International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Efficient Feature Engineering-Based Anomaly Detection for Network Security

Mayukha S, Dr.R.Vadivel

Submitted: 03/02/2024 **Revised**: 11/03/2024 **Accepted**: 17/03/2024

Abstract: With the rapid advancement of internet technology, network-based attacks have become increasingly prevalent, posing significant challenges to ensuring the security of network infrastructures. In response, feature selection and feature reduction have emerged as essential techniques for dealing with the large volumes of data inherent in network security applications. However, traditional feature selection methods may not always suffice when all attributes are crucial for anomaly detection. To address this, we propose a feature engineering-based approach that combines feature selection and feature extraction to effectively reduce dimensionality while preserving relevant attributes. Specifically, we introduce a Stochastic-based Feature Engineering (S_FE) algorithm tailored for both manual packet capture and real-time payload datasets. In manual packet capture datasets, our algorithm extracts Trust Value, Byte Frequency Analysis (BFA), Byte Entropy (BE), Payload Length (PL), and Stream Index features, while for real-time payload datasets, it focuses on Trust Value, direction, and Hash Value features. We compare the performance of our S_FE algorithm against widely used Feature Engineering (FE) algorithms using key metrics such as accuracy, precision, recall, and F1-score. Experimental results demonstrate the superior performance of our proposed algorithm, highlighting its efficacy in network anomaly detection. This research contributes to the development of efficient techniques for enhancing network security in the face of evolving cyber threats.

Keywords: *Network security, Feature engineering, Anomaly detection, Dimensionality reduction, Packet capture*

1. Introduction

In today's interconnected digital landscape, ensuring the security of network infrastructures has become paramount, given the incessant rise in sophisticated cyber threats and network-based attacks. As the internet continues to evolve and expand, so too do the methods employed by malicious actors to exploit vulnerabilities and compromise sensitive data. In this context, effective anomaly detection mechanisms play a critical role in safeguarding networks against potential breaches and intrusions.

Anomaly detection involves the identification of deviations from normal behavior within network traffic or system activity, signaling potential security threats or malicious activities. However, the sheer volume and complexity of data generated by modern networks present significant challenges for traditional anomaly detection approaches. Conventional methods often struggle to efficiently process and analyze large datasets, leading to performance bottlenecks and reduced detection accuracy.

To address these challenges, researchers and practitioners have turned to feature engineering as a promising avenue for enhancing anomaly detection in network security. Feature engineering encompasses a range of techniques aimed at extracting and selecting relevant features from raw data, thereby improving the performance and efficiency of anomaly detection algorithms. By transforming complex data into meaningful features, feature engineering enables more effective detection of anomalous patterns and behaviors within network traffic.

One particularly promising approach within the realm of feature engineering is the development of efficient algorithms tailored specifically for anomaly detection in network security applications. These algorithms leverage advanced techniques such as dimensionality reduction, feature selection, and custom feature extraction to optimize the detection process while minimizing computational overhead. By intelligently selecting and engineering features that capture the underlying characteristics of network traffic, these algorithms can significantly enhance the accuracy and reliability of anomaly detection systems.

In this context, the present study focuses on the development and evaluation of an efficient feature engineering-based anomaly detection framework for network security. The proposed framework leverages state-of-the-art techniques in feature selection and extraction to identify and prioritize relevant features within network traffic data. By integrating these features into a robust anomaly detection pipeline, the framework aims to improve the detection capabilities of existing security systems while mitigating the impact of false positives and false negatives.

Central to the proposed framework is the utilization of stochastic-based feature engineering algorithms, which

[1] *Department of Information Technology, School of Computer Science and Engineering, Bharathiar University, Coimbatore – 641 009*
ORCID ID : 0000-0002-3463-2722
[2] *Department of Information Technology, School of Computer Science and Engineering, Bharathiar University, Coimbatore – 641 009*
ORCID ID : 0000-0001-9684-909X

offer a principled approach to feature selection and extraction in the context of network security. These algorithms employ probabilistic models and optimization techniques to identify the most informative features while accounting for the inherent uncertainty and variability present in network data. By adaptively adjusting feature selection criteria based on the observed data distribution, stochastic-based algorithms can effectively adapt to evolving network environments and emerging security threats.

