Paper Review of

Grammar-Based Anomaly Detection of Microservice Systems

CSE713 - Advance Pattern Recognition

Tanvir Rahman (22241134)

BRAC University

PAPER OVERVIEW

Core Identification

Title: Grammar-Based Anomaly Detection of Microservice Systems Execution Traces

Authors: Andrea D'Angelo & Giordano d'Aloisio

Conference: ICPE '24 Companion

Publication Context

Published in May 2024, focusing on distributed trace analysis for modern software infrastructures.

Research aimed at bridges the gap between ML accuracy and human interpretability.

1. SUMMARY - MOTIVATION

The Problem

Trace volumes in microservices are exploding. Manual investigation is no longer feasible or scalable.

Black-Box ML

Current high-accuracy ML models are computationally expensive and lack interpretability for operators.

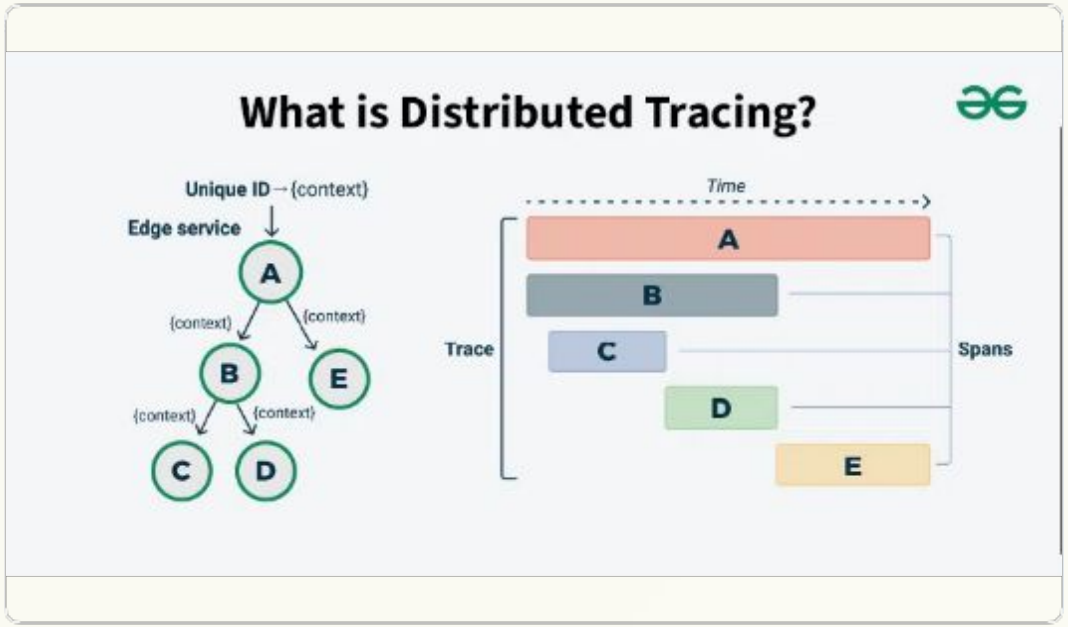
The Objective

Deliver efficient, explainable anomaly detection that reduces training overhead while maintaining accuracy.

1. SUMMARY – CONTRIBUTION

SAX + Sequitur Approach

- Converts numerical latency into discrete symbolic patterns.
- 15s faster training than Logistic Regression with ~5% accuracy loss.
- Provides human-readable Context-Free Grammar rules.



1. SUMMARY - METHODOLOGY

Stage 1: SAX

Latency discretization using Symbolic Aggregate Approximation into 5 specific bins.

Stage 2: Induction

Sequitur algorithm constructs Context-Free Grammar from the symbolic traces.

Stage 3: Testing

Traces are checked for membership in the Grammar. Non-members are flagged as anomalies.

1. SUMMARY - RESULTS

Empirical Evaluation

Evaluated on **E-Shopper** and **Train-Ticket** datasets.

- Significant reduction in training latency.
- Competitive F1, Precision, and Recall scores.

