ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

Anomaly Detection and Explainability with RAG and System Behaviour

Sudharsan R[*1] and Thara L[2]

[*1] II MCA, [2] Associate Professor & HOD

Master of Computer Applications, PSG College of Arts & Science, Coimbatore, Tamil Nadu, India

DOI: https://doi.org/10.51583/IJLTEMAS.2026.151100037

Received: 08 November 2026; Accepted: 13 November 2026; Published: 25 November 2026

ABSTRACT

Signature and rules based security software is good for ob. Served activities are similar to what was mentioned before; however they have some differences). Coverage is low for unknown behaviors. This project This paper discusses an alternative method for detecting an omalies within systems. Activity helps make the sea lertsmore understandable by activity itself. An The unsupervised Isolation Forests are used for synthetic data. generated dataset containing CPU usage, network traffic, login Attempts, counts of processes and files accessed QUESTBY. For each flagged obser Vation is an algorithm that transforms features to for, as a text-based search engine.And fetches relevant information about security awareness wareness BYC. base. Transformer Sentences Embedding Support for Simulation (FAISS).Search for illicit activities on content like MITRE ATTCK, etc. OWASP Top 10, malware behavior and network anomaly. A LangChain-based retrieval-augmented generation (RAG) work The flow goes through it to retrieve a context for rules-based threats detection.To an open source language model hosted on groq for producing readability of text.explanation. A Streamlit dashboard presents anomaly scores Chartsarts Disposter charts, an output sheet of data analysis report tables, explanation sheets and questions answerers. individual anomalies. Reported, as a synthetic data test report. Accuracy was 0.9990, Precision is 0.9524, and Specificity is 1.0000. Recall of 0.9756 F1-score and one false positives without any other errors. False negatives for 1,020 cases. These results A controlled controlled synthetic evaluation instead of an actual product. security performance. The prototype illustrates how behavioral Detection of detection and retrieval of domain knowledge for better results. Alerts are investigated by L1 IT professionals and other non-specialists. Validation BY European validation of representative real world log is still needed.

Keywords: anomaly detection, Isolation Forest, retrievalaugmented generation, explainable AI, cybersecurity, MITRE ATT&CK, FAISS, system behavior analytics

INTRODUCTION

Rule-based security detection is used for detecting activities that are linked with existing threat information. These techniques are still effective; however, an event which doesn't fit into a predefined model might go unnoticed. The gap is important because of changes to systems' behaviors, software environment and attacks techniques. Therefore detecting anomalous behavior would be an effective supplement for identifying known signs since there may not always be available labels on each type of threat. Machine learning offers ways to identify statistical deviations in system records. For example, unlabeled data may be used for training models through techniques such as self-supervision or semi-supervised learning where there are no explicit labels provided by humans. However, detectors usually provide scores or labels instead of explaining what makes them important for us to pay attention to. A blind alert is hard to prioritize by an L1 IT professional or administrator without specialized knowledge of cybersecurity. High scores/anomaly flags alone do not prove there was any kind of attack involved. The purpose of this work is to detect and interpret it. An Isolation Forests are used for detecting anomalies among system activity feature sets where an augmentation-based text-generation process follows it by obtaining information about cyber security issues

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

before providing explanations in human language form. The MITRE ATTCK model is part of this database whereas an attribute level mapping links some attributes to certain types of attacks. This tool will be used for human analysis instead of making independent decisions about incidents. This paper proposes an approach called ”Contribution” which combines unsupervised behavioral anomaly detection, retrieval-based explanations of anomalies and an interactive streamlit web application for analysis purposes. The evaluation of this paper is based upon its synthetic activities as per a given document called ”project report”. This article provides these findings without considering their relevance to performance on an operational network.

Related Work

Signature-and-rules based monitoring is a method of encoding known patterns beforehand. It gives a good starting point to match things, but there could be some cases where it doesn't work properly due to changes or new behaviors. The anomaly detection approach is based on evaluating an observation by its degree of conformity with existing pattern models from available datasets. An analysis may reveal behavior that is not associated with any particular pattern but does not necessarily mean it was done for nefarious purposes. Unsupervised learning applies when there are few available security labels due to high cost of production or lack thereof. Liu, Ting, and Zhou proposed Isolation Forest as a machine learning algorithm for anomaly detection which partitions data randomly to isolate outliers; these are usually found after few iterations compared with other methods [1].

