iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.82032

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Fake News Detection and Recommendation System Using BERT and Federated Learning

Adithya A. A[1], Goutham Krishnan[2], Hana Habeeb Rahman[3], Sylasree V. A[4], Reshma P. D[5]
Dept. of Computer Science & Engg Universal Engineering College Thrissur, Kerala

Abstract: ***The rapid spread of misinformation on digital plat-forms necessitates effective news verification systems. This project presents a Hybrid Fake News Detection System that combines BERT-based text classification with semantic similarity analysis using Sentence-BERT (SBERT). The BERT model analyzes linguistic patterns to classify news as real or fake, while SBERT compares the news with content obtained through web scraping from trusted sources to validate its authenticity. A combined decision mechanism improves overall detection accuracy.***

The system also incorporates federated learning to enhance news recommendation while preserving user privacy by avoiding centralized data sharing. Additionally, it provides social features such as posting, liking, commenting, and following, along with an admin module for monitoring and controlling misinformation. The application is developed using HTML, CSS, and JavaScript for the frontend and Python Django for the backend. This approach improves both reliability and privacy compared to traditional methods.

Index Terms: Fake News Detection, BERT, Sentence-BERT (SBERT), Semantic Similarity, Web Scraping, Federated Learn-ing, Deep Learning, Natural Language Processing, Django, HTML CSS JavaScript

I. INTRODUCTION

With the rapid growth of digital media and social network-ing platforms, the spread of misinformation and fake news has become a significant global concern. Fake news can influence public opinion, create social unrest, and mislead individuals, making it essential to develop reliable systems for automatic news verification. Traditional approaches to fake news detec-tion relied on manual verification or basic rule-based methods, which are time-consuming and often ineffective for large-scale data.

Recent advancements in deep learning and Natural Lan-guage Processing (NLP) have enabled automated detection of fake news using contextual understanding of text. In particular, transformer-based models such as BERT have shown strong performance in distinguishing real and fake news by analyzing linguistic patterns and contextual relationships within text [1]. Additionally, deep learning techniques have been widely applied in various detection systems, demonstrating their ef-fectiveness in handling complex data patterns [14]–[16].

However, relying solely on content-based classification is not sufficient, as fake news can sometimes mimic real news writing styles. To address this limitation, semantic similarity techniques such as Sentence-BERT (SBERT) are used to compare user-submitted news with real-time news content obtained from trusted online sources through web scraping. This allows the system to validate whether similar news exists in reliable media, thereby improving detection accuracy.

In this paper, we propose a Hybrid Fake News Detection System that combines AI-based classification with seman-tic similarity verification. The system integrates BERT for content analysis and SBERT for real-world validation, us-ing a combined decision mechanism to improve reliability. Furthermore, federated learning is incorporated to provide personalized news recommendations while preserving user privacy. The platform also includes social interaction features and an administrative module for monitoring and controlling misinformation.

The proposed system is implemented using a web-based frontend developed with HTML, CSS, and JavaScript, and a backend powered by Python Django, with machine learning models integrated using PyTorch. This approach aims to provide an efficient, scalable, and user-centric solution for combating fake news in modern digital environments.

II. RELATED WORKS

The problem of fake news detection has gained significant attention in recent years, leading to the development of various machine learning and deep learning-based approaches. This section reviews key contributions relevant to our work, including text classification, deep learning-based detection systems, and semantic analysis techniques.

©IJRASET: All Rights are Reserved

A. Fake News Detection using Deep Learning

Early approaches to fake news detection relied on traditional machine learning techniques and manual feature extraction. However, with the advancement of deep learning, more so-phisticated models have been developed to automatically learn complex patterns from textual data. Transformer-based models such as BERT have demonstrated strong performance in iden-tifying fake news by capturing contextual relationships and linguistic features within text [1]. Additionally, deep neural networks and convolutional architectures have been success-fully applied in various classification tasks, highlighting their effectiveness in handling large-scale and complex datasets [14]–[16].

