Refining Time Series Anomaly Detectors

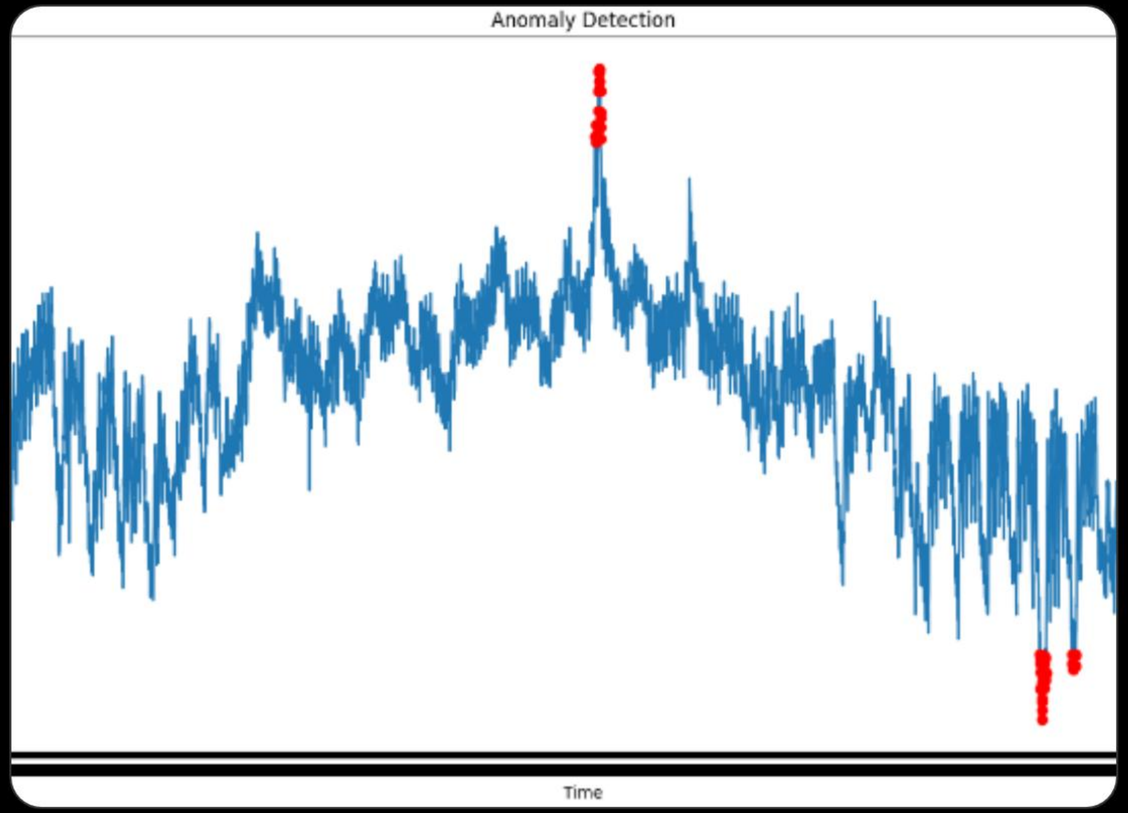
A Human-in-the-Loop Approach using Multimodal Large Language Models (LLMs) to Reduce False Positives.

Based on the paper by Yang et al. (2025)

The Problem: False Alarm Fatigue

Time series anomaly detection (TSAD) is critical for finance, healthcare, and manufacturing. However, current automated methods suffer from a major flaw:

- Detectors are "paranoid" to avoid missing real issues.
- This leads to a massive volume of **False Positives**.
- Human experts must manually review thousands of charts.
- **Result:** High operational costs and "alert fatigue."



The Solution: Human-in-the-Loop Automation

Don't Replace, Refine

Instead of building a new, expensive detector, we keep the existing "cheap" detector (like k-NN) to cast a wide net.

LLM as Analyst

We use a Multimodal LLM (Llama 3.2 Vision) as a "virtual analyst" to review the candidate anomalies found by the first detector.

Filter False Positives

The LLM's job is simple: Look at the chart, read the context, and decide if it's a real anomaly or just noise.

