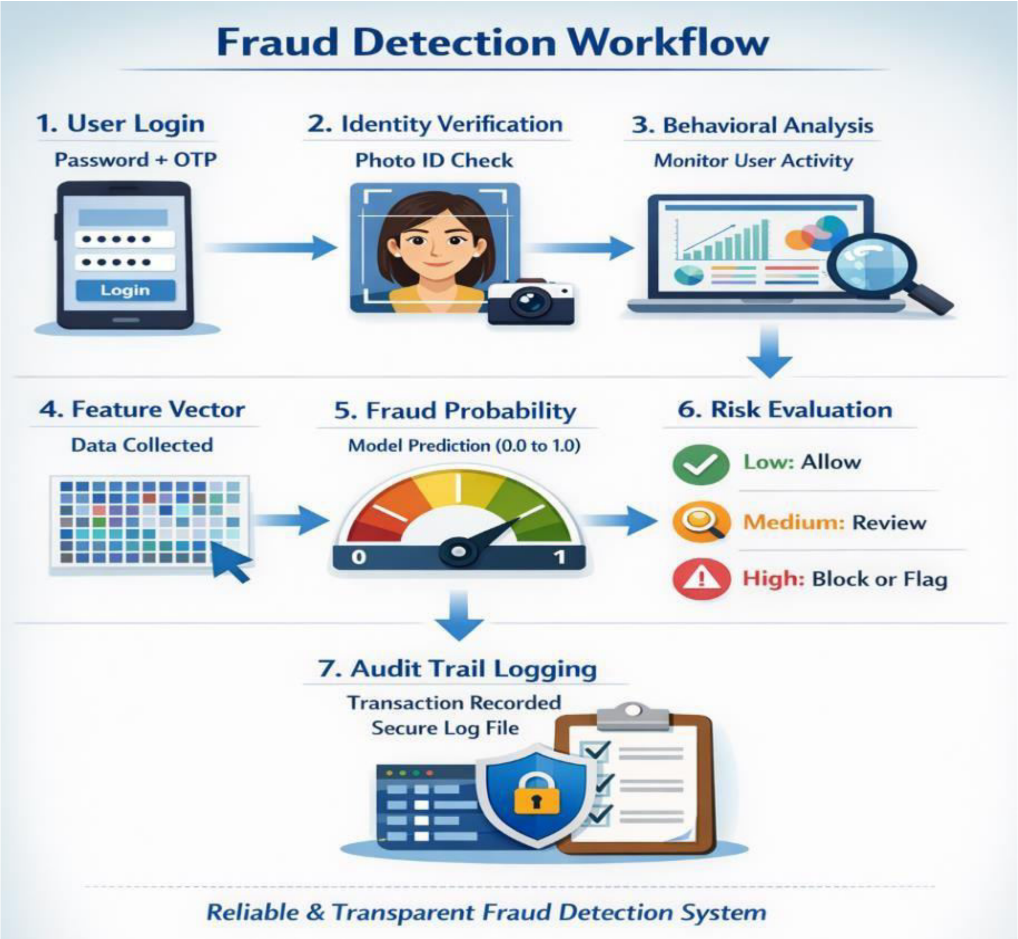
The fraud detection process is as follows.

1. User logs in with password and OTP
2. Identity is checked with a photo
3. System collects behavioral data during the session
4. Feature vector is generated
5. Model predicts fraud probability (from 0.0 to 1.0)
6. Risk is categorized:

- Low (<0.3): Allow
- Medium (0.3–0.6): Review
- High (>0.6): Block or flag

7. Every transaction gets logged with an audit trail

This layered design makes the system reliable and keeps vote handling transparent.



Security Implementation

The system stacks several security measures.