B. Detection Systems and AI-based Analysis

Deep learning has been widely used in detection systems across multiple domains, including anomaly detection and event recognition. Several studies have demonstrated the effec-tiveness of AI-based models in identifying complex patterns and behaviors in data [7], [13]. These approaches emphasize the importance of automated systems for real-time analysis and decision-making, which can be extended to fake news detection scenarios. Such systems inspire the development of intelligent frameworks that can process large volumes of data efficiently.

C. Semantic Similarity and Content Verification

While classification models are effective, they may fail when fake news closely resembles real news in writing style. To overcome this limitation, semantic similarity techniques have been introduced. Sentence-BERT (SBERT) enables com-parison of textual meaning by converting sentences into embeddings and measuring similarity scores. This approach allows verification of news by comparing it with trusted sources, improving detection reliability. Semantic analysis plays a crucial role in validating information beyond surface-level features.

D. Privacy-Preserving and Advanced Learning Approaches

Recent research has also explored privacy-preserving tech-niques such as federated learning, which allows models to learn from decentralized data without sharing sensitive user information. This is particularly important in applications involving user-generated content, where privacy and data secu-rity are critical concerns. Integrating such techniques enhances both scalability and user trust in intelligent systems.

E. Motivation

Although existing approaches provide strong foundations for fake news detection, most systems rely solely on either content-based classification or external validation methods. Few systems effectively combine both approaches to improve accuracy and reliability. In this work, we address this gap by proposing a hybrid framework that integrates BERT-based classification with SBERT-based semantic similarity and real-time web data, providing a more robust and comprehensive solution for fake news detection.

III. SYSTEM ARCHITECTURE

The proposed system follows a modular client-server archi-tecture consisting of two primary modules: the User Module and the Admin Module. The frontend is developed using HTML, CSS, and JavaScript, while the backend is imple-mented using Python Django, integrating machine learning models for fake news detection.

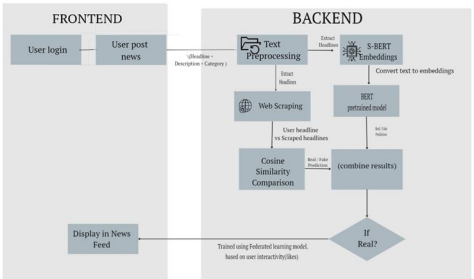


Fig. 1. Hybrid Fake News Detection System Architecture and Data Flow

A. User Module

The user module provides functionalities such as registra-tion, login, news posting, and interaction with news content. Users can submit news articles in the form of headline, description, and category. These inputs are forwarded to the backend for processing. Users can also view news in the feed, like or comment on posts, and follow other users.

B. Backend Architecture (Django AI Models)

The backend acts as the core processing unit, handling user requests, database operations, and AI-based fake news detection.

1) *Text Preprocessing:* The submitted news content is cleaned and normalized to remove noise and extract meaning-ful textual features. This step ensures improved performance of downstream machine learning models.
2) *BERT-based Classification:* The processed text is passed through a BERT-based classification model, which analyzes contextual and semantic relationships to classify the news as real or fake [1]. Transformer-based models like BERT have shown strong performance in text classification tasks due to their deep contextual understanding.
3) *Web Scraping for Real-world Validation:* To enhance reliability, the system collects related news headlines from trusted online sources using web scraping. This provides real-time external data for verification.
4) *SBERT Embedding and Similarity Analysis:* The system converts both user-submitted news and scraped headlines into embeddings using Sentence-BERT (SBERT). Cosine similarity is computed between these embeddings to measure semantic similarity:

$$Similarity = \frac{A \cdot B}{||A||, ||B||} \quad (1)$$

If similarity exceeds a predefined threshold, the news is considered consistent with real-world sources; otherwise, it is flagged as suspicious.

5) *Hybrid Decision Mechanism:* The final decision is ob-tained by combining the outputs of the BERT classifier and SBERT similarity analysis. This hybrid approach improves accuracy by leveraging both content-based classification and external validation.

