**Architectural Decision Record (ADR)**

**1. Title**

**Adoption of AI-Based Fraud Detection Using FAISS, LSTM, and Isolation Forest**

**2. Context**

Fraudulent activities such as **answer collusion, time-based answer manipulation, and automated bot submissions** pose risks to the integrity of the aptitude test system. Currently, there is no automated fraud detection mechanism, and expert review is manual and time-consuming.

We need an **AI-based fraud detection system** that:

* **Detects collusion** (candidates copying answers from each other).
* **Flags time-based answer manipulation** (candidates modifying answers suspiciously).
* **Identifies bot-like automated submissions**.
* **Consolidates results for expert review**.

The fraud detection system will analyze data from **two databases**:

1. **Aptitude Test Ungraded Database** (Stores ungraded MCQ & short-answer responses).
2. **Submission Ungraded Database** (Stores candidate submission timestamps and modification history).

A **Fraud Processing Module** will run a **daily cron job** to detect suspicious activity using multiple AI models and store flagged cases for **expert review**.

**3. Decision**

We have decided to integrate a fraud detection pipeline using the following components:

**AI Models Chosen**

| **Fraud Type** | **Detection Method** | **Why This Model?** |
| --- | --- | --- |
| **Answer Collusion (Copy-Pasting)** | **FAISS + SBERT** | Detects **semantic similarity** between different candidates' responses. |
| **Time-Based Answer Manipulation** | **LSTM (Recurrent Neural Network)** | Identifies **sequential answer changes** over time. |
| **AI-Based Fraud Detection (Bot Detection)** | **Isolation Forest (Anomaly Detection)** | Flags **unusual submission patterns** (e.g., high-speed multiple submissions). |

**Workflow**

1. **Fraud Processing Module** runs a **daily cron job** on **Aptitude Test Ungraded Database + Submission Ungraded Database**.
2. **FAISS (with SBERT embeddings) detects answer collusion**.
3. **LSTM detects candidates who repeatedly change answers in suspicious patterns**.
4. **Isolation Forest flags bot-like rapid submissions**.
5. **Flagged cases are stored for expert review**.
6. **Expert manually verifies fraud alerts before taking action**.

**4. Architecture Impact**

* **New Components:**
  + **FAISS Vector Database**: Stores **SBERT-generated embeddings** for answer similarity detection.
  + **LSTM Model**: Tracks **sequential answer modifications** for time-based fraud detection.
  + **Isolation Forest Model**: Detects **bot-like activity** in submission logs.
  + **Fraud Processing Module**: Runs **AI models via a daily cron job** and stores flagged results.
  + **Expert Review Dashboard**: Allows administrators to **verify flagged fraud cases**.
* **Integration Points:**
  + **FAISS integrates with the Aptitude Test Ungraded Database** to **identify similar answers**.
  + **LSTM uses the Submission Ungraded Database** to **analyze suspicious answer modification sequences**.
  + **Isolation Forest processes both databases** to **detect automated answer submission patterns**.
  + **All flagged fraud cases are stored in the Fraud Review Database**.
* **Performance Considerations:**
  + FAISS **vector search must return results in <2 seconds**.
  + LSTM must efficiently **track sequential answer modifications**.
  + Isolation Forest must detect anomalies **without excessive false positives**.

**5. Risks & Mitigation**

| **Risk** | **Impact** | **Mitigation Strategy** |
| --- | --- | --- |
| FAISS may falsely flag paraphrased answers as collusion. | Medium | **Set a similarity threshold (e.g., cosine similarity < 0.95)** to reduce false positives. |
| LSTM may misidentify valid answer modifications as fraudulent. | Medium | Fine-tune LSTM **with expert-verified training data** to improve accuracy. |
| Isolation Forest may generate too many false positives for rapid submissions. | High | **Adjust contamination level** to **optimize anomaly detection sensitivity**. |

**6. Acceptance Criteria**

| **Requirement** | **Description** | **Acceptance Threshold** |
| --- | --- | --- |
| **FAISS Accuracy** | Correctly detects copied answers | **≥ 90% match with expert-reviewed cases** |
| **LSTM Accuracy** | Flags manipulated answers correctly | **≥ 85% match with expert evaluations** |
| **Bot Detection Accuracy** | Flags automated answer submissions | **≥ 80% anomaly detection success** |
| **Processing Time** | Fraud detection pipeline runtime | **≤ 5 minutes per daily batch** |
| **Expert Review Accuracy** | Agreement between AI and human fraud review | **≥ 85% precision in flagged cases** |

**7. Implementation Plan**

* **Phase 1:** Set up FAISS database and SBERT embeddings.
* **Phase 2:** Train & fine-tune LSTM model for answer modification tracking.
* **Phase 3:** Implement Isolation Forest for bot detection.
* **Phase 4:** Develop **Fraud Processing Module** to automate fraud detection.
* **Phase 5:** Integrate **Expert Review Dashboard** for fraud validation.
* **Phase 6:** Deploy and monitor fraud detection performance.

**8. Decision Status**

* 🟢 **Accepted** (Implementation in progress).

**9. Related Documents**

* AI-Based Fraud Detection System Architecture Diagram.
* FAISS Clustering & Answer Similarity Detection Report.
* AI Model Performance Evaluation & Threshold Tuning Document.