Anomaly Detection for System Logs Literature Overview[*]

Justas Juknys[1,*,†]

[1] Vytautas Magnus University, Universiteto str. 10–202, 53361 Akademija, Kaunas, Lithuania

Abstract

This paper describes analysis results of system log Anomaly Detection literature from time period of 2018 to 2023. The literature was found using keywords “log anomaly”, “machine learning”, “neural network”. A total of 80 different scientific papers have been analyzed. It has been determined that most popular neural networks are LSTM/BiLSTM; most common datasets are HDFS, BGL and Thunderbird; Most popular evaluation metrics include F1, precision and accuracy. Most of research sought to address issues of improving model detection accuracy, lowering system resource use and making model more suitable real time detection.

Keywords

Deep learning; Neural networks; Machine Learning; Log messages; Literature Review; Anomaly Detection; Cyber Security; Classification

1. Introduction

As time goes by complexity and scale of software systems is rapidly increasing, which forces the necessity for new anomaly detection methods to be developed[1]. As the amount of data needed to be analyzed increases, the need to fully automate detection process increases[2]. As treats to system security become more sophisticated, the amount of needed to be analyzed data points keeps increasing as well, which at the same time makes it harder to use supervised training approach and properly interpret received data[3]. Another major issue is prevalence of 0 day exploits which are usually impossible to predict in advance[4]. All of the aforementioned issues are normally addressed through use of anomaly detection methods.

To limit the scope of research it was chosen to focus on the specific keywords: “Log”, “Anomaly detection”, “machine learning”, “neural networks”. arxiv.org[5] and sciencedirect.com[6] databases have been used for research paper collection. A total of 117 different research papers have been analyzed. All papers have been written during 2000-2023 time period with 78 of the papers being from 2018-2023 time period. Only research and conference papers have been analyzed.

For the purposes of this paper the following has been chosen to analyze:

1. Which neural network and machine learning approaches are being used?
2. What metrics have been used to evaluate suggested approaches and how do different approaches compare to each other?
3. Which data sets are being used to train models?
4. What problems in anomaly detection have been identified?
5. What findings/conclusions have been made?

2. Key definitions

[*] IVUS2024: Information Society and University Studies 2024, May 17, Kaunas, Lithuania

[1,*] Corresponding author

[†] These author contributed equally.

justas.juknys@vdu.lt (J. Juknys)

0009-0005-7913-3934 (J. Juknys)

1) *Anomaly detection*: It is an approach seeking to identify unusual events based on comparisons to standard situation. The anomalous event is something which cannot be fully anticipated in advance and as result cannot be detected via traditional pattern based detection methods. To declare an anomaly an outlier needs to be found. This outlier could appear through various contexts like statistical outlier, situation/sequence outlier, timing outlier and so on…

It is usually assumed that the amount of anomalous data is much less numerous than normal data. Most popular approach to solving anomaly detection problems is use of semi supervised training, where models are trained exclusively on normal data[3].

2) *Log data*: this is information gathered in sequential order and presented in lines. Each log entry contains all the necessary information to identify various system states at given time moments. Data is usually saved in either string or numerical values and is saved in easily readable text files. By following log entries it should be possible to reconstruct how system continuously functioned in the past, so if system deviates from expected behavior, log analysis should identify the moment of system malfunction.

Log data can be used to determine in advance if there are any risks for system failure and also can be used to detect possible intrusions. In order to achieve this, multiple data entries need to be analyzed at once in order to identify any abnormal patterns[3].

3) *Neural Networks* are subset of Artificial Intelligence (AI) research. They are algorithms based on neuroscience seeking to replicate function of human brain. These networks consist of many input units, which are arranged in sets of layers. Initially preproccessed data is fed to initial layer and after performing initial data transformations, layer results are passed to subsequent layers7. Over time Neural Network discovers patterns within its data and then can use it to classify data into various categories.

4) *Machine Learning* (ML) is a subset of AI research, seeking to imitate human intellect through self learning algorithms. Firstly it is provided with preprocessed data, then a chosen model is applied to discover any meaningful patterns within given data[8]. The given data can either be labeled to enhance model accuracy, which is called "Supervised Learning". In case of Unsupervised training provided data is unlabeled and patterns need to be discovered using statistical methods.