C. Admin Module

The admin module provides control and monitoring ca-pabilities over the system. The administrator can view all registered users, manage user accounts, and monitor uploaded news content. Admins can also identify fake news posts, remove inappropriate content, and block users who repeatedly post misleading information. Additionally, reported news and user feedback are reviewed to maintain system integrity. This centralized control ensures reliability and prevents misuse of the platform.

D. Federated Learning-based Recommendation

The system incorporates federated learning to generate personalized news recommendations based on user interactions such as likes and engagement. Instead of sharing raw user data, the model learns from decentralized data sources, preserving user privacy while improving recommendation quality. Such decentralized learning approaches are gaining importance in modern AI systems for privacy-aware applications [14]–[16].

E. Output and Visualization

The final classification result (Real/Fake) is stored in the database and displayed in the news feed. Users can interact with the content through likes, comments, and sharing, while the admin ensures proper moderation and system management.

IV. IMPLEMENTATION DETAILS

A. Frontend Architecture (HTML, CSS, JavaScript)

The frontend of the system is developed using HTML, CSS, and JavaScript to provide a responsive and user-friendly interface. It includes pages for user registration, login, news posting, profile management, and news feed interaction.

1) User Interface: Provides forms for submitting news (headline, description, category) and displaying results in a structured news feed.
2) Dynamic Interaction: JavaScript is used to handle asyn-chronous requests such as liking news, commenting, and following users without reloading the page.

3) Session Handling: User authentication is managed using session-based login, ensuring secure access to features.

B. Backend Implementation (Django)

The backend is implemented using the Django framework, which handles routing, database operations, and integration with machine learning models. The system follows an MVC (Model-View-Controller) pattern for clean separation of con-cerns.

1) *Request Processing Workflow:* When a user submits a news article, the backend performs the following steps:

- Input Collection: The system collects headline, cate-gory, and news content from the user.
- Preprocessing: The text is cleaned and normalized to remove noise and improve model performance.
- BERT Classification: The processed text is passed to a BERT-based model to classify the news as real or fake based on contextual understanding [1].
- Web Scraping: Relevant news headlines are fetched from external sources such as Google News and Bing News.
- SBERT Similarity: Both user input and scraped head-lines are converted into embeddings using SBERT, and cosine similarity is calculated to measure semantic sim-ilarity.
- Hybrid Decision: The results from BERT and SBERT are combined to produce the final classification.

2) *Database Integration:* The system uses an SQL-based database managed through Django ORM. It stores user details, news posts, comments, likes, reports, and feedback. Relation-ships such as followers and interactions are efficiently handled using relational models.

3) *Admin Control Implementation:* The admin module is implemented within Django to provide system-level control. It includes functionalities for:

- Viewing and managing registered users
- Blocking/unblocking users
- Monitoring and deleting fake or inappropriate news
- Reviewing reported news and feedback

This ensures proper moderation and maintains the integrity of the platform.

4) *Federated Learning Integration:* The system incorpo-rates a federated learning-based recommendation mechanism, where user interactions such as likes are used to improve personalized news suggestions. Instead of sharing raw user data, the model updates are aggregated, preserving privacy while improving recommendation quality [14]–[16].

C. System Workflow Execution

The overall workflow begins when a user submits news through the frontend. The backend processes the input using classification and similarity models, generates a final decision, and stores the result. The processed news is then displayed in the news feed, where users can interact with it. Meanwhile, the admin module continuously monitors system activity to ensure reliability and prevent misuse.

V. EXPERIMENTAL SETUP & EVALUATION

To evaluate the performance of the proposed Hybrid Fake News Detection System, experiments were conducted using a locally hosted Django backend integrated with machine learning models. The system was tested using multiple user-submitted news samples along with real-time web data col-lected through scraping.

A. Experimental Setup

The backend system was implemented using Python Django, with machine learning components integrated using transformer-based models. The BERT model was used for classification, while Sentence-BERT (SBERT) was used for semantic similarity analysis. Web scraping was performed on sources such as Google News and Bing News to obtain real-time headlines for validation.