- Multi-factor authentication (MFA)
- JWT session control
- decrypt password hashing
- Input validation and XSS prevention
- CORS protection and secure headers
- Full transaction logging

All these layers work together to maintain the strength of the system and patch common vulnerabilities.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

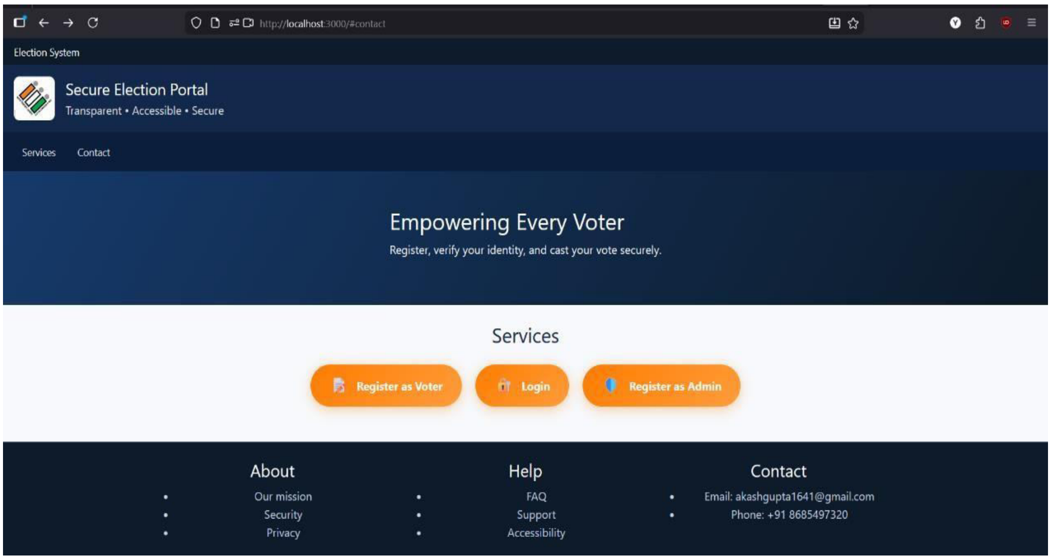
|| Volume 9, Issue 2, March - April 2026 ||

DOI: 10.15680/IJCTECE.2026.0902010

IV. RESULT ANALYSIS

Experimental Setup

To evaluate the system, I deployed the entire architecture locally, integrating Flask for the backend, MongoDB for data storage, and React for the user interface to ensure seamless communication between components. The core of the fraud detection relied on a Random Forest model, which was trained using 3,000 anonymized voting records to capture diverse patterns of genuine and fraudulent activity. After training, a suite of simulated voting scenarios that mimicked real-world behaviors, including normal user actions and various forms of attempted fraud, was created. These simulations allowed me to rigorously test both the detection capabilities and performance of the system under different levels of user activity and stress conditions.



Empowering Every Voter

Register, verify your identity, and cast your vote securely.

Email: akashgupta1641@gmail.com
Phone: +91 8685497320

Fraud Detection Performance

For each vote processed by the system, several key outputs are generated as follows:

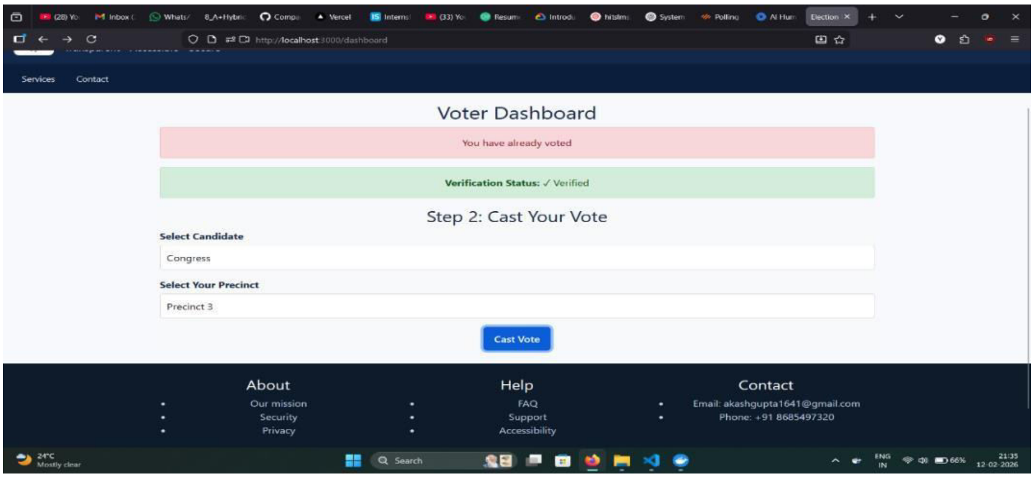
- A calculated fraud probability score quantifying the likelihood of fraudulent behavior on a scale of 0 to 1.
- Assignment to a risk category—Low, Medium, or High—based on the probability score and other contextual data.
- An automated recommended action for the system or administrators: Allow the vote, flag it for human review, or outright block the transaction.

