Advanced risk assessment using machine learning and sentiment analysis on log data

Nidal Turab[1], Abdelrahman Abushattal[1], Jamal Al-Nabulsi[2], Hamza Abu Owida[2]
[1]Department of Networks and Cyber Security, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan
[2]Department of Medical Engineering, Faculty of Engineering, Al-Ahliyya Amman University, Amman, Jordan

Article Info

Article history:

Received Sep 8, 2024
Revised Jul 5, 2025
Accepted Aug 6, 2025

Keywords:

K-nearest neighbors
Natural language processing
Sentiment analysis
Risk assessment
Support machines

ABSTRACT

Standard risk assessment approaches are sometimes time-consuming and subjective. In order to overcome these challenges an innovative method will be presented in this article by mixing sentiment analysis and machine learning (ML). The suggested technique improves the effectiveness, precision, and scope of risk insights when it comes to the detection of feelings in logs via the use of automated data collection. The research examines several different ML classifiers and makes use of a deep learning model that has been pre-trained to evaluate risks in logs that are multi-linguistic. This proves the adaptability and scalability of our technique when used in a multilanguage setting. This combination of sentiment analysis and ML are a significant advancement in comparison to traditional approaches since it enables real-time processing and delivers important insights into the management of organizational risks.

Corresponding Author:

Nidal Turab
Department of Networks and Cyber Security, Faculty of Information Technology
Al-Ahliyya Amman University
Amman 19328, Jordan
Email: n.turab@ammanu.edu.jo

1. INTRODUCTION

Data logs are essential documentation of computer systems or network events that provide an audit trail for assessing and addressing problems. Logs play a vital role in various functions such as audit and compliance to ensure compliance with regulations. They also aid in troubleshooting by providing detailed information to find the root causes of failures [1]. In addition, logs are used for security monitoring to detect suspicious activities and potential breaches [2]. Furthermore, logs are valuable for performance analysis, helping to understand system performance and find areas that need improvement. Logs are essential instruments for preserving integrity and perfecting the performance of technological infrastructures due to their diverse nature [3].

Within the landscape of cyber security, risk assessment is quite an important piece to ensure crucial information is secured and the IT information systems are fully functional and available [4]. Several conventional risk techniques are related to cybersecurity. Expert-based risk assessment (EBRA) uses experts' knowledge to evaluate, prioritize, and define the risks that the system has. However, it is susceptible to biases and contradictions [5]. Compliance-based risk assessment (CBRA) defines the risks by guaranteeing compliance with regulations like the health insurance portability and accountability act (HIPAA) but lacks flexibility for dynamic threats [6]. Red team/blue team exercises (RTBTE) decide risk in cases with attackers (red team) versus defenders (blue team) [7]; however, it is time-consuming for data collection and pre- and post-participation status characterization [7].

Artificial intelligence (AI) has proven to be a valuable tool in evaluating the cybersecurity risks ranging from AI-based attacks to deepfake videos [8]. The speech's most potential application is use of natural language processing (NLP), which allows one to retrieve important data from written documents. NLP can help to increase the efficiency of cybersecurity risk assessment by transforming unstructured data into structured and useful information [9]. Event log data is often rich in free-text information, making it particularly useful to understand risk assessment in great detail, and NLP greatly enhances the capability of assessing these risks [9]. The importance of using sentiment analysis is using the computer algorithms to measure the psychological strain in text [10]. Sentiment analysis evaluates the subjective information included in words, phrases, or even longer text passages to find if the attitude expressed is good, negative, or neutral. It is extensively used for monitoring company reputation, client feedback, market research, and enhancing customer service via emotion-based responses [11]. Thus, sentiment analysis compensates for the time-consuming nature of risk assessment by automating it and providing real-time processing that integration improves corporate risk management efficiency, accuracy, and response.