To validate the efficacy of the proposed framework, extensive experiments and performance evaluations are conducted using real-world network datasets and benchmarking against existing anomaly detection methods. Key performance metrics such as accuracy, precision, recall, and F1-score are utilized to assess the effectiveness of the framework in detecting various types of network anomalies. The results of these evaluations provide valuable insights into the strengths and limitations of the proposed approach, paving the way for future advancements in feature engineering-based anomaly detection for network security.

1.1. Feature Engineering

Feature engineering is a fundamental aspect of machine learning and data analysis that involves creating or selecting relevant features from raw data to improve the performance of predictive models or analytical algorithms. In essence, it is the process of transforming raw data into a format that is more suitable for modeling, thereby enhancing the accuracy, interpretability, and efficiency of machine learning algorithms.

The importance of feature engineering lies in its ability to extract meaningful information from data, enabling algorithms to effectively capture patterns and relationships that contribute to the desired outcome. By carefully selecting or creating features that capture the underlying structure of the data, feature engineering can significantly enhance the predictive power of machine learning models and facilitate better decision-making in various domains, including finance, healthcare, cybersecurity, and more.

Key techniques in feature engineering include:

1. Feature Selection: Identifying and selecting the most relevant features from the original dataset, often based on statistical measures, domain knowledge, or algorithmic criteria. Feature selection helps reduce dimensionality, improve model performance, and mitigate the risk of overfitting.

2. Feature Extraction: Generating new features from existing ones through techniques such as dimensionality reduction, transformation, or aggregation. Feature extraction aims to uncover hidden patterns or relationships in the data, making it easier for algorithms to learn and generalize from the information provided.

3. Feature Construction: Creating new features by combining or transforming existing ones based on domain knowledge or heuristic rules. Feature construction allows for the incorporation of domain-specific insights or assumptions into the modeling process, enhancing the relevance and interpretability of the features.

Overall, feature engineering plays a crucial role in the success of machine learning projects, as the quality and relevance of features directly impact the performance and effectiveness of predictive models. By leveraging domain expertise, algorithmic techniques, and creative problem-solving, practitioners can harness the power of feature engineering to unlock valuable insights and drive innovation in data-driven decision-making.

2. Existing Algorithms

Existing algorithms such as Principal Component Analysis (PCA), Lasso Regression, and Linear Discriminant Analysis (LDA) have been widely employed in the domain of anomaly detection in network security, each offering distinct advantages and methodologies for feature engineering and dimensionality reduction.

Principal Component Analysis (PCA) is a dimensionality reduction technique commonly used in anomaly detection to transform high-dimensional data into a lower-dimensional subspace while preserving as much variance as possible. PCA achieves this by identifying the principal components of the data, which are orthogonal vectors that capture the directions of maximum variance. By projecting the data onto these principal components, PCA effectively reduces the dimensionality of the feature space, making it easier to detect anomalies and patterns within the data. However, PCA assumes linear relationships between variables and may not perform optimally when dealing with non-linear data distributions or sparse datasets.

Lasso Regression, also known as L1 regularization, is a regression technique that introduces a penalty term based on the absolute magnitude of the coefficients, encouraging sparsity in the resulting model. In the context of anomaly detection, Lasso Regression can be used to select a subset of the most informative features while shrinking the coefficients of irrelevant or redundant features to zero. By enforcing sparsity in the feature space, Lasso Regression effectively reduces the dimensionality of the data and enhances the interpretability of the model. However, Lasso Regression may struggle with multicollinearity and may not always select the most relevant features for anomaly detection.

Linear Discriminant Analysis (LDA) is a supervised dimensionality reduction technique commonly used for

classification tasks but can also be adapted for anomaly detection. LDA seeks to find a linear combination of features that maximizes the separation between different classes while minimizing the within-class variance. By projecting the data onto this discriminant subspace, LDA can effectively reduce dimensionality while preserving class-specific information, making it well-suited for anomaly detection tasks where class imbalance or class-specific anomalies are present. However, LDA assumes that the data distributions are Gaussian and may not perform well with non-linear or non-Gaussian data distributions.

