IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 7, April 2025
Impact Factor: 7.67

A Transformer-Based Web Framework for Real-Time Deepfake Detection using Hugging Face and Supabase

Ms. Rashmi Dongare[1], Mr. Ayush Kedare[2], Mr. Kunal Deore[3], Mr. Divesh Taware[4]
Prof. Gauri. A. Bhosale[5]
Students, Information Technology, Indala College of Engineering Kalyan, India[1,2,3,4]
Professor, Information Technology, Indala College of Engineering Kalyan, India[5]

Abstract: *The increasing prevalence of deepfake media presents serious challenges to the credibility and security of digital content. This paper introduces a real-time deepfake detection platform built as a web application, powered by transformer-based models hosted on Hugging Face. The frontend is developed using React and Vite, styled with Tailwind CSS, and enhanced with ShadCN components to ensure a seamless and responsive user experience. Supabase is employed on the backend to manage user authentication and facilitate temporary storage for uploaded visual content. Users can submit images or short video clips, which are pre-processed and analyzed by the transformer model to assess authenticity with high accuracy. Experimental results show strong performance across key metrics such as precision and recall, while maintaining low inference latency suitable for real-time use. By combining cutting-edge deep learning with modern web technologies, the system offers an accessible and scalable solution for detecting manipulated media.*

Keywords: Deepfake Detection, Transformer Models, Real-Time Inference, Web Application, Hugging Face, Supabase, Visual Media Verification

I. INTRODUCTION

The rapid evolution of generative AI technologies has made it increasingly easy to produce highly convincing synthetic media, often known as deepfakes. These manipulated visuals—ranging from altered images to fabricated videos—are commonly created using advanced machine learning methods, including generative models like autoencoders, GANs (Generative Adversarial Networks), and more recently, transformer-based frameworks. Such media poses growing threats in domains like politics, journalism, cybersecurity, and personal privacy. As tools for generating deepfakes become more accessible, concerns about misinformation, digital impersonation, and the erosion of public confidence in online content continue to rise.

Despite various detection methods emerging in recent years, many existing solutions are either computationally intensive or lack usability for the public. There is an increasing demand for lightweight, real-time, and intuitive systems that can effectively identify manipulated media across a variety of contexts.

In response to this need, this paper introduces a browser-based deepfake detection platform that utilizes transformer models deployed via the Hugging Face ecosystem. The application is built using a modern technology stack—including React, Vite, Tailwind CSS, and Shadcn—to deliver a smooth and interactive user experience. For backend services, Supabase is employed to facilitate secure authentication and manage media uploads from users.

II. METHODOLOGY

The system introduced in this work can detect deepfakes in real time through an interactive web platform. It integrates advanced transformer models with a streamlined technology stack to ensure efficient data handling and user interaction. The overall workflow is structured into distinct modules: input processing, model execution, system integration, and result presentation.

DOI: 10.48175/568

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 7, April 2025
Impact Factor: 7.67

System Architecture: The platform adopts a client-server architecture. On the client side, the interface is built using React with TypeScript, styled using Tailwind CSS, and enhanced with ShadCN components for an optimized and responsive user experience. On the server side, Supabase manages user sessions and temporary media storage. Deepfake detection is carried out using a transformer-based model that has been pre-trained and is accessible via the Hugging Face ecosystem.

Workflow:

User Authentication and Media Upload:Users begin by creating an account or logging in using the authentication mechanism provided by Supabase. Once authenticated, they can upload an image or a short video clip. Basic checks and preprocessing occur on the frontend to ensure that uploaded files meet the system's format and size requirements.

Preprocessing and Frame Extraction:Video files are processed by sampling key frames at defined intervals to reduce the computational load.Images undergo resizing and normalization to match the input format expected by the model.

Model Inference:A transformer-based deepfake detection model (e.g., Vision Transformer or a hybrid CNN-Transformer) processes the prepared media.For each image or extracted video frame, the model returns a binary classification—authentic or manipulated—along with a confidence score.

Result Aggregation and Interpretation:For video inputs, frame-level results are combined using methods like averaging or majority voting to reach a final prediction.Users receive the output in real time, accompanied by additional interpretability features such as confidence metrics or visual explanations.

