ADBA Computer Science
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8544
RESEARCH ARTICLE
Vol.1 / No.1 / 2024 / pp.14-18
https://doi.org/10.69882/adba.cs.2024073

Enhancing Anomaly Detection in Large-Scale Log Data Using Machine Learning: A Comparative Study of SVM and KNN Algorithms with HDFS Dataset

Yusuf Alaca [*,1], **Erdal Başaran** [α,2] **and Yüksel Çelik** [β,3]

[*]Department of Computer Engineering, Hitit University, 19030, Corum, Turkiye, [α]Department of Computer Engineering, Agrı İbrahim Cecen University, Agrı, Turkiye, [β]Department of Computer Engineering, Karabük University, Karabük, Turkiye.

ABSTRACT As information technology rapidly advances, servers, mobile, and desktop applications are easily attacked due to their high value. Therefore, cyber attacks have raised great concerns in many areas. Anomaly detection plays a significant role in the field of cyber attacks, and log records, which record detailed system runtime information, have consequently become an important data analysis object. Traditional log anomaly detection relies on programmers manually inspecting logs through keyword searches and regular expression matching. While programmers can use intrusion detection systems to reduce their workload, log data is massive, attack types are diverse, and the advancement of hacking skills makes traditional detection inefficient. To improve traditional detection technology, many anomaly detection mechanisms, especially machine learning methods, have been proposed in recent years. In this study, an anomaly detection system using two different machine learning algorithms is proposed for large log data. Using Support Vector Machines (SVM) and K-Nearest Neighbors (KNN) algorithms, experiments were conducted with the Hadoop Distributed File System (HDFS) log dataset, and experimental results show that this system provides higher detection accuracy and can detect unknown anomaly data.

KEYWORDS
Anomaly detection
KNN
SVM
Machine learning
HDFS

INTRODUCTION

In information technology infrastructures, many components and assets are interconnected and continuously interacting. Therefore, determining the cause of cyber attacks is challenging (A. Oliner and Xu 2012). Log records are considered a primary data source because they capture the runtime information of software (Sillito and Kutomi 2020). Detecting anomalies in log records is difficult due to several factors. The primary reasons include the rapidly increasing volume of logs (H. Mi and Cai 2013), the simultaneous generation of diverse log records (W. Xu and Jordan 2009), and changes in the nature of log recording due to software updates (Elbasani and Kim 2021).

In the existing literature, anomaly detection has been performed on various types of log records, including failure prediction and management (Tan and Gu 2010), RAS logs (Z. Zheng and Beckman 2010), health logs (Elbasani and Kim 2021), event logs (T. Pitakrat and Hoorn 2014), activity logs (H. Saadatfar and Deldari 2012), transactional and operational log records (T. Jia and Xu 2017), and more. Additionally, parsing log records has been achieved using frequency pattern mining (Vaarandi 2003), clustering (H. Hamooni and Mueen 2016), and natural language processing (NLP) techniques (X. Duan and Yin 2021).

In this study, the machine learning algorithms K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) are used for fast and effective anomaly detection. Analyses were conducted using the Hadoop Distributed File System (HDFS) dataset, which has been employed in numerous studies (M. Du and Srikumar 2017), achieving high success rates.

Manuscript received: 20 May 2024,
Revised: 27 June 2024,
Accepted: 28 June 2024.

[1]yusufalaca@hitit.edu.tr(**Corresponding author**)
[2]erdalbasaran@agri.edu.tr
[3]yukselcelik@ karabuk.edu.tr

DATASET DESCRIPTION AND PREPARATION

In this study, experiments were conducted using the HDFS dataset. This dataset has been labeled as normal and abnormal by Hadoop experts. Table 1 shows the time span, number of log lines, and the amount of labeled abnormal data in this dataset. The HDFS log dataset was collected from over 200 heterogeneous sources of Amazon and consists of 11,175,629 lines of log data. The HDFS log data records operations such as partitioning, replicating, and deleting within a specific block using block_id. This dataset comprises 575,061 log blocks with 16,838 labeled as abnormal by Hadoop experts (M. Du and Srikumar 2017).

