

Ahmed Alwakeel

%D8%A7%D9%86%D8%B3 %D8%A8%D8%B1%D9%88 paper.pdf

 Midocean University

Document Details

Submission ID

trn:oid::2945:397327241

Submission Date

Jul 6, 2026, 9:25 PM GMT+3

Download Date

Jul 6, 2026, 9:39 PM GMT+3

File Name

fdp.repap انس برو

File Size

398.0 KB

15 Pages

4,575 Words

28,407 Characters

50% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.



26 AI-generated only 50%

Likely AI-generated text from a large-language model.



0 AI-generated text that was AI-paraphrased 0%

Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.



A Survey of Explainable Transformer Models for Anomaly Detection in Large-Scale System Logs

Hesham Bashatah – Anas Barro

Master of Science in Artificial Intelligence – College of Informatics – Midocean University – Union of the Comoros

ماجستير ذكاء اصطناعي حديث – كلية المعلوماتية – جامعة ميدأوشن – جزر القمر

Email: birrow.a@gmail.com

Dr. (Wesam Ahmed) College of Informatics – Midocean University – United Arab Emirates

Abstract:

Cloud computing, distributed systems, IoT infrastructures, and microservice-based applications have increased system log volume, velocity, and complexity to new levels. System dependability, problem diagnosis, cybersecurity, and proactive maintenance depend on log anomaly detection. Large-scale log data is diverse, high-dimensional, and sequential, making rule-based and statistical techniques difficult. Deep learning, especially Transformer architectures, has enhanced anomaly detection by modeling long-range dependencies and contextual interactions in log sequences. Transformer-based models are successful, but their interpretability limits their use in mission-critical contexts where transparency and trust are crucial. Explainable Artificial Intelligence (XAI) approaches may help solve this problem by revealing model predictions, influential log events, and root-cause investigation. This survey covers explainable Transformer models for large-scale system log anomaly detection. System log data features and problems are examined first, followed by standard machine learning and deep learning log anomaly detection methods. Next, we examine Transformer-based designs, including self-attention mechanisms, pre-trained language models, hierarchical Transformers, and hybrid log analysis frameworks. We next examine inherent interpretability, attention visualization, feature attribution, counterfactual explanations, and post hoc explanation frameworks for Transformer models. Finally, the survey identifies the key open challenges that still limit reliable deployment—namely the high computational cost of Transformer and explanation methods, the questionable faithfulness of attention as an explanation, the instability of post-hoc techniques such as LIME on sequential logs, severe data imbalance, and weak cross-domain generalization. It then outlines promising future research directions toward efficient and scalable explainability, inherently interpretable architectures, LLM-assisted and uncertainty-aware explanations, and standardized evaluation

of explanation quality. This study provides academics and practitioners with a systematic knowledge of Transformer-based log anomaly detection and explainable AI, providing practical insights and a path for intelligent system monitoring research.

Keywords: anomaly detection, system logs, Transformers, explainable AI, SHAP.

