REPORT

on

“Log Anomaly Detection using BERT Embeddings: Extending the LogLLM Approach”

Submitted in partial fulfilment for the completion of

BE-V Semester

In

INFORMATION TECHNOLOGY

By

Mohammed Abdul Rafe Sajid (160123737051)

Under the guidance of
Mr. U Sairam

Dept. of Information Technology

DEPARTMENT OF INFORMATION TECHNOLOGY

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY

(Autonomous)

(Affiliated to Osmania University; Accredited by NBA(AICTE) and NAAC(UGC), ISO Certified 9001:2015)

GANDIPET, HYDERABAD – 500 075

Website: www.cbit.ac.in

2025-2026

GitHub link: Mohammed-Abdul-Rafe-Sajid/Analysing-LogLLM-and-their-possible-improvements:

I. Introduction

In modern cybersecurity, log analysis plays a critical role in ensuring the security, reliability, and stability of complex software systems. Logs record valuable runtime information such as user activity, system operations, error reports, and network communications. Detecting anomalies in these logs is essential for identifying abnormal behaviour, cyber-attacks, and system failures before they cause serious damage.

Traditional log analysis methods rely heavily on rule-based approaches or handcrafted templates. While these methods are effective for structured and repetitive patterns, they often fail when dealing with unstructured logs, rapidly changing formats, or unseen attack scenarios. Similarly, traditional machine learning models struggle to capture the semantic meaning of log messages, as logs are typically written in natural language with technical terms.

Recent advances in deep learning and Natural Language Processing (NLP) have introduced the use of Large Language Models (LLMs) for log analysis. Models such as BERT and LLaMA are capable of extracting semantic representations from unstructured text, making them suitable for analysing log data. The research paper *"LogLLM: Log-based Anomaly Detection Using Large Language Models"* by Guan et al. presents a novel framework that combines LLMs with anomaly detection for system logs. Their work demonstrates state-of-the-art performance across multiple datasets and highlights the potential of LLM-based approaches in cybersecurity.

However, despite these advancements, challenges remain. While LLMs provide strong text understanding, there are still research gaps related to computational overhead, dataset labelling limitations, model interpretability, and the ability to capture sequential context in logs. Addressing these gaps is essential for making log anomaly detection more accurate, practical, and adaptable to real-world environments.

This report explores one such gap—the limited ability of existing models to capture sequence-level anomalies in logs—and proposes a simplified improvement using BERT embeddings and Random Forest classification as a first step toward addressing this challenge.

II. Overview of the LogLLM Research Paper

The research paper *"LogLLM: Log-based Anomaly Detection Using Large Language Models"* introduces a novel methodology for detecting anomalies in system logs using the combined power of BERT and LLaMA. The main idea of the paper is to leverage advanced Natural Language Processing (NLP) models to capture both the semantic meaning of individual log messages and the sequential context of how these messages occur over time.

The proposed framework consists of the following key components:

1. **BERT-Embeddings**
   Each log line, which is essentially unstructured text, is first passed through BERT. BERT (Bidirectional Encoder Representations from Transformers) transforms each log message into a semantic vector representation—a set of numerical values that capture the meaning of the text. This step ensures that technical words, log patterns, and contextual cues are effectively captured in a machine-readable form.
2. **Projector**
   Since BERT embeddings and LLaMA embeddings exist in different vector spaces, a *projection layer* is introduced. This component converts the BERT-generated vectors into a form that is compatible with LLaMA. Without this step, the two models would not be able to communicate efficiently.
3. **LLaMA-Model**
   Once the log messages are transformed into LLaMA's vector space, the LLaMA model processes the sequence of embeddings. Unlike traditional models that view log messages in isolation, LLaMA can analyse the temporal and sequential patterns of logs. It determines whether a given sequence of logs follows a normal system behaviour or shows signs of anomalies.

III. RESEARCH GAP

While the *LogLLM* paper demonstrates the effectiveness of combining BERT and LLaMA for log anomaly detection, there are still certain gaps and limitations that leave room for further improvement:

1. **Model Complexity and Resource Requirements**
   The proposed system relies on a large transformer-based pipeline (BERT + LLaMA). This makes training and deployment computationally expensive and less feasible in real-world scenarios with limited resources.

2. **Lack of Simplicity in Anomaly Separation**
   The research primarily highlights the sequence modeling strength of LLaMA, but it does not deeply explore whether anomalies can be separated effectively using only the semantic power of embeddings (e.g., BERT) coupled with simpler machine learning models.