The analysis of log data involves using numerical and categorical data as input, which requires the raw log data to be cleaned, sorted, and normalized. Figure 1 shows the log parsing steps. Each raw log entry consists of two parts: a timestamp and a complementary log part. The timestamp records the time of each log entry. Since timestamps in different formats are regular expressions, they can be easily extracted from raw log data during the log parsing stage. The log identifier is a token that identifies multiple processes or message exchanges within the system.

081109 203615 148 INFO dfs.DataNode$PacketResponder: PacketResponder 1 for block blk_38865049064139660 terminating

Date	081109
Time	203615
Level	INFO
Component	dfs.DataNode$PacketResponder
Event Template	PacketResponder 1 for block <*> terminating
Parameters	blk_38865049064139660

Figure 1 Steps of Log Parsing

After the log parsing steps, the data needs to be digitized. The word2vec (Church 2017) algorithm has been used to convert the textual parts of the log data into numerical values. The Mean/Mode method commonly used in the literature has been employed to address missing data, and to mitigate the impact of missing data, all missing values have been replaced with zero (Lin and Tsai 2020). Following digitization, anomaly labels generated by Hadoop experts have been appended to the end of the dataset. In the label column, 0 is used for normal data and 1 for abnormal data.

PROPOSED METHOD

Detecting anomalies in log analysis is quite challenging because log data consists of both numerical and categorical data. To enable the analysis of this data, it first undergoes preprocessing. Through log parsing, features are extracted from the dataset and transformed into a vectorized form. Subsequently, this vectorized dataset is analyzed using machine learning algorithms to detect anomalies.

Figure 2 illustrates the architecture of the proposed method. Particularly, the utilization of the word2vec algorithm for digitization during log parsing has had a significant impact on the high performance of experimental results. By employing this method, multiple machine learning algorithms have been utilized for anomaly detection from log records, resulting in high success rates. The

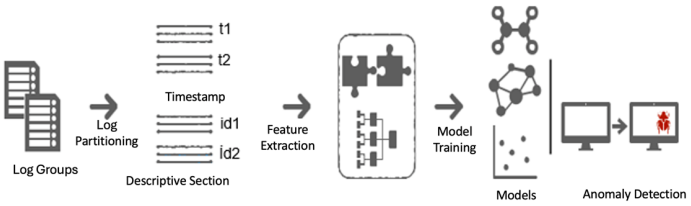

Figure 2 Architecture of the Proposed Anomaly Detection Method

log parsing process is crucial for using data in machine learning algorithms.

SUPPORT VECTOR MACHINES

Support Vector Machines (SVMs) are used for classification problems using supervised learning. Typically, they classify by drawing a line on a plane to maximize the distance between points of two classes (M. A. Hearst and Scholkopf 1998). The main objective of classification is to determine which class future data belongs to. In Figure 3, the data is divided into two classes, black and white. A line is drawn to separate these two classes, and the area between them is called the margin. The larger the margin, the better the two classes are separated. W denotes the weight vector, x denotes the input vector, and b denotes the bias. Using these values, the margin region remains between ±1.

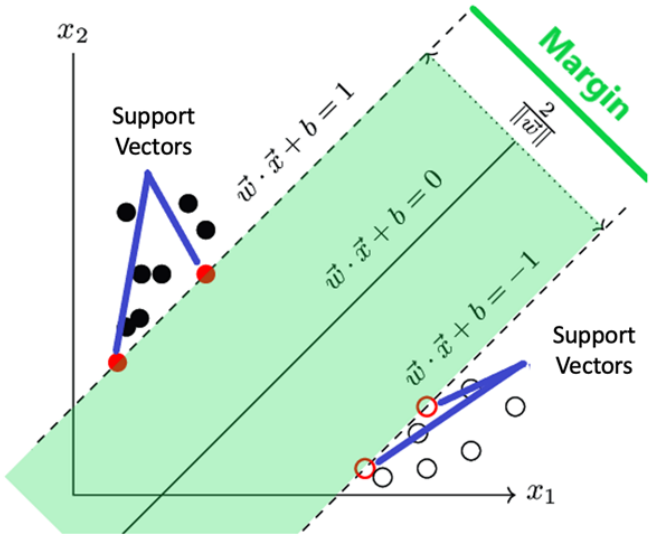

Figure 3 Working principle of Support Vector Machines

To classify low-dimensional data more efficiently, the kernel method is employed. This method expands the available data by multiplying it with kernel functions without increasing the dimensionality of the data, making it more meaningful (Steinwart and Christmann 2008). Two of the kernels used are the Polynomial and the Gaussian RBF cubic kernel. The Polynomial kernel enables processing of data from 2 dimensions to 3 or more dimensions (Moghaddam and Hamidzadeh 2016). It classifies by calculating the similarity of each point to a specific point using a normal distribution. The spread of the distribution is controlled by the gamma hyperparameter. A smaller gamma parameter leads to a wider distribution. To avoid overfitting, the gamma value should be reduced while for underfitting, it should be increased. In this

