

Explainable Multimodal AI Framework for Deepfake Detection and Digital Content Authenticity Verification

R S Geethanjali

Assistant Professor

Dept. of Computer Science and
Engineering

K S School of Engineering and
Management
Bengaluru, India

Sneha K S

Student

Dept. of Computer Science and
Engineering

K S School of Engineering and
Management
Bengaluru, India

Vaidehi Vasudev Arundekar

Student

Dept. of Computer Science and
Engineering

K S School of Engineering and
Management
Bengaluru, India

Vinutha B R

Student

Dept. of Computer Science and
Engineering

K S School of Engineering and
Management
Bengaluru, India

Abstract - The rapid growth of synthetic media produced by generative models has made it difficult to verify the authenticity of images, videos, audio recordings, news text and online reviews. Most available detectors handle a single modality and give no reason for their output. This paper presents DeepShieldAI, a web-based multimodal system for detecting and explaining manipulated digital content. The system uses a MobileNetV2 classifier for images, an R3D-18 three-dimensional convolutional network for videos, a Wav2Vec2-based classifier for audio, and DistilBERT-based classifiers for news text and reviews. Explainability is provided through Grad-CAM heatmaps, SHAP feature attributions and LIME word-level explanations that are generated from each prediction and stored with the analysed file. A Retrieval-Augmented Generation module built with LangChain and ChromaDB answers user questions from an indexed knowledge base of uploaded documents, and a review forensics and bot analysis module extends the system beyond media classification. The platform is delivered through a React frontend, a layered FastAPI backend and a PostgreSQL database that keeps user-specific history, predictions and reports. The paper describes the architecture, module design and processing workflow, and reports the functionality demonstrated by the implemented system. Quantitative benchmark evaluation and unified multimodal fusion are identified as the remaining work.

Keywords - Deepfake detection; explainable AI; multimodal learning; Retrieval-Augmented Generation; Grad-CAM; SHAP; LIME; FastAPI; content authenticity

I. INTRODUCTION

Generative models such as Generative Adversarial Networks (GANs) and their successors can now synthesise realistic faces, voices and text. These capabilities have legitimate uses in entertainment and education, but they also enable deepfakes: manipulated media that support misinformation, identity misuse and financial fraud. As synthetic content becomes harder to tell apart from genuine material, manual inspection is no longer a dependable safeguard.

Real-world digital content rarely arrives in a single modality. A post may combine a video, an audio track and a caption, and a detector that examines only one of these can miss manipulation in the others. In addition, many deployed deep learning models act as black boxes that return a label without indicating which evidence led to it, which limits user trust and forensic usefulness. In a discussion held on 23 April 2026 with social media influencers and content creators, the participants described fake videos, misleading content and identity misuse as problems that

damage audience trust. Journalists, law enforcement agencies, social media platforms and the general public were identified as other affected groups.

To address these gaps, this paper presents DeepShieldAI, an implemented multimodal system for analysing potentially manipulated digital content. Its main contributions are:

- A working multimodal analysis platform that handles images, videos, audio, news text and reviews with modality-specific deep learning models.
- Explainable-AI (XAI) outputs that are generated from each model prediction and stored and displayed with the analysed file.
- A Retrieval-Augmented Generation (RAG) knowledge-base module that answers user questions from indexed documents.
- A review forensics and bot analysis module that goes beyond binary classification of media.
- A full-stack architecture with authentication, per-user history, reports and administration.

The remainder of the paper is organised as follows. Section II reviews related work, Section III states the problem and objectives, Section IV presents the architecture, Section V describes the implementation, Section VI discusses the results demonstrated so far, and Section VII concludes.

II. RELATED WORK

A. Retrieval-Augmented Verification

Bai and Fu [1] built a fake news detection framework that uses claim extraction, retrieval-augmented generation and reasoning-based verification. They report 88% accuracy and an F1-score of 89.4%, outperforming BERT, RoBERTa and DeBERTa baselines, but the approach depends heavily on retrieval quality and is slower because of its multi-stage pipeline.

B. Explainable Deepfake Detection

Abinash et al. [2] combined transfer learning with Grad-CAM, LIME and Integrated Gradients and reported approximately 98% accuracy on a dataset of about 140,000 images; the approach has high computational cost and limited cross-dataset generalisation. Patel et al. [3] surveyed explainable deepfake detection, covering LIME, SHAP, Grad-CAM and

multimodal frameworks and providing a taxonomy and a view of the field's evolution from 2017 to 2024, but their survey offers no experimental validation and gives mostly theoretical insights.