B. Evaluation of Detection Methods

The system uses two primary methods for fake news detec-tion:

- BERT Classification: Predicts whether the news is real or fake based on textual content [1].
- SBERT Similarity Analysis: Compares user news with real-world headlines using cosine similarity.

TABLE I
PERFORMANCE COMPARISON OF DETECTION METHODS

Method	Accuracy (%)
BERT Classification	86.5%
SBERT Similarity	82.3%
Hybrid Approach	91.2%

The results show that the hybrid approach outperforms individual methods by combining contextual understanding with real-world validation.

C. Similarity Threshold Analysis

The cosine similarity score plays a key role in determining whether a news article matches existing real-world news. A threshold value (e.g., 0.6) is used to classify similarity.

TABLE II
SIMILARITY THRESHOLD EVALUATION

Threshold Value	Detection Accuracy (%)
0.5	85.4%
0.6	89.8%
0.7	84.2%

It is observed that a threshold of 0.6 provides the best balance between precision and recall.

D. System Performance Analysis

The system was evaluated based on response time and efficiency. Since web scraping and similarity computation are performed dynamically, slight delays may occur. However, the system maintains acceptable performance for real-time usage, typically providing results within a few seconds.

E. Discussion

The experimental results demonstrate that combining BERT classification with SBERT-based similarity improves detection accuracy compared to standalone approaches. The inclusion of real-time web data enhances reliability, while federated learning contributes to personalized recommendations without compromising user privacy. Overall, the system provides an effective and scalable solution for fake news detection.

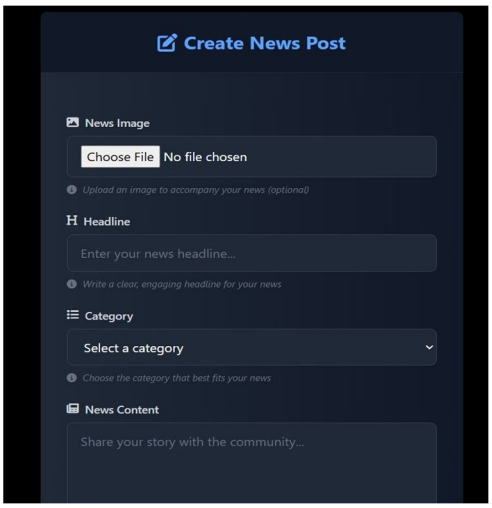


Fig. 2. User Posting News



Fig. 3. Labelling News As REAL

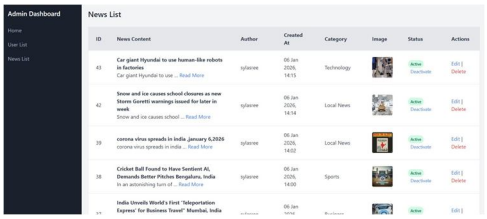


Fig. 4. Admin Dashboard

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed a Hybrid Fake News Detection System that combines BERT-based classification with SBERT-based semantic similarity analysis to improve the accuracy and reliability of fake news detection. By integrating real-time web scraping, the system validates user-submitted news against trusted sources, reducing the chances of misclassification caused by misleading writing styles.

The inclusion of a hybrid decision mechanism enables the system to leverage both contextual understanding and real-world verification, resulting in better performance compared to standalone approaches. Furthermore, the integration of federated learning allows personalized news recommendation while preserving user privacy, addressing critical concerns in modern data-driven applications. The addition of user and admin modules ensures effective interaction, monitoring, and control of misinformation within the platform.

Experimental results demonstrate that the hybrid approach outperforms individual detection methods in terms of accuracy and reliability. The system is scalable, efficient, and suitable for real-time applications.

In future work, we aim to enhance the model by incorporating larger and more diverse datasets, improving real-time performance, and exploring advanced deep learning techniques for better accuracy. Additionally, optimizing the recommendation system and strengthening misinformation detection using multimodal data (such as images and videos) can further improve the overall effectiveness of the platform.