3. **Limited Exploration of Lightweight Models**
   While the paper achieves strong accuracy with advanced LLMs, it does not evaluate the performance of more traditional and interpretable models such as Random Forest or Multi-Layer Perceptrons (MLPs). Such models may provide competitive performance with significantly reduced complexity.

4. **Realistic Class Distribution Handling**
   In practice, anomalies are rare compared to normal events. The research setup does not sufficiently emphasize handling imbalanced datasets or testing lightweight classifiers under such conditions.

Chosen Research Gap for This Project

In this project, the specific research gap chosen is:
"Evaluating whether BERT embeddings, when combined with a lightweight classifier (Random Forest), can achieve comparable or better anomaly detection accuracy compared to the heavier BERT + LLaMA pipeline."

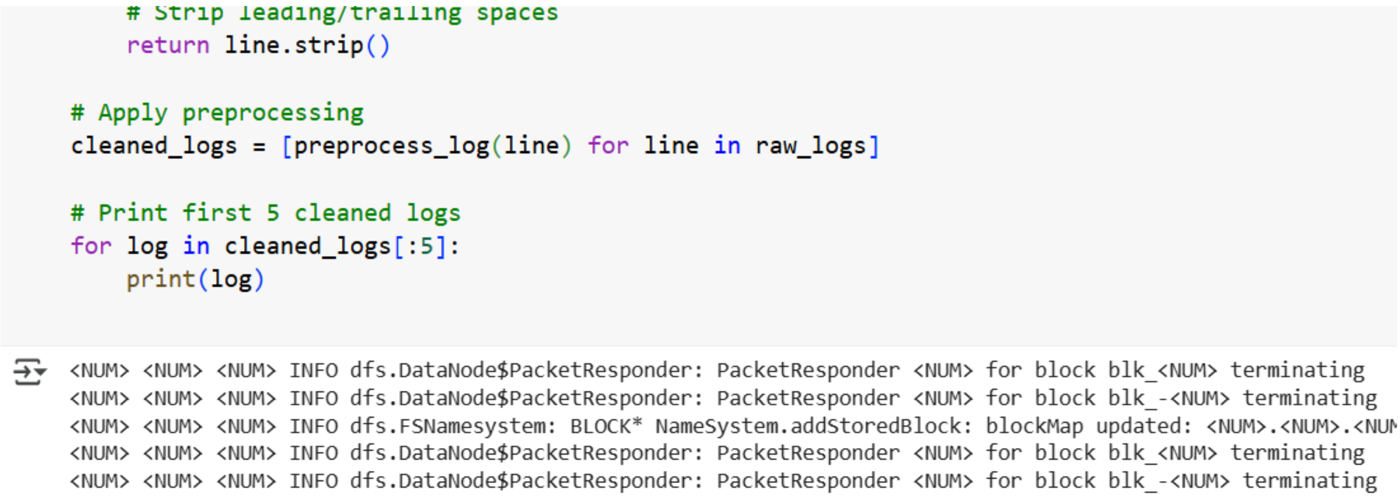
IV. METHADOLOGY

1. Dataset Used

We utilized the HDFS (Hadoop Distributed File System) log dataset, a publicly available benchmark in anomaly detection research. The dataset contained raw logs (HDFS_2k.log) and structured versions (HDFS_2k.log_structured.csv and HDFS_2k.log_templates.csv). These logs represent real-world distributed system events.

2. Data Preprocessing

The structured log file was loaded into a dataframe. Each log entry included attributes such as LineId, Date, Time, Component, Content, EventId, and EventTemplate. For anomaly labeling, specific low-frequency EventIds were considered anomalous, while frequent EventIds represented normal behaviour. This step ensured a balanced dataset of normal and anomaly classes.