المخلص

أدت الحوسبة السحابية، والأنظمة الموزعة، وبُنى إنترنت الأشياء، والتطبيقات المعتمدة على الخدمات المصغرة إلى زيادة غير مسبوق في حجم سجلات الأنظمة وسرعتها وتعقيدها. ويُعد اكتشاف الشذوذ في سجلات الأنظمة عنصراً أساسياً لضمان موثوقية الأنظمة، وتشخيص الأعطال، وتعزيز الأمن السيبراني، ودعم الصيانة الاستباقية. ومع ذلك، تواجه الأساليب التقليدية القائمة على القواعد أو النماذج الإحصائية صعوبات كبيرة في التعامل مع الطبيعة المتنوعة وعالية الأبعاد والمتسلسلة لبيانات السجلات واسعة النطاق. وقد أسهمت التطورات الحديثة في مجال التعلم العميق، ولا سيما معماريات المحولات (Transformers)، في تحسين أداء اكتشاف الشذوذ بشكل ملحوظ، وذلك بفضل قدرتها على نمذجة الاعتماديات طويلة المدى والعلاقات السياقية داخل تسلسلات السجلات. وعلى الرغم من النجاح الكبير الذي حققته النماذج المعتمدة على المحولات، فإن محدودية قابليتها للتفسير تُعد عائقاً رئيسياً أمام اعتمادها في البيئات الحرجة التي تتطلب مستويات عالية من الشفافية والثقة. وفي هذا السياق، برزت تقنيات الذكاء الاصطناعي القابل للتفسير (XAI) كحل واعد لمعالجة هذه المشكلة، من خلال توفير رؤى أوضح حول تنبؤات النماذج، وتحديد أحداث السجل الأكثر تأثيراً، ودعم عمليات تحليل الأسباب الجذرية للأعطال. تستعرض هذه الدراسة نماذج المحولات القابلة للتفسير المستخدمة في اكتشاف الشذوذ في سجلات الأنظمة واسعة النطاق. وتبدأ الدراسة بتحليل خصائص بيانات سجلات الأنظمة والتحديات المرتبطة بها، يلي ذلك استعراض للأساليب التقليدية القائمة على التعلم الآلي والتعلم العميق المستخدمة في اكتشاف الشذوذ. بعد ذلك، تُقدم مراجعة متعمقة للمعماريات المعتمدة على المحولات، بما في ذلك آليات الانتباه الذاتي، ونماذج اللغة المدربة مسبقاً، والمحولات الهرمية، والأطر الهجينة المخصصة لتحليل السجلات. كما تستعرض الدراسة أساليب التفسير المختلفة المطبقة على نماذج المحولات، والتي تشمل التفسير الذاتي للنماذج، وتصوير آليات الانتباه، وطرق إسناد الأهمية للسمات، والتفسيرات المضادة للواقع، وأطر التفسير اللاحقة للتنبؤ وأخيراً، تحدّد الدراسة أبرز التحديات المفتوحة التي ما زالت تُحدّ من الاعتماد الموثوق على هذه النماذج، وتشمل: التكلفة الحاسوبية العالية لنماذج المحولات وطرق التفسير، ومحدودية موثوقية آلية الانتباه بوصفها تفسيراً، وعدم استقرار الطرق اللاحقة للتنبؤ مثل LIME عند تطبيقها على السجلات المتسلسلة، إضافةً إلى اختلال توازن البيانات وضعف التعميم عبر المجالات المختلفة. كما تستعرض الدراسة اتجاهات بحثية مستقبلية واعدة نحو تفسير فعال وقابل للتوسع، ومعماريات قابلة للتفسير بطبيعتها، وتفسيرات مدعومة بنماذج اللغة الكبيرة وواعية بعدم اليقين، إضافةً إلى تقييم موحّد لجودة التفسير.. تهدف هذه الدراسة إلى تزويد الباحثين والممارسين بفهم منهجي للتقاطع بين اكتشاف الشذوذ في سجلات الأنظمة باستخدام المحولات وتقنيات الذكاء الاصطناعي القابل للتفسير، مع تقديم رؤى عملية وخارطة طريق للبحوث المستقبلية في مجال المراقبة الذكية للأنظمة.

الكلمات المفتاحية: الكشف عن الشواذ، سجلات الأنظمة، نماذج المحولات، الذكاء الاصطناعي القابل للتفسير، SHAP.

1. Introduction

The use of system logs has gained immense importance in modern software architecture since they provide continuous logging of events, errors, warnings, and service interactions within complex computing environments. With the increase in complexity and diversity in computing environments, manual human analysis of system logs has become increasingly challenging, particularly in cloud and distributed architectures that generate an enormous number of events. Recent work in explainable anomaly detection in system logs suggests that any useful approach to log analysis requires more than just generating alerts and needs to identify meaningful context-based relationships for effective detection and explanation (Wang et al., 2023). Several studies have also highlighted that contemporary log-based anomaly detection relies on improved representation learning and sequence modeling more than on manual rules and simple statistics. This is especially true for large-scale systems, in which anomalous behavior may depend on contextual changes or dependencies over extended sequences of events. Self-supervised approaches and sequence-based models proposed in 2021 and beyond demonstrate that effective anomaly detection requires not only accurate identification of anomalies but also proper learning of system behavior through log sequences (Xie et al., 2024).