It is suitable to use this as an effective example of behavior-based systems analysis due to its simplicity. The process is still required for selecting features, reviewing alerts, and validating a detector against similar environments' data. Explanation is about how to bridge this divide between what models produce versus decisions made by analysts. Anomaly detection requires an effective description of observed behaviors to differentiate between data records and interpretations thereof. Currently there is no distinction between context-based retrieval and an attribute attribution technique; instead of that we provide related document information along with anomalies for use by a language model. The retrieval augmented generation technique is based upon combining a language model with an external source of relevant documents for generating text that can be conditioned on retrieved context [2]. SentenceBERT style sentence embedding is an approach of representing texts as vectors for semantic comparison [3] whereas FAISS enables fast similarity search on vector collection [4].

Knowledge sources like MITRE ATTCK describe adversary tactics and techniques [5] while OWASP Top 10 is an overview of web application security threats [6]. These sources provide common terms; however, they don't show whether an anomaly is due to any given method of analysis. This is an integrated solution that combines all those tasks within this application for you to use easily. The main constraint for such a system would be that its anomalies may include false positives while retrieving data from indexes might suffer due to low index completeness or poor accuracy, and there could be an issue where generated texts overestimate retrieved information without proper review. Therefore this paper is concerned with generating explanation for research purposes instead of verifying incidents.

Problem Statement and Objectives

Problem definition: This is an issue of detecting unknown patterns through machine learning anomaly alerts that are hard to understand by humans. An effective prototype should detect anomalies of system behavior without needing label information to fit models and provide context so that users could review them.

There are many issues related to this project such as lack of labeled cyber security information; unseen or zero day attacks; false positive results which might be difficult for analysts to handle; changes in system behavior; problems with interpretability; and large volumes of system logs. Anomaly based detection is motivated by these problems; however it limits conclusions drawn from an artificial example. For example, an anomaly detection algorithm may separate out some types of anomalies from others but it doesn't mean there is always going to be a consistent