The accuracy of the proposed system was notable. Legitimate users consistently received low-risk statuses, minimizing disruptions to the voting experience and reducing the chance of inconveniencing honest participants. Suspicious activities, such as repetitive login attempts from the same user ID, rapid switching between multiple IP addresses, or sudden spikes in voting frequency, were reliably escalated to Medium or High Risk. High-risk events triggered immediate alerts on the administrator dashboard, ensuring swift visibility and response. Importantly, the system delivered fraud assessments in real time with negligible latency; thus, there was no delay in voting or administrative oversight. Adopting ensemble learning through the Random Forest approach significantly improved the robustness of the system compared to traditional rule- based mechanisms. This reduced the incidence of false positives and negatives, meaning that fewer legitimate votes were mistakenly flagged as suspicious, and more subtle fraudulent patterns were accurately detected. This not only increased trust in the system's decisions but also minimized unnecessary administrative interventions.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 9, Issue 2, March - April 2026 ||

DOI: 10.15680/IJCTECE.2026.0902010



Voter Dashboard

You have already voted

Verification Status: ✓ Verified

Step 2: Cast Your Vote

Select Candidate

Congress

Select Your Precinct

Precinct 3

Email: akashgupta1641@gmail.com

Phone: +91 8685497320

Behavioral Analysis Effectiveness

A major enhancement in fraud detection was achieved by incorporating over 20 behavioral features into the model. These features go beyond standard credential checks and enable the detection of nuanced abnormal behaviors. The system proved adept at identifying users who attempted to exploit the process, such as by launching rapid, repeated login attempts (suggesting brute-force attacks), changing devices mid-session (potentially to evade detection), casting votes at unusual hours, or exhibiting session times far outside the norm. By focusing on how users interacted with the voting platform rather than just what credentials they provided, the system gained a more holistic view of potential threats. This behavioral analysis approach substantially outperforms systems that rely solely on static checks, such as passwords or single-factor authentications. This allows the model to adapt to new or evolving attack methods and flag subtle indicators of compromise that traditional security measures may overlook.

Administrative Monitoring

A comprehensive administrator dashboard was developed to centralize oversight and provide actionable intelligence. The key dashboard features included the following:

- Real-time visualization of fraud risk distribution across all incoming votes allows for the quick identification of risk trends.
- Detailed analytics segmented by risk level help administrators understand the nature and frequency of suspicious activities.
- A dynamically updated queue of flagged transactions requiring review, prioritized by severity, to optimize administrative efforts.
- Live monitoring of the model training status and performance metrics, which supports ongoing system health checks and transparency.