VII. ACKNOWLEDGMENT

We would like to thank the Department of Computer Science and Engineering at Universal Engineering College for their continued guidance, support, and the resources provided throughout the development of this project.

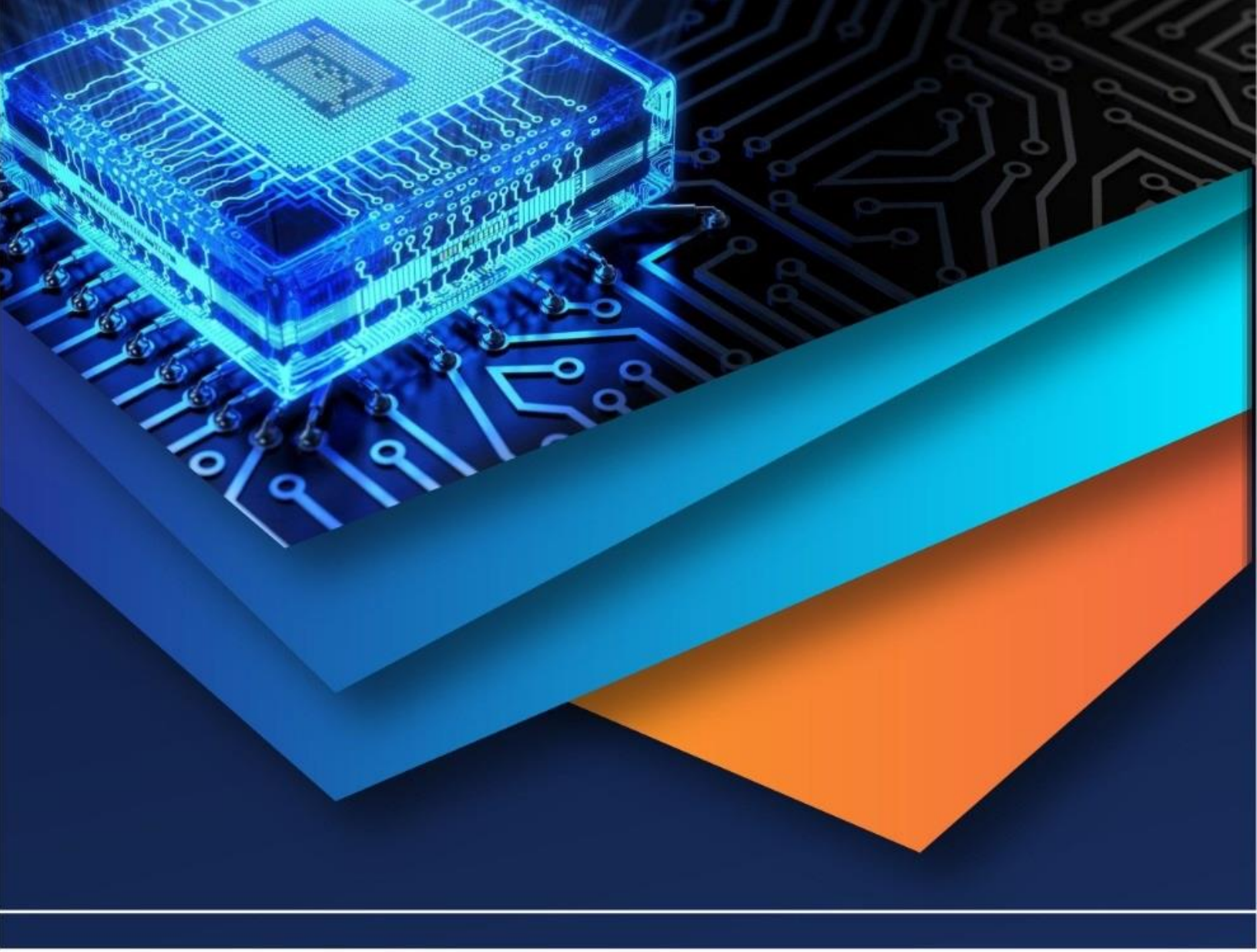
REFERENCES

[1] T. H. C. Chiang, C. S. Liao, and W. C. Wang, "Investigating the Difference of Fake News Source Credibility Recognition between ANN and BERT," Journal of Information Technology and Media Studies, vol. 9, no. 2, pp. 45–52, 2022.