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

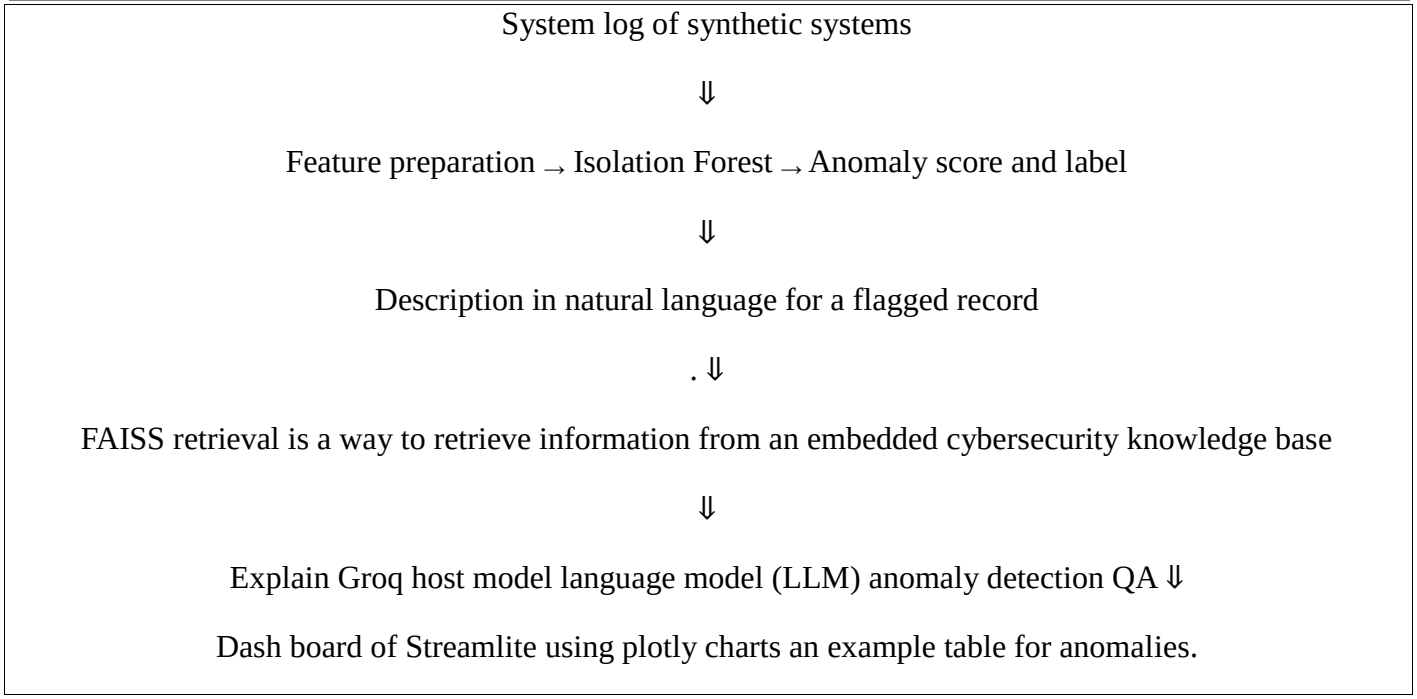


Fig. 1. Text-based architecture of the proposed anomaly detection and explanation workflow.

threshold between them for real world applications. Objectives include detecting anomalous behavior via unsupervised ML detection methods for identifying suspicious activities; providing human-readable contextualized explanations of flagged items; relating selected attributes to security models such as MITRE ATTCK methodology; displaying scorecards, charts, and anomaly logs on a web-based interface; and enabling users to inquire about specific incidents.

This document is for L1 IT professionals and non-experts to understand what they should do when an alert occurs before escalation or investigation.

Proposed Methodology

System Architecture

The system comprises a synthetic-data and detection path, a knowledge-retrieval and explanation path, and a user-facing dashboard. Activity records are representedbynumericalfeaturesandprocessedbyIsolation Forest. Flagged records are converted into textual descriptions. The text query is embedded and compared with embedded cybersecurity documents in FAISS; selected features are also mapped to candidate techniques using predefined rules. Retrieved content, the record description, and the mapping are provided to the language model, whose response is shown in the dashboard.

Synthetic Data and Features

Synthetic system activities are used for modeling both regular and irregular behavior of systems. This paper is about labeling of generated data sets for testing purposes but doesn't cover actual logging from an organization. Five features are given by this table. eftab:features. CSV files contain these data sets that will be used by our models to process them.

Table I. System Activity Features Described in the Project

Feature	Role in the activity record
CPU usage	Represents processor utilization.

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

Network traffic	Represents the volume of network activity.
Login attempts	Represents authentication activity.
Process count	Represents the number of active processes.
File access	Represents the frequency of file operations.

Data Preprocessing

Structured features are inputs to a model. There is no mention of normalization, missing value treatment, feature scaling, or an explicit definition for parameters in this paper so we cannot assume anything about them from it. The current step provides an uninterrupted model for detection purposes.

Isolation Forest Based Anomaly Detection

The following article will discuss about Isolated nodes and their role of being a part of tree data structures. According to this document, a machine learning algorithm is trained on an activity dataset where it classifies each observation as either normal or abnormal. The label used to generate synthetic data is supported by this study; however it isn't an appropriate description of a necessary labeling task for unsupervised classification problems.

Anomaly Score Generation

Each entry is given a value for anomalies by this model which then classifies them as either normal or abnormal. The anomaly occurs for low-scoring items whereas high scoring ones are found among average values of this dataset. Precise score transformation and configurable thresholds are unspecified here. Threshold control is mentioned on this page; however there are no values for it provided.

Query Generation

A detected record is represented by numeric values like CPU utilization rate and internet data flow rates in simple words. The question generation process links numerical detection and text retrieval; thus, a generated description may match what is found within an ontology database. The document doesn't claim to be a malicious act.