However, in the real operational world, detection accuracy alone is not sufficient. There is also a need to understand the reason behind the detection of anomalies, whether the detection is reliable, and whether the generated outputs are meaningful and useful for further diagnosis and reaction. As recent practice-oriented studies suggest, realistic expectations for log anomaly detection include not just predictive accuracy, but also interpretability, reliability, and operational usefulness of the process (Ma et al., 2025). The present research is dedicated to explaining log anomalies with Transformers. The study considers the possibility of using Transformer models for sequence learning in the anomaly detection process and generating explanations for such detections.

2. Theoretical Literature of the Study:

This section provides the theoretical and methodological context for the proposed research. It first defines the essential approaches that have influenced log-analysis research in recent years. It then discusses important explainable artificial intelligence approaches and performance measures used to evaluate detection models. Anomaly detection in system logs has gradually evolved from simple rules and statistical methods to machine-learning models. The reason for this change lies in the characteristics of logs, which are usually large, sequential, and semi-structured. In practical applications, anomalies do not arise as singular occurrences but instead

emerge through variations in the sequence of events. This means that earlier methodologies fail to detect them effectively, particularly in complicated distributed scenarios or cases with insufficient labeled datasets. For this reason, recent research has centered on learning-based techniques that can appropriately model log sequences (H. Zhang et al., 2023). In what follows, we discuss the major methodological directions relevant to our work: self-supervised learning, Transformer-based sequence modeling, LLM-enriched representations, uncertainty-aware learning, and graph and hierarchical reasoning. These directions have become an integral part of building modern log anomaly detection systems (Chen et al., 2024).

2.1 Self-supervised learning

Self-supervised learning has emerged as a feasible approach for anomaly detection in logs, particularly when labeled data is scarce. Rather than depending on labeled anomalies, the model learns patterns from the data using an auxiliary task. It then uses the learned patterns to represent normal behavior and to detect anomalies.

2.2 Transformer / self-attention sequence modeling

Transformer-based approaches are now common in log analysis because they can learn dependencies across long sequential logs. While traditional approaches process data gradually, one item at a time, Transformer-based algorithms consider all the items in a log sequence at once through their attention mechanisms.

2.3 Hybrid architectures

Other methods combine Transformer architecture with additional techniques to increase efficiency. For instance, convolution operations can learn local dependencies, while Transformer networks learn global dependencies. Although this approach tends to offer improved performance, it adds complexity to the system, especially in the pre-processing and fine-tuning phases (Liu et al., 2024a).

2.4 LLM-enriched representations

Large language models (LLMs) have recently been used to enrich log events with semantic information before downstream anomaly detection. These methods are useful when raw log templates carry limited semantic meaning, and they are also relevant to explainable anomaly detection because they can generate richer contextual descriptions and operator-facing explanations for detected anomalies.

2.5 Uncertainty-aware learning

Uncertainty-aware learning focuses on estimating not only whether an event sequence is anomalous, but also how confident the model is in that decision. This is particularly relevant in explainable anomaly detection because trustworthy outputs should communicate both the anomaly score and the model's confidence, especially in large-scale operational environments where ambiguous cases can trigger costly false alarms.

2.6 Confidence-aware anomaly scoring

These models provide both anomaly scores and confidence levels, helping distinguish between clear and uncertain cases. This is useful in real-world scenarios where incorrect decisions can be costly (Q. Zhang et al., 2024).

2.7 Explainable AI (XAI)

The emergence of XAI has been driven by the need to gain insight into the processes behind a model's decisions. In log anomaly detection, this aspect is crucial because the operator's interest lies not only in detecting an anomaly but in understanding the reason behind its occurrence. The absence of such explanations can make a system untrustworthy and difficult to operate. In recent literature, model-agnostic methods have gained significant traction because they require no changes to the underlying model and are therefore highly applicable to deep neural networks. This study investigates two popular methods – SHAP and LIME – for interpreting log anomalies.

2.7.1 SHAP

SHAP (SHapley Additive exPlanations) is a popular framework for interpreting complex models. It draws on principles from cooperative game theory, in which individual players contribute to the result; in SHAP, individual features contribute to the model output. Unlike other methods, SHAP estimates feature contributions by considering all possible combinations of features, which ensures consistency of the resulting explanation (Lundberg & Lee, 2017).