C. Real-Time and Spatio-Temporal Approaches

Rao et al. [4] proposed a real-time misinformation detection framework that combines BERT-based language analysis, convolutional networks, GAN fingerprinting and blockchain. They report improved real-time detection and robustness against adversarial attacks, but the system is complex, has high resource requirements and incurs blockchain overhead. Wagh et al. [5] used spatial and temporal analysis with Grad-CAM and LIME and reported accuracy of about 96% to 98% depending on the model, but their approach is computationally expensive and lacks standard benchmarks.

D. Gap Addressed

Among the works reviewed, most are restricted to one modality or treat explanation, contextual retrieval and deployment as separate concerns. Few are delivered as an interactive system in which a user can upload content, obtain a prediction with its explanation, consult a knowledge base and revisit earlier results. DeepShieldAI targets this gap by integrating these functions in one web application.

III. PROBLEM STATEMENT AND OBJECTIVES

The rise of deepfakes generated with techniques such as GANs makes it difficult to distinguish real from fake content, which leads to misinformation and fraud. Existing detectors often cover a single modality, give no explanation of their decisions and perform poorly in real-world conditions. The objectives of the project are:

- to develop a multimodal AI system for deepfake detection;
- to detect manipulated content in video, audio and text data;
- to identify AI-generated media with deep learning techniques;
- to provide transparency and interpretability through XAI;
- to support content authenticity verification through RAG; and
- to detect fake reviews, bot activity and misinformation campaigns.

Scope. The implemented system analyses images, videos, audio, news text and reviews, provides explanations and a RAG knowledge base, and offers a web interface with user accounts, history and reports. Unified multimodal fusion and large-scale benchmark evaluation are outside the work reported here.

IV. SYSTEM ARCHITECTURE

A. Backend Architecture

The backend follows a layered design, shown in Fig. 1. The React frontend communicates with a FastAPI application over

HTTP/REST. The API layer exposes authentication, document, prediction, report, RAG and administration routes. Requests pass to a service layer that holds the business logic and then to a repository layer that performs database operations. Data are persisted in a PostgreSQL database containing users, documents, predictions, reports, audit logs, and roles and permissions, while uploaded files and generated XAI outputs are kept in file storage. The API layer also connects to the AI/ML services (image, video, audio and text models and the XAI components) and to the RAG service, which uses LangChain, embeddings, ChromaDB and a large language model (LLM). The architecture reserves a place for a fusion model; this component is not part of the current inference path and is listed as future work.

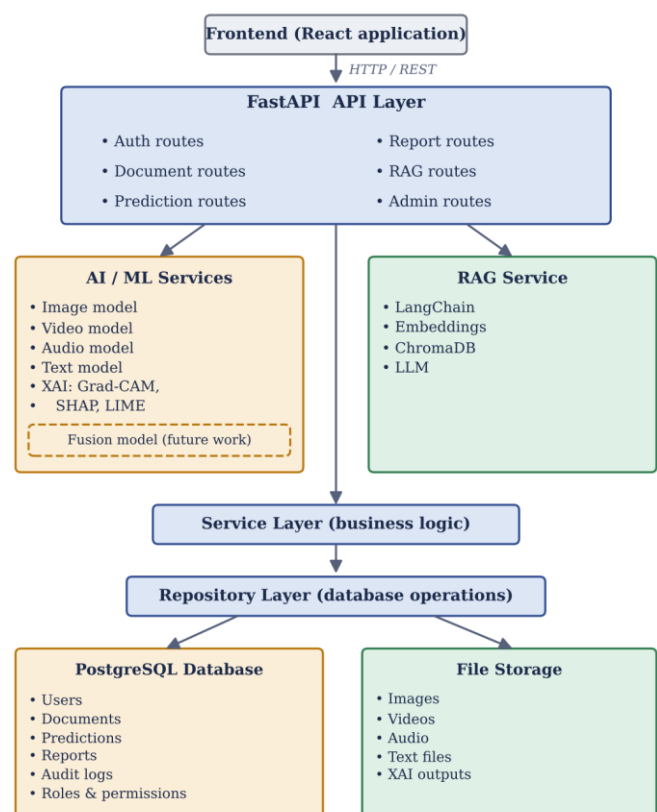
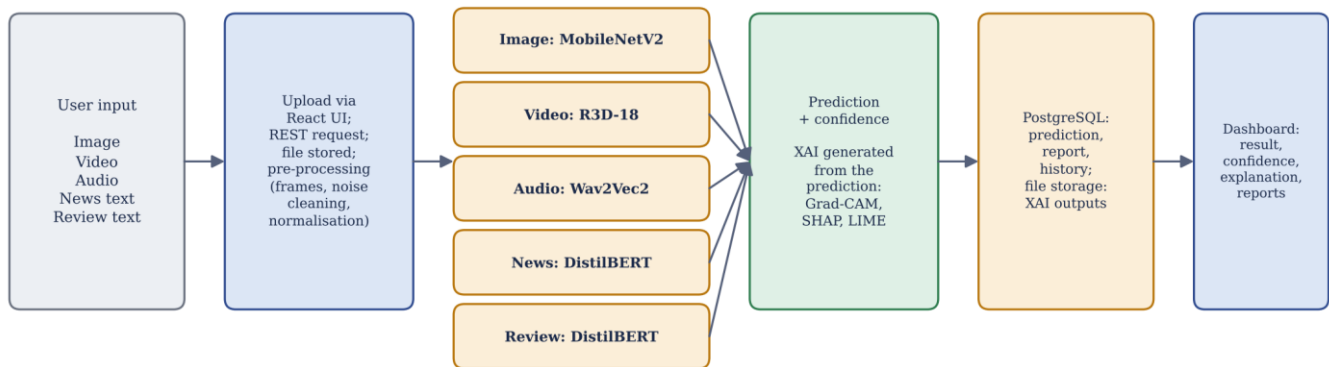