RAG Knowledge Retrieval and FAISS

Knowledge base includes information about cybersecurity like MITRE ATTCK, OWASP Top 10, malware behavior and network anomalies [5] [6]. As per this paper, texts are broken down into pieces for embedding generation by means of Sentence Transformer models.

Vectors are stored in FAISS; when an anomaly query is formed, its embedding is used for a similarity search to retrieve relevant chunks [3], [4]. The LangChain

Table II. Illustrative Rule-Level Threat Mappings in the Project

Observed feature pattern	**Candidate mapping in project material**
High CPU usage	Resource Hijacking (T1496)
High login attempts	Brute Force (T1110)

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

High network activity	Data Exfiltration (T1041)

manages an RAG process. The retrieval process gives you some information about what comes after this stage but doesn't confirm whether or not there are any threats to be found.

MITRE ATT&CK Threat Mapping

Alongside retrieval, there are some pre-defined features that link certain activities to specific methods of analysis. Project slide examples can be found on this page of table. eftab:mapping. These are examples of projects rather than generalizations or confirmed classification for incidents. For instance, high traffic on a network is not proof of data theft but needs more context for analysis by investigators.

Explanation Generation Using Groq

Retrieved data is provided along with an anomaly definition for use by groq api hosted models. This document is about structured outputs such as threat analysis, potential attacks techniques, signs of intrusion detection system, investigative procedures and evaluation reports. They're just some kind of suggestion based on what was provided for this task rather than being a standalone verification indicator or an automated answer choice.

Streamlit Dashboard and Visualization Streamlite Dashboard shows detection outcomes for users to interact with them. The Plotly chart shows a graph of CPU usage vs network traffic as well as anomalies score distributions while providing tables for anomaly detection statistics, graphs showing performance metrics, thresholds setting tools to detect outliersalongwithexplanations. Thefiguresareshown together as a group called "Stages". effig:architecture carry activity data from detection into a reviewable explanation.

Interactive Question Answering

Question answering allows people to inquire of an identified problem and get information related to it from its description along with relevant documents. For example, why is there an excessive amount of CPU usage. It provides an option for seeking clarifications without interpreting them or making decisions on behalf of analysts.

Table III. Technologies Identified in the Project Materials

Technology	**Reported role**
Python, Scikit-learn	Data handling and Isolation Forest implementation.
Streamlit, Plotly	Interactive dashboard and visual analytics.
LangChain	RAG workflow coordination.
Sentence Transformers	Text embedding generation.
FAISS	Vector storage and similarity retrieval.
Groq API	Language-model explanation and Q&A.

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

Implementation and System Design

Python and scikit-learn are used to build a model of data and an Isolation Forest algorithm [7] while streamlit and plotly are utilized for creating an application interface and performing statistical analysis on it [8]. LangChain is responsible for retrieving information from it and generating output [9]; Sentence Transformers are used to create vector representations of sentences [3]; FAISS indexes these vectors for efficient searching [4]; while the Groq API provides a language model that generates answers [10].

The detectors are responsible for identifying unusual features of interest whereas retrievals and generators offer descriptions to them. Source texts for a knowledge base are broken down to individual words or phrases that can be represented by high-dimensional vectors.

During queries, features are described as anomalies by means of embeddings through an embedding process while vectors are selected based on their similarities. It's an extraction rather than verification by our knowledge base of anomalies. OWASP Top 10 is an extension to MITRE ATTCK which includes a lexicon for tactics and techniques while also providing information about malware behavior and network anomalies.

There are no details about version numbers of documents, size chunks for this paper; also there isn't any information on an embedder id FAISS index types names like a language models etc. thus nothing can be said here. After that, it merges extracted passages with an anomaly definition and a predetermined risk map to request feedback from the language model (LLM).

Output is shown on this page along with a chart of data points from records. The interactive QA feature of this presentation allows users to ask questions like ”Why is there so much CPU usage?” Or ”Which attack concept does it suggest for that?”. Answers are still based upon retrieval of information from a given text as well as an AI-generated response to it.

Results and Performance Analysis Reported is based on synthetic generation of system activities for both normal and abnormal cases. Table is, provided here for reference. E ef, tab: metrics; fig.