When compared to neural networks, classical, or "non-deep", machine learning is more dependent on human intervention to learn. Human experts determine the set of features to understand the differences between data inputs, usually requiring more structured data to learn[9]. Traditional machine learning methods include Isolation Forest, SVM, kNN, Naive Bayes, Polynomial/Linear Regression, PCA and other methods.

3. Survey Results

Table 1 showcases the amount of publications released during recent years. Publication amount is the exact number of research papers released during that year. Any papers which also include research into neural network use are counted as well.

Table 1
Publications per year

Year	Total Publication Amount	Neural Network Papers
2023 (first half)	6	5
2022	15	10
2021	26	17
2020	13	10
2019	10	4

2018	8	3
2017	9	3
2016	3	0
2015	5	3

It can be said that during recent 5 years the anomaly detection field has received an increased amount of attention from the research community. During last 3 year period majority of written literature covers Neural Network methods and standard machine learning methods (like Knn, decision trees, SVM...) are becoming less popular.

Table 2 lists all the different neural networks which have been mentioned in at least at least 2 separate research papers. All the remaining methods are included in "other" category. By far the most popular neural network models were LSTM or BiLSTM. The primary reason for this is that log data is normally represented in time series where usually previous log entries have influence over later entries[10].

Table 2
Most common Neural Network approaches

Neural Network	Amount
LSTM/BiLSTM	11
Autoencoder	7
CNN/TCN	6
Deeplog/LogAnomaly/LogRobust	6
Transformer	6
GNN/eGNN/eGFC	5
RNN	4
MLP	3
Siamese Neural Network	2
Other	13

Table 3 provides the list of all commonly used machine learning methods. Any method which only has been used once within researched literature has been included in "other" category. It has been determined that SVM is the most frequently used machine learning method. Its primary advantage over Neural Networks is its significantly faster computational speed, which is important when it's necessary to detect system anomalies as soon as possible. Some other notable advantages include ability to handle high dimensional data and low risk of overfitting[11].

Table 3
Most Common Machine Learning Approaches

Method Name	Amount
SVM	14
Isolation Forest	10
Logistic/Linear Regression	6
PCA	6
Word2Vec	6
Bayesen	5
kNN	4
Decision Tree	3
Drain Algorithm	2
Other	14

Table 4 contains amounts of all most commonly used datasets. Industrial category refers to unnamed datasets which used specific industrial process log data. Private category includes all datasets, which cannot be disclosed due to a non disclosure agreement. Generated category includes all synthetic datasets which were generated specifically for the research study. Any dataset which didn't fall into previous 3 categories and was only mentioned once within all research papers, has been included in "other" category.

HDFS is a key component of Hadoop, offering reliable storage through data replication, integrates with big data frameworks and supports batch processing[12]. Within reviewed literature it appeared the most frequently and often was simultaneously used with BGL and Thundebird[13], both of which are popular supercomputer log datasets.

Table 4
Most Common Datasets

Dataset Name	Amount
HDFS	20
BGL	17
Thunderbird	10
Openstack	8
Spirit	6
NSL-KDD	4
DARPA	3
Hadoop	3
Lanl	3
Mnist	3
CIFAR	2
Huawei Cloud	2
KDD CUP 99	2
Industrial	4
Private	11
Generated	4
Other	79

Table 5 showcases all frequently used evaluation metrics. Any research metric which has only been used once within all research papers is included in "other" category. It has been determined that F1 Score (Formula 1) was the most commonly used evaluation metric. This metric is calculated using Recall (Formula 2) and Precision (Formula 3) metrics, so in most of research papers all 3 metrics have been used simultaneously. Within these formulas True Positive stands for all correctly identified elements, False Negative stands for all elements which have been incorrectly labeled as false, False Positive are all elements incorrectly labeled as true.

Precision is a good way of determining reliability of individual results which helps to minimize the risk of spending unnecessary resources on managing false alarms. Recall is useful for determining how much of an impact false negatives might have which is very important as all it takes is one missed anomaly to cause massive system damage. As both Precision and Recall are important, F1 ensures that both of them can be represented using a single metric[14].