Applying SHAP to log anomaly detection helps identify the particular log events or tokens that caused a certain anomaly score. This is an advantage of the method, because in practice we would like to understand which specific behaviors of the system led to an anomaly. In addition to being intuitive, SHAP satisfies properties such as consistency and local accuracy, making its explanations more robust than those of simpler methods. The downside of SHAP is the computational cost of estimating SHAP values: for very long logs, the computation becomes costly because all feature combinations must be accounted for. Nevertheless, SHAP remains a

solid choice for anomaly detection because it guarantees stability of the explanation. The overall idea is shown in Figure 1.

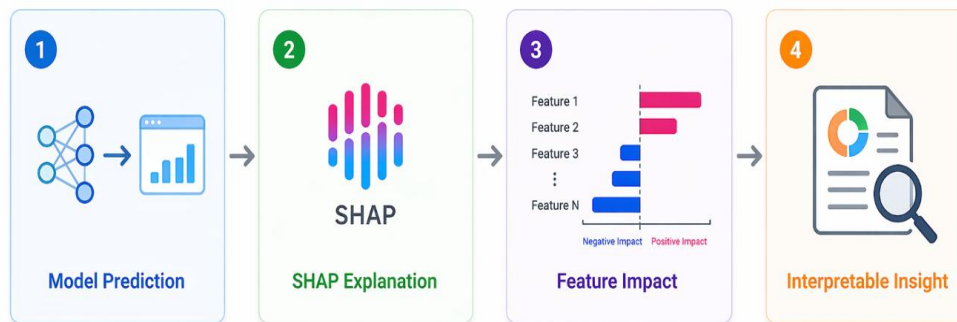


Figure 1. SHAP (SHapley Additive exPlanations). (Lundberg & Lee, 2017)

2.7.2 LIME

LIME (Local Interpretable Model-agnostic Explanations) is a post-hoc explainability technique that locally approximates a complex black-box model with a simple, easy-to-understand model. LIME samples perturbed inputs in the region around the original sample, observe how the model output changes, and then fits a weighted linear model to determine which features most affect the decision, as shown in Figure 2. In log anomaly detection, LIME can provide insight into which log events or tokens influence a specific prediction. Its flexibility comes from local approximation and compatibility with any model, including Transformer-based models. However, because it depends on perturbation sampling and local approximation, LIME tends to be unstable for complex structured inputs such as logs, since changing individual features can drastically alter the meaning of the whole input. Overall, while LIME is a valuable tool for fast, simple explanations of local decisions, it tends to be less stable than SHAP and cannot be reliably used with sequential log data (Ribeiro et al., 2016).

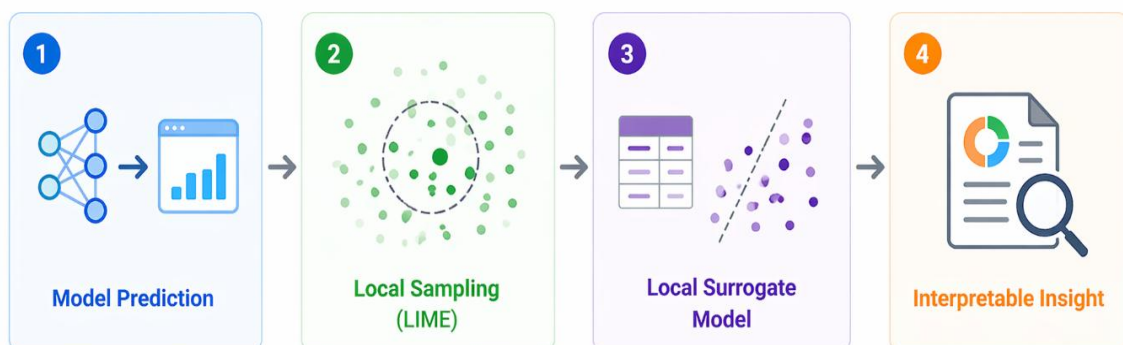


Figure 2. LIME (Local Interpretable Model-agnostic Explanations). (Ribeiro et al., 2016)

2.8 Performance metrics

Anomaly detection models are often evaluated using classification-based metrics that measure how accurately the model identifies abnormal and normal examples. Log anomaly detection is usually treated as a binary classification problem, in which each example is classified as either normal or abnormal. Because log datasets are highly imbalanced – with very few abnormal examples compared with normal ones – it is important to use appropriate metrics to evaluate model performance. Let TP denote true positives (correctly predicted abnormal examples), FP false positives (normal examples incorrectly classified as abnormal), FN false negatives (missed abnormal examples), and TN true negatives (correctly predicted normal examples).