Fig. 1. Layered backend architecture of DeepShieldAI (redrawn from the project design).

B. End-to-End Data Flow

Fig. 2 summarises the two main flows. In the detection flow, uploaded content is stored, pre-processed and passed to the model for its content type; the prediction and its explanation are saved and returned to the dashboard. In the knowledge-base flow, uploaded documents are indexed in a vector store and later retrieved to answer user questions.

A. Detection and explanation pipeline



B. Knowledge-base (RAG) pipeline

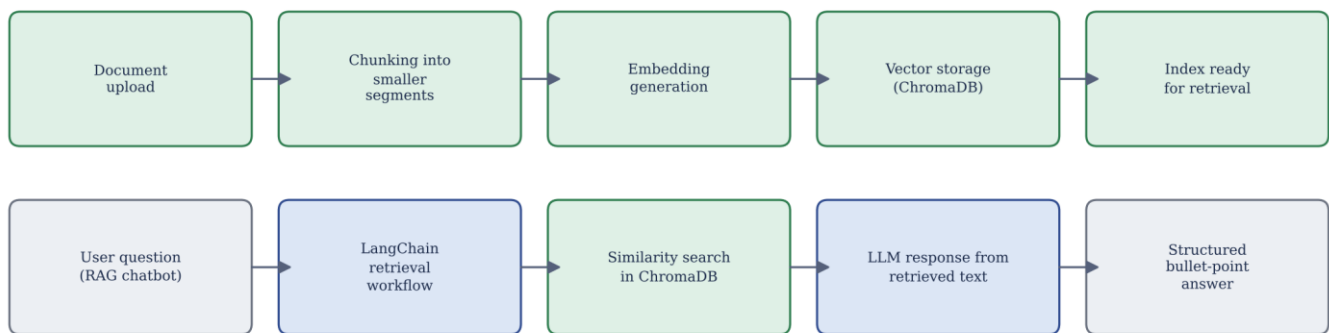


Fig. 2. End-to-end processing flow: (A) detection and explanation pipeline and (B) knowledge-base (RAG) pipeline.

C. Frontend Architecture

As shown in Fig. 3, a user first reaches the login or registration page. After successful authentication the user enters a dashboard from which the upload, prediction, history, explainability, RAG chatbot and administration sections are available. All sections communicate with the FastAPI backend through an API service layer based on an HTTP client (Axios). Knowledge-base management and reports are also reachable from the interface.

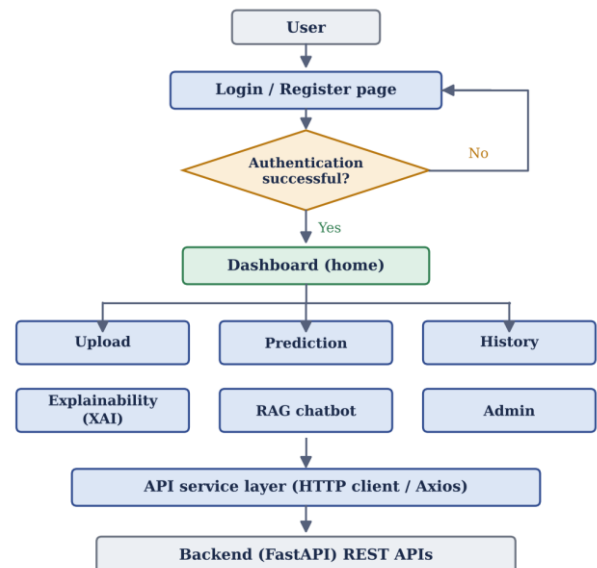


Fig. 3. Frontend navigation flow of the React application.