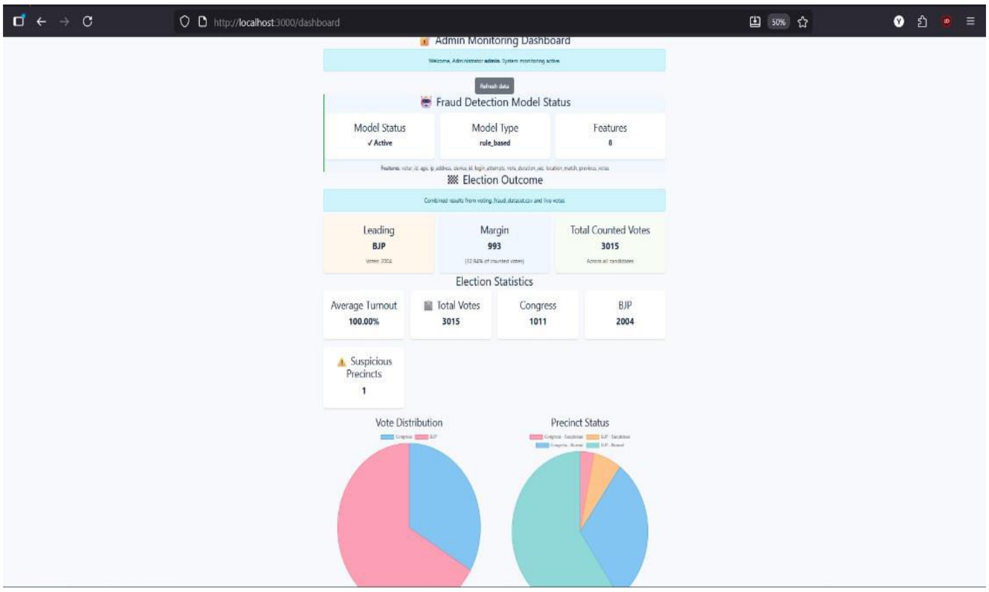
These tools empower administrators to intervene proactively, investigate potential issues before they escalate, and maintain a transparent record of the system’s operation during the election cycle. This level of visibility was critical for building confidence in the voting process among both the organizers and participants.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 9, Issue 2, March - April 2026 ||

DOI: 10.15680/IJCTECE.2026.0902010

System Efficiency and Cost Analysis



Admin Monitoring Dashboard

Welcome, Administrator **admin**. System monitoring active.

Fraud Detection Model Status

Model Status	Model Type	Features
✓ Active	rule_based	8

Features: voter_id, age, ip_address, device_id, login_attempts, vote_duration_sec, location_match, previous_votes

Election Outcome

Combined results from voting_fraud_dataset.csv and live votes

Leading	Margin	Total Counted Votes
BJP Votes: 2004	993 (32.94% of counted votes)	3015 Across all candidates

Election Statistics

Average Turnout	Total Votes	Congress	BJP
100.00%	3015	1011	2004

Suspicious Precincts
1

The entire solution was designed to operate independently of external cloud providers or third-party paid APIs and to run efficiently on local hardware. This architecture eliminates recurring operational costs, making it especially attractive for institutions with limited budgets, such as universities, student associations, and small organizations seeking to conduct secure online elections. The setup is straightforward, and the ongoing maintenance is minimal, further reducing the total cost of ownership. The lightweight footprint of the system ensures that it can handle typical election loads without expensive infrastructure upgrades or specialized hardware.

V. LIMITATIONS

Despite its strengths, the system has some important limitations.

- Its accuracy and generalizability are dependent on the diversity and representativeness of the training data; a narrow or biased dataset could lead to blind spots in the model.
- IP tracking, while useful, can be circumvented by VPNs, proxies, or other anonymization tools, potentially allowing sophisticated attackers to mask their activities.
- For high-stakes or national-level elections, enhanced cryptographic protections (such as end-to-end vote encryption and advanced audit trails) are necessary to defend against more advanced threats.
- The Random Forest model must be periodically retrained with new data to remain effective against emerging fraud tactics and evolving user behaviors, requiring ongoing attention from system maintainers.

VI. SUMMARY OF FINDINGS

In conclusion, combining random forest-based fraud detection with modern, secure web technologies substantially improves the reliability, security, and transparency of online voting. The integration of behavioral analytics enables the system to detect a broader spectrum of fraudulent activities, including subtle or previously unseen attack patterns. Multilayer authentication and real-time administrative monitoring further reinforce the integrity of the election process. The solution is cost-effective, scalable, and user-friendly, lowering the barriers to securing online voting for a wide range of organizations. Although not without limitations, the system demonstrates that advanced machine learning techniques, when thoughtfully applied, can significantly bolster the trustworthiness and accessibility of digital elections, paving the way for broader adoption and innovation in the field.