Diagnostic Output

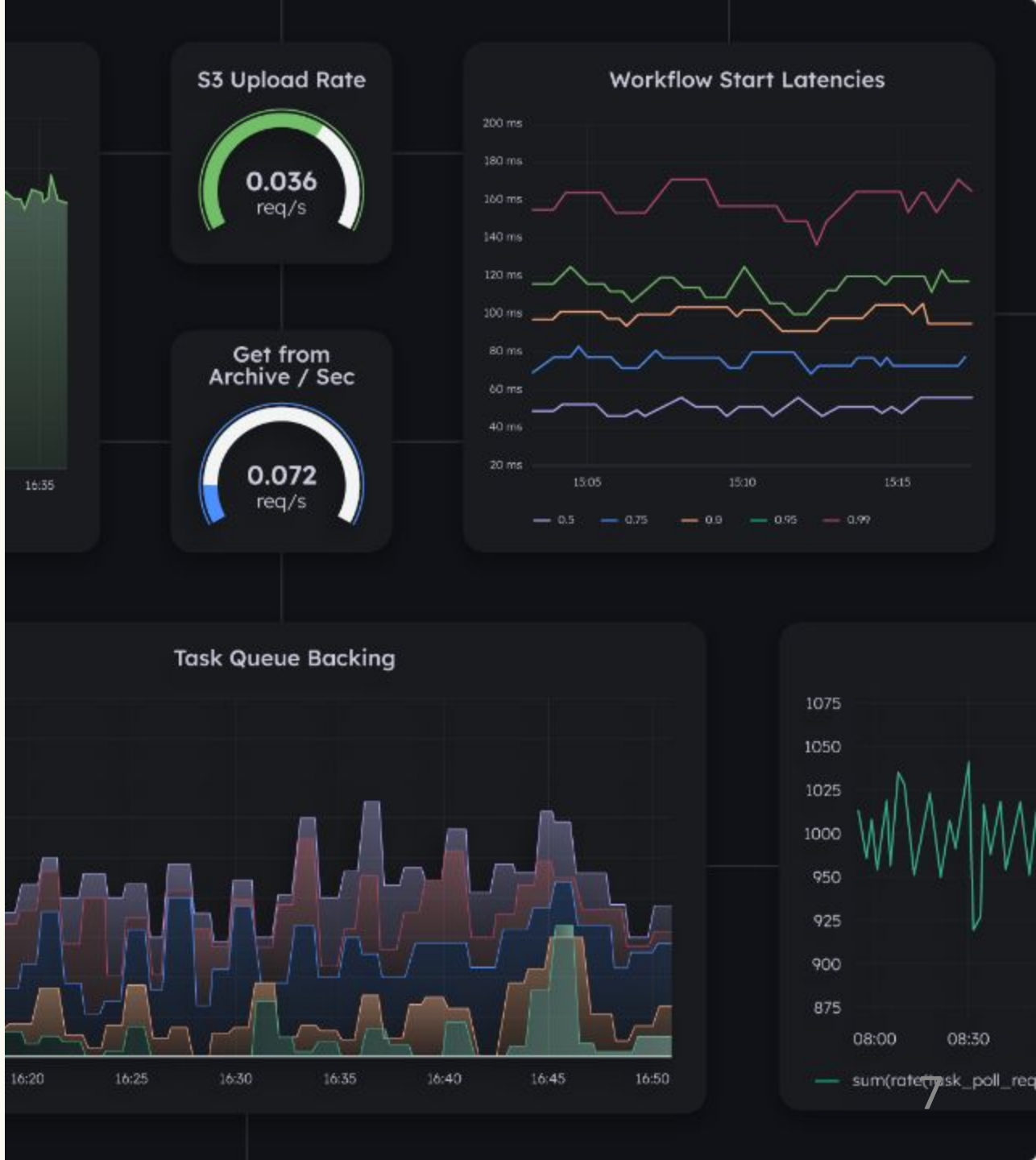
The system generates parse trees that help pinpoint where pattern deviations occur in the execution flow.