The Two-Stage Pipeline

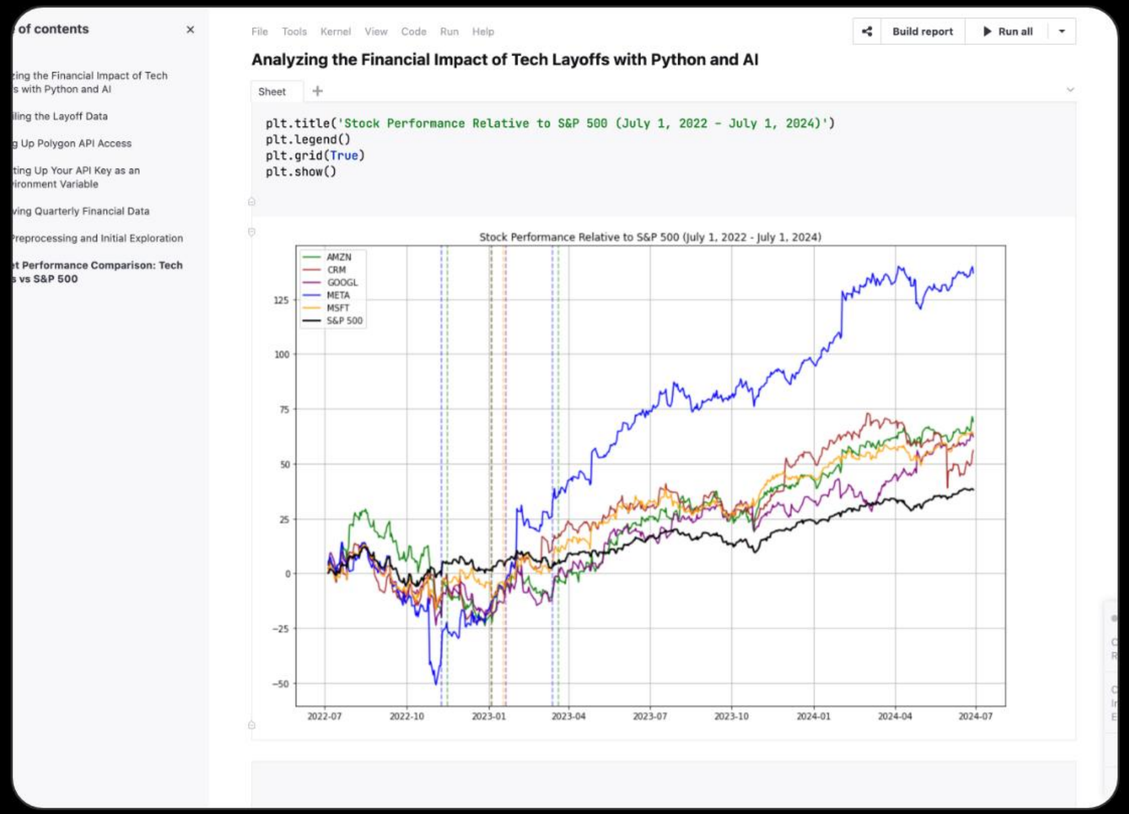
Stage 1: The Net (Detector)

Goal: High Recall. Catch everything.
Tool: k-Nearest Neighbor (k-NN).
Output: A list of "Candidate Intervals" that *might* be anomalies.

Stage 2: The Filter (LLM)

Goal: High Precision. Remove fakes.
Tool: Llama 3.2 Vision.
Input: Visual plot + Textual Context.
Output: "True Positive" or "False Positive".

Why Vision Matters



LLMs need to "See" the Data

Traditional text-only LLMs struggle with time series data because they just see a long string of numbers (e.g., "0.5, 0.6, 0.2..."). They can't perceive "shape."

The Visual Prompt:

- **Blue Line:** The actual observed data.
- **Green Line:** The predicted/expected behavior.
- The LLM visually compares the deviation between the two.

Context is King

The Prompt Structure:

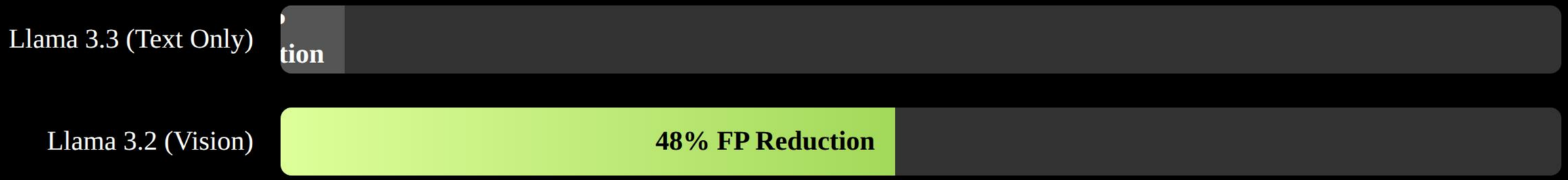
"Does the blue time series have the same shape as the green time series?

If 'Yes', the data should match the following description:
[Dataset Context]: 'This data consists of a mixture of normal heartbeats and premature ventricular contraction...'"

By explicitly telling the LLM *what* the data represents (e.g., "Heartbeats"), we ground its reasoning in domain knowledge.

Ablation Study: Vision vs. Text

Does the model really need to see the chart? Yes. Text-only models failed to filter correctly.



Vision-enabled models were nearly 10x more effective at identifying false alarms.

Key Results

48%

Reduction in False Positives

85%

True Positives Retained

2

Modalities (Vision + Text)

The system successfully filters out nearly half of the noise while keeping the vast majority of critical alerts.

Case Study: Heartbeat Analysis

The Setup

Dataset: ECG Heartbeat data.
Context: "Periodic normal beats vs. PVC (Premature Ventricular Contraction)."

The Outcome

The k-NN detector flagged a slightly irregular beat as an anomaly. The LLM looked at the shape, compared it to the "Predicted" normal beat, and correctly identified it as a **False Positive** because the deviation wasn't significant enough to be a PVC.

LLM Reasoning:
"The blue line follows the general trend of the green line..."

Future Directions

Conversational Agents

Moving from simple "Yes/No" classification to a chatbot that can explain *why* an anomaly is happening in plain English.

Autonomous Tuning

Using the LLM's feedback to automatically tune the parameters (k, threshold) of the base detector.

Multivariate Data

Expanding from single-variable charts to complex, multi-sensor dashboards for industrial IoT.

Conclusion

We don't need to replace traditional algorithms; we need to augment them.

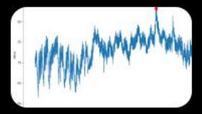
By giving LLMs **Vision** and **Context**, we can automate the "human" part of the anomaly detection loop.

Questions?

Refining Time Series Anomaly Detectors using Large Language Models

Thank you for watching!

Image Sources



https://media.geeksforgeeks.org/wp-content/uploads/20230506140813/anamoly-detection-using-line-plot.png

Source: www.geeksforgeeks.org

https://blog.jetbrains.com/wp-content/uploads/2024/07/Tech_layoffs.png

Source: blog.jetbrains.com