V. IMPLEMENTATION

A. Data Acquisition and Pre-processing

The methodology collects multimodal data comprising videos, audio, text, fake reviews and bot-related data from datasets and real-world sources. Pre-processing prepares each modality for its model by extracting frames from video, cleaning noise and normalising inputs.

B. Modality-Specific Detection Models

Separate deep learning models analyse each content type (Table I). A MobileNetV2 model classifies images as real or manipulated. An R3D-18 network, a three-dimensional convolutional network, analyses spatial and temporal information in videos. A Wav2Vec2-based model classifies audio as real or potentially manipulated. DistilBERT is used for two text tasks: classifying news and other textual content as fake or real, and classifying reviews as genuine or suspicious.

TABLE I
DETECTION MODELS USED IN DEEPSHIELD AI

Content	Model	Output class
Image	MobileNetV2	Real / manipulated
Video	R3D-18	Real / manipulated
Audio	Wav2Vec2-based	Real / potentially manipulated
News / text	DistilBERT	Fake / real
Review	DistilBERT	Genuine / fake (suspicious)

C. Explainable AI

Three XAI techniques are integrated so that predictions can be inspected. Gradient-weighted Class Activation Mapping (Grad-CAM) produces heatmaps that mark the image regions contributing to a prediction. SHapley Additive exPlanations (SHAP) estimate the contribution of individual features, tokens or segments. Local Interpretable Model-agnostic Explanations (LIME) identify the words or phrases that influence text and review classifications. The explanations are computed from the model prediction rather than being predefined messages, are associated with the corresponding prediction and uploaded file, and are shown in the Explainability section.

D. Retrieval-Augmented Generation Knowledge Base

The RAG module provides contextual information from documents that users add to a knowledge base. Algorithm 1 describes its workflow. LangChain manages the retrieval and question-answering process, and ChromaDB stores the document embeddings for similarity-based retrieval. Answers are presented as structured bullet points so that they are easy to read.

Algorithm 1. Knowledge-base indexing and question answering

- 1) The user uploads a document in the Knowledge Base section and the entry is registered.
- 2) The document is divided into smaller meaningful chunks.
- 3) An embedding is generated for each chunk and stored in ChromaDB.
- 4) For a user question, LangChain retrieves the most similar chunks from ChromaDB.
- 5) The LLM generates a response from the retrieved information.

- 6) The response is displayed as structured bullet points.

E. Review Forensics and Bot Analysis

A separate module analyses suspicious or potentially fake reviews. It classifies a submitted review as genuine or fake, identifies the words and patterns that contribute to the prediction, and reports additional characteristics of the review. A bot analysis feature highlights patterns that may indicate automated review activity. Results are displayed together with the prediction and the supporting explanation, which gives more detail than a bare label.

F. Backend, Database and Request Processing

The backend is implemented in FastAPI with REST APIs for authentication, document management, predictions, reports, RAG operations and administration. PostgreSQL holds users, uploaded documents, predictions, reports and audit-related information, and records are tied to the authenticated user so that each user sees only the corresponding history. Algorithm 2 summarises how a detection request is processed.

Algorithm 2. Processing of a detection request

- 1) The user logs in and the backend authenticates the credentials.
- 2) The user uploads an image, video, audio file, news text or review, and the frontend sends it to the backend through a REST request.
- 3) The backend stores the uploaded file and registers it for the user.
- 4) The content is pre-processed according to its modality.
- 5) The model for the content type returns a class label and a confidence value.
- 6) The XAI module generates the explanation for this prediction and the output is stored with the file.
- 7) The prediction and related information are saved in PostgreSQL against the user.
- 8) The frontend displays the prediction, confidence and explanation, and the record becomes available under History and Reports.