In summary, PCA, Lasso Regression, and LDA are three existing algorithms that can be utilized for feature engineering and dimensionality reduction in anomaly detection for network security. Each algorithm offers distinct advantages and trade-offs, and the choice of algorithm depends on the specific characteristics of the data and the desired performance criteria. By leveraging these algorithms in combination with advanced feature engineering techniques, researchers and practitioners can develop more robust and effective anomaly detection systems for safeguarding network infrastructures against emerging cyber threats.

3. Proposed Algorithm

The proposed Stochastic-based Feature Engineering (S_FE) algorithm aims to enhance anomaly detection in network security by extracting relevant features from both manual packet capturing, and real-time payload extraction datasets as described in Figure 3.1.

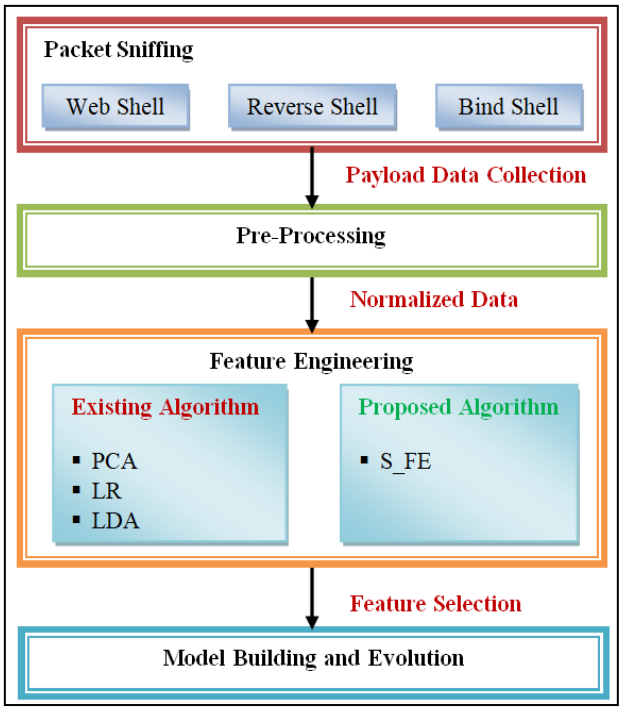

Fig. 1. Proposed System Architecture

The algorithm offers several key contributions:

1. Trust Value (Customized value): The S_FE algorithm incorporates a customizable Trust Value feature, which enables the quantification of the trustworthiness or reliability of network traffic. This feature allows for the prioritization of trustworthy data while identifying potentially malicious or anomalous behavior.

2. Byte Frequency Analysis (BFA): By performing Byte Frequency Analysis, the algorithm extracts information regarding the frequency distribution of byte values within network packets. This feature provides insights into the composition and structure of network traffic, facilitating the detection of abnormal patterns or anomalies.

3. Byte Entropy (BE): The S_FE algorithm calculates Byte Entropy, which measures the randomness or uncertainty of byte sequences within network payloads. High entropy values indicate a greater degree of randomness, while low entropy values suggest more predictable or structured data. By incorporating Byte Entropy as a feature, the algorithm can identify anomalous data patterns that deviate from expected entropy levels.

4. Payload Length (PL): Extracting Payload Length information allows the algorithm to analyze the size and distribution of payloads within network packets. Variations in payload length may indicate abnormal network activity, such as data exfiltration or denial-of-service attacks. By including Payload Length as a feature, the algorithm enhances its ability to detect such anomalies.

5. Stream Index: The algorithm incorporates Stream Index as a feature, which provides information about the sequence or order of network packet transmissions. Analyzing stream indices enables the detection of suspicious or out-of-sequence packet flows, which may indicate attempts to evade detection or exploit vulnerabilities in network protocols.

6. Direction: For real-time (automatic) payload extraction, the S_FE algorithm extracts the Direction feature, which denotes the directionality of network traffic (e.g., incoming, or outgoing). This feature facilitates the identification of asymmetrical or irregular traffic patterns that may indicate potential security threats or anomalies.

7. Hash Value: Additionally, the algorithm extracts Hash Value features from real-time payload extraction datasets. Hash values enable the efficient comparison and identification of identical or similar payloads, aiding in the detection of known attack signatures or patterns.