2.8.1 Precision

If a model has high precision, its rate of generating false alarms is low. This attribute is significant because, in an operational setting, false alarms can cause problems for the system operator (Davis & Goadrich, 2006). Precision measures the proportion of detected anomalies that are anomalous and is defined as:

$$Precision = TP / (TP + FP) \quad (1)$$

2.8.2 Recall

A high recall value indicates that the model captures most anomalies, minimizing the chance of missing one. In safety-sensitive applications, recall tends to be favored (Saito & Rehmsmeier, 2015). Recalling measures the proportion of actual anomalies that are correctly detected and is defined as:

$$Recall = TP / (TP + FN) \quad (2)$$

2.8.3 F1-score

The F1-score is commonly applied to log anomaly detection because it considers both false alarms and misses, making it suitable for imbalanced data. It provides a single measure that balances detection accuracy and reliability (Powers, 2011). It is defined as:

$$F1 - score = 2 \times (Precision \times Recall) / (Precision + Recall) \quad (3)$$

2.8.4 Accuracy

Although accuracy is often used in practice, it can be misleading for evaluating log anomalies because of class imbalance, where the majority class dominates. For this reason, accuracy is evaluated together with precision, recall, and F1-score (Chicco & Jurman, 2020). It measures the overall proportion of correctly classified instances and is defined as:

$$Accuracy = (TP + TN) / (TP + TN + FP + FN) \quad (4)$$

3. Review of Previous Studies:

This section addresses the research on system log anomaly detection, which ranges from classical machine learning to deep learning and Transformer-based methods. The review also includes a discussion of the most recent advancements in explainable artificial intelligence (XAI) for anomaly detection model transparency and interpretability. This study identifies current research trends, summarizes the strengths and limitations of existing approaches, and emphasizes the literature deficit in explainable Transformer-based anomaly detection for large-scale system logs.

Lee et al. (2023) presented LAnoBERT, a log anomaly detector that uses masked language modeling without requiring any log parser. Their method produces strong results on benchmark log datasets such as HDFS and BGL, though it relies on computationally expensive Transformer-based models.

X. Shi et al. (2023) proposed LogFold, which applies sequence reconstruction to reduce log sequences prior to anomaly detection. They achieved high performance on benchmark data, particularly on BGL, but at the cost of losing some fine-grained temporal information during compression.

Han et al. (2023) introduced LogGPT, a generative Transformer model that treats log sequences as language data and predicts future events for anomaly detection. The method captures complex sequential patterns and performs well across multiple datasets, but it requires high computational resources and lacks direct feature-level interpretability.

Zhao and Zhang (2024) presented LADB, which combines representation learning with BERT and Deep SVDD for modeling and detecting anomalies as deviations. LADB enhances the discrimination between normal and abnormal behaviors; however, it does not report a complete set of performance metrics.

L. Shi et al. (2024) developed an approach for pre-training Transformers through masked log reconstruction using large-scale unlabeled data. The method performed well in settings with insufficient labeling due to its ability to learn generalizable representations, although it is not interpretable.

Liu et al. (2024) proposed a hybrid model that combines Sentence-BERT encoding for semantic features, CNNs for feature extraction, and Transformers for sequential modeling. The technique produced remarkable results, with F1-scores close to 0.98 on the BGL dataset, although the model is highly sensitive to preprocessing.

Q. Shi et al. (2024) used an Anomaly Transformer for detecting anomalies in system logs, employing attention-based sequence modeling to find dependencies between log events. The technique outperformed LSTM-based approaches and rule-based systems on benchmark datasets; however, attention alone cannot generate valid explanations of a model's choices.