Table 1 Characteristics of the HDFS Log Dataset

Dataset	Duration	Number of Log Lines	Number of Anomalies (Blocks)
HDFS	38.7 hours	11,175,629	16,838

Table 2 Data with Missing Values Completed by Digitizing Using Word2Vec

Column 1	Column 2	Column 3	...	Column 23	Column 24	Labels
5	5	5	...	0	0	0
5	22	9	...	23	21	0
22	5	5	...	0	0	1
22	26	26	...	4	21	0
5	9	11	...	23	21	1
5	26	3	...	21	0	1

study, classification methods using normal distribution along with polynomial and cubic kernels were employed, resulting in a high success rate.

K NEAREST NEIGHBORS ALGORITHM

kNN is a supervised learning algorithm used for both classification and regression problems. It finds the k nearest neighbors to a new point and makes predictions based on those neighbors (G. Guo and Greer 2003; Ö. Tonkal and Kocaoğlu 2021). Three different distance calculation methods have been used in this study. The Euclidean distance is used to measure proximity in the kNN algorithm. Euclidean distance linearly measures the distance between two points. The calculation of Euclidean distance between points P=(x1, x2, ..., xn) and Q=(y1, y2, ..., yn) is given in Equation 1.

$$D_{PQ} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{1}$$

The Minkowski distance is expressed with a general formula and is used to define various distance metrics for different values of p. It is a generalization of distance metrics such as the Euclidean distance commonly used in machine learning, clustering, and data mining applications. The Minkowski distance between any two points P and Q, where P=(x1, x2, ..., xn) and Q=(y1, y2, ..., yn), is calculated according to Equation 2.

$$D_{PQ} = \left(\sum_{i=1}^{n}|x_i - y_i|^p\right)^{1/p} \tag{2}$$

The Mahalanobis Distance is a distance measurement system used in computer science and many other fields. Its main difference from other measurement systems is that it performs distance separation on an elliptical plane. The Mahalanobis distance is calculated as the square root of the product of the difference between the value vector and the mean, the inverse of the covariance matrix, and the transpose of the difference between the value vector and the mean. Equation 3 illustrates the calculation of the Mahalanobis distance.

$$D_M(x) = \sqrt{(x-\mu)^T S^{-1}(x-\mu)} \tag{3}$$

PERFORMANCE METRICS FOR EVALUATING THE PROPOSED METHOD

In this study, the success of the proposed method was assessed using the following criteria sequentially. Accuracy and Precision measurements were conducted according to Equations 4 and 7, respectively. These equations utilize parameters such as TN (true negatives), TP (true positives), FN (false negatives), and FP (false positives). The F-Score derived from the cumulative sum of Accuracy and Precision was calculated in Equation 8. Additionally, Precision was computed in Equation 7, and Specificity was determined in Equation 6.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{5}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{6}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

$$\text{F-Score} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{8}$$

Table 3 Classification Test Results

Classification Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	Precision (%)	F-Score (%)
SVM					
Linear	0.9794	0.9944	0.9644	0.9655	0.9797
Polynomial	0.9958	0.9978	0.9939	0.9939	0.9958
Cubic	0.9978	1	0.9956	0.9956	0.9978
kNN					
Euclidean	0.9694	0.9806	0.9583	0.9592	0.9698
Minkowski	0.9725	0.9856	0.9594	0.9605	0.9729
Mahalanobis	0.9761	0.9950	0.9572	0.9588	0.9766

CONCLUSION

Detecting anomalies from large log data is quite challenging. In this study, log parsing was conducted using word2vec on datasets containing both numerical and categorical data such as the HDFS dataset. Experimental test results have demonstrated high success using machine learning algorithms such as SVM and kNN. In the future, testing success results with different machine learning algorithms is planned.

Availability of data and material

Not applicable.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The authors have no relevant financial or non-financial interests to disclose.