Furthermore, the proposed research work implements the S_FE algorithm in the context of various shell attacks, including Web Shell, Reverse Shell, and Bind Shell. By evaluating the algorithm's performance across different attack scenarios, the study provides insights into its effectiveness in detecting and mitigating a range of

network-based security threats. Overall, the S_FE algorithm represents a comprehensive approach to feature engineering in network security, offering enhanced capabilities for anomaly detection and threat mitigation in both manual and real-time network environments.

3.1. Pseudocode

The proposed algorithm outlines a systematic approach to enhancing anomaly detection in network payload data through stochastic-based feature engineering. Beginning with the initialization of the stochastic-based feature engineering function, the algorithm proceeds to randomly select and extract a specified number of feature indices without replacement from the payload data. This stochastic selection ensures diversity in feature extraction, allowing for a comprehensive representation of the data. Subsequently, the main function is executed to perform stochastic-based feature engineering, initializing an empty dictionary to store one-hot encoded features. Iterating through each payload in the dataset, categorical features are extracted and encoded using a one-hot encoding scheme. Additionally, the algorithm groups payloads by length, facilitating the analysis of payload characteristics and their relationship to network anomalies. Key features such as Trust Value (TV), Byte Frequency Analysis (BFA), Byte Entropy (BE), Payload Length (PL), Stream Index (SI), Hash Value (HV), and direction are then selected to capture various aspects of payload data. By incorporating these features, the algorithm aims to provide comprehensive insights into network traffic patterns and abnormalities. Finally, the performance of the model is evaluated to assess its effectiveness in detecting anomalies within network payload data. This algorithm offers a comprehensive framework for improving anomaly detection in network security applications by systematically extracting and encoding relevant features from payload data, thereby enhancing the accuracy and reliability of anomaly detection systems.

Input : Payload Dataset

Output : Select appropriate features from payload dataset

Procedure

Start
Step 1. To perform stochastic-based feature engineering function on the payload data.
Step 2. Randomly selects a specified number of feature indices without replacement and extracts the corresponding features from the data.
Step 3. Execute the main function to perform S_FE
Step 4. Initialize an empty dictionary to store the one-hot encoded features
Step 5. Iterate through each payload in the dataset
Step 6. Initialize an empty dictionary to store the one-hot encoded features for the current payload
Step 7. Mask with one-hot encoded labels
Step 8. function ENCODE_LABELS
Step 9. Determine size of one-hot vectors with maximum value of integer label
Step 10. for mask in ENCODE_LABELS do
Step 11. encode integer label to one-hot vector
Step 12. end for
Step 13. return
Step 14. one-hot encoded masks
Step 15. end function
Step 16. Extract categorical features from the payload
Step 17. Iterate through each categorical feature
Step 18. Check if the feature is already present in the dictionary
Step 19. If not present, initialize an empty list for the feature
Step 20. Check if the feature value is already encoded
Step 21. If not encoded, append it to the list and encode it
Step 22. Initialize a dictionary to store payloads grouped by length
Step 23. Calculate the length of the payload
Step 24. Check if the length is already present in the dictionary
Step 25. If not present, initialize an empty list for the length
Step 26. Append the payload to the list corresponding to its length
Step 27. Select the TV, BFA, BE, PL, SI, HV and direction features.
Step 28. Evaluate the performance of the model
End

4. Experimental Results

All the experiments are done in Python and a Tensor Flow on a system equipped with Intel Core i5, 2.53 GHz, 8 GB of RAM, 64-bit, and running on Windows 10.

4.1. Dataset Details

The result outcome of the pre-processed payload data is the input of this phase. The features used in the work are demonstrated in Table 4.1.

Table 1. Selected Engineered Features

1	BE	Byte Entropy
2	BFA	Byte Frequency Analysis
3	Class_Type	Class Type
4	Direction	Direction
5	Hash_Value	Hash Value
6	Stream_Index	Stream Index
7	Trust_Value	Trust Value

4.2. Training and Testing Data

Training Data is the subset of data for training purposes and Test data is the subset of data for testing the trained data and for this experiment 80:20 is the ratio chosen for training and test data.

4.3. Evaluation

The performance of the Feature Engineering (FE) algorithms is evaluated using various performance metrics such as accuracy, precision, recall, and F1-score. These metrics provide insights into the FE ability to select the significant features from payloads.