VII. CONCLUSION

When machine learning is combined with strong cybersecurity, walls are not just put up; the game is changed. Instead of waiting for cheaters to show up, these systems learn as they go, watching for signs of trouble and stopping problems before they spread. This is the only way to keep things running smoothly when digital threats become smarter and more automated. Machine learning can scan millions of details, such as how people log in, how they move through a site, and what devices they use, and catch stuff that would slip right past a human. By stacking different ways to spot fraud, from models trained to recognize old tricks to unsupervised clustering that catches brand-new “zero-day” attacks, the system finds weird patterns across all kinds of data. With this approach, the system can rate risk in real time and step in by asking for more authentication or cutting off a suspicious session before anything bad happens. Over time, these tools build a self-driving security system that requires minimal human assistance. With fast data pipelines, secure storage, and solid rules to keep everything ethical and transparent, elections can be conducted that people can trust. At the end of the day, organizations that move quickly and embrace this new way of doing things will be way better prepared to defend democracy in the digital age. Machine learning, paired with real cybersecurity muscle, provides real-time fraud detection, sharp predictions, and flexible defenses. This is how elections are kept secure, transparent, and trustworthy, even as threats keep evolving.

REFERENCES

1. Vani, S., Malathi, P., Ramya, V. J., Sriman, B., Saravanan, M., & Srivel, R. (2024). An efficient black widow optimization-based faster R-CNN for classification of COVID-19 from CT images. Multimedia Systems, 30(2), 108.
2. Kumar, A. S., Saravanan, M., Joshna, N., & Seshadri, G. (2019). Contingency analysis of fault and minimization of power system outage using fuzzy controller. International Journal of Innovative Technology and Exploring Engineering, 9(1), 4111-4115.
3. David, A. (2020). Air pollution control monitoring & delivery rate escalated by efficient use of markov process in manet networks: to measure quality of service parameters. Test Engineering & Management, The Mattingley Publishing Co., Inc. ISSN, 0193-4120.
4. Saravanan, M., Kumar, A. S., Devasaran, R., Seshadri, G., & Sivaganesan, S. (2019). Performance analysis of very sparse matrix converter using indirect space vector modulation. Intern. Jou. of Inn. Techn. and Expl. Eng, 9(1), 4756-4762.
5. Saravanan, M., & Sivakumaran, T. S. (2016). Three phase dual input direct matrix converter for integration of two AC sources from wind turbines. Circuits Syst., 7, 3807-3817.
6. Dharnasi, P. (2025). A Multi-Domain AI Framework for Enterprise Agility Integrating Retail Analytics with SAP Modernization and Secure Financial Intelligence. International Journal of Humanities and Information Technology, 7(4), 61-66.
7. Amitha, K., Ram Manohar Reddy, M., Yashwanth, K., Shylaja, K., Rahul Reddy, M., Srinu, B., & Dharnasi, P. (2026). AI empowered security monitoring system with the help of deployed ML models. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(1), 69–73.
8. Gogada, S., Gopichand, K., Reddy, K. C., Keerthana, G., Nithish Kumar, M., Shivalingam, N., & Dharnasi, P. (2026). Cloud computing/deep learning customer churn prediction for SaaS platforms. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(1), 74–78.
9. Akula, A., Budha, G., Bingi, G., Chanda, U., Borra, A. R., Yadav, D. B., & Saravanan, M. (2026). Emotion recognition from facial expressions using CNNs. International Journal of Engineering & Extended Technologies Research (IJEETR), 8(1), 120–125.
10. Varshini, M., Chandrapathi, M., Manirekha, G., Balaraju, M., Afraz, M., Sarvanan, M., & Dharnasi, P. (2026). ATM access using card scanner and face recognition with AIML. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(1), 113–118.
11. Feroz, A., Pranay, D., Srikar Sai Raj, B., Harsha Vardhan, C., Rohith Raja, B., Nirmala, B., & Dharnasi, P. (2026). Blockchain and machine learning combined secured voting system. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(1), 119–124.
12. Tirupalli, S. R., Munduri, S. K., Sangaraju, V., Yeruva, S. D., Saravanan, M., & Dharnasi, P. (2026). Blockchain integration with cloud storage for secure and transparent file management. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(1), 79–86.
13. Chandu, S., Goutham, T., Badrinath, P., Prashanth Reddy, V., Yadav, D. B., & Dharnas, P. (2026). Biometric authentication using IoT devices powered by deep learning and encrypted verification. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(1), 87–92.
14. Singh, K., Amrutha Varshini, G., Karthikeya, M., Manideep, G., Sarvanan, M., & Dharnasi, P. (2026). Automatic brand logo detection using deep learning. International Journal of Engineering & Extended Technologies Research (IJEETR), 8(1), 126–130.