Table IV. Reported Detection Performance

Metric	**Reported value**
Accuracy	0.9990
Precision	0.9524
Recall	1.0000
F1 score	0.9756
True negatives	999
False positives	1
False negatives	0
True positives	20

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

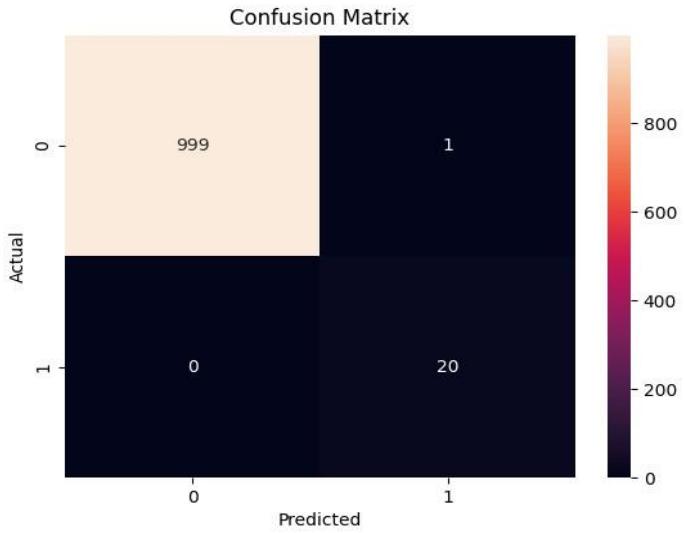


Fig. 2. Confusion matrix from the reported synthetic-data evaluation.

Effig: confusion. This is just a description of it; there are no reports about tests conducted by us for production use, public benchmarks or actual organization records.

With an accuracy rate of 0.9990, almost all evaluated records are accurately classified by this synthetic dataset. A precision value of 0.9524 indicates that a large proportion of flagged records are actually anomalies from the provided evaluation, while there is one reported false positive case. The recall for 1.0000 is equivalent to detecting all 20 labeled anomalies without any false negatives. F1 score is 0.9756 which represents a good trade-off between precision and recall. The metrics are consistent with the reported confusion rates, 999 true negatives, 1 false positive, 0 false negatives, and 20 true positives, 1, 020 records in total, because they depend upon how the synthetic examples were created and evaluated.

The score-distribution plot (Fig. The figure, fig., scores, shows an abundance of normal data points at high values while there are fewer anomalies at low levels as per our current model's prediction about these results. CPU network scatterplot Fig. The figure, fig, scatter, shows that low-activity data points are clustered together while high-cpu and high-network anomalies appear to be spread out over a larger area of space. The graphs show an example of what was produced but are unrelated to any other dataset.

This graph shows accuracy rate, sensitivity and

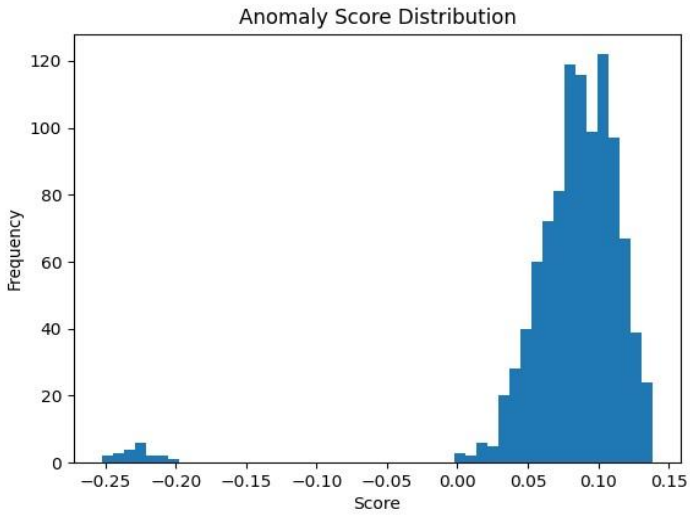


Fig. 3. Anomaly-score distribution reported for the synthetic system-activity data.