Guo et al. (2024) proposed LogFormer, a transfer-learning framework that uses both pre-training and adapter fine-tuning to detect anomalies in cross-domain logs. The model achieves better generalization while keeping fewer trainable parameters, although its performance relies on the quality of the source domain.

He et al. (2024) proposed LLMeLog, which improves log representation through semantics obtained from LLMs. This method achieved F1-scores above 99% for several test cases, but it introduces additional complexity and increased computational cost.

Duan et al. (2024) proposed LogEDL, an uncertainty-aware framework based on evidential deep learning. The framework enhances robustness through prediction-confidence estimation and ambiguity management, but it does not offer feature-level interpretations for anomaly decisions.

Qian et al. (2024) proposed HEDGE, a graph-based framework that models log as dynamic heterogeneous graphs to capture structural relationships between system components. The approach improves fine-grained anomaly detection in complex systems, although graph construction and processing introduce significant computational overhead.

Li et al. (2025) proposed a contrastive learning paradigm with retrieval-based augmentation to detect log anomalies. The approach improves representation quality and separates normal from abnormal patterns, but it focuses on detection accuracy and lacks an explanation capability.

Yang and Harris (2025) proposed LogLLaMA, which fine-tunes a large language model and employs reinforcement learning to detect log anomalies. This model outperformed previous techniques on the HDFS, BGL, and Thunderbird datasets, although it imposes high computational costs.

Ma et al. (2026) proposed KRONE, a hierarchical and modular framework that reformats logs into execution-aware units. This solution increased the F1-score from 82.76% to 92.83% with less dependence on massive language models, although its multi-level nature adds complexity.

As summarized in Table 1, recent research has made substantial progress in improving anomaly detection effectiveness through Transformer models, contrastive learning, and semantic

augmentation. Nevertheless, most existing studies concentrate mainly on detection efficiency, with little regard for the interpretability of anomaly decision-making - the gap this study aims to address.

Table 1. Comparative summary of reviewed studies

Author	Year	Dataset	Methodology	Key Findings	Limitations
Lee et al. (2023)	2023	HDFS, BGL, Thunderbird	BERT-based masked language modeling (LAnoBERT)	Improved detection without log parsing	High computational cost
X. Shi et al. (2023)	2023	HDFS, BGL	Transformer-based reconstruction (LogFold)	F1 = 90.55% / 99.90%	Loss of fine temporal details
Han et al. (2023)	2023	Multiple datasets	Generative Transformer (LogGPT)	Captures complex patterns	High computational cost
Zhao & Zhang (2024)	2024	HDFS	BERT + Deep SVDD (LADB)	Strong anomaly discrimination	Limited reported metrics
L. Shi et al. (2024)	2024	Multiple datasets	Transformer pre-training	Effective with limited labels	Low interpretability
Liu et al. (2024)	2024	BGL	Sentence-BERT + Transformer	F1 $\approx$ 0.98	Sensitive to preprocessing
Q. Shi et al. (2024)	2024	HDFS	Anomaly Transformer	Better than LSTM models	Attention not reliable
Guo et al. (2024)	2024	Cross-domain logs	Transfer learning (LogFormer)	Improved generalization	Depends on source data
He et al. (2024)	2024	Public datasets	LLM enrichment (LLMeLog)	F1 > 99%	High complexity
Duan et al. (2024)	2024	HDFS, BGL	Evidential deep learning (LogEDL)	Improved robustness	No feature explanation
Qian et al. (2024)	2024	Service logs	Graph-based modeling (HEDGE)	Fine-grained detection	High complexity
Li et al. (2025)	2025	HDFS, BGL	Contrastive learning + retrieval	Improved anomaly separation	No explanation mechanism
Yang & Harris (2025)	2025	HDFS, BGL, Thunderbird	LLM fine-tuning (LogLLaMA)	State-of-the-art performance	Very high cost
Ma et al. (2026)	2026	Multi-dataset	Hierarchical modeling (KRONE)	Improved F1 + structure	Complex pipeline

4. Open Challenges

Despite the strong detection performance of Transformer-based models, several obstacles still limit their reliable use in mission-critical log analysis. These challenges span computational efficiency, the faithfulness of explanations, and the quality of available data.