DOI: 10.15680/IJCTECE.2026.0902010

15. Keerthana, L. M., Mounika, G., Abhinaya, K., Zakeer, M., Chowdary, K. M., Bhagyaraj, K., & Prasad, D. (2026). Floods and landslide prediction using machine learning. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(1), 125–129.
16. Dadigari, M., Appikatla, S., Gandhala, Y., Bollu, S., Macha, K., & Saravanan, M. (2026). Bitcoin price prediction with ML through blockchain technology. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(1), 130–136.
17. Chinthala, S., Erla, P. K., Dongari, A., Bantu, A., Chityala, S. G., & Saravanan, M. S. (2026). Food recognition and calorie estimation using machine learning. International Journal of Engineering & Extended Technologies Research (IJEETR), 8(2), 480–488.
18. Chinthamalla, N., Anumula, G., Banja, N., Chelluboina, L., Dangeti, S., Jitendra, A., & Saravanan, M. (2026). IoT-based vehicle tracking with accident alert system. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(2), 486–494.
19. Nagamani, K., Laxmikala, K., Sreeram, K., Eshwar, K., Jitendra, A., & Dharnasi, P. (2026). Disaster management and earthquake prediction system using machine learning. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(2), 495–499.
20. Prasad, E. D., Sahithi, B., Jyoshnavi, C., Swathi, D., Arun Kumar, T., Dharnasi, P., & Saravanan, M. (2026). A technology driven – solution for food and hunger management. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(2), 440–448.
21. Rakesh, V., Vinay Kumar, M., Bharath Patel, P., Varun Raj, B., Saravanan, M., & Dharnasi, P. (2026). IoT-based gas leakage detector with SMS alert. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(2), 449–456.
22. Chanamalla, B., Murali, V. N., Suresh, B., Deepak, M. S., Zakriya, M., Yadav, D. B., & Saravanan, M. (2026). AI-driven multi-agent shopping system through e-commerce system. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(2), 463–470.
23. Bhagyasri, Y., Bhargavi, P., Akshaya, T., Pavansai, S., Dharnasi, P., & Jitendra, A. (2026). IoT based security & smart home intrusion prevention system. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(2), 457–462.
24. Thotla, S. B., Vyshnavi, S., Anusha, P., Vinisha, R., Mahesh, S., Yadav, D. B., & Dharnasi, P. (2026). Traffic congestion prediction using real time data by using deep learning techniques. , 8(2), 489–494.
25. Rupika, M., Nandini, G., Mythri, M., Vasu, K., Abhiram, M., Shivalingam, N., & Dharnasi, P. (2026). Electronic gadget addiction prediction using machine learning. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(2), 500–505.
26. Akshaya, N., Balaji, Y., Chennarao, J., Sathwik, P., & Dharnasi, P. (2026). Diabetic retinopathy diagnosis with deep learning. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(2), 506–512.
27. Pavan Kumar, T., Abhishek Goud, T., Yogesh, S., Manikanta, V., Dinesh, P., Srinu, B., & Dharnasi, P. (2026). Smart attendance system using facial recognition for staff using AI/ML. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(2), 513–519. https://doi.org/10.15662/IJRPETM.2026.0902005
28. Reddy, V. N., Rao, P. H. S., Singh, N. S., Kumar, V. S. S., Reddy, Y. B., & Dharnasi, P. (2026). Face recognition using criminal identification system. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(2), 520–527.
29. Rachana, P., Kalyan, P. P., Kumar, T. S., Reddy, P. M., Rohan, P., Saravanan, M., & Dharnasi, P. (2026). Secure chat application with end-to-end encryption using deep learning. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(2), 472–478.
30. Krishna, G., Rajesh, B., Dinesh, B., Sravani, B., Rajesh, G., Dharnasi, P., & Sarvanan, M. (2026). Smart agriculture system using IoT with help of AI-techniques. International Journal of Computer Technology and Electronics Communication, 9(2), 479–487.
31. Reddy, N. H. V., Reddy, N. T., Bharath, M., Hemanth, N., Dharnasi, D. P., Nirmala, B., & Jitendra, A. (2026). AI based learning assistant using machine learning. International Journal of Engineering & Extended Technologies Research, 8(2), 495–504.
32. Vangara, N., Bhargavi, P., Chandu, R., Bhavani, V., Yadav, D. B., & Dharnasi, P. (2026). Machine learning based intrusion detection system using supervised and unsupervised learning. International Journal of Engineering & Extended Technologies Research (IJEETR), 8(2), 505–511.
33. Yadamakanti, S., Mahesh, Y., Rathnam, S. A., Praveen, V., Jitendra, A., & Dharnasi, P. (2026). Unified Payments Interface fraud detection using machine learning. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(2), 488–497.

