PROJECT REPORT ON TOPIC

TITLE

Machine Learning based approach for:

Log Analysis or Review, Anomaly Detection.

Minor Project Report by:

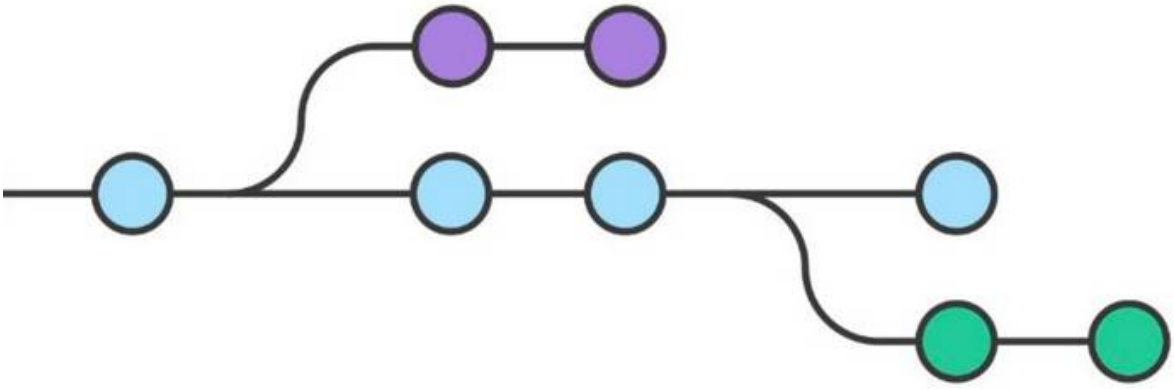
Name: Vicky Dattatray Jadhav.

Course: MCA 3rd Semester. Year 2023

Enrolment Number: A9929722000210(el)

Table of Contents

1. Abstract:



Since AI / ML started evolving, it has been really helpful in resolving and handling complex task. Out of which one of the task is Long Review, Log Analysis, Root cause finding, Anomaly Detection, In fact my Focus is to Analysis the set of logs from larger number of Machines or Network of Machines to find out what is the root cause of the issue or to find the error and resolve the issue based on the findings, but to perform such an automation on logs is a very complex task and still an on going challenge. With this Abstract, I am offering the precise log analysis by Using Machine Learning and finding the root cause of the issue, Challenges, to examine the fundamental principles and further developments in this area.

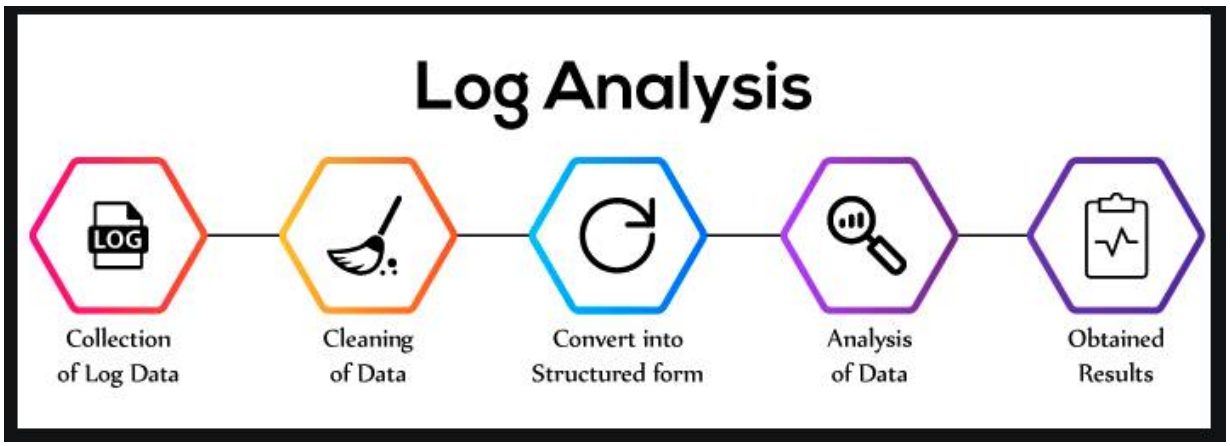


Image 2 - Log analysis process traditional method.

How log review is done currently (Traditional method): As per the traditional method, we as Infrastructure Engineer, Software Quality tester, Developers, Site Reliability Engineer (SRE's) and Support Engineers would manually copy the logs from few affected machine and review those logs manually, by opening the logs in Text editors or log review tools, this is a time consuming and a slow process and one should have enough patience to go through these logs line by line.

Why log review is a problem: On a windows machine we have lots of different types of logs, like EventViewer logs which are in .etl file format, Windows Dump, Application Dump, which requires WinDgb like tools, other system logs, application logs, process logs etc.. Which is not at all easy and difficult to read in plain english format, these are mostly unstructured and very large data sets. In our company we have 100's of sites located across the world, we have a huge variety of users : they are using huge amount of softwares, applications, on various platforms like Virtual, could, on-prem, saas, daas, which includes multiple application vendors, and these type of system datasets contains millions of lines across tons of files.

Example few of the pain points:

- Some of the traditional methods is to write custom, powershell scripts, VB script, BasScript or Shell script using grep, awk etc., to find specific data. This may not always yield what you were hoping to find from the logs.
- Trying to understand what's happening across multiple different log files is challenging task at best requires a high performance machine, to open lots of windows, files and scrolling through lakh's of lines of logs needs lot of patience.
- We are currently using the tools like Aternity, ServiceFirst, ElasticSearch, Splunk, Kibana, Grafana etc.. to speed-up finding something when you know what you are looking for, but troubleshooting is not like that. And these tools requires that you understand the basic underlying data structures and parse out the parts that are important. We also have to write custom powershell scripts, SQL Queries to fetch the data from the logs. This is a complex and time consuming task.

- One of the challenge to list out is to perform the anomalies detection where the log data is Unlabelled data. Solution for this task is to label the data manually, but at times these logs are huge and not easy to understand the logs in the first go.