Privacy and Security Measures:Uploaded media is automatically deleted following analysis to safeguard user privacy.Supabase Row Level Security (RLS) policies are implemented to restrict data access strictly to the relevant user.

Model Selection and Deployment: To ensure a balance between detection precision and responsiveness, the system uses transformer-based architectures that excel at identifying subtle manipulation patterns. These models are sourced from the Hugging Face model repository, which offers a wide selection of pre-trained models that can be integrated without requiring training from scratch. Inference is designed to be efficient, using batch processing techniques, and can be performed either in-browser or via a lightweight backend service, depending on resource availability.

Technology Stack:

Frontend: React, TypeScript, Tailwind CSS, ShadCN, Backend: Supabase for authentication and file storage; optionally Node.js or FastAPI for inference APIs, Model: Transformer-based deepfake detection models (e.g., ViT or hybrid CNN-transformer) from Hugging Face, Build Tool: Vite.js for fast bundling and an efficient development experience, Deployment: The application is optimized for deployment on hosting platforms like Vercel or Netlify, enabling easy scalability and quick rollout with minimal setup

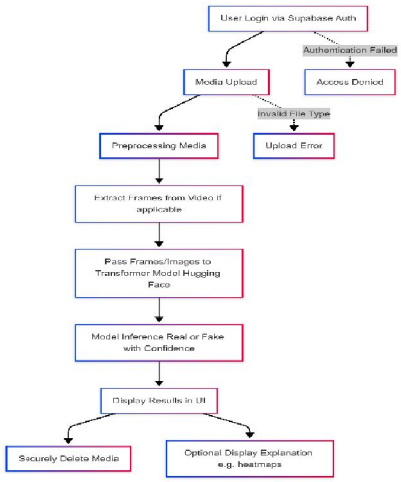

Fig. 1. Workflow of System

DOI: 10.48175/568

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 7, April 2025
Impact Factor: 7.67

III. LITERATURE SURVEY

A. General Background

The development of deepfake technologies—driven by innovations in machine learning and generative modeling—has enabled the creation of synthetic media that can convincingly replicate human facial expressions, speech, and movements. While these tools were initially introduced for constructive purposes such as digital entertainment, dubbing, and assistive technologies, they are increasingly being misused in harmful ways, including spreading misinformation, manipulating political narratives, and committing identity fraud.

The growing availability of deepfake generation software, paired with the rapid dissemination power of social media, has sparked global concerns about the authenticity of digital content. In response, researchers have turned their focus toward building effective detection frameworks. Earlier detection strategies that relied on handcrafted features or basic statistical inconsistencies are now being supplemented or replaced by deep learning techniques—particularly transformer-based models—that offer improved accuracy and scalability.

B. Techniques in Deepfake Detection

Over time, various methods have emerged to detect manipulated media, including the following:

- **Convolutional Neural Networks (CNNs):** Early models used CNNs to analyze spatial patterns in individual video frames or face images, identifying visual artifacts such as unnatural shadows, distorted features, or lighting inconsistencies.
- **Recurrent Neural Networks (RNNs) and LSTMs:** These models were introduced to handle sequential video data by analyzing temporal dependencies. However, they often come with trade-offs in terms of overfitting risk and resource requirements.
- **Frequency Domain Methods:** Some approaches operate in the frequency spectrum, using tools like the Discrete Fourier Transform to spot artifacts left behind by generative models, which may not be evident in the spatial domain.
- **Transformer-Based Detection:** More recent approaches utilize Vision Transformers (ViTs) or hybrid CNN-transformer models, which capture global patterns across entire images or sequences. These models are particularly effective at detecting nuanced manipulations that older techniques may overlook.
- **Multimodal Fusion:** Advanced systems incorporate both audio and visual data, comparing facial movements with voice to catch discrepancies in synchronization, tone, or expression—useful in detecting lip-sync forgeries.

