ADROSONIC®

www.adrosonic.com

AI-Powered Synthetic Identity & Deepfake Claim Detection System

Detecting AI-generated and manipulated content in digital insurance claims.

24-Hour Hackathon Challenge · Fraud Detection and AI

24 hrs	2–4	Pre-trained	Working UI
CHALLENGE DURATION	TEAM SIZE	MODELS ALLOWED	REQUIRED

01 PROBLEM BACKGROUND

Insurance companies process thousands of digital claims every day: accident photos, vehicle damage images, medical reports, and identity documents. With the rise of generative AI tools like Midjourney, Adobe Firefly, and open source deepfake models, fraudsters can now fabricate convincing fake content at scale.

Traditional fraud detection systems were not built for AI-generated content. This creates a critical gap in claim verification, and that is the gap you are being asked to close.

02 YOUR CHALLENGE

Build a working prototype of an AI-powered fraud detection system that can analyze uploaded images and documents and flag potentially synthetic or manipulated content with a risk scoring engine. You may either build and train your own models from scratch or leverage pre-trained models, depending on project requirements, available timelines, and resource constraints. The primary focus should remain on designing an effective detection pipeline, developing robust risk-scoring logic, and delivering an intuitive user experience.

> **KEY PRINCIPLE**
> While custom model development is an option, greater emphasis should be placed on overall solution design and product execution. A well-integrated, end-to-end pipeline, whether powered by a pre-trained or custom-trained deepfake detection model, demonstrates stronger engineering and product thinking than a sophisticated model that is poorly integrated into the user workflow.

03 SCOPE

A. Image Manipulation Detection (Core, Must Have)

- Use either a pre-trained deepfake detection model or a custom-trained image manipulation detection model to classify uploaded images as authentic or AI-generated / manipulated.
- Provide a confidence score for each prediction to indicate the likelihood of manipulation.
- Flag images that exceed a defined risk threshold and present the corresponding confidence level to the user.
- Optionally highlight suspicious regions if your chosen model supports it.

B. Document Tampering Detection (Core, Must Have)

- Analyze uploaded PDF or image documents for signs of tampering: inconsistent fonts, altered values, metadata anomalies.
- Use OCR (pytesseract or similar) to extract text, then apply rule-based or ML checks for inconsistencies.
- Flag suspicious fields with a reason.

C. Risk Scoring (Core, Must Have)

- The system must generate an image authenticity score, a document authenticity score, and an overall fraud likelihood, along with clear and interpretable reasons for flagging suspicious cases.
- The scoring logic must be explainable: show why a score was assigned.

D. Demo Interface (Core, Must Have)

- A simple web UI where a user can upload files and see results.
- Streamlit, Gradio, or a basic React / Flask frontend are all acceptable.
- Show fraud score, flagged indicators, and a short plain-English explanation.

E. Identity Comparison (Bonus, Nice to Have)

- Compare faces across uploaded ID documents and selfie images to detect mismatches or morphing.

> **SCOPE BOUNDARY**
> You may choose to either build and train your own models or leverage pre-trained models, based on your team's expertise, available data, project timeline, and resource constraints. The primary objective is to demonstrate the effectiveness of the fraud detection workflow and analysis logic through a functional prototype using representative sample data.

04 CHALLENGES

Hackathon Challenge

Teams are expected to build a functional prototype or demonstrable system within the allocated hackathon duration. The use of open-source tools, frameworks, and publicly available models is encouraged. Submissions should be complete enough to demonstrate core functionality and feasibility.

AI and Model Challenge

Participants are required to design solutions that effectively detect AI-generated and manipulated content using advanced techniques. Purely rule-based approaches are discouraged. Models should demonstrate the ability to generalize across unseen or novel fraud patterns, considering the rapidly evolving nature of generative AI technologies.

Data Challenge

Participants will have access to a limited set of authentic images, and real-world fraud datasets will not be fully available. Teams are expected to generate synthetic or tampered data for training and validation purposes. Solutions must also be robust enough to handle variations in data quality, including low-resolution images, noise, compression artifacts, and partially incomplete inputs. Few resources for reference have been provided.

05 EVALUATION CRITERIA

Criterion	Target	Weight
Detection accuracy on sample data	Correctly flags known fakes	25%
Risk scoring logic and explainability	Clear reasoning per flag	25%
UI / demo quality	Usable upload and results view	20%
Code quality and architecture	Readable, modular, documented	15%
Innovation and bonus features	Identity comparison, novel techniques	15%

06 DEMO REQUIREMENTS

Your 10-minute final demo must walk evaluators through:

1. Upload an image; show the detection output and confidence score.
2. Upload a document; show tampering flags and reasons.

3. Show the composite risk score (low, medium, high) with an explanation.
4. Briefly walk through your architecture and key decisions.
5. If implemented, show identity mismatch detection.

07 IMPACT

- Reduces insurance fraud losses by catching AI-generated fake claims early.
- Minimises manual review effort for claims teams.
- Builds explainable AI outputs that investigators can trust and act on.

ADROSONIC Build · adrosonic.com

ADROSONIC®

THANK YOU

www.adrosonic.com