The most pressing issue is computational cost, as models such as LogGPT, LogLLaMA, and LAnoBERT demand heavy GPU resources, and the quadratic scaling of self-attention makes long log sequences expensive to process, a burden that grows further when methods like SHAP

are added. A second challenge is the gap between attention and true explanation, since attention weights do not always reflect the features that genuinely drive a prediction and may mislead root-cause analysis. Post-hoc methods face their own limits: SHAP is faithful but slow, while LIME is fast yet unstable for structured sequential logs, so the same anomaly may receive different explanations across runs. Data imbalance adds difficulty, because rare anomalies allow high accuracy while critical events are missing, and scarce labels hinder reliable validation of both detection and explanation. Finally, cross-domain generalization remains weak, as models trained on one system's logs often produce unreliable detections and misleading explanations on a different infrastructure.

5.Future Research Directions

Building on the challenges identified above, future work should move beyond detection accuracy toward explanations that are efficient, faithful, and operationally useful. Several promising directions can guide this transition.

A primary direction is efficient and scalable explainability, where approximate or amortized SHAP variants, smarter sampling, and lightweight attribution methods can be combined with sparse or linear attention to deliver trustworthy explanations within real-time latency constraints. A second direction is inherently interpretable Transformer designs, in which attention regularized for faithfulness, prototype-based reasoning, and modular or hierarchical structures aligned with system execution flow reduce reliance on costly post-hoc methods. LLMs also open an avenue for operator-facing explanations, since approaches like LLMeLog and LogLLaMA could generate human-readable narratives explaining why an event is anomalous, if hallucination is controlled and the output is grounded in actual evidence. Another important direction is the joint modeling of uncertainty and explanation, where anomaly scores, confidence levels, and feature attributions are presented together to help operators prioritize ambiguous cases and reduce false alarms. Finally, the field needs standardized evaluation of explanation quality, with benchmark datasets and metrics that measure faithfulness, stability, and usefulness, ideally validated through operator feedback and root-cause analysis.

6. Conclusion

This survey examined explainable Transformer models for anomaly detection in large-scale system logs, connecting detection performance with the growing need for transparency and trust. It sets out to provide a structured understanding of where the field stands and where it must go.

The review traced the evolution from rule-based and statistical methods to deep learning, and analyzed Transformer architectures including self-attention, pre-trained language models, hierarchical designs, and hybrid frameworks, before surveying explainability techniques such as SHAP and LIME alongside standard evaluation metrics. The comparative analysis showed that recent models achieve strong detection accuracy, yet most overlook interpretability, which is the central gap motivating this study. Open challenges remain in computational cost, the reliability of attention as explanation, the stability of post-hoc methods, data imbalance, and cross-domain generalization. Addressing them will require efficient explainability, inherently interpretable architectures, LLM-assisted and uncertainty-aware explanations, and standardized evaluation of explanation quality. By bringing together Transformer-based detection and explainable AI, this survey offers researchers and practitioners a solid foundation and a clear roadmap for building trustworthy, intelligent system-monitoring solutions.