2. RELATED WORK

An anomaly detection system using NLP approaches for log analysis, term frequency inverse document frequency (TF-IDF), polarity score, and Word2Vec for vectorization [12]. Studiawan *et al.* [13] proposed using deep learning to decide abnormalities in operating system logs using sentiment analysis. Unbalanced class distribution may be a=handle using a gated recurrent unit (GRU) layer and Tomek connection. A long short-term memory (LSTM) network was used to perform spatiotemporal sentiment analysis on disaster- related tweets [14]. The proposed risk assessment sentiment analysis (RASA) model had superior performance compared to earlier algorithms in the task of sentiment categorization. Han *et al.* [15] analyzed and compared different machine learning (ML) classifiers for the purpose of assessing software risk. Potential possibilities for enhancing the use of ML models in risk assessment were also discussed. Improved detection and less time-consuming operation are both provided by the proposed technology. Almahadin *et al.* [16] propose a physical layer secure technique using phase index RSM to overcome eavesdroppers employing k-nearest neighbors (KNN) supervised pattern recognition.

The use of support vector machines (SVM) in conjunction with recurrent neural networks (RNN) has been shown to improve the accuracy of sentiment categorization, reaching a maximum of 93.6% overall [17]. Compared to conventional approaches, transformer-based models, such as DistilBERT, have shown superior performance, obtaining accuracy rates of 96.10% in the categorization of emotion [18]. NLP is used to improve network security log analysis by making complex unstructured data into usable information using the LSTM RNN model, which provides an F1 score of about 90 [19]. Almodovar *et al.* [20] introduce LogFiT, a self-supervised ML model that enables the detection of anomalies in logs. The model is implemented based on bidirectional encoder representations from transformers (BERT). Pham and Lee [21] propose TranSentLog methods that merge transformers with sentiment analysis to provide anomaly detection of the event logs. Table 1 provides a summary of the contributions and limitations of earlier works.

Table 1. Summary of contributions and limitations from previous works

References contribution	Contribution
Deep learning-based GRU network sentiment analysis of OS is proposed [10].	Methods include manual searching or predetermined rules.
Focus on global spatiotemporal characteristics to find log anomalies [13].	Current techniques concentrate on distribution, system log temporal or geographical characteristics
Sentiment anomalies-based semi supervised [14].	Global characteristics cannot be extracted accurately for anomaly detection using current approaches. Semi-supervised anomaly detection and time-dependent confusion matrix for imbalanced dataset assessment.

3. MACHINE LEARNING CLASSIFICATION

One of the most important uses of ML is predictive and classification data. ML algorithms are every occurrence in every dataset with the same attributes [22]. ML classification could be categorized into various techniques, mainly supervised and unsupervised learning [22], [23]. When instances are provided with known labels (the outputs that correspond to them), the learning process is referred to be supervised. In this section, we will briefly illustrate some of the ML techniques that are used in our research.

3.1. Logistic regression

Although logistic regression was made to work solely for binary classification, it can also be changed to manage multiclass classifications through the use of multinomial logistic regression. This version enables the model to predict probabilities of its outcomes for more than three classes. It does so by using the softmax function instead of the sigmoid function [24], [25].

3.2. K-nearest neighbors

The KNN algorithm is predicated on the idea that every instance that is included inside a dataset would, in most cases, be found near other instances with similar characteristics. If the instances are assigned to a classification label, the label value of an unclassified instance can be decided by examining the classification of the instances closest to it. This decision is based on the classifications of the neighboring instances [26].

3.3. Support vector machines

The SVMs are a recently developed method of supervised ML. SVMs are based on the concept of a ”margin”, which refers to the distance on each side of a hyperplane that divides two classes of data. It has been proved that maximizing the margin, which is the longest possible distance between the separating hyperplane and the instances on either side, reduces the predicted generalization error [27].

3.4. Random forest classifier

A ML technique known as the random forest classifier handles constructing many decisions. It then combining them in order to provide a forecast that is both more correct and more stable. The predicted accuracy is improved by the use of averaging, which also helps to reduce overfitting [28].

4. METHODOLOGY

In this section, the system design step will be illustrated step by step. As shown in Figure 1, the information and the data frame are processed. This processing is done using Spyder 5.4.3 Python Environment.