Real world scenario is - Assume we have lakh's of computers in a network and huge number of machines having some sort of Application installation failure error or some service specific error. Or Assume we are deploying some application and on 50k devices the deployment is failing. As one of the subject matter expert in this field, I would manually access few (5 to 10) devices at the max and review the associated logs like: **Event viewer, Deployment logs, Application failure logs etc**.. and trying to find out what could be the root cause of the deployment failure.

Here comes the actual issue – Hypnotically, based on the investigation & findings done on these 10 to 15 devices, I am going to assuming, that the issue or the root cause is same for all 50k devices which reported installation failure.

In reality, it is possible that few of them are failing because of network failure issue, Disk Full issue, or could have some conflict with other application, or possibly the machine is turned off during deployment, or could be any other issue. But as a Human, I have not reviewed all 50 devices and it is not possible for a human to go through such a huge data set.

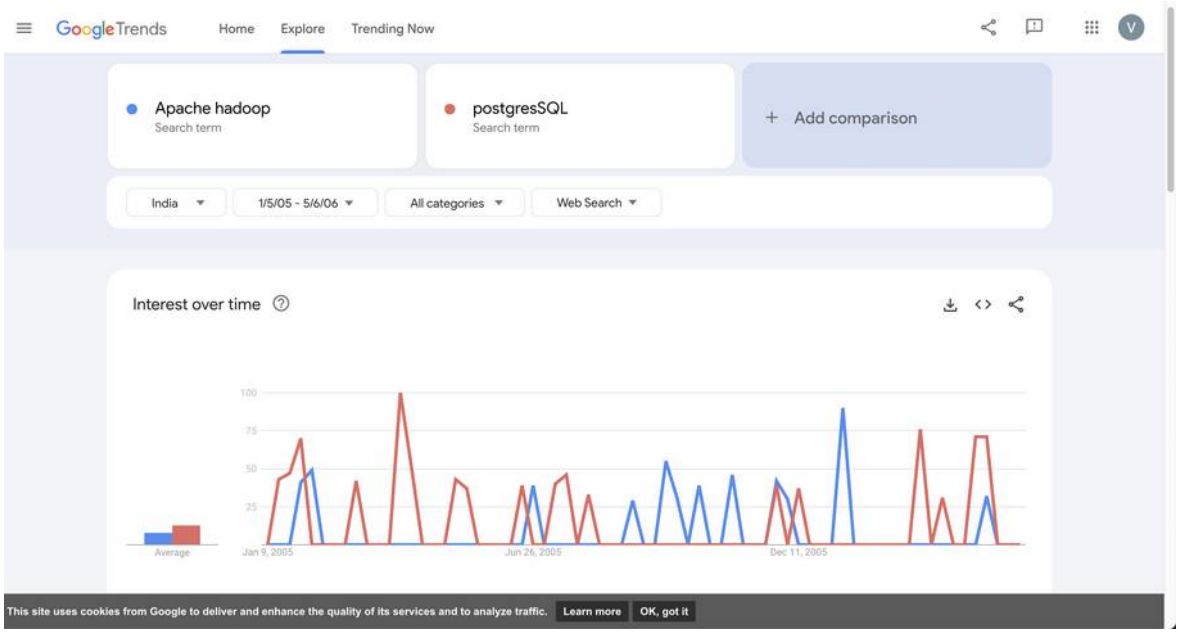


Image 3: Searches on the most popular open-source tools for Structured Data (Red) and Unstructured (blue).

Solution: (I am talking about the problem statement which I have described above) Is to use Machine learning (Structure is the key), to review the logs on all 50k machines or lakhs of machine to find out similar patters or similar errors in the logs. We can train the machine learning module by feeding in the logs from these 50k affected devices and other devices logs where the installation was successfully done. This algorithm will then perform the log analysis from both working and non working machines and find out what is the actual root cause of the issue. This will be more accurate and precise findings. Will save lot of time, money and result will be happy customer who get to use a reliable solution.

We absorb, structure, and auto-analyse system data: logs, stats, events and configs which are collected from the test runs. We then create and load a relational database on which you can query. The machine learning pipeline not only structures the data into tables with columns for the variables, it does anomalies detection that actually works, this then keeps the db schema and signatures up-to-date. Because the logged format of the given event might change over the time.

2. Introduction:

2.1 Purpose of this document

The purpose of this document is to provide a detailed project documentation of the Solution called Automate the Log Analysis and Anomaly Detection to find the Root Cause of the Issue Using Machine Learning. This is designed to provide automated solution using Machine learning for Technical Support Engineer, Software quality testers, Engineers, Developers, on finding the root cause and automate log review on large scan and Find the anomalies, error patterns in the logs. Which will then save lot of time and money and help us find the accurate root cause of the issue in less time. Users/customers will have a reliable solution to use.

2.2 Intended Audience

This document can be used in all phases of the project as a guideline.

Intended audiences of this project will be:

- Technical Support engineer.
- Software quality testers.
- Individual Computer user.
- Developers.
- Project Managers. Etc..

2.3 Scope

This Document defines the Minor Project Plan of the solution called, Automate the Log Analysis and Anomaly Detection to find the Root Cause of the Issue Using Machine Learning.

Introduction of the Study:

In windows operating system mostly, every application and process is capable of generating the logs and events. These are then used to find the root cause of the issue or the error. Based on these log findings we can troubleshoot , debug any issues related to software failure, installation failures etc., These logs and events can be used for anomaly detection or to find the specific pattern in the logs. **For example Time Stamp, Event ID's, Error codes, Severity of the logs, log levels,** But these are not sufficient information, some times few applications requires a few advance and verbose levels logs to be enabled manually, in order to capture the logs in more details. One need to be a domain expert in order to enable, generate such levels of logs. As of now these are all done manually, which is not a suitable method when it comes to lakh's of machine where the issue is getting reproduced. Therefore an automated method is needed to analyse these logs and identify the anomalies or root cause of the issue. The main aim behind this is to increase speed in solving issues, and quality of troubleshooting and accurate information. As per my understanding there is no such solution in real world yet. This is very complex and still an ongoing challenge. On the real world platform this problem is very complex when it comes to understand which is normal and anomalous classes.