2. CRITIQUE: FEATURE MYOPIA

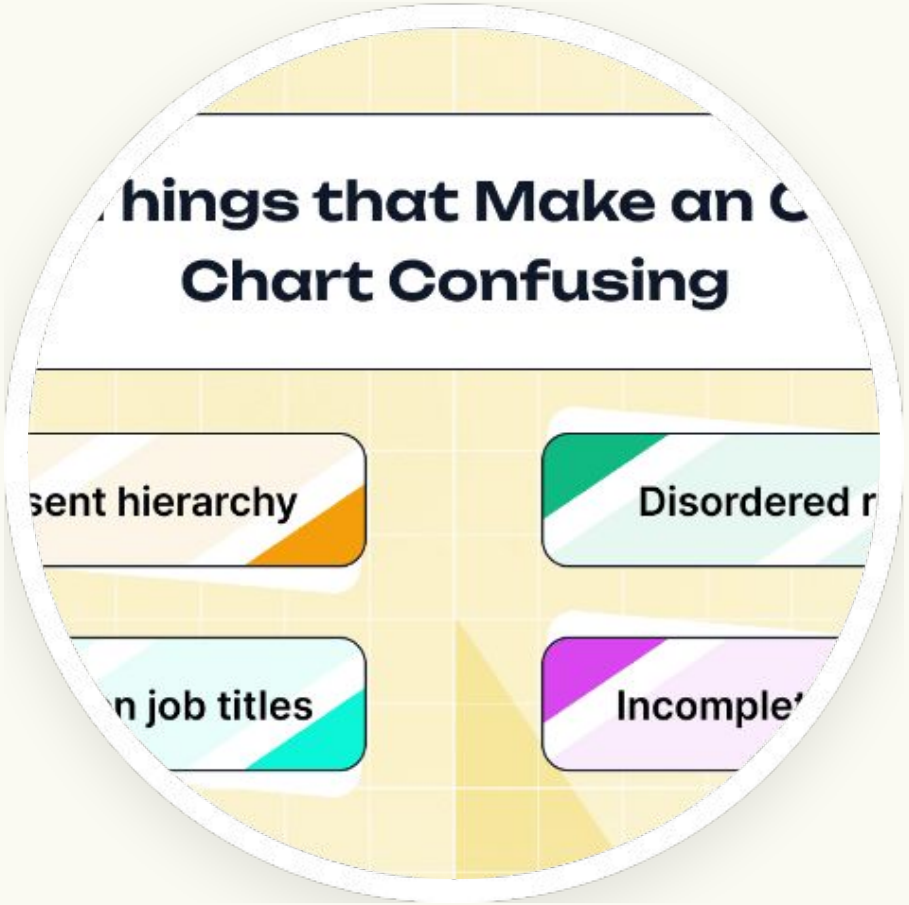
Latency-Only Limitation

The paper relies **exclusively** on temporal (latency) data. It fails to account for service topology or call graph dependencies.

Impact: Bottlenecks caused by circular dependencies or cascading failures are missed if they don't produce unique latency signatures.



2. CRITIQUE: THE DIAGNOSTIC GAP



Ambiguous Root Causes

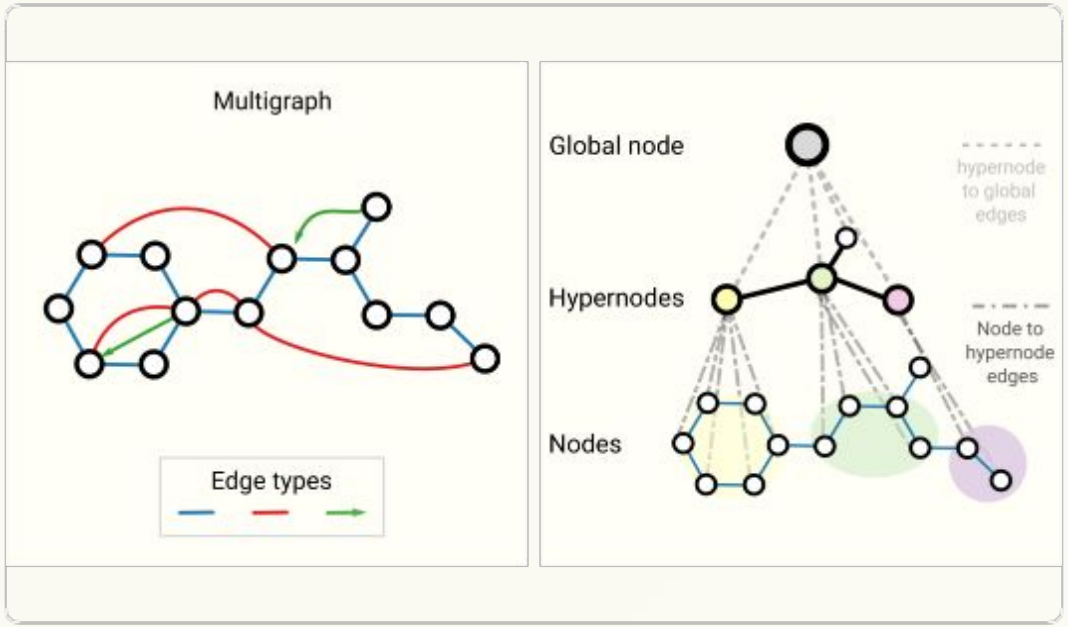
Parse trees identify **pattern violations**, but they do not identify **which** specific service failed or **why**.

The Sequitur grammars are often large and non-optimal, creating a "data fog" rather than clear diagnostic instructions for SREs.

3. SYNTHESIS: THE XHYBRID PROJECT

A Multi-Modal Future

- **GNN:** Incorporates service topology and dependencies.
- **LSTM:** Enhances temporal sequence recognition.
- **SHAP:** Quantifies feature contributions for true RCA.



SYNTHESIS: IMPLEMENTATION STACK

Frameworks

PyTorch, PyTorch Geometric, and NetworkX for graph modeling.

Processing

saxpy and sksequitur for grammar-based feature extraction.

Explainability

SHAP integration to highlight critical services for operators.

Thank you

Tanvir Rahman (22241134)

Brac University | CSE713 Pattern Recognition

Paper DOI: 10.1145/3629527.3651844