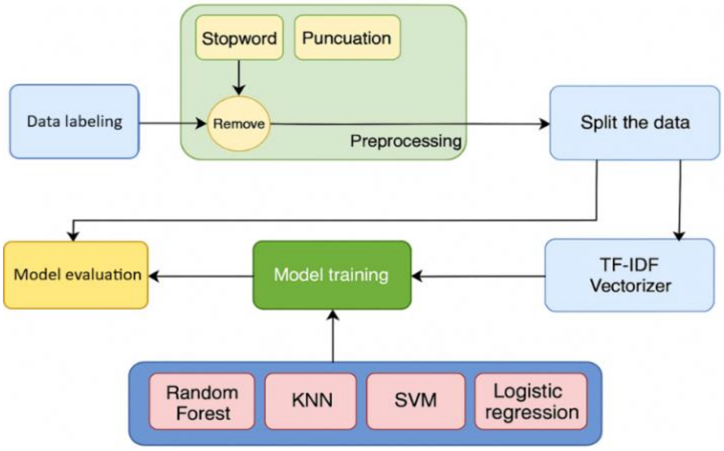


Figure 1. Block diagram for a ML classification process

4.1. Data labeling

For sentiment analysis with three sentiment classes (high, moderate, and low), corresponding to scores of 10, 5, and 2 respectively, data labeling involves categorizing text descriptions based on sentiment intensity. In this paper, manual labeling was adopted where the description of the log was generated using ChatGPT. The reason for that is the limited resources available for log datasets. Table 2 provides a sample of the data logs from the dataset, including a description, severity, and score. Figure 2 illustrates the distribution of sentiment classes in your dataset. Specifically, it shows that 40% of the dataset is classified as ”very high sentiment,” while 30% is classified as ”moderate” sentiment, and another 30% is classified as ”low” sentiment. The wide-ranging distribution of feelings is helpful for training sentiment analysis models since it enhances the model's ability to effectively categorize sentiments in unfamiliar texts by including a variety of sentiment intensities.

Table 2. Logs data sample dataset

Height time stamp	Description	Severity	Score
08/06/2024 13:20	Critical system overload due to DDoS attack. All online services are down.	Very high	10
08/06/2024 14:49	Suspicious email attachments opened in the network, moderate risk of malware spread	Moderate	5
08/06/2024 16:55	suboptimal performance was noticed in cafeteria POS systems, with a minimal risk of affecting service speed	Low	2

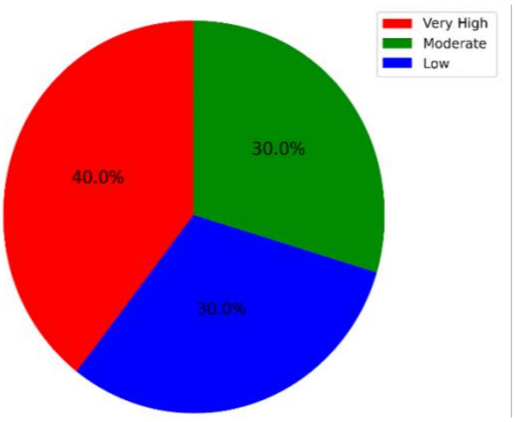


Figure 2. Data visualization for labeled data

4.2. Data preprocessing

Data preprocessing is an essential stage in sentiment analysis that focuses on improving the accuracy and efficiency of the model. The process begins with text cleansing, special characters, and capital letters disappear to ensure consistency. Next, tokenization is performed, which involves dividing the text into separate words or tokens. Stop words, which are common terms that provide little meaning, are later eliminated [29].

4.3. Data normalization and standard scaling

In order to guarantee that features are on a comparable scale, normalization and scaling are crucial preprocessing steps that should be taken. This makes it less complicated for ML models to be trained and generalized successfully. where standard x̂ is evaluated as in (1) and where μ is the mean value and σ is the standard deviation, and it could be calculated as in (2). Additionally, there should be an identical implementation of the normalizing and scaling processes for both the training dataset and the test dataset. This process should be generated using the training data.

$$\check{x} = \frac{x - \mu}{\sigma} \tag{1}$$

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(mi - \mu)^2} \tag{2}$$

4.4. Data splitting

It is essential to divide the data before training and testing ML models. To train the model, confirm its performance, and test its generalization to new data it has not seen, the conventional technique includes partitioning the dataset into two or more subsets to do these various goals. In the result section, the effect of the spitting ratio will be considered and illustrated.