LITERATURE CITED

A. Oliner, A. G. and W. Xu, 2012 Advances and challenges in log analysis. Communications of the ACM **55**: 55–61.

Church, K. W., 2017 Word2Vec. Natural Language Engineering **23**: 155–162.

Elbasani, E. and J. D. Kim, 2021 LLAD: Life-Log Anomaly Detection Based on Recurrent Neural Network LSTM. Journal of Healthcare Engineering **2021**.

G. Guo, D. B. Y. B., H. Wang and K. Greer, 2003 KNN model-based approach in classification. In *OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"*, pp. 986–996, Springer.

H. Hamooni, J. X. H. Z.-G. J., B. Debnath and A. Mueen, 2016 Logmine: Fast pattern recognition for log analytics. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 1573–1582, ACM.

H. Mi, Y. Z. M. R.-T. L., H. Wang and H. Cai, 2013 Toward fine-grained unsupervised scalable performance diagnosis for production cloud computing systems. IEEE Transactions on Parallel and Distributed Systems **24**: 1245–1255.

H. Saadatfar, H. F. and H. Deldari, 2012 Predicting job failures in AuverGrid based on workload log analysis. New Generation Computing **30**: 73–94.

Lin, W.-C. and C.-F. Tsai, 2020 Missing value imputation: a review and analysis of the literature (2006–2017). Artificial Intelligence Review **53**: 1487–1509.

M. A. Hearst, E. O.-J. P., S. T. Dumais and B. Scholkopf, 1998 Support vector machines. IEEE Intelligent Systems and their Applications **13**: 18–28.

M. Du, G. Z., F. Li and V. Srikumar, 2017 DeepLog: Anomaly detection and diagnosis from system logs through deep learning. In *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 1285–1298.

Moghaddam, V. H. and J. Hamidzadeh, 2016 New Hermite orthogonal polynomial kernel and combined kernels in support vector machine classifier. Pattern Recognition **60**: 921–935.

Sillito, J. and E. Kutomi, 2020 Failures and Fixes: A Study of Software System Incident Response. In *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 185–195, IEEE.

Steinwart, I. and A. Christmann, 2008 *Support Vector Machines*. Springer Science & Business Media.

T. Jia, P. C. Y. L.-F. M., L. Yang and J. Xu, 2017 Logsed: Anomaly diagnosis through mining time-weighted control flow graph in logs. In *2017 IEEE 10th International Conference on Cloud Computing (CLOUD)*, pp. 447–455, IEEE.

T. Pitakrat, O. K. F. K., J. Grunert and A. V. Hoorn, 2014 A framework for system event classification and prediction by means of machine learning. In *Proceedings of the 8th International Conference on Performance Evaluation Methodologies and Tools*, pp. 173–180, ACM.

Tan, Y. and X. Gu, 2010 On predictability of system anomalies in real world. In *2010 IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pp. 133–140, IEEE.

Vaarandi, R., 2003 A data clustering algorithm for mining patterns from event logs. In *Proceedings of the 3rd IEEE Workshop on IP Operations & Management (IPOM 2003)*, pp. 119–126, IEEE.

W. Xu, A. F. D. P., L. Huang and M. I. Jordan, 2009 Detecting large-scale system problems by mining console logs. In *Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles*,

pp. 117–132, ACM.
X. Duan, H. C. W. Y., S. Ying and X. Yin, 2021 OILog: An online incremental log keyword extraction approach based on MDP-LSTM neural network. Information Systems **95**: 101618.
Z. Zheng, R. G. S. C., Z. Lan and P. Beckman, 2010 A practical failure prediction with location and lead time for blue gene/p. In *2010 International Conference on Dependable Systems and Networks Workshops (DSN-W)*, pp. 15–22, IEEE.
Ö. Tonkal, E. B. Z. C., H. Polat and R. Kocaoğlu, 2021 Machine Learning Approach Equipped with Neighbourhood Component Analysis for DDoS Attack Detection in Software-Defined Networking. Electronics **10**.

How to cite this article: Alaca, Y., Basaran, E., and Celik, Y. Enhancing Anomaly Detection in Large-Scale Log Data Using Machine Learning: A Comparative Study of SVM and KNN Algorithms with HDFS Dataset. *ADBA Computer Science*, 1(1), 14-18, 2024.