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

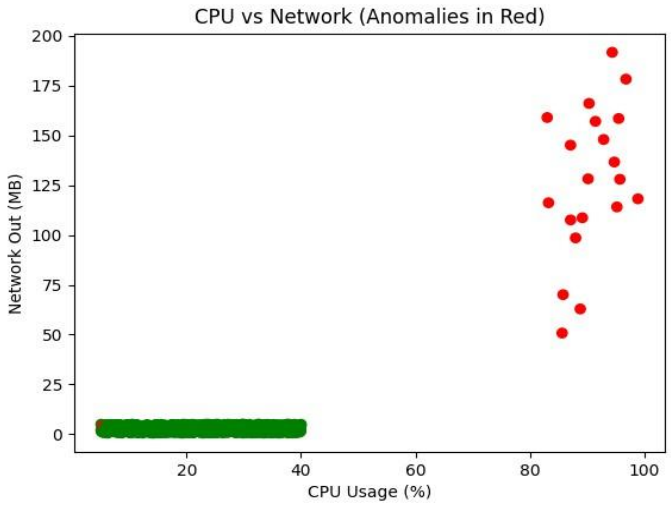


Fig. 4. CPU usage versus network activity, with anomalies highlighted in the source visualization.

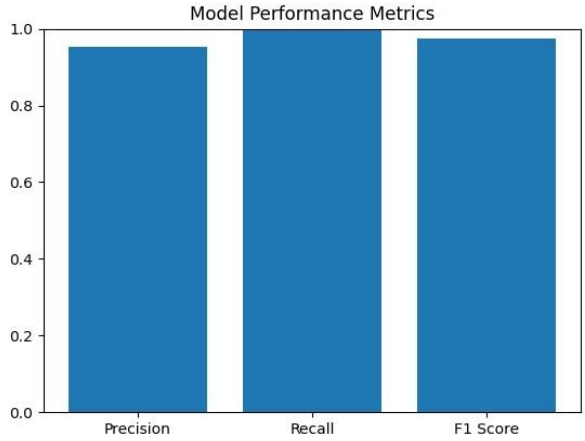


Fig. 5. Performance-metric visualization supplied with the project report.

F1 score (Fig.). Fig.: The actual measured results such as precision and recall can be found here (Table). eftab:metrics. Anomaly table on this dashboard gives you an overview at the level of detection along with explanations for further investigation through questions answering capabilities. There are no reports of an independent measure for latency, throughput, or operational efficiency.

DISCUSSION

Evaluation shows how Isolation Forest separates labeled normal and anomalous data points from a given synthetic dataset where each anomaly is identified by itself while there are no anomalies present among normals. It is an indication that this model can detect objects along its trajectory; however it does not mean there will always be perfect accuracy for all cases. The synthetic data may have clear separation but does not represent noise, variability, or changing baseline for operational systems. Unsupervised detectors are appropriate to be used on this task since they do not require labeled attacks as an input from a model. Label comparison between predictions of a mode land its own synthetic data is needed for this task. Reported recalls are relevant to this study as there were no unlabeled anomalies detected during testing but they may need additional analysis for real-world applications such as false positives costs, class imbalances and attacks diversity separately. In addition, changes of normal workload might be considered as an anomaly before we assess them by comparing it with current models and thresholds under different operational conditions. A RAG model is not concerned about detecting objects like an object detector does. The Isolation Forest is used to identify an anomaly as it stands out among all records while providing potential

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

security details through retrievals; meanwhile, the language model converts descriptions and retrieved contexts into prose. The division will allow us as analysts to know why there is an issue so that we could look at it further if needed. Mapping is an example of starting vocabulary rather than causality. CPU load high, logins rate low; also there is no way to remove this from context retrieval. For L1 IT professionals and nonspecialists, showing an anomaly along with a plot, providing a plain language explanation, and offering an opportunity for further questions might be easier to do than just giving scores. Benefit is due to visibility of information and human oversight. An explanation may sound good but might not be accurate; therefore it is advisable to check facts from reliable sources of information about incidents prior to acting upon them.