4.5. Frequency-inverse document frequency vectorizer

The terminology TF-IDF calculates the relevance of a phrase in a document within a collection. Text analysis and NLP use vectorizers to translate text into ML -compatible numerical representations. It measures the significant of a word in a document compared to other texts [28]. The technique uses term frequency (TF) to quantify the frequency of a phrase in a text and inverse text frequency (IDF) to reduce terms that appear often in multiped documents. TF-IDF multiplies these two metrics to highlight unique and essential words in texts. This is used with document classification and clustering [30].

4.6. Model training

Up to this point, the training set will be fed to the selected ML classifiers and choosing a suitable method, preparing the data, initializing the model, iterating through training epochs to update parameters, and assessing the model's performance on validation and test sets are all steps that are involved in the process of training a ML model. It is an essential stage in ML, and it calls for careful attention.

4.7. Model evaluation

Model evaluation is an essential step that must be taken in order to guarantee that the trained model satisfies the necessary performance requirements and is proper for the application for which it was designed. Selecting relevant metrics, evaluating performance on test data, and iteratively changing the model as required are all activities that are included in this process. There are various metrics usually used to evaluate the model, like accuracy in (3), precision in (4), recall in (5), and the F1 score (6). The following equation is each one of them [30]:

$$Accuracy = \frac{TP+TN}{Tp+TN+FP+FN} \tag{3}$$

$$Precision = \frac{TP}{Tp+FP} \tag{4}$$

$$Recall = \frac{TP}{Tp+FN} \tag{5}$$

$$F1\ Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{6}$$

4.8. Prediction

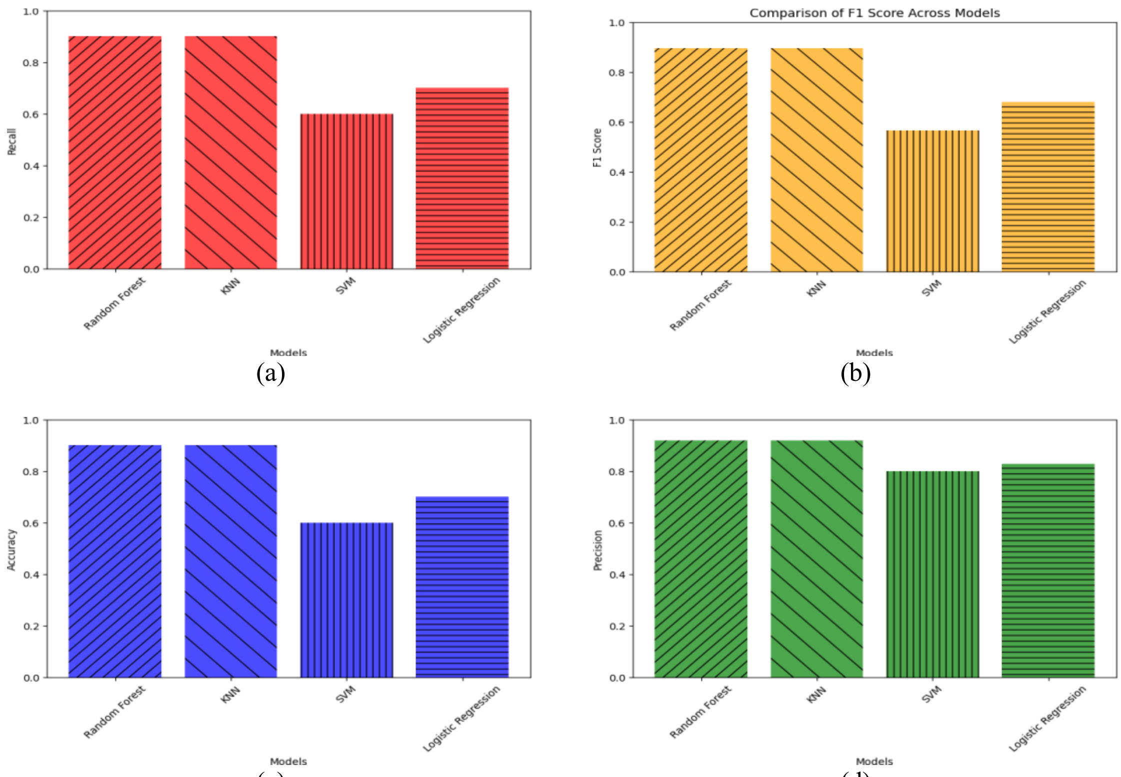
Having finished the training and evaluation of your model, and being delighted with its performance, you can employ it to generate predictions on new, unobserved data. in this paper, the uncertain binary stars in the catalog will go through the same process that the training set did and finally, predict the wanted target data.