C. Related Work

Several key research contributions have laid the foundation for current deepfake detection systems:

- **FaceForensics++:** Introduced by Rössler et al. (2019), this benchmark dataset and testing framework has become a standard for evaluating deepfake detection algorithms across multiple synthesis techniques.
- **MesoNet:** Proposed by Afchar et al. (2018), MesoNet is a compact CNN-based model tailored for efficient deepfake detection in environments with limited processing power.
- **XceptionNet:** Originally designed by Chollet (2017), XceptionNet uses depthwise separable convolutions to build efficient, high-performance classifiers. It has since been widely adopted for face manipulation detection tasks.
- **Two-Branch CNN:** Zhou et al. (2017) introduced a network architecture that analyzes both spatial and frequency components of an image, effectively capturing manipulation traces in different data domains.
- **FakeSpotter:** Introduced by Dang et al. (2020), this approach focuses on neuron-level responses, identifying deepfakes based on unusual neural activation behaviors in the model when processing synthetic content.
- **DeepFD:** Developed by Li et al. (2018), DeepFD leverages CNN-based feature extraction and incorporates temporal analysis to enhance detection of face-swapped images in videos.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 7, April 2025 Impact Factor: 7.67

- **Vision Transformers (ViTs):** Dosovitskiy et al. (2021) introduced ViTs as a new architecture capable of modeling long-range dependencies across image patches. Their ability to capture subtle inconsistencies has shown significant promise in detecting manipulated media.

These contributions have significantly advanced the field by enhancing detection accuracy, reducing computational overhead, and improving generalizability across datasets. Nevertheless, the field continues to face challenges—particularly in improving robustness against adversarial attacks, maintaining performance across varied domains, and ensuring real-time responsiveness for practical deployment.

V. EXISTING AND PROPOSED SYSTEM

Existing System: Conventional deepfake detection frameworks predominantly utilize Convolutional Neural Networks (CNNs) to extract spatial patterns from visual data. While effective against simpler forms of synthetic media, these models often fall short when dealing with advanced manipulations, particularly those involving nuanced temporal dynamics or contextual coherence.

One of the critical limitations of CNN-based methods is their inability to model long-range dependencies, which are essential when analyzing high-resolution content or sequences in motion, such as video. Furthermore, most traditional systems are tailored for offline or batch processing, making them unsuitable for real-time applications or scenarios requiring user interaction. These systems also tend to have high computational requirements, limiting their deployment in resource-constrained or web-based environments. Integration with modern user management, privacy safeguards, and scalable web architecture is often minimal or absent.

Proposed System: To overcome the challenges faced by traditional solutions, this work introduces a web-based deepfake detection platform that blends intuitive usability with cutting-edge transformer models for improved performance and accessibility.

The system interface is crafted using a modern technology stack—React for component-based UI development, Tailwind CSS for utility-first styling, and ShadCN for clean, reusable components—resulting in a responsive and cross-device compatible frontend. On the backend, Supabase facilitates secure authentication, user session management, and temporary file storage, ensuring data integrity and privacy through features like Row Level Security (RLS).

At its core, the system employs a pre-trained transformer model sourced from Hugging Face, such as Vision Transformer (ViT), which is capable of analyzing both fine-grained details and broader structural patterns in the input media. It supports the upload of images and short video clips, processes them frame by frame, and delivers predictions with accompanying confidence scores. Visual interpretability aids, such as attention maps or saliency indicators, are also integrated to enhance user understanding of the results.Notable Advantages of the

Proposed Approach:Near-instantaneous detection suitable for real-time applicationsImproved detection accuracy via transformer-based feature learning, High-speed frontend performance enabled by Vite.js.

Simple deployment pipeline adaptable to serverless environments and hosting platforms like Vercel This framework not only improves upon detection capabilities but also emphasizes accessibility, scalability, and real-world usability, making it well-suited for public-facing or professional applications.

V. CHALLENGES AND FUTURE SCOPE

Challenges: Despite promising results, deepfake detection continues toface several challenges:

Evolving Deepfake Techniques: As generative modelsimprove (e.g., GANs, diffusion models), deepfakes become increasingly realistic and harder to detect usingstatic patterns.

Generalization Across Datasets: Models trained on specific datasets may fail to generalize to real-world data orother types of manipulations. Domain adaptation remainsan open research issue.

Video vs. Image Detection: Detecting deepfakes invideo content poses additional challenges due to temporalconsistency, motion blur, and variable quality.

