Assessment of Real-World Incident Detection Through a Component-Based Online Log Anomaly Detection Pipeline Framework

Scott Lupton¹,², Lena Yu¹, Hironori Washizaki², Nobukazu Yoshioka², and Yoshiaki Fukazawa²

¹Nomura Securities Co., Ltd., Tokyo, Japan
²Waseda University, Tokyo, Japan
scott.lupton@nomura.com, lena.yu@nomura.com, washizaki@waseda.jp
nobukazuy@acm.org, fukazawa@waseda.jp

Abstract—This study introduces an open-source, component-based pipeline framework for online log anomaly detection. It implements popular parsing, encoding, and anomaly detection methods as replaceable components, and compares their performance to industry-standard, rule-based methods using real-world incident log data. The goal of this study is to assess the suitability of using modern log anomaly detection methods for industry system monitoring.

Keywords—anomaly detection, online, logs, component, framework, pipeline

1. INTRODUCTION

Rule-based approaches to process monitoring have many weaknesses: rule maintenance is time-consuming and it is difficult to craft rule definitions that are capable of detecting unforeseen future incidents. Modern log anomaly detection techniques have the potential to address these shortcomings, and thus a comparison of these methods using real-world incident log data would be beneficial. The motivation for this study is to assess the appropriateness of using log anomaly detection workflows for real-world industry system monitoring.

There is "a lack of anomaly detection frameworks that can be deployed and used online" [1]. In this study, we introduce such a framework and provide it open-source¹. While online log parsers and anomaly detection methods have been frequently tested with public log datasets, there is a shortage of real-world use-case studies [2]. To fill this gap, we implement online parser, encoder, and anomaly detection methods as components for use in the pipeline and compare them to current industry-standard, rule-based monitoring methods using real-world log data. Through this comparison, we assess the practicality of using anomaly detection methods for system monitoring and provide suggestions for future research.

2. RESEARCH QUESTIONS

The objective of this paper is to address the following research questions:

- **RQ1**: How do modern log anomaly detection methods perform in comparison to industry-standard, rule-based monitoring approaches when applied to real-world log data?

- **RQ2**: What challenges exist in utilizing log anomaly detection methods for production-grade process monitoring and what future work would help address these challenges?

3. FRAMEWORK DESIGN

The proposed pipeline framework (Fig. 1) consists of a sequential collection of **Encoders** and **Detectors**. Encoder components inherit from an abstract parent class implementing **encode** and **feedback** methods. These components are responsible for performing transformations on input data. Examples of encoders include regex replacement functions, semantic embedding methods, and filters. Log parsers (i.e., template extraction methods) also function as encoders within this design. Detectors are responsible for returning detected signals (i.e., anomalies) from input data through the implementation of the abstract **detect** method. Like with encoders, detectors can perform incremental model updates or other corrective adjustments through the implementation of the **feedback** method. Note it is possible to configure multiple detectors in a series but ensemble methods are out of the scope of this study.

4. EXPERIMENT

We compared log anomaly detection methods to rule-based monitoring using our component-based pipeline framework. The experiment used a front-office industry process log containing one year of data (57,951 entries) covering the occurrence of one known production incident: a disruption of service caused by high resource utilization. We manually extracted anomalous logs from the incident period and labeled them for use as ground truth. All experiments were performed on the same workstation with the following specifications: Intel(R) Xeon(R) Silver 4110 CPU @ 2.10 Ghz (2 processors), 64 GB RAM.

He et al. stress that "the industry is more concerned with whether any failure is observed and disrupts the system functionalities" than what particular log entries are detected as anomalous [3]. Indeed what we deem most important is that failures are detected as early as possible and false positive signals are minimized. To measure for these criteria, we have included false positive counts (FP) and incident detection delay (DD) as metrics in our experiment along with standard accuracy measures (i.e., precision, recall, accuracy, and F1).

We test two versions of case-insensitive, rule-based detection using our standard alert keyword set (containing the
one incident can produce multiple anomalous logs, discovering the first of those signals is arguably more important than the total volume of anomalies detected. Thus, judging the strength of a model on accuracy alone can be said to be misguided.

Regarding RQ2, high false positive rates and log drift are the most significant challenges to utilizing anomaly detection methods for industrial log monitoring. Feedback mechanisms could potentially address these issues, and future studies evaluating the effectiveness of such mechanisms would be beneficial. Another area of concern is the time and resources necessary to train deep-learning models. GPU resources are costly and CPU-based training can be slow. The mitigation of these resource-related issues is another significant topic of importance.

6. CONCLUSION

In this study, we introduced a component-based online log anomaly detection pipeline framework and used it to compare modern anomaly detection methods to rule-based monitoring approaches. We performed an evaluation using a real-world industry process log and found that while anomaly detection methods have the potential to better detect unforeseen issues (i.e., high false positive rates, log drift, and long training times). We hope that this study helps to provide direction for future research and supports the industrialization of log anomaly detection methods going forward.

REFERENCES


478