5. RESULTS

The performance matrix that is considered is the accuracy metric which reflects the accuracy that the model predicts the correct target values. Comparisons are made between the performance metrics of four distinct classifiers, namely SVM, KNN, random forest, and logistic regression. These comparisons are made across four major criteria: recall, accuracy, and precision, as well as the F1 score.

Figure 3 provides the performance assessment results of several classification models using various metrics. Figure 3(a) shows the accuracy, with the random forest classifier achieving the best accuracy of 90%. About 60% is the lowest level achieved by SVM. Figure 3(b) shows how each model precision decides positive instances while minimizing false positives. The results show that random forest has the greatest precision (92%) while SVM has the lowest, around 80%. Figure 3(c) compares models using recall metric, balancing positive event detection with missing and found ones. According to the results, the random forest classifier has the highest recall rate at 94%. Additionally, SVM yields the lowest level, 60%. The balanced classifier is decided by the F1-score, which combines precision and recall performance in Figure 3(d). Superior performance is achieved with random forest (94.5%).

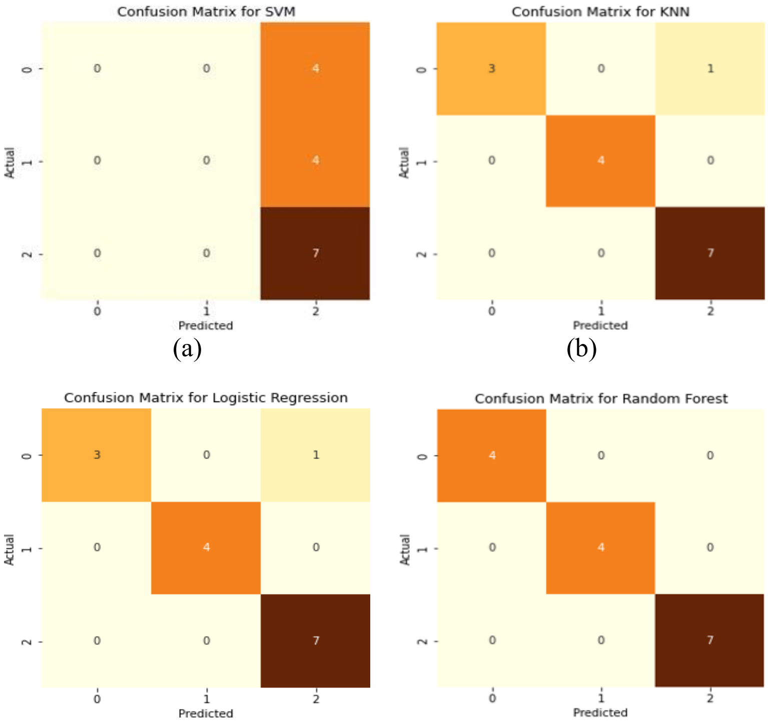
The confusion matrices in Figure 4 show that how the four ML models' performance was categorized into three groups: low (0), moderate (1), and high (2). Figure 4(a) presents the confusion matrix for the KNN classifier, where it missed classifying the low class as moderate. On the other hand, Figure 4(b) shows the confusion matrix of SVM, where it has a poor performance classification that is biased to ”high” classes. Figure 4(c) displays the confusion matrix for logistic regression, where it provides excellent classification with one missing (low) class. Finally, Figure 4(d) provides the random forest classifier, it has the best classification for all instances. The results show that the random forest classifier performs high scores where it decides all the instances correctly. Thus, the random forest classifier is the most suitable model for sentiment analysis for risk assessment.