G. Technology Stack and Environment

Table II lists the technologies of the project. The software requirements also list TensorFlow or PyTorch, HuggingFace Transformers, NLTK, spaCy, OpenCV, MediaPipe, Librosa and PyDub as supporting libraries. The stated hardware requirement is an Intel Core i5 or AMD Ryzen 5 processor or better, at least 8 GB of RAM (16 GB recommended) and at least 256 GB of SSD storage. A GPU is not mandatory, since the system can run on a CPU, and cloud GPUs such as Google Colab or Kaggle can be used to speed up training.

TABLE II
TECHNOLOGY STACK

Layer	Technology
Frontend	React; Axios HTTP client
Backend	Python; FastAPI (REST APIs)
Database	PostgreSQL; file storage for uploads and XAI outputs
Detection models	MobileNetV2, R3D-18, Wav2Vec2, DistilBERT
Explainability	Grad-CAM, SHAP, LIME
RAG	LangChain; ChromaDB; embeddings; LLM

VI. RESULTS AND DISCUSSION

A. Demonstrated Functionality

The implemented system was demonstrated at the project review. Its results are functional: for each supported content type, the platform accepts an upload, returns a prediction with a confidence value, and displays the corresponding explanation. Table III lists the demonstrated functions.

TABLE III
 FUNCTIONALITY DEMONSTRATED BY THE IMPLEMENTED SYSTEM

Module	Demonstrated function
Authentication	Registration and login; dashboard access after successful authentication
Detection	Image, video, audio, news and review upload with real/fake (or genuine/suspicious) prediction and confidence
Explainability	Grad-CAM, SHAP and LIME results shown with the prediction
Knowledge base	Document upload and indexing; question answering with structured bullet-point responses
Review forensics	Genuine/fake classification, influential words and bot-analysis information
History and reports	Previous analysis results available to the authenticated user

Quantitative measures such as accuracy, precision, recall, F1-score and inference latency have not yet been reported for the implemented models, so none are claimed here. Each of these measures requires a defined test set and protocol, which is the next experimental step.

B. Discussion

The results show that detection, explanation, retrieval and review analysis can be combined in one usable application. Because explanations are generated from each prediction and saved with the file, a user can review the evidence behind a result after the session. The RAG module gives users access to context from their own indexed documents, but it provides background information rather than a verdict, and the quality of its answers depends on the documents in the knowledge base. Storing predictions and reports per user supports repeated review of results.

C. Limitations

The main limitations are as follows. First, there is no quantitative evaluation, so detection performance and cross-dataset generalisation are unknown. Second, the models operate separately and the fusion model reserved in the architecture is not implemented, so the system does not yet give a single combined verdict across modalities. Third, the explanation methods have not been assessed for their reliability. Fourth, the details of the bot analysis feature are limited.

VII. CONCLUSION AND FUTURE WORK

This paper presented DeepShieldAI, an implemented multimodal system for analysing potentially manipulated digital content. It supports image, video, audio, news and review analysis with deep learning models, provides Grad-CAM, SHAP and LIME explanations, offers a RAG knowledge base, and includes review forensics and bot analysis, all within a React, FastAPI and PostgreSQL web platform. The results presented are functional demonstrations rather than benchmark measurements.

Future work includes training and testing the models on larger and more diverse datasets to measure robustness, introducing multimodal fusion to give a unified prediction across audio, video, image and text, and optimising the system for faster real-time audio and video analysis. Further directions are enhancing bot detection with additional behavioural and linguistic features, expanding the RAG knowledge base with verified and trusted sources, and providing stronger evidence citation and source verification for generated answers.

REFERENCES

- [1] Yangxiao Bai and Kaiqun Fu, "A large language model-based fake news detection framework with RAG fact-checking," in *Proc. 2024 IEEE Int. Conf. Big Data*, IEEE, 2024.
- [2] Abinash P., N. Krishnamoorthy, and V. Vinodkumar, "Interpretable deepfake detection: Enhancing real vs. fake face classification with explainable AI and transfer learning," in *Proc. IEEE ICHET*, IEEE, 2025.
- [3] Krupanshi Patel, Meet Sutariya, and Palak Parmar, "A comprehensive survey on explainable deepfake detection: Techniques, challenges, and future directions," in *Proc. IEEE AIMV*, IEEE, 2025.
- [4] D. Nageswar Rao *et al.*, "AI-powered real-time misinformation detection: A deep learning framework for combating fake news and deepfakes," in *Proc. IEEE ICMCTC*, IEEE, 2025.
- [5] Abhishek Wagh, Rashmi Rane, and Latika Pinjarkar, "Enhancing deepfake detection through deep learning and explainable AI," in *Proc. IEEE AIMV*, IEEE, 2025.