Table 5
Most Common Evaluation Metrics

Metric	Amount
F1	40
Precision	25
Recall	24
Accuracy	20
AUC	11
Computation Time/Resource Reduction	5
Error Rate	2

Standard Deviation	2
Other	11

$$F1=\frac{2\cdot Precision\cdot Recall}{Precision+Recall} \tag{1}$$

$$Recall=\frac{True\ Positive}{True\ Positive+False\ Negative} \tag{2}$$

$$Precision=\frac{True\ Postive}{True\ Positive+False\ Positive} \tag{3}$$

Table 6 lists most common anomaly detection problems described within research papers. Any problem which has only been mentioned once has been assigned to "other" category. The largest concern specified by research literature is that due to increase in data amount, the extent to which log data analysis should be automated should increase as well[15][16][17]. Another major issue being brought up is that by itself log data does not include a sufficient amount of data to effectively determine new treats[18]. Often while relying on log data, only time context is established and additional data context is ignored[19]. Further issues could also be introduced while parsing log data, which could further degrade anomaly detection accuracy[20].

Table 6
Most common problems

Problem type	Count
Need for better data processing	19
Need better context extracting	18
Excess computing resource use	12
Excess information amount	8
Changing environment/software updates	8
Cloud computing optimization	7
Unfit for novel anomaly detection	6
Need more data points	5
Insufficiently tested models	4
Flawed datasets	4
Insufficient detection rate	2
Log data by itself is insufficient	2
Other	5

One of the main requirements for successful anomaly detection is timely discovery of new treats. In order to comply with it and provide near real time detection, some necessary compromises need to be done. For example often this means only relying on most simple log data analysis and ignoring additional system analysis tools[21][22]. Furthermore state of the art anomaly detection methods with highest detection accuracy are usually unfit for time sensitive issue detection[23]. Another concern is that due to amount of information needed to be processed, cloud computing becomes necessary, which introduces issues of data transfer speeds[24][25]. To add on top of that due to software updates, models designed for previous software versions might severely degrade in accuracy[26].

Some additional issues being brought up in literature included having difficulty to perform simultaneous parallel analysis when each input is part of time series and requires proper understanding of its context[27]; not all problems might be reflected within logs and the issues of software program itself might be overlooked[28]; anomaly detection methods do not get sufficiently compared to each other[29];

traditional machine learning methods such as SVM are unable to perform sufficiently accurate analysis of temporal information of discrete log messages[30]; Certain anomaly detection models have not been sufficiently tested in real life application[31]; models based on statistical methods might be insensitive to importance of log entry order sequences[32].

3.1. Primary Findings

The following were the main findings of analyzed literature:

1.Embedding multi-core point-by-point convolution and global average pooling achieves significant advantages in terms of arithmetic power, memory and high availability, while ensuring detection accuracy [23].

2.Gumbel Noise Score Matching model demonstrated the capability of score matching for anomaly detection on categorical types in both tabular and image datasets. It also provided a unified framework for modeling mixed data types via score matching [33].

3.In transformer based models adapter-based tuning consistently outperforms training and fine-tuning models[16].

4.Dividing log events into dependent and independent types is an effective way to boost model accuracy [17].

5.Taking a character-based approach to process log events (lines) contributes to higher performance as the model may take advantage of characters deleted in word-based approaches, such as numbers and punctuation. Merging the parser, vectorizer, and classifier components into one deep neural network, allows model to learn log data at the language level [34].

6.Models trained on multi-project datasets are not only more accurate in standard tests but also more robust to sequence evolutions and more accurate in ahead of time anomaly predictions [34].

7.Though the presence of critical logs often indicates problems, their absence does not necessarily imply a healthy system status. An important reason is that sometimes determining where and how to place an informative log statement is difficult. In some cases, faults do not affect metrics, while in other cases, metrics exhibit unusual patterns (e.g., jitters) even if the system is experiencing minor performance fluctuations instead of faults. Hence, simply identifying anomalous metric patterns is insufficient [1].

8.Faults can cause unexpected behaviors involving either logs or metrics, or both of them. So the two data sources should be analyzed comprehensively to reveal the actual anomalies [1].

9. Intrinsic structure of host-based logs, as captured by persistence images and the spectrum of graph and hypergraph Laplacians, contains discriminative information about whether or not the logs are anomalous[35].

10. Data augmentation can simulate deviations in log data that occur from service updates over time which contribute to successful anomaly detection[25].

11. Multimodal approach can improve the scores for anomaly detection for multiple modalities in comparison to the single modalities of logs and traces [36].