(a) (b)

(c) (d)

Figure 3. The comparison of classification techniques: (a) accuracy comparison, (b) precision comparison, (c) recall comparison, and (d) F1 score comparison



(a) (b)

(c) (d)

Figure 4. The confusion matrix for classification techniques: (a) confusion matrix for KNN, (b) confusion matrix for SVM, (c) confusion matrix for logistic regression, and (d) confusion matrix for random forest

6. CONCLUSION

The findings of the research revealed that the combination of sentiment analysis and ML classifiers has the potential to considerably improve the effectiveness and precision of risk assessments using both methods. The system was able to automate the identification of sentiment in logs by using methods such as SVM, KNN, and random forest classifiers. This allows for a more in-depth understanding of possible dangers. A further illustration of the flexibility of the model across a variety of languages and datasets is provided by the use of a pre-trained deep learning model for the analysis of non- English logs. To increase the predicted accuracy of the classification algorithms.

FUNDING INFORMATION

Authors state there is no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration. The following table provides a summary of the author's contributions to this research paper. Each author's responsibilities in the advancement of this research are listed in detail.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Nidal Turab	✓	✓	✓	✓	✓	✓		✓	✓	✓		✓	✓	
Abdelrahman Abushattal	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓			
Jamal Al-Nabulsi	✓					✓	✓		✓		✓	✓	✓	✓
Hamza Abu Owida	✓	✓								✓		✓		✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis

I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting

Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that supports the findings of this study are available from the corresponding author, [NT], upon reasonable request.

REFERENCES

[1] B. Chen and Z. M. J. Jiang, "A survey of software log instrumentation," *ACM Computing Surveys*, vol. 54, no. 4, pp. 1–34, May 2022, doi: 10.1145/3448976.

[2] R. Ávila, R. Khoury, R. Khoury, and F. Petrillo, "Use of security logs for data leak detection: a systematic literature review," *Security and Communication Networks*, vol. 2021, pp. 1–29, Mar. 2021, doi: 10.1155/2021/6615899.

[3] O. Yapar, "Blockchain-based database management for national security: ensuring data integrity and privacy," *SSRN Electronic Journal*, 2024, doi: 10.2139/ssrn.5032985.

[4] A. Al-Hawamleh, "Cyber resilience framework: strengthening defenses and enhancing continuity in business security," *International Journal of Computing and Digital Systems*, vol. 15, no. 1, pp. 1315–1331, Mar. 2024, doi: 10.12785/ijcds/150193.

[5] M. Krisper, J. Dobaj, and G. Macher, "Assessing risk estimations for cyber-security using expert judgment," *Communications in Computer and Information Science*, pp. 120–134, 2020, doi: 10.1007/978-3-030-56441-4_9.

[6] D. Itani, R. Itani, A. A. Eltweri, A. Faccia, and L. Wanganoo, "Enhancing cybersecurity through compliance and auditing: a strategic approach to resilience," in *2024 2nd International Conference on Cyber Resilience (ICCR)*, Feb. 2024, pp. 1–10, doi: 10.1109/ICCR61006.2024.10532959.

[7] J. Straub, "Assessment of cybersecurity competition teams as experiential education exercises," *ASEE Annual Conference and Exposition, Conference Proceedings*, 2020, doi: 10.18260/1-2--34187.

[8] A. Nassar and M. Kamal, "Machine learning and big data analytics for cybersecurity threat detection: a holistic review of techniques and case studies," *Journal of Artificial Intelligence and Machine Learning in Management*, vol. 5, no. 1, pp. 51–63, 2021.

[9] F. Ekundayo, I. Atoyeb, A. Soyele, and E. Ogunwobi, "Predictive analytics for cyber threat intelligence in fintech using big data and machine learning," *International Journal of Research Publication and Reviews*, vol. 5, no. 11, pp. 5934–5948, 2024, doi: 10.55248/gengpi.5.1124.3352.