Limitations

Synthetic data usage is a major problem for this project. Allows for a controlled test of this system without testing it against live data sets, varying loads, or new attacks that have never been encountered before. This paper is not an example of actual implementation, no live logs are available for analysis; there isn't any reference to an external standardization test result nor testing on other platforms. Thus, report values cannot be passed on for use by manufacturing processes.

There may still be false positives because there is more than one report of this kind from a single test result. The dynamic nature of a system may change its pattern for normal activities while affecting an anomaly score. There is no information on how to select a threshold for scores from detectors; also there are none about their accuracy when they have been affected by drift. Scalability of large scale and real time logging is a problem while throughput and latency measurement remains an issue. Quality of explanation is an independent variable. The following are some examples of how to write an essay on a topic you're interested in writing about. There is no information about index version of this paper; there are also no embedded models' specifications for it, its retrieving parameters and quality score definition. The LLM may be inaccurate or too general for its purpose. Review is required from an expert person for these reasons especially when there are any issues that need attention such as escalation of incidents etc.

CONCLUSION

The paper describes this work as aprototype of using an unsupervised Isolation Forest to screen out-of-sample data points followed by providing explanations through retrievals on those samples. Its feature set covers CPU usage, network traffic, login attempts, process count, and file access. The RAG layer fetches information on cyber security such as MITRE ATTCK and OWASP from its source then provides these data points for mapping against rules of a groq hosted language model. With Streamlit and Plotly you can view score charts graphs tables reports explanation questions answers anomalies all at once.

Reported findings show good performance of this model but it is not an alternative to validate against actual data from live systems. Central to this approach is matching anomalies against context so users may start investigating them. Presenting results instead of verdicts is like giving you tools for exploring how detection and explanations are interdependent without taking over from humans' judgement.

Future Work

Project documents include various enhancements such as live logging integration, deployment on a cloud platform, connection with SIEM systems for detection of anomalies like intrusion attempts by malware programs (advanced) or other types of attacks; automatic score generation based upon detected threats along with better visualization techniques; also includes an approach towards obtaining information from external sources via Shodan web service. Shodan will be used for analyzing an ip address and a domain, exposing services to port scanning, checking vulnerabilities etc.

Future direction is a different thing from current capability to this model. Also further research needed to evaluate accuracy of detection and explanations through sample from actual life situations along with models documents retrievals thresholds settings.

ISSN No. 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV Issue XI November 2026

REFERENCES

1. F. T. Liu, K. M. Ting, and Z.-H. Zhou, '"Isolation Forest" by Zhou is published in Extit Proceedings of the Eighth IEEE International Conference on Data Mining, 2008, pp.
2. P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in extitAdvances in Neural Information Processing Systems, vol. 33, 2020, pages 9459-9474.
3. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in extitProceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 2019, pp. 3982–3992, doi: 10.18653/v1/D19-1410.
4. J. Johnson, M. Douze, and H. Jegou, "Billion-scale´ similarity search with GPUs," IEEE Transactions on Big Data, vol. 7, no. 3, pp. 535–547, 2021, doi: 10.1109/TBDATA.2019.2921572.
5. J. Johnson, M. Douze, and H. Jegou, "Billion-scale´ similarity search with GPUs," extitIEEE Transactions on Big Data, vol.7, no. 3, pp. 535–547, 2021, doi: 10.1109/TBDATA.2019.2921572
6. OWASP Foundation, "OWASP Top 10," [Online]. Available: https://owasp.org/ www-project-top-ten/. Accessed: Oct. 7,2026.
7. Scikit-learn Developers, "IsolationForest," Scikitlearn User Guide and API Reference. [Online]. Available: https://scikit-learn.org/stable/ modules/generated/sklearn.ensemble. IsolationForest.html. Accessed: Oct. 7, 2026.
8. Streamlit, Inc., "Streamlit Documentation." [On- line]. Available: https://docs.streamlit.io/. Accessed: Oct. 7, 2026.
9. LangChain, "LangChain Documentation." [Online]. Available: https://python.langchain. com/docs/introduction/. Accessed: Oct. 7, 2026.
10. Groq, Inc., "Groq API Documentation." [Online]. Available: https://console.groq.com/docs/. Accessed: Oct. 7, 2026.