For example these are few challenges given below:

- The rate at which these logs are generated.
- The lack of centralized or unified structure, as logs and events data are in larger variations and has many different patterns.
- The classification of the errors or the absence of labelled records.
- The current method is to manually labelled the data and this requires lot of human efforts and time. This is not a suitable solution for large and huge data set.

I am proposing the automated method using Machine learning (Structured) techniques with insight from domain experts and knowledge to provide more effective and better performing patterns and anomaly detection model.

What is its relevance in the current business scenario?

Using machine learning for log review and anomaly detection has significant relevance in the current business scenario. Here is why:

1. **Efficient Log Analysis:** Log files generated by systems, applications, and network devices contain valuable information about their operations and performance. However, manually reviewing and analyzing large volumes of log data is time-consuming and prone to human error. Machine learning algorithms can automate log analysis, enable efficient processing, identification of patterns, and extraction of actionable insights.
2. **Anomaly Detection:** Machine learning algorithms can be trained to detect anomalies in log data. By learning from historical log patterns, ML models can identify deviations from normal behaviour, indicating potential security threats, system failures, or unusual activities. Anomaly detection can help organizations proactively address issues, prevent downtime, and improve overall system reliability.
3. **CyberSecurity:** Log analysis and anomaly detection play a crucial role in cybersecurity. Machine learning can help identify malicious activities, such as intrusion attempts,

unauthorized access, or data breaches, by analyzing log data for patterns indicative of security threats. ML models can learn from known attack patterns and adapt to emerging threats, enhancing the detection and response capabilities of security systems.

4. **Operational Efficiency:** By automating log analysis, machine learning can streamline troubleshooting and problem resolution processes. ML algorithms can quickly identify root causes of system issues by analyzing log data.
5. **Business insights:** Machine learning applied to log data can uncover valuable business insights. By identifying usage patterns, customer behaviour, or operational trends from logs, organizations can gain a better understanding of their systems, applications, and user preferences. These insights can be used to optimize processes, improve user experience, and make data-driven business decisions.
6. **Proactive Maintenance:** Machine learning models trained on log data can help predict failures or performance degradation in advance. By detecting early warning sings in logs, organizations can take proactive maintenance actions, schedule repairs or replacements, and prevent unplanned downtime. This approach enhances system reliability, reduces costs associated with reactive maintenance, and improves customer satisfaction.

To summarised the proposed solution:

In Summary, machine learning for log review and anomaly detection offers numerous benefits in the current business scenario, including efficient log analysis, enhanced cybersecurity, improved operational efficiency, actionable insights, proactive maintenance, and better decision-making capabilities. By leveraging machine learning techniques, organizations can gain valuable insights from their log data, mitigate risks, and optimize their operations.

- Fast and efficient methods and algorithms, in real-time situations to detect the root cause in such time critical situations and infrastructures.

- Accuracy using Machine learning algorithms. Is the key.
- No need to take any decision based on assumptions. Like review 10 to 15 machine logs and assume the issue will be same in other lakhs of machine across the network.

3. Objectives of the ML-Based log review and anomaly detection:

Objective:

The Objective of studying machine learning-based log review and anomaly detection is to harness the power of advanced computational techniques to enhance the analysis, interpretation, and understanding of log data. Here are some key objectives of studying this field:

- **Improved Log Analysis Efficiency:** By studying machine learning-based log review, researchers aim to develop algorithms and methodologies that automate the analysis of log data. The Objective is to reduce manual effort, increase efficiency, and handle large volumes of log files more effectively.
- **Enhanced Anomaly Detection:** An important objective is to develop machine learning models that can identify anomalies in log data with higher accuracy. The goal is to improve the detection of security breaches, system failures, or unusual actives, enabling organizations to take timely actions and mitigate potential risks.
- **Early anomaly detection:** In most disastrous events, there's always an initial anomaly that wasn't detected. Machine learning can detect the anomaly before it creates a major problem.
- **Real-time Monitoring and Response:** Studying machine learning-based log review seeks to enable real-time monitoring of log data and the development of automated systems that provide instant alerts or notifications for critical events. The objective is to facilitate proactive response and minimize the impact of security incidents or operational disruptions.
- **Pattern Recognition and Insight Generation:** Another objective is to leverage machine learning techniques to identify meaningful patterns and insights within log data. By studying

log review methodologies, researchers aim to develop models that can extract valuable information, detect trends, and facilitate decision-making processes.

- **Reduced False Positive and Noise:** Researchers aim to improve the accuracy of anomaly detection algorithms by reducing false positive and noise in log data analysis. The objective is to develop machine learning models that can filter out irrelevant information and prioritize alerts based on their significance and potential impact.
- **Stability and Adaptability:** Studying machine learning-based log review focuses on developing scalable and adaptable systems that can handle diverse log sources, varying log formats, and evolving IT environments. The objective is to create methodologies that can seamlessly integrate with different systems and accommodate the growing complexity of log data.
- **Continuous Learning and Improvement:** An objective is to explore techniques for continuous learning and improvement of machine learning models for log review. The aim is to develop methodologies that can adapt to changing log patterns, incorporate feedback and update their knowledge base to enhance accuracy and performance over time.

Overall, the objective of studying machine learning-based log review and anomaly detection is to leverage advanced computational techniques to automate an enhance the analysis of log data. By achieving this objective, organization can improve security, operational efficiency, and decision-making processes while effectively managing the challenges associated with the ever-increasing volume and complexity of log information.

4. Background and Literature:

The background and literature review for the topic machine learning-based log review and anomaly detection involve exploring existing research, methodologies, and advancements in the field. Here are some key aspects to consider:

Background:

The importance of log analysis and anomaly detection lies in its ability to provide valuable insights, enhance security, improve operational efficiency, and support decision-making processes. Log analysis involves examining and interpreting log data generated by various components of an information system, such as servers, applications, network devices, and databases. Logs contain detailed records of events, errors, transactions, and user activities. By analyzing these logs, organizations can gain valuable visibility into system behavior, identify trends, troubleshoot issues, and optimize performance.