- True Positive – Total number of payloads are accurately estimated.
- False Positive – Total number of missing payloads that are wrongly predicted.
- True Negative – Total number of relevant payloads that are accurately estimated.
- False Negative – Total number of payloads that are wrongly predicted.

Precision

It is the ratio of the number of retrieved payloads and the total number of irrelevant and relevant payloads retrieved as follows,

$$Precision = \frac{TP}{TP+FP}$$

Recall

It is the ratio of the number of relevant payloads retrieved from different shells.

$$Recall = \frac{TP}{TP+FN}$$

F-Score

It is measured by computing the weighted harmonic mean of the precision, and recall values.

$$F - measure = 2 . \frac{Precision\ X\ Recall}{Precision+Recall}$$

Accuracy

It calculates the global prediction rate by calculating the ratio of correct results to the total number of characters as follows,

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

Table 2. Performance Analysis of Feature Selection - Manual Packet Capturing Data

Shells	Algorithms	Pre.	Rec.	F-Sc.	Acc.
Web Shell	LDA	86.85	85.05	87.93	87.45
	LR	91.99	90.61	92.27	92.59
	PCA	92.76	91.24	93.98	93.36
	S_FE	**94.01**	**93.11**	**95.04**	**94.61**
Reverse Shell	LDA	81.85	80.05	82.93	82.45
	LR	85.99	84.61	86.27	86.59
	PCA	92.76	91.24	93.98	93.36
	S_FE	**95.01**	**94.11**	**96.04**	**95.61**
Bind Shell	LDA	82.12	84.44	84.23	82.72
	LR	88.44	87.74	89.57	89.04
	PCA	93.75	92.58	94.92	94.35
	S_FE	**96.25**	**95.08**	**97.24**	**96.85**

Table 2. demonstrates the performance analysis of Feature Selection for Manual Packet Capturing Data. From the experimental results, it is noticed that the proposed S_FE algorithm is performs better than other Feature Engineering algorithms.

Table 3. Performance Analysis of Feature Selection – Real-Time Payload Data

Shells	Algorithms	Pre.	Rec.	F-Sc.	Acc.
Web Shell	LDA	87.15	85.35	88.23	87.75
	LR	92.29	90.91	92.57	92.89
	PCA	93.06	91.54	94.28	93.66
	S_FE	**94.31**	**93.41**	**95.34**	**94.91**

Reverse Shell	LDA	82.15	80.35	83.23	82.75
	LR	86.29	84.91	86.57	86.89
	PCA	93.06	91.54	94.28	93.66
	S_FE	**95.31**	**94.41**	**96.34**	**95.91**
Bind Shell	LDA	82.42	84.74	84.53	83.02
	LR	88.74	88.04	89.87	89.34
	PCA	94.05	92.88	95.22	94.65
	S_FE	**96.55**	**95.38**	**97.54**	**97.15**

Table 3. represents the performance analysis of Feature Selection for Real-Time payload data. From the experimental results, it is noticed that the proposed S_FE algorithm performs better than other Feature Engineering algorithms.

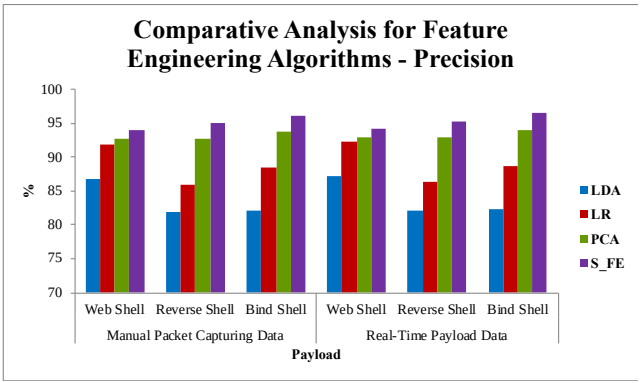

Fig. 2. Comparative Analyses of Feature Engineering Algorithms using Precision

Fig. 2. shows the Comparative Analysis of Feature Engineering Algorithms using Precision. From the result observation, it is found that the proposed S_FE algorithm produced a higher precision rate than other Feature Engineering Algorithms with respect to Manual Packet Capturing Data and Real-Time payload data.