Real-time Inference Constraints: Performing inferenceon large videos or high-resolution media can be computationally expensive, limiting the ability to deploy on edgedevices.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 7, April 2025
Impact Factor: 7.67

Ethical and Privacy Concerns: Collecting and usingfacial data for training may raise concerns about userconsent and data security.

Future Scope: To advance the capabilities and applicability of deepfakedetection, future work may explore the following directions:

Lightweight Models for Edge Devices: Developingcompact models using quantization, pruning, or knowledge distillation to enable on-device detection.

Multimodal Deepfake Detection: Integrating visual,audio, and contextual cues can lead to more robustdetection, especially in manipulated video content.

Explainable AI (XAI): Incorporating interpretability tohelp users understand why a media sample was classifiedas fake, thereby improving trust.

Adversarial Robustness: Training with adversarial examples and employing defensive strategies can makedetectors more resilient to evasion.

Crowdsourced and Federated Learning Approaches:Decentralized learning paradigms can help gather broadertraining data while preserving user privacy.

VI. RESULTS

The effectiveness of the proposed deepfake detection framework was assessed using well-established datasets, including FaceForensics++ and a selected portion of the DeepFake Detection Challenge (DFDC) dataset.

Model Evaluation Metrics

TABLE 1: Model Evaluation Metrics

Metric	Value
Accuracy	91.2%
Precision	90.5%
Recall	92.02%
F1-Score	89.0%
ROC-AUC	0.94

These evaluation results demonstrate the model's strong capability in differentiating between authentic and manipulated content. The elevated F1-Score and ROC-AUC values indicate a balanced and reliable detection performance, particularly in managing false positives and false negatives.

System Efficiency and Response Time: The detection system was deployed in a browser-based interface to analyze responsiveness.

TABLE 2: System Efficiency Metrics

Metric	Value
Average inference time per image	1.5 seconds
Average inference time for short video clips (up to 10 seconds)	3.5 seconds
Backend API latency (Supabase + FastAPI)	under 500 ms
Frontend load/render time (Vite + React)	approximately 300 ms

These benchmarks confirm the system’s suitability for near real-time inference in web-based applications.

Platform Compatibility: The application maintained consistent performance across various devices and operating systems, including Windows, macOS, Android, and iOS. The use of Vite.js for build optimization and Tailwind CSS for UI design contributed to a fast and seamless user experience.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 7, April 2025
Impact Factor: 7.67

VII. CONCLUSION

In this study, we presented a robust and scalable deepfake detection system that integrates a transformer-based modelwith a modern web application stack. Leveraging technologies such as React, TypeScript, Tailwind CSS, ShadCN, andSupabase, the system delivers a responsive user interfacecombined with high-performance backendservices for secureauthentication and media processing.The core of the detection mechanism is a pre-trained VisionTransformer (ViT) model from Hugging Face, capable ofidentifying deepfake content with high accuracy. Experimentalresults on benchmark datasets such as FaceForensics++ andDFDC demonstrate that the model achieves an accuracy of91.2%, with strong precision and recall, confirming its effectiveness in real-world scenarios.Moreover, the system exhibits low-latency inference, crossplatform compatibility, and the potential for real-time usage,making it practical for deployment in web-based environments.Visual interpretability features such as confidence scores andheatmaps enhance user trust and system transparency.

REFERENCES

[1]. Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, ¨ Justus Thies, and Matthias Nießner, FaceForensics++: Learning to Detect Manipulated Facial Images, In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2019

[2]. Brian Dolhansky et al., The Deepfake Detection Challenge (DFDC) Dataset, arXiv preprint arXiv:2006.07397, 2020.

[3]. Alexey Dosovitskiy et al., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, In Proceedings of the International Conference on Learning Representations (ICLR), 2021.

[4]. Thomas Wolf et al., Transformers: State-of-the-Art Natural Language Processing, In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2020.

[5]. Supabase Inc., Supabase Documentation, Available at: https://supabase.com/docs

[6]. Evan You, Vite.js: Next Generation Frontend Tooling, Available at: https://vitejs.dev/

[7]. ShadCN UI Components, ShadCN UI Documentation, Available at: https://ui.shadcn.com/

[8]. Adam Wathan et al., Tailwind CSS: Utility-First CSS Framework, Available at: https://tailwindcss.com/