Anomaly detection, on the other hand, focuses on identifying abnormal patterns or behaviours within log data. It helps organizations detect potential security breaches, system failures, or performance bottlenecks that may indicate malicious activities, technical glitches, or emerging issues. Anomalies can include unusual network traffic, atypical user behaviours, excessive resource utilization, or unexpected error patterns.

The Importance of log analysis and anomaly detection lies in their ability to provide early warning, prevent system failures, mitigate risks, and improve overall systems performance.

Traditional methods of log analysis in large organizations:

Traditional methods, typically involve manual review, rule-based approaches, and statistical analysis. First we need to define log analysis itself, and see why it is crucial for companies. In fact, log analysis is reviewing and making sense of computer-generated log messages, such as log events or audit trial records (generated from computers, networks, firewalls, application servers, and other IT systems).

It’s used by organizations to improve performance and solve issues. It also mitigate a variety of risks, responds with security policies, comprehends online user behavior, and conduct forensics during an investigation.

Manual log analysis depends on the expertise and experience of the person running the analysis. If they are expert in domain, they may gain some momentum in reviewing the logs manually. However, this has serious limitations. Company has to be dependent on single person.

Current Techniques:

Machine learning-Based approaches:

Machine learning techniques have gained significant popularity in log analysis. Supervised learning algorithms, such as decision trees, support vector machines, or random forests, can be trained on labeled log data to classify and detect anomalies. Unsupervised learning algorithms, such as clustering or anomaly detection methods, can discover patterns or anomalies in log data without the need for labeled data. Deep learning approaches, such as neural networks, can capture complex relationships and patterns in log sequences. By adopting current techniques such as machine learning, NLP, time series analysis etc.. organizations can overcome the limitations of traditional log analysis methods. These approaches enable more efficiency, accuracy and scalable log analysis, facilitate timely anomalies detection, incident response, and proactive security measures.

Research Methodology of Machine learning based Log analysis and Anomalies detection:

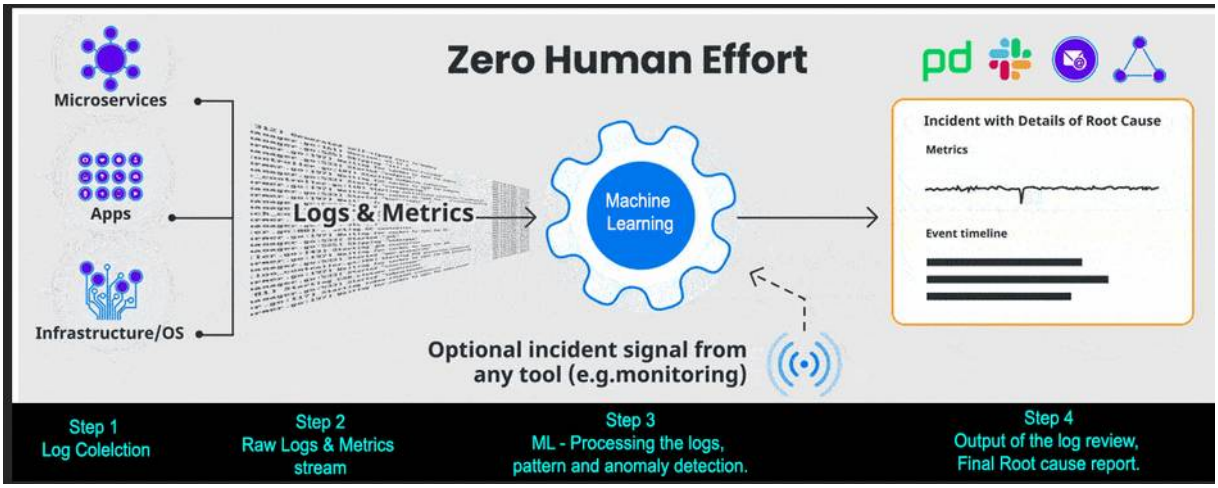


Image 4: Log and Metrics go In, Incidents and root cause come out.

5. Research Methodology: for Machine learning based log analysis and Anomalies detection research: Describing the problem:

Humans, troubleshoot software problems by digging into the logs and events and finding the error, root cause or fault, unexpected events etc.. based on these findings we as humans try to fix the problem, but there is a huge limitation here and it is very tough job as a human to review 1000'+ lines of logs and events and also keep a track of time, date and other check points through out the process. This is one of the complex task and we need to spend hours in going through these logs, its a time consuming and the major problem or the primary problem here is, Scenario is: In multinational organization, Imagine lakhs of machine across various different location faces issues with some software or while deploying some updates or patches few lakhs of machine crashes and starts facing issue. Then, as a Human we cannot review the logs from all those lakhs of machine. So currently, what we are doing is from the list of those affected machines we take 10 to 15 machines randomly and review the logs from those remotely and try to find out what is the root cause of the issue. Based on the review of those 10 to 15 minutes we assume that's what ever we found in these machine we take one common point and assume that's the same issue on all those remaining affected machine, which could be in lakh's across the various different location in planet earth.

We as human has limitation and we are slow when it comes to data. We need help from machine to perform such complex task, on such plethora of data.

Thus, machine learning is one of the suitable solution in this situation. As mentioned above, we feed the data to machine learning and it will learn the data and give us the results and predictions.

How it will be in real life?