12. Filtering out common log entries can noticeably improve anomaly detection accuracy [37].

4. Conclusions

During this survey it has been determined that over recent years the popularity of this topic has been increasing. The problems identified within research papers still need to be addressed and no universal solution has been discovered which would allow anomaly detection methods to keep up with ever increasing amount of generated log data and general increasing complexity of system software. It has also been determined that neural networks are continuously increasing in popularity, while traditional machine learning methods are becoming less popular. It has been determined that the most popular neural network model is LSTM/BiLSTM, most commonly used dataset is HDFS and most frequently used evaluation metric is F1 score.

Bibliography

[1] Cheryl Lee, Tianyi Yang, Zhuangbin Chen, Yuxin Su, Yongqiang Yang and Michael R. Lyu, Heterogeneous Anomaly Detection for SoftwareSystems via Semi-supervised Cross-modal Attention, 2023. URL: https://arxiv.org/pdf/2302.06914.pdf

[2] Thorsten Wittkopp, Dominik Scheinert, Philipp Wiesner, Alexander Acker, and Odej Kao, PULL: Reactive Log Anomaly DetectionBased On Iterative PU Learning, 2023. URL: https://arxiv.org/pdf/2301.10681.pdf

[3] Max Landauer, Sebastian Onder, Florian Skopik, and Markus Wurzenberger, Deep Learning for Anomaly Detection in Log Data:A Survey, 2023. URL: https://arxiv.org/pdf/2207.03820.pdf

[4] Rasheed Ahmad, Izzat Alsmadi, Wasim Alhamdani, Lo'ai Tawalbeh, Zero-day attack detection: a systematic literature review, 2023. URL: https://link.springer.com/article/10.1007/s10462-023-10437-z

[5] , About arXiv, 2024. URL: https://info.arxiv.org/about/index.html

[6] , About Science Direct, 2024. URL: https://www.elsevier.com/products/sciencedirect

[7] Chris Woodford, Neural networks, 2023. URL: https://www.explainthatstuff.com/introduction-to-neural-networks.html

[8] Sara Brown, Machine learning, explained , 2021. URL: https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

[9] IBM, What is machine learning?, 2024. URL: https://www.ibm.com/topics/machine-learning

[10] Amir Farzada,, T. Aaron Gullivera, Log Message Anomaly Detection and Classification UsingAuto-B/LSTM and Auto-GRU, 2021. URL: https://arxiv.org/pdf/1911.08744.pdf

[11] Suthar Mudra Bhavikkmuar, Advantages of Support Vector Machines (SVM), 2023. URL: https://iq.opengenus.org/advantages-of-svm/

[12] Donal Tobin, The Ultimate Guide to HDFS for Big Data Processing, 2023. URL: https://www.integrate.io/blog/guide-to-hdfs-for-big-data-processing/

[13] Adam Oliner, Jon Stearley, What Supercomputers Say: A Study of Five System Logs, 2007. URL: https://ieeexplore.ieee.org/document/4273008/

[14] Nikolaj Buhl, F1 Score in Machine Learning, 2023. URL: https://encord.com/blog/f1-score-in-machine-learning/

[15] Jasmin Bogatinovski, Gjorgji Madjarov, Sasho Nedelkoski, Jorge Cardoso, Odej Kao, Leveraging Log Instructions in Log-based AnomalyDetection, 2022. URL: https://arxiv.org/pdf/2207.03206.pdf

[16] Hongcheng Guo, Xingyu Lin, Jian Yang, Yi Zhuang, Jiaqi Bai, TieqiaoZheng, Liangfan Zheng, Weichao Hou, Bo Zhang, Zhoujun Li, TRANSLOG: A Unified Transformer-based Framework forLog Anomaly Detection, 2022. URL: https://arxiv.org/pdf/2201.00016.pdf

[17] Yongzheng Xie, Hongyu Zhang, Bo Zhang, Muhammad Ali Babar, Sha Lu, LogDP: Combining Dependency and Proximityfor Log-based Anomaly Detection, 2021. URL: https://arxiv.org/pdf/2110.01927.pdf

[18] He Cheng, Depeng Xu, Shuhan Yuan, Xintao Wu, Fine-grained Anomaly Detection in Sequential Datavia Counterfactual Explanations, 2022. URL: https://arxiv.org/pdf/2210.04145.pdf

[19] Saswati Ray, Sana Lakdawala, Mononito Goswami, Chufan Gao, Learning Probabalistic Graph Neural Networks forMultivariate Time Series Anomaly Detection, 2021. URL: https://arxiv.org/pdf/2111.08082v1.pdf