[10] S. M. Mohammad, "Sentiment analysis: detecting valence, emotions, and other affectual states from text," in *Emotion Measurement*, Elsevier, 2016, pp. 201–237, doi: 10.1016/B978-0-08-100508-8.00009-6.

[11] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: a survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, Dec. 2014, doi: 10.1016/j.asej.2014.04.011.

[12] D. A. Bhanage and A. V. Pawar, "Improving classification-based log analysis using vectorization techniques," *Lecture Notes in Networks and Systems*, vol. 612, pp. 271–282, 2023, doi: 10.1007/978-981-19-9228-5_24.

[13] H. Studiawan, F. Sohel, and C. Payne, "Anomaly detection in operating system logs with deep learning-based sentiment analysis," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2136–2148, Sep. 2021, doi: 10.1109/TDSC.2020.3037903.

[14] M. Parimala, R. M. S. Priya, M. P. K. Reddy, C. L. Chowdhary, R. K. Poluru, and S. Khan, "Spatiotemporal-based sentiment analysis on tweets for risk assessment of event using deep learning approach," *Software: Practice and Experience*, vol. 51, no. 3, pp. 550–570, Mar. 2021, doi: 10.1002/spe.2851.

[15] P. Han, H. Li, G. Xue, and C. Zhang, "Distributed system anomaly detection using deep learning-based log analysis," *Computational Intelligence*, vol. 39, no. 3, pp. 433–455, Jun. 2023, doi: 10.1111/coin.12573.

[16] G. AlMahadin, M. O. Hiari, A. H. Hussein, N. M. M. Turab, A. Alkhresheh, and M. A. B. Al-Tarawneh, "Performance evaluation of an intelligent and optimized machine learning framework for attack detection," *International Journal of Communication Networks and Information Security*, vol. 14, no. 3, pp. 358–371, 2022.

[17] A. Srivastava, V. Srivastava, K. Kumar, S. Srivastava, and N. Garg, "Hybrid machine learning method for sentiment analysis," in *3rd International Conference on Innovative Mechanisms for Industry Applications, ICIMIA 2023-Proceedings*, Dec. 2023, pp. 646–652, doi: 10.1109/ICIMIA60377.2023.10426420.

[18] S. Bayat and G. Işik, "Evaluating the effectiveness of different machine learning approaches for sentiment classification," *Iğdır Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, vol. 13, no. 3, pp. 1496–1510, Sep. 2023, doi: 10.21597/jist.1292050.

[19] J. Wu and J. Xiao, "Application of natural language processing in network security log analysis," *Journal of Computer Technology and Applied Mathematics*, vol. 1, no. 3, pp. 39–47, 2024, doi: 10.5281/zenodo.13366745.

[20] C. Almodovar, F. Sabrina, S. Karimi, and S. Azad, "LogFiT: log anomaly detection using fine-tuned language models," *IEEE Transactions on Network and Service Management*, vol. 21, no. 2, pp. 1715–1723, Apr. 2024, doi: 10.1109/TNSM.2024.3358730.

[21] T. A. Pham and J. H. Lee, "TransSentLog: interpretable anomaly detection using transformer and sentiment analysis on individual log event," *IEEE Access*, vol. 11, pp. 96272–96282, 2023, doi: 10.1109/ACCESS.2023.3311146.

[22] A. Booker, V. Chiu, N. Groff, and V. J. Richardson, "AIS research opportunities utilizing machine learning: from a meta-theory of accounting literature," *International Journal of Accounting Information Systems*, vol. 52, Mar. 2024, doi: 10.1016/j.accinf.2023.100661.

[23] O. Abualghanam, O. Adwan, M. A. Al Shariah, and M. Qatawneh, "Enhancing the speed of the learning vector quantization (LVQ) algorithm by adding partial distance computation," *Cybernetics and Information Technologies*, vol. 22, no. 2, pp. 36–49, 2022, doi: 10.2478/cait-2022-0015.