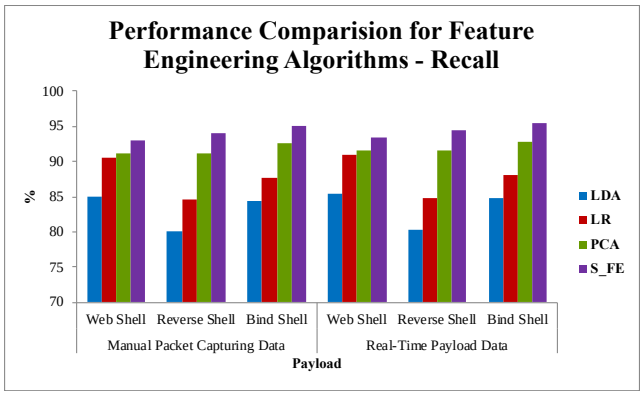

Fig. 3. Performance Comparison of ML Classifiers using Recall

Fig. 3. illustrates the Performance Analysis of Feature Engineering Algorithms using F-Score. From the experimental results, it is found that the proposed S_FE algorithm produced a higher F-Score rate than other Feature Engineering Algorithms for Manual Packet Capturing Data and Real-Time payload data.

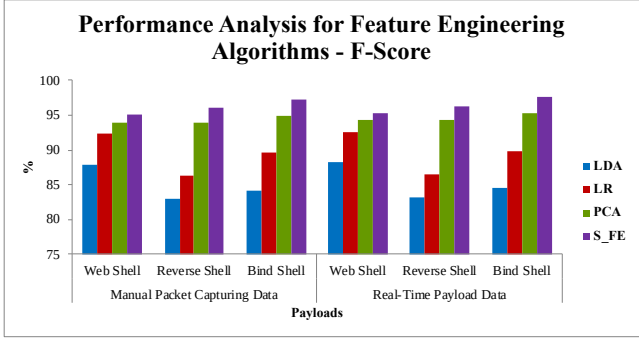

Fig. 4. Performance Analysis of Feature Engineering Algorithms – F-Score

Fig. 4. illustrates the Performance Analysis of Feature Engineering Algorithms using F-Score. From the experimental results, it is found that the proposed S_FE algorithm produced a higher F-Score rate than other Feature Engineering Algorithms with respect to Manual Packet Capturing Data and Real-Time payload data.

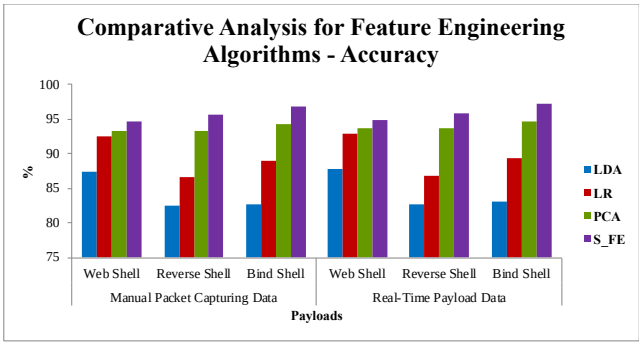

Fig. 4. Comparative Analysis for Feature Engineering Algorithms – Accuracy

Fig. 4. illustrates the Performance Analysis of Feature Engineering Algorithms using Accuracy. From the result outcome, it is proved that the proposed S_FE algorithm produced a higher Accuracy rate than other Feature Engineering Algorithms with respect to Manual Packet Capturing Data and Real-Time payload data.

5. Summary

In this phase of the study, a novel Stochastic-based Feature Engineering (S_FE) algorithm is introduced to enhance anomaly detection in both manual packet capturing and real-time payload datasets, pivotal for bolstering network security measures. Notably, the manual packet capture dataset is characterized by five significant features: Trust Value (Customized value), Byte Frequency Analysis (BFA), Byte Entropy (BE), Payload Length (PL), and Stream Index features. Leveraging the S_FE algorithm, these features are meticulously extracted, providing a comprehensive understanding of the underlying characteristics of network traffic.

For real-time payload extraction, the proposed algorithm

identifies three essential features: Trust Value, direction, and Hash Value. These features are crucial for promptly detecting anomalies and potential security threats in dynamic network environments. Through the innovative application of the S_FE algorithm, real-time data streams are efficiently processed