[20] Van-Hoang Le, Hongyu Zhang, Log-based Anomaly Detection Without Log Parsing, 2021. URL: https://arxiv.org/pdf/2108.01955.pdf

[21] Davide Sanvito, Giuseppe Siracusano, Sharan Santhanam, Roberto Gonzalez, Roberto Bifulco, syslrn: Learning What to Monitor for EfficientAnomaly Detection, 2022. URL: https://arxiv.org/pdf/2203.15324.pdf

[22] Yipeng Ji, Jingyi Wang, Shaoning Li, Yangyang Li, Shenwen Lin, Xiong Li, An Anomaly Event Detection Method Based on GNN Algorithmfor Multi-data Sources, 2021. URL: https://arxiv.org/pdf/2104.08761.pdf

[23] Zumin Wang, Jiyu Tian, Hui Fang, Liming Chen, Jing Qin, LightLog: A lightweight temporal convolutional network for log anomaly detection on the edge, 2022. URL: https://www.sciencedirect.com/science/article/abs/pii/S1389128621005119

[24] Bruno Wassermann, David Ohana, Ronen Schaffer, Robert Shahla, Elliot K. Kolodner, Eran Raichstein, Michal Malka, DeCorus: Hierarchical Multivariate Anomaly Detection atCloud-Scale, 2022. URL: https://arxiv.org/pdf/2202.06892.pdf

[25] Thorsten Wittkopp, Alexander Acker, Sasho Nedelkoski, Jasmin Bogatinovski, Dominik Scheinert, Wu Fan, Odej Kao, A2Log: Attentive Augmented Log Anomaly Detection, 2021. URL: https://arxiv.org/pdf/2109.09537.pdf

[26] Harold Ott, Jasmin Bogatinovski, Alexander Acker, Sasho Nedelkoski, Odej Kao, Robust and Transferable Anomaly Detection in LogData using Pre-Trained Language Models, 2021. URL: https://arxiv.org/pdf/2102.11570.pdf

[27] Prateek Chanda, Malay Bhattacharya, Distributed Anomaly Detection in Edge Streams usingFrequency based Sketch Datastructures, 2021. URL: https://arxiv.org/pdf/2111.13949.pdf

[28] Yukyung Lee, Jina Kim, Pilsung Kang , LAnoBERT : System Log Anomaly Detectionbased on BERT Masked Language Model, 2023. URL: https://arxiv.org/pdf/2111.09564.pdf

[29] Zhuangbin Chen, Jinyang Liu, Wenwei Gu, Yuxin Su, Jieming Zhu, Yongqiang Yang, Michael R. Lyu, Experience Report: Deep Learning-based System Log Analysisfor Anomaly Detection, 2022. URL: https://arxiv.org/pdf/2107.05908.pdf

[30] Haixuan Guo, Shuhan Yuan, Xintao Wu, LogBERT: Log Anomaly Detection via BERT, 2021. URL: https://arxiv.org/pdf/2103.04475.pdf

[31] Jonghyeon Ko, Marco Comuzz, Online anomaly detection using statisticalleverage for streaming business process events, 2021. URL: https://arxiv.org/pdf/2103.00831.pdf

[32] Yicheng Guo, Yujin Wen, Congwei Jian, Yixin Lian, Yi Wan, Detecting Log Anomalies with Multi-Head Attention (LAMA), 2021. URL: https://arxiv.org/pdf/2101.02392.pdf

[33] Ahsan Mahmood, Junier Oliva, Martin Styner, Anomaly Detection via Gumbel Noise Score Matching, 2023. URL: https://arxiv.org/pdf/2304.03220.pdf

[34] Shayan Hashemi, Mika Mäntylä, OneLog: Towards End-to-End Training in Software Log Anomaly Detection , 2021. URL: https://arxiv.org/pdf/2104.07324v1.pdf

[35] Thomas Davies, Topological Data Analysis for Anomaly Detection in Host-Based Logs , 2022. URL: https://arxiv.org/pdf/2204.12919.pdf

[36] Jasmin Bogatinovski, Sasho Nedelkoski, Multi-Source Anomaly Detection in Distributed IT Systems, 2021. URL: https://arxiv.org/pdf/2101.04977.pdf

[37] Siavash Ghiasvand, Florina M. Ciorba, Anomaly Detection in High Performance Computers: A Vicinity Perspective, 2019. URL: https://arxiv.org/pdf/1906.04550.pdf