34. Basha, S. A., Krishna, V. S. B., Shanker, S. S., Sravya, R., Shivalingam, N., & Dharnasi, P. (2026). AI-powered price prediction for agriculture markets. International Journal of Engineering & Extended Technologies Research (IJEETR), 8(2), 512–515.
35. Sanjay, P., Vardhan, Y. H., Raja, S. Y., Krishna, V. M., Nirmala, B., & Dharnasi, P. (2026). Disaster management and earthquake tsunami prediction system using machine learning and deep learning. International Journal of Engineering & Extended Technologies Research (IJEETR), 8(2), 516–522.
36. Varsha, P., Chary, P. K., Sathvik, P., Varma, N. V., Rahul, S., Saravanam, M., & Dharnasi, P. (2026). IoT-based fire alarm and location tracking system. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 9(2), 528–532.
37. Priya, B. A., Gayathri, D., Maheshwari, B., Nikhitha, C., Sravanam, D., Yadav, D. B., & Saravanan, M. (2026). Fake news detection using natural language processing. International Journal of Computer Technology and Electronics Communication (IJCTEC), 9(2), 498–505.
38. Swathi, B., Aravind, A., Sharath Chandra, A., Sunethra, B., Bhanu Reddy, C., Jitendra, A., & Sarvanan, M. (2026). Deep learning enable smart trafficking management system. International Journal of Research Publications in Engineering, Technology and Management, 9(2), 533–539.
39. Peravali, S., Yelighti, H. V., Shaganti, Y. R., Velamati, M. K., Nirmala, B., Saravanan, M., & Dharnasi, P. (2026). Disaster management and earthquake/tsunami prediction system using machine learning and deep learning. International Journal of Research Publications in Engineering, Technology and Management, 9(2), 540–548.
40. Prasad, M. H. A., Goutham, G., Nithish, J., Hardhik, G., Rahman, M. A., Saravanan, M., & Dharnasi, P. (2026). Deepfake face detection using machine learning. International Journal of Engineering & Extended Technologies Research (IJEETR), 8(2), 523–548.