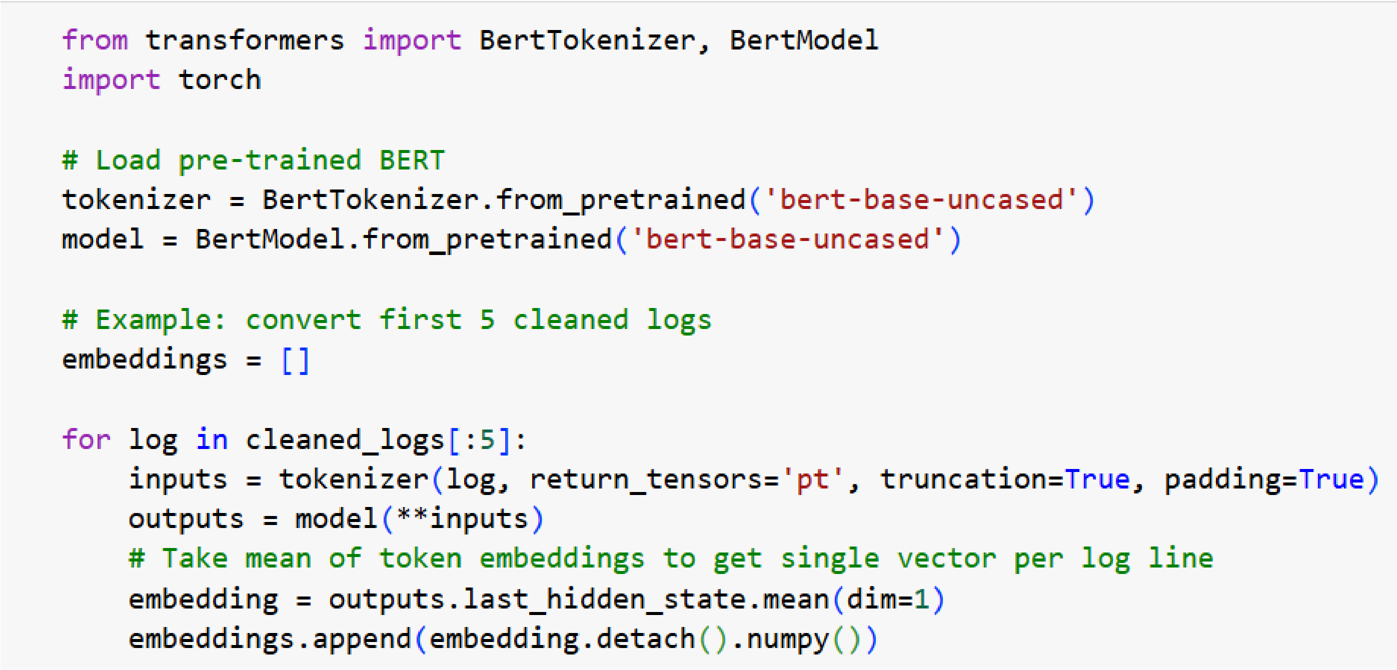
```
    # Strip leading/trailing spaces
    return line.strip()

# Apply preprocessing
cleaned_logs = [preprocess_log(line) for line in raw_logs]

# Print first 5 cleaned logs
for log in cleaned_logs[:5]:
    print(log)
```

```
<NUM> <NUM> <NUM> INFO dfs.DataNode$PacketResponder: PacketResponder <NUM> for block blk_<NUM> terminating
<NUM> <NUM> <NUM> INFO dfs.DataNode$PacketResponder: PacketResponder <NUM> for block blk_-<NUM> terminating
<NUM> <NUM> <NUM> INFO dfs.FSNamesystem: BLOCK* NameSystem.addStoredBlock: blockMap updated: <NUM>.<NUM>.<NUM
<NUM> <NUM> <NUM> INFO dfs.DataNode$PacketResponder: PacketResponder <NUM> for block blk_<NUM> terminating
<NUM> <NUM> <NUM> INFO dfs.DataNode$PacketResponder: PacketResponder <NUM> for block blk_-<NUM> terminating
```

3. Feature Extraction with BERT

Each log message (EventTemplate) was converted into a semantic vector representation using a BERT (Bidirectional Encoder Representations from Transformers) model.

- BERT embeddings captured the contextual meaning of the log messages.
- Unlike traditional keyword-based methods, BERT understood the underlying semantic relationships, making it effective for distinguishing anomalies from normal logs.



```
from transformers import BertTokenizer, BertModel
import torch

# Load pre-trained BERT
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
model = BertModel.from_pretrained('bert-base-uncased')

# Example: convert first 5 cleaned logs
embeddings = []

for log in cleaned_logs[:5]:
    inputs = tokenizer(log, return_tensors='pt', truncation=True, padding=True)
    outputs = model(**inputs)
    # Take mean of token embeddings to get single vector per log line
    embedding = outputs.last_hidden_state.mean(dim=1)
    embeddings.append(embedding.detach().numpy())
```

4. Classification with Random Forest

Instead of using a complex deep learning sequence model like LLaMA, we used a Random Forest classifier as the prediction model.