[2] V. D. Husza´r, V. K. Adhikarla, I. Ne´gyesi, and C. Krasznay, "Toward Fast and Accurate Violence Detection for Automated Video Surveillance Applications," IEEE Access, vol. 11, pp. 18772–18793, 2023.

[3] M. Khan, A. E. Saddik, W. Gueaieb, G. De Masi, and F. Karray, "VD-Net: An Edge Vision-Based Surveillance System for Violence Detection," IEEE Access, vol. 12, pp. 43796–43808, 2024.

[4] B. Wang and X. Zhang, "An Efficient Violence Detection Method Based on Temporal Attention Mechanism," Instrumentation, vol. 10, no. 2, 2023.
[5] V. Lomte, S. Singh, S. Patil, and D. Pahurkar, "Video Anomaly Detection through Live Surveillance," IJRESM, vol. 2, no. 4, 2019.
[6] M. Ahmed et al., "Real-Time Violent Action Recognition Using Key Frames Extraction and Deep Learning," 2021.
[7] Z. Trabelsi et al., "Real-Time Attention Monitoring System for Class-room: A Deep Learning Approach for Student's Behavior Recognition," Big Data and Cognitive Computing, vol. 7, no. 1, 2023.
[8] J. H. Piater, S. Richetto, and J. L. Crowley, "Event-Based Activity Analysis in Live Video Using a Generic Object Tracker," in Proc. IEEE Int. Workshop on Performance Evaluation of Tracking and Surveillance, 2002, pp. 1–8.
[9] J. Xiao, S. Li, and Q. Xu, "Video-Based Evidence Analysis and Extraction in Digital Forensic Investigation," IEEE Access, vol. 7, pp. 55432–55442, 2019.
[10] A. Patwardhan and G. Knapp, "Aggressive Actions and Anger Detection from Multiple Modalities Using Kinect," arXiv preprint arXiv:1607.01076, 2016.
[11] L. Zhang, X. Ruan, and J. Wang, "WiVi: A Ubiquitous Violence Detection System With Commercial WiFi Devices," IEEE Access, vol. 8, pp. 66626–66672, 2020.
[12] B. Wang and X. Zhang, "An Efficient Violence Detection Method Based on Temporal Attention Mechanism," Instrumentation, vol. 10, no. 2, 2023.
[13] D. Tsiktsiris, A. Lalas, M. Dasygenis, and K. Votis, "Multimodal Abnormal Event Detection in Public Transportation," IEEE Access, vol. 12, pp. 133469–133480, 2024.
[14] A. Dembele, R. W. Mwangi, and A. O. Kube, "A Lightweight Convo-lutional Neural Network with Hierarchical Multi-Scale Feature Fusion for Image Classification."
[15] A. G. Howard et al., "Searching for MobileNetV3," in Proc. IEEE/CVF ICCV, 2019.
[16] B. Jacob et al., "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference," in Proc. IEEE CVPR, 2018, pp. 2704–2713.
[17] H. Fan, K. Zhao, and X. Wang, "VideoMix: Rethinking Data Augmentation for Video Classification," arXiv preprint, 2020.
[18] A. Arnab et al., "Video Vision Transformers for Violence Detection," arXiv preprint, 2022.
[19] C. Wang and Y. Liu, "Exploring Temporally Dynamic Data Augmentation for Video Recognition," 2022.
[20] M. Patel, "Real-Time Violence Detection Using CNN-LSTM," 2021.
[21] J. Wang et al., "DETRs Beat YOLOs on Real-Time Object Detection," arXiv preprint, 2023.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)