[24] M. A. Alsharaiah *et al.*, "Neural network prediction model to explore complex nonlinear behavior in dynamic biological network," *International Journal of Interactive Mobile Technologies*, vol. 16, no. 12, pp. 32–51, 2022, doi: 10.3991/ijim.v16i12.30467.

[25] M. S. U. Sourav *et al.*, "Transformer-based text classification on unified bangla multi-class emotion corpus," in *2024 25th International Arab Conference on Information Technology, ACIT 2024*, Dec. 2024, pp. 1–7, doi: 10.1109/ACIT62805.2024.10877210.

[26] R. K. Halder, M. N. Uddin, M. A. Uddin, S. Aryal, and A. Khraisat, "Enhancing k-nearest neighbor algorithm: a comprehensive review and performance analysis of modifications," *Journal of Big Data*, vol. 11, no. 1, Aug. 2024, doi: 10.1186/s40537-024-00973-y.

[27] M. H. Ibrahim, E. A. Badran, and M. H. Abdel-Rahman, "Detect, classify, and locate faults in DC microgrids based on support vector machines and bagged trees in the machine learning approach," *IEEE Access*, vol. 12, pp. 139199–139224, 2024, doi: 10.1109/ACCESS.2024.3466652.

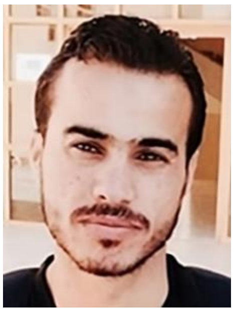
[28] F. Bin, S. Hosseini, J. Chen, P. Samui, H. Fattahi, and D. Jahed Armaghani, "Proposing optimized random forest models for predicting compressive strength of geopolymer composites," *Infrastructures*, vol. 9, no. 10, 2024, doi: 10.3390/infrastructures9100181.

[29] M. C. Hinojosa Lee, J. Braet, and J. Springael, "Performance metrics for multilabel emotion classification: comparing micro, macro, and weighted F1-scores," *Applied Sciences*, vol. 14, no. 21, 2024, doi: 10.3390/app14219863.

[30] N. Rajagukguk, I. P. E. N. Kencana, and I. G. N. L. W. Kusuma, "Application of term frequency-inverse document frequency in the Naive Bayes algorithm for ChatGPT user sentiment analysis," *Proceedings of the First International Conference on Applied Mathematics, Statistics, and Computing (ICAMSAC 2023)*, pp. 29–40, 2024, doi: 10.2991/978-94-6463-413-6_4.

BIOGRAPHIES OF AUTHORS

Nidal Turab is Ph.D. in computer science Professor at the Networks and Cyber Security Department, Al-Ahliyya Amman University, Jordan. His research interests include WLAN security, computer networks security and cloud computing security, eLearning, and internet of things. He can be contacted at email: n.turab@ammanu.edu.jo.



Abdelrahman Abushattal is a IEEE student member that received a B.Sc. in communication engineering from Al-Hussein Bin Talal University in 2012 and an M.Sc. from Mutah University in 2016. He is pursuing a Master's in Cybersecurity at Al-Ahliyya Amman University, Jordan, and a Ph.D. in Electrical and Electronics Engineering at Karadeniz Technical University, Trabzon, Turkey. Research interests include physical-layer security, power-domain NOMA, orthogonal time-frequency space, spatial modulation, robotic design, AI, fluid control networks, natural language processing, and computer vision. He can be contacted at email: ceabushattal@gmail.com.

Jamal Al-Nabulsi is Ph.D. in Biomedical Engineering, Professor at the Medical Engineering Department, Al-Ahliyya Amman University, Jordan. His research interests are biomedical sensors, digital signal processing, and image processing. He can be contacted at email: j.nabulsi@ammanu.edu.jo.

Hamza Abu Owida is Ph.D. in Biomedical Engineering, Assistant Professor at the Medical Engineering Department, Al-Ahliyya Amman University, Jordan. Research interests focused on biomedical sensors, nanotechnology, and tissue engineering. He can be contacted at email: h.abuowida@ammanu.edu.jo.