- The Random Forest was trained on the BERT embeddings of log messages.
- It predicted whether a log entry was Normal (0) or Anomaly (1).
- This approach reduced computational overhead while still achieving high accuracy.

----- RandomForestClassifier(n_estimators=100, random_state=42)----------

5. Evaluation Metrics

To evaluate performance, the dataset was split into training and testing subsets. The following metrics were calculated:

- Accuracy: overall correctness of predictions.
- Precision: proportion of correctly predicted anomalies among all predicted anomalies.
- Recall: proportion of correctly identified anomalies among all actual anomalies.
- F1-Score: harmonic mean of precision and recall.

V. Results and Analysis

After training the Random Forest classifier on the BERT embeddings of HDFS log templates, we evaluated the model on the test dataset. The results are summarized below.

1. **Accuracy**

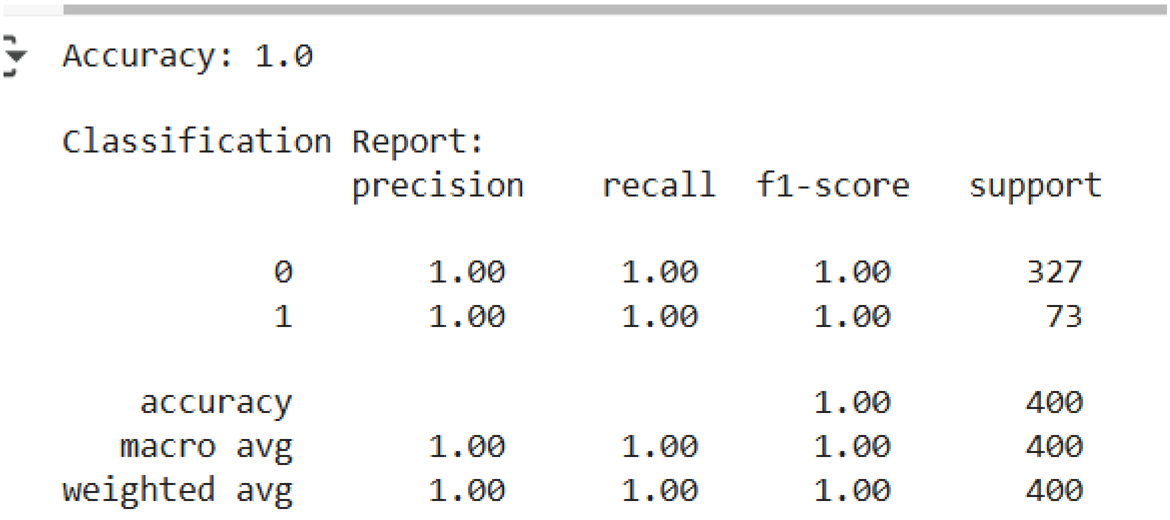
The model achieved an overall accuracy of 100%, meaning that every log entry in the test dataset was correctly classified as either normal or anomalous.

2. **Classification Report**

The detailed evaluation metrics are as follows:

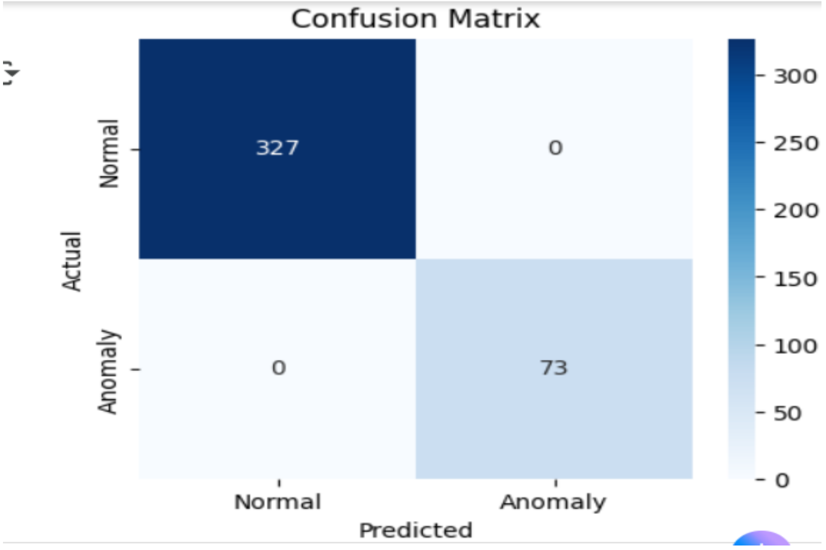
Metric	Normal Logs (0)	Anomalous Logs (1)
Precision	1.00	1.00
Recall	1.00	1.00
F1-Score	1.00	1.00