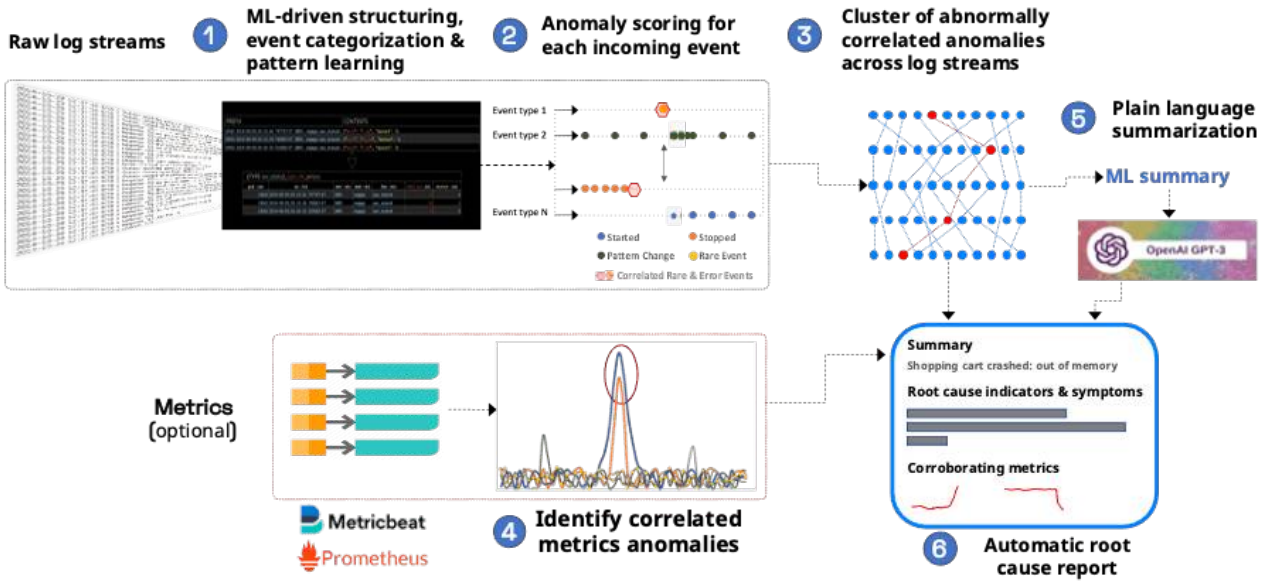


Image 5: Complete overview of machine learning based log analysis and anomaly detection.

Solving the above problem:

The flow of the over all process as shown in the above Image 5:

1. Collect and feed the raw log data.
2. ML-Driven Structuring. Event categorization & pattern learning.
3. Anomaly Scoring for each incoming event.
4. Cluster of abnormally correlated anomalies across log streams.
5. Identify correlated metrics anomalies.
6. Plain language summarization.
7. Automatic root cause report.

Understanding the above 7 steps in details:

5.1. Data Collection.

Data will be collected with various method here, basically every company has its own centralized log collection system. We can connect our system via cloud based storage and suggest the user to configure their log collection tool to send the logs directly to our cloud based solution.

Some of the Of the shelf solutions available: as given below

Step 1 – Ingest the logs and events and categorization:

1. Using Fluentd which is an open-source data collector for a unified logging layer. Fluentd allows you to unify data collection and consumption for better use and understanding of data.
2. Or Alternative is using Prometheus - Prometheus collects and stores its metrics as time series data, i.e. metrics information is stored with the timestamp at which it was recorded, alongside optional key-value pairs called labels.
3. Or Fork a copy of logs using LogStash - Logstash is a free and open server-side data processing pipeline that ingests data from a multitude of sources, transforms it, and then sends it to your favourite "stash."

One doesn't need any kid of parsers, code changes, rules or config are need. After this let the machine learning take over the entire process ahead.

Further Machine learning learns the structures of the logs and categorization of each event into a dictionary of unique xevent types. Here categorization is very important for accuracy of pattern learning of the logs and metrics.

5.2. DATA SAMPLING:

Step 2 – Patter and Anomalies Detection:

The above collected log data and data sets are further sent for Data Sampling: Say like assume within the first couple of hours, the patterns of each type of log event and metric are learnt well here the learning will keep on improvising based on the data we feed, more the data more the learning will improve.

When the pattern of any log event or metric changes (Say like for example: Changes in period or frequency, new or rare messages starts, etc..), it is scored as to how anomalous it is, but these anomalies tend to be very noisy. In order to separate signal from noise, the ML then looks for hotspot of abnormally correlated anomalies across the matric and logs.

6. DATA ANALYSIS & INTERPRETATION:

Step 3 - Augment (Make (Something) greater by adding to it. Increase)

If some one use an Incident Management tool like ServiceFirst, SalesForceCRM, PagerDuty, Opsgenie or Slack, or an existing log management or monitoring too, then this system can also augment any INC with a characterization of root cause mentioned in its subject or description etc.. We can configure it to send the signal to the ML based system when an Incident occurs. Or one can manually trigger the signals. After this, this system with start finding any root cause reports or sets of anomalies or anomalous log / metric patterns then coincide with the signal, and automatically feeds the information back to your Incident management tool. Basically, here someone who really cares and knows about the Mean Time to Detect (MTTD) and Mean Time to Resolve (MTTR). Then you are familiar with monitoring, incident response, war rooms, and MIM calls etc.. 1) All software system have bugs and 2) It's all about how you respond. While some companies (users / customers) may be sympathetic to #1, without exception, all of them still expect early detection, acknowledgement of the issue and near-immediate resolution. Well this solution is going to be an Unsupervised Machine learning to automatically detect and correlate anomalies and patterns across both logs & metrics.

Example: If you are using the PagerDuty INC Response system:

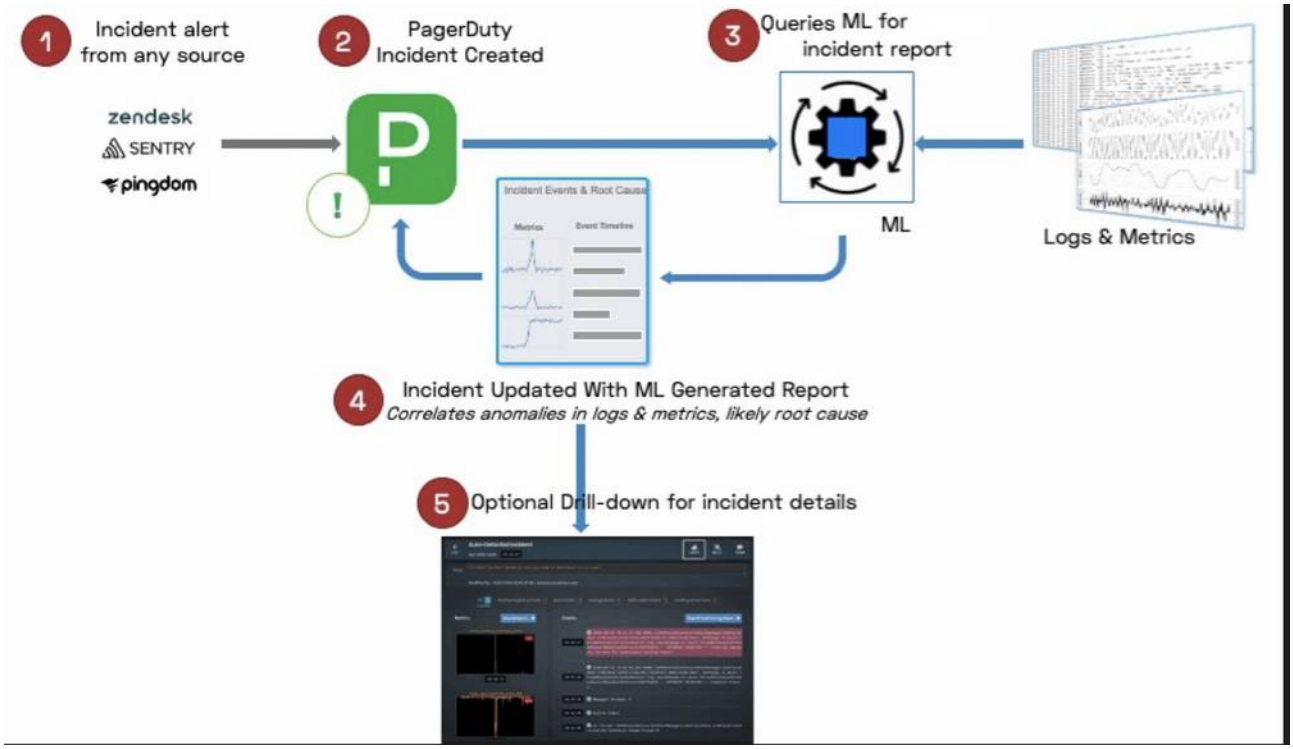
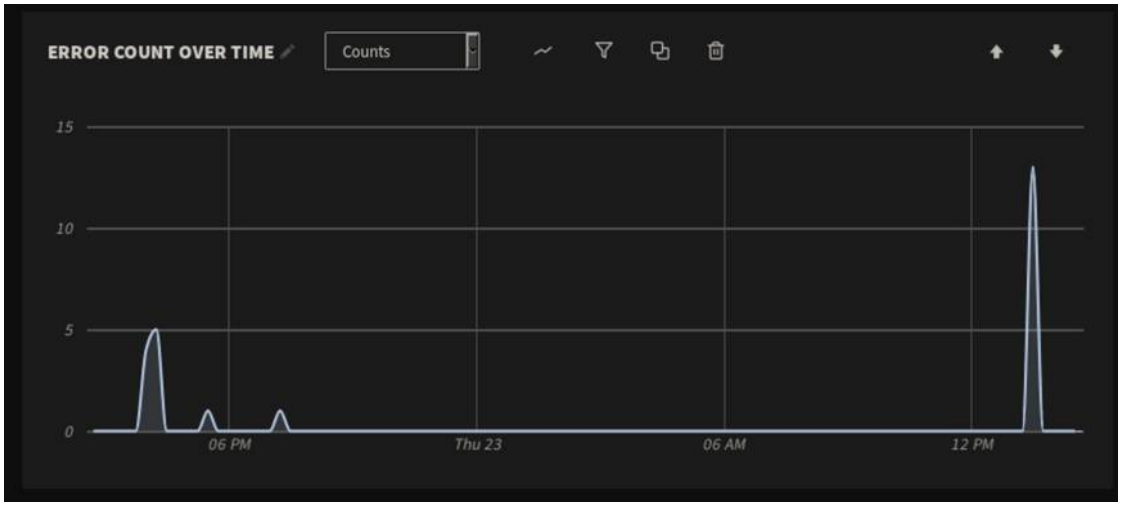


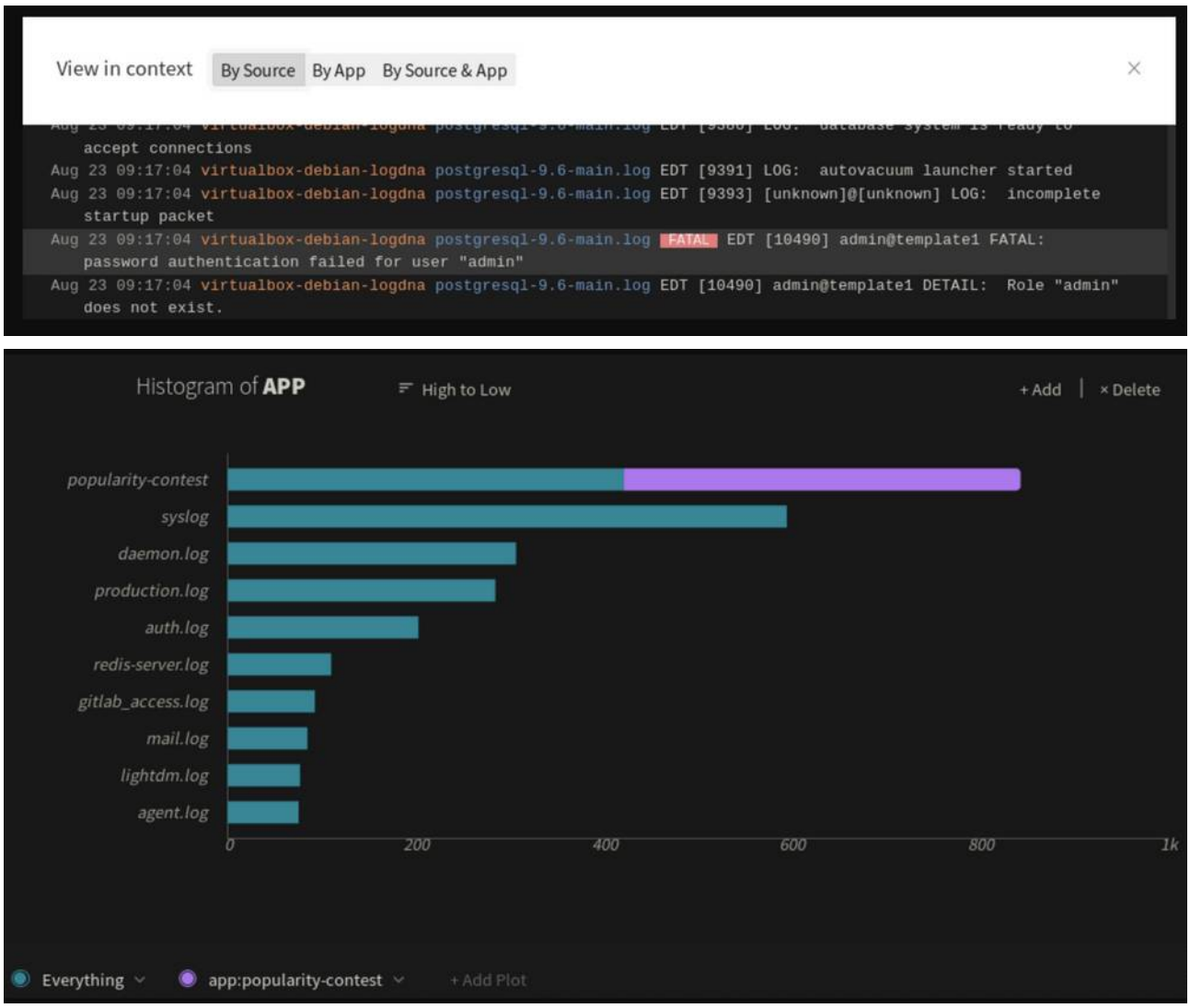
Image 6: Quick example of an Automate the Incident Response and Machine Learning to detect the root cause.

