LOG 6309E - Intelligent DevOps

Group Assignment: *Log-based Anomaly Detection*

Notes

- This assignment is done in a group of three or four students.

- Weight: 15% of the final grade.

- Deliverable: A 5-8 page IEEE report in PDF. A template is available on Moodle.

- Assignment Presentation – Tuesday, OCT 17, 2023

- Assignment Report Due – Monday, OCT 23 11:59 PM

Objectives

- Understand and apply the concepts and techniques of log parsing.

- Understand and apply the concepts and techniques of log-based anomaly detection.

- Perform an empirical evaluation study of existing techniques.

- Understand and apply basic

- Improve the skill of writing a scientific report.

- Practice teamwork in a research project.

Specification

This assignment requires you to partially replicate and extend a recent paper [1] related to log-based anomy detection. The paper describes a pipeline for log-based anomaly detection and evaluates the impact of different components (parsing, representation, etc.) within the pipeline.

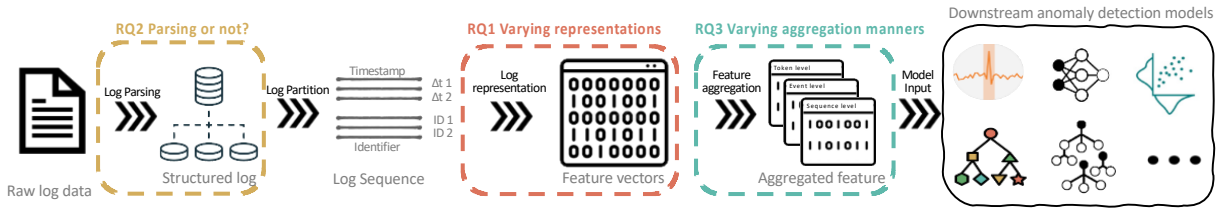

Figure 1: A pipeline for log-based anomaly detection [1]

Replication. Replicate RQ1 of the paper [1] – build log-based anomaly detection models through the following three steps.

1) Parse the raw log data into event templates (*a.k.a.*, log parsing or log abstraction). You can use Drain or other log parsers (you may find a list of log parsers here). Note: pre-processing and post-processing may be before/after using a log parser.
2) Represent the parsed log data as numeric features. Choose the appropriate log representation methods (e.g., *Message Count Vector*) according to your downstream models. You only need to consider one representation method for each model.
3) Construct and evaluate log-based anomaly detection models. You are required to consider four machine learning models, including at least one classical model and at least one deep leaning model. You should use four standard evaluation metrics: precision, recall, F1-score and AUC).

You should perform the experiments on two log datasets, out of the four used in the paper. You can leverage the replication package of the paper in your work. Compare your results with that in the original paper.

Extension. Extend the analyses performed in the replication. Perform the following analyses on one log dataset.

1) Use statistical ranking (e.g., Scott-Knott test) to rank the models by their performance (e.g., F1-score or AUC). Note: you can produce a statistical distribution of the performance of each model through resampling the training/testing datasets multiple times.
2) Analyze and explain the important features of the models (e.g., permutation importance or Gini importance). You may also use LIME or Shapley values to explain individual predictions (optional). You may only consider the model with the best performance or the best interpretability.
3) Apply correlation analysis (e.g., hierarchical clustering) and redundancy analysis (e.g., variance inflation factor, or VIF analysis) to remove collinearity and multi-collinearity among the features before constructing the models. Compare the performance of the models with and without this step.
4) Compare the performance evaluation results when using a random splitting and a time-based splitting. The same splitting ratio (e.g., 70%/30%) should be used for the comparison.

Writing tips:

- Describe your approach in a concise and unambiguous way: others should be able to repeat your experiments following your report.
- Justify your design decisions (the selection of the models, representation techniques, evaluation metrics, and statistical analysis methods).
- Don't just present your results, but also explain your results and discuss their implications. If you cannot explain a result, then something is probably wrong.
- Highlight a few take-home messages that you want readers to learn from your results.

Paper to be replicated and extended:

[1] X. Wu, H. Li, and F. Khomh, "**On the Effectiveness of Log Representation for Log-based Anomaly Detection**," *Empirical Software Engineering* (EMSE), 2023. PDF

Anomaly detection models to be considered:

Classical models:

Ref: Shilin He, Jieming Zhu, Pinjia He, and Michael R. Lyu. "**Experience report: System log analysis for anomaly detection**." In 2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE), pp. 207-218. IEEE, 2016.

Repo: https://github.com/logpai/loglizer

Deep learning models:

Ref: Chen, Zhuangbin, Jinyang Liu, Wenwei Gu, Yuxin Su, and Michael R. Lyu. "**Experience Report: Deep Learning-based System Log Analysis for Anomaly Detection**." arXiv preprint arXiv:2107.05908 (2021).

Repo: https://github.com/logpai/deep-loglizer

Datasets:

Use two of the datasets used in the replicated paper [1] (e.g., the HDFS log and the BGL log datasets). The datasets can be found in the following repo:

Datasets: https://github.com/logpai/loghub