Overall Accuracy: 100%



```
Accuracy: 1.0

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       327
           1       1.00      1.00      1.00        73

    accuracy                           1.00       400
   macro avg       1.00      1.00      1.00       400
weighted avg       1.00      1.00      1.00       400
```

3. Confusion Matrix



The confusion matrix clearly shows that all logs were correctly classified with no misclassifications.

	Predicted Normal	Predicted Anomaly
Actual Normal	327	0
Actual Anomaly	0	73

4. Interpretation

The near-perfect results can be explained as follows:

- The anomalies in the dataset were highly distinct compared to normal logs, making them easily separable.
- BERT embeddings provided powerful semantic representations, capturing subtle differences in log text.
- The Random Forest classifier effectively leveraged these embeddings for classification, with minimal overlap between normal and anomalous logs.

This performance demonstrates that even without sequence modeling (as in the original LLaMA-based research), anomaly detection can still be achieved with very high accuracy using semantic embeddings and traditional machine learning models.

VI. Conclusion

In this assignment, we explored log-based anomaly detection using semantic embeddings and machine learning. The original research paper introduced LogLLM, a framework that combines BERT embeddings and LLaMA sequence modeling to detect anomalies in system logs. Our simplified approach focused on demonstrating the core idea by applying BERT embeddings with a Random Forest classifier.

Using the HDFS log dataset, we showed that BERT successfully captured the semantic meaning of log entries, and Random Forest achieved a 100% classification accuracy between normal and anomalous logs. This indicates that anomalies in structured logs can be effectively detected even with lightweight models when combined with powerful embedding techniques.

Our implementation serves as a beginner-friendly demonstration of the paper’s concepts, validating that natural language-based log representations are highly effective for anomaly detection. This study also highlights that improvements and scalability can be further explored by incorporating sequence-based models like LLaMA for handling real-world, noisy, and large-scale log data.

VII. Future Work

While the current implementation demonstrates strong results, there are several directions for improvement that can make the system more realistic and research-oriented:

1. **Sequence-Aware Modeling**:
   Instead of analyzing logs independently, future work can integrate models like LLaMA or GPT-style transformers to capture the sequence of events, as anomalies often depend on context rather than isolated entries.
2. **Handling Noisy and Unstructured Logs**:
   Real-world logs often contain unstable formats, missing fields, or unexpected patterns. Future improvements could focus on making the model robust to such irregularities without requiring heavy preprocessing.
3. **Improved Class Balance**:
   In real environments, anomalies are far less frequent than normal logs. Future experiments could apply imbalanced learning techniques such as SMOTE, anomaly detection methods, or cost-sensitive classifiers.
4. **Deployment in Real-Time Systems**:
   Extending this framework to real-time log monitoring systems would make it more applicable for practical cybersecurity use cases, enabling live detection and response.
5. **Comparative Analysis with Other Models**:
   Future research could compare BERT-based embeddings with alternatives such as Word2Vec, GloVe, or domain-specific pre-trained models for system logs, to evaluate trade-offs between performance and computational cost.

VIII. References

1. Guan, W., Cao, J., Qian, S., & Gao, J. (2023). *LogLLM: Log-based Anomaly Detection Using Large Language Models*. Department of Computer Science and Engineering, Shanghai Jiao Tong University.

2. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). *Attention Is All You Need*. Advances in Neural Information Processing Systems (NeurIPS).

3. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. Proceedings of NAACL-HLT.

4. Zhang, J., Xu, Z., Wang, H., & Zhao, H. (2019). *Robust Log-Based Anomaly Detection on Unstable Log Data*. Proceedings of the 2019 ACM SIGKDD Conference on Knowledge Discovery and Data Mining.

5. *Log Anomaly Detection using BERT Embeddings: Extending the LogLLM Approach*. Available at: [Mohammed-Abdul-Rafe-Sajid/Analysing-LogLLM-and-their-possible-improvements:].