Step 4: Root cause Report or End Result in report format.

The hotspots detected above are compiled and packaged into concise root cause reports that contain root cause indicators, symptoms and correlated anomalous metrics.

Well we can also create a plain simple english language summary of the report using OpenAI's GPT-3 Language Model.





As presented in the above figures, The pre-processing requires, log uploader to be setup. If incase you have any external log collector or log monitoring tool, you can link it with this solution.

The logs needs to be collected based on the time stamp. Time stamp is very important in order to find the actual root cause of the issue.

I collected the logs from the lakhs of client machine located internally in our organization which is located all across the world. Basically, we have a centralized log collection tool configured within the organization like Splunk, Service First, Aternity, Tanium etc. These log collection database has plethora of data. Using these tools (Aternity, Tanium, Splunk) I can query and run the reports to fetch the logs in various formats: csv, excel, pdf or text.

This logs then gets stored on a Cloud Storage, depends on how huge the data is. It will take some time to upload the data.

Once the logs are stored, it will then start processing the data. This will then start labelling the data based on the time stamp, type of logs etc..

Next will be, the Machine learning Algorithms will start training its model and will start finding the anomalies and patterns and errors in the logs. This model will basically perform the Unsupervised machine learning on log to automatically find the root cause of the issue in the softwares or services. It does not requires any manual rules or training.

The Artificial-Intelligence machine learning AI / ML engine analyses the logs, events looking for similar patters, abnormal log line clusters, then resemble problems, such as abnormally correlated rate and error events from across all log streams.

Next Based on its analysis, it will generate the compressive report , suggestions and will display on the portal. This report will show us the root cause of the issue, and will let you know what is the actual error.

This trained models will be then able able to predict if there will be any more issues etc..

Domain experts and engineers will then take appropriate action based on these reports.

7. Discussion & Result

In todays time, the systems are running with highly critical applications, which are mostly connected to the cloud platform and always in continuos availability mode. When any of the service crashes or if any deployment fails, its a critical situations when the severity of these issues starts taking a critical and system down issues. In such cases, we don’t have the root cause of the issue right away. We have to fetch the logs, review the logs and based on analyses we assume that this could be the same reason for all the issues reported across the network. As a human we cannot spend time in reviewing huge log files across all the devices which has reported the issue, unless we have some automation in log review.

We are monitoring and setting up dashboards to find the common known or top 10 issues reported in the network. But in order to find the root cause we have to dig deep into the logs and find out what is happening. That's when this kind of Machine Learning based Solution is helpful. We feed the logs and data to these ML based Algorithm, train the model and it will start producing the results which is more accurate and easy to fix. We can also get the predictions from these in ML based algorithms and we can perform proactive measures, before applying any patches or updates on the machine etc. This will save Time, Money and increase the efficiency of the engineers. Lets take a quick look in this very basic example:

Perfectly structuring logs without parsing:

Developers, Testers, Engineers & DevOps constantly use log files and metrics to find and troubleshoot failures. But their lack of structure makes extracting useful information without data wrangling, regexes and parsing scripts and challenge.

Lets check this example here, this is a typical log line entry in /var/logs/system.log on my MAC machine, that contains useful information:

```
2   Jul  3 14:01:09 192 login[65867]
3   (com.apple.sandboxd[7344]): Service exited due to SIGALRM | sent by kernel_task[0]
```

Now, let's say you want to build a report that shows which services exited, and for what reason. For example, something like this:

REASON	SERVICE
SIGALRM	com.apple.sandboxd
SIGKILL	QA2G25RMZ4.com.wunderkinder.wunderlist-helper
SIGKILL	com.apple.AirPlayUIAgent
SIGKILL	com.apple.LocalAuthentication.UIAgent
SIGKILL	com.apple.OSDUIHelper
. . .	. . .
. . .	. . .

First you need to understand the structure of log lines like the one above, so that you can get the data needed for the report. With a bit of inspection (and some guessing) you might come up with (underlined CAPS represent parameters):

```
MONTH DAY HH:MM:SS HOST PROCESS1[PID1] (SERVICE[PID2]):  Service exited due to SIGNALTYPE | sent by PROCESS2[PID3]
```

You also need to be able to uniquely identify all occurrences of this "event type" in order to effectively parse out the data. This can be done by searching for some identifying text, for example "Service exited". I tried this with grep and it turned out to be too general as it also matched other similar events like this one:

```
21
22  MONTH DAY HH:MM:SS HOST PROCESS1[PID1] (SERVICE[PID2]):  Service exited due to SIGNALTYPE | sent by PROCESS2[PID3]
23
24  Jul  3 14:01:09 192 login[65867] (com.apple.WebKit.Networking.A4992C6F-8F69-4EA9-A031-76B032FB964F[7720]): Service exited with abnormal code: 1
25
```

Changing the search term to "Service exited due to" seemed to solve the problem, but I only tested this on a small log file, so it's possible that there are other event types that would also match this term. The important point here is that you need a way to uniquely identify an event type or your parsing will generate incorrect data.