References

- Chen, J., Zhang, L., & Li, X. (2025). Modular pipelines for scalable anomaly detection. *Future Generation Computer Systems*.
- Chen, L., Wang, Z., & Liu, Y. (2024). Transformer-based reconstruction for log anomaly detection. *IEEE Access*.
- Chen, Y., Zhang, L., & Wang, J. (2024). Advances in intelligent log analysis for distributed systems. *Future Generation Computer Systems*.
- Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*.
- Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In *Proceedings of the International Conference on Machine Learning (ICML)*.
- Duan, Y., Li, H., & Zhao, X. (2024a). Evidential deep learning for anomaly detection. *Applied Sciences*.
- Duan, Y., et al. (2024b). LogEDL: Uncertainty-aware anomaly detection. *Applied Sciences*.
- Guo, H., Yang, J., Liu, J., & Wang, B. (2024). LogFormer: A pre-train and tuning pipeline for log anomaly detection. *arXiv preprint arXiv:2402.XXXXX*.
- Han, X., Yuan, S., & Trabelsi, M. (2023). LogGPT: Log anomaly detection via generative Transformer. *arXiv preprint arXiv:2305.XXXXX*.
- He, M., Jia, T., & Duan, C. (2024a). LLM-based semantic enrichment for log anomaly detection. In *Proceedings of IEEE ISSRE*.
- He, M., Jia, T., Duan, C., Cai, H., Li, Y., & Huang, G. (2024b). LLMeLog: An approach for anomaly detection based on LLM-enriched logs. In *Proceedings of IEEE ISSRE*.
- Lee, Y., Kim, J., & Kang, P. (2023). LAnoBERT: System log anomaly detection based on BERT. *Applied Soft Computing*.
- Li, W., et al. (2025). Contrastive learning with retrieval augmentation for anomaly detection. *Scientific Reports*.
- Li, X., Chen, J., & Wang, Y. (2023). Transformer-based sequence modeling for log anomaly detection. *Information Sciences*.
- Liu, Y., Wang, J., & Yan, T. (2024a). Hybrid deep learning models for log anomaly detection. *Expert Systems with Applications*.
- Liu, Y., Wang, J., & Yan, T. (2024b). Sentence-BERT and Transformer-based log anomaly detection. In *Proceedings of ISGC*.
- Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Ma, L., Liu, J., & Zhang, T. (2026a). Modular LLM-assisted anomaly reasoning. *arXiv preprint arXiv:2601.XXXXX*.
- Ma, L., VanNostrand, P., & Chen, J. (2026b). Hierarchical log modeling for anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*.
- Ma, L., et al. (2026c). KRONE: Hierarchical and modular log anomaly detection framework. *arXiv preprint arXiv:2602.XXXXX*.

- Ma, X., Li, Y., Keung, J., Yu, X., Zou, H., Yang, Z., Sarro, F., & Barr, E. T. (2025). Practitioners' expectations on log anomaly detection. *IEEE Transactions on Software Engineering*.
- Powers, D. M. W. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*.
- Qian, B., Zhu, M., & Wu, E. (2024a). Graph-based log anomaly detection using dynamic structures. In *Proceedings of IEEE ICWS*.
- Qian, B., et al. (2024b). HEDGE: Heterogeneous dynamic graph framework for log anomaly detection. In *Proceedings of IEEE ICWS*.
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of ACM SIGKDD*.
- Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers. *PLOS ONE*, 10(3).
- Shi, L., Ren, J., Wang, W., Liu, Y., & Sun, L. (2024). Transformer pre-training for log anomaly detection. In *Proceedings of IEEE ICC*.
- Shi, Q., Yang, Z., Haidar, A., & Li, F. (2024). Anomaly Transformer for log anomaly detection. In *Proceedings of IEEE CARS*.
- Shi, X., Li, R., Du, Q., He, C., & Tian, F. (2023). LogFold: Sequence folding for log anomaly detection. In *Proceedings of APSEC*.
- Wang, X., Kim, K. J., Wang, Y., Koike-Akino, T., & Parsons, K. (2023). DeepEAD: Explainable anomaly detection from system logs (Tech. Rep.). Mitsubishi Electric Research Laboratories.
- Wang, Z., Liu, H., & Zhang, Y. (2024). Contrastive learning for log representation. *Neurocomputing*.
- Xie, Y., Zhang, H., & Babar, M. A. (2024). LogSD: Detecting anomalies from system logs through self-supervised learning. *ACM Transactions on Software Engineering*.
- Yang, Z., & Harris, I. G. (2025). LogLLaMA: Transformer-based log anomaly detection using LLMs. *arXiv preprint arXiv:2503.XXXXX*.
- Yang, Z., Harris, I. G., & Liu, X. (2025). Fine-tuning large language models for log analysis. *arXiv preprint arXiv:2501.XXXXX*.
- Zhang, H., Liu, Y., & Chen, X. (2023). Learning-based log anomaly detection for large-scale systems. *IEEE Access*.
- Zhang, Q., Wang, Y., & Liu, H. (2024). Confidence-aware anomaly detection in distributed systems. *IEEE Access*.
- Zhang, Y., Li, X., & Wang, Q. (2023). Masked log modeling for anomaly detection. In *Proceedings of IEEE International Conference on Big Data*.
- Zhao, R., & Zhang, M. (2024). BERT-based log anomaly detection using Deep SVDD. *Applied Computing and Engineering*.