Thus, this Machine Learning based solution is suitable in this scenario.

- Identify all unique event types (Typically thousands per application)
- Build a schema for the event type (Structuring, fixed text and variable parameters)
- Place each event type into a view where each parameter has its own typed column.
- All of this is built for simple query and is self-maintained as log file structure changes.

The Result will be:

Drag the log file referenced (/var/log/system.log) into this ML based log review solution's UI. Using just this file, and with no pre-learning or data wrangling, it will then perform the further analysis:

- It should be able to identify all unique event types and create a view for each.
- It should be able to build an index, and categorized each event type by topic, which made it really easy to find what I was looking for.

- Then it should be able to build a simple SQL query example :

```
SELECT DISTINCT _to AS Reason, _str AS Service
FROM v_service_exited_due_sent_by
ORDER BY Reason, Service;


Reason    | Service

----------+-----------------------------------------

 SIGALRM | com.apple.sandboxd

 SIGKILL | QA2G25RMZ4.com.wunderkinder.wunderlist-helper

 SIGKILL | com.apple.AirPlayUIAgent

 SIGKILL | com.apple.LocalAuthentication.UIAgent

 SIGKILL | com.apple.OSDUIHelper

 SIGKILL | com.apple.SystemUIServer.agent

 SIGKILL | com.apple.UserEventAgent-Aqua

 SIGKILL | com.apple.ViewBridgeAuxiliary
```

8. Recommendations and Conclusion:

Machine learning-based log review and anomaly detection is a powerful tool for identifying and troubleshooting problems in complex software systems. By analyzing large volumes of log data, machine learning models can learn to identify patterns that indicate anomalous behaviour. This can help to pinpoint the root cause of problems, even when they are difficult to identify manually.

There are number of different machine learning algorithms that can be used for anomaly detection. Some of the most common include:

- **Isolate forest:** This algorithm identifies anomalies by randomly partitioning the data into “Trees”. Anomalies are more likely to be found in the leaves of the trees, as they are more isolated from the rest of the data.
- **Local Outlier Factor (LOF):** This algorithm identifies anomalies by measuring the degree to which a point is isolated from its Neighbors. Anomalies are more likely to have a high LOF score, as they will be surrounded by points that are more similar to each other.

- **Autoencoders:** Autoencoders are neural networks that are trained to reconstruct their input data. Anomalies are more likely to be found in the data that they autoencoder is unable to reconstruct accurately.

The choice of machine learning algorithm will depend on the specific characteristics of the data and the desired level of accuracy.

In addition, to the machine learning algorithm, there are a number of other factor that can affect the accuracy of anomaly detection models. These include:

- The quality of the log data: The data must be clean and well-formatted in order for the models to learn effectively.
- The size of the training dataset: The larger the training dataset, the more accurate the models will be.
- The frequency of retraining: The models should be retrained regularly to account for changes in the behavior of the system.

As mentioned earlier, machine learning-based log review and anomaly detection is a powerful tool that can help to improve the reliability and security of complex software systems. By carefully considering the factors that can affect the accuracy of the models, organizations can use this technology to identify and troubleshoot problems more effectively.

Here are my recommendations for using machine learning for log review and anomaly detection:

- Use a variety of machine learning algorithm to improve the accuracy of the models.
- Use a large and representative training dataset to train the models.
- Retrain the models regularly to account for changes in the behavior of the system.
- Use a visualization tool to help identify and troubleshoot problems.

In this report I have presented the Automated log review, anomaly detection model combining, supervised and unsupervised Machine Learning with Domain Knowledge expert.

This model is composed of these following steps:

- Collect and feed the raw log data.
- ML-Driven Structuring. Event categorization & pattern learning.
- Anomaly Scoring for each incoming event.
- Cluster of abnormally correlated anomalies across log streams.
- Identify correlated metrics anomalies.
- Plain language summarization.
- Automatic root cause report.

There will be sill some challenges in this Automated log review process, like the logs and events are not same in every environment. It also varies based on the versions of softwares installed on the machine. Even though most of the enterprises uses same software but their environment is completely different then each other. In such cases, the ML algorithm will take a while to train the models based on the log sets. Also the process is going to be time consuming but fast enough as compare to humans.

My next focus will be to resolve the above challenges and to improve the model to make it more user-friendly and suitable for multiple environments.

I believe that machine learning has the potential to revolutionize the way that we monitor and troubleshoot complex software systems. By automating the process of log review and anomaly detection, machine learning can help us to identify and fix problems more quickly and efficiently. This can lead to improved reliability, security, and performance for our systems.

9. Bibliography & references

(1) Book:

Fuzzy Systems and Data Mining VIII

Anomaly Detection in Software Systems by Sebastian Raschka

Machine Learning for Anomaly Detection by Jason Brownlee

Log Analysis and Anomaly Detection with ELK by Matt Stansby

(2) **Journal Article:**

https://dzone.com/articles/using-machine-learning-for-log-analysis-and-anomal

https://www.researchgate.net/publication/370948902_LightESD_Fully-Automated_and_Lightweight_Anomaly_Detection_Framework_for_Edge_Computing

https://www.researchgate.net/publication/364983188_Automated_Log_Analysis_and_Anomaly_Detection_Using_Machine_Learning

(3) **Website:**

https://docs.fluentd.org/

https://prometheus.io/docs/introduction/overview/

https://www.elastic.co/logstash/

https://openai.com/blog/gpt-3-apps

https://www.loggly.com/

https://logentries.com/product/anomaly-detection/

https://sofecta.com/customize-siem/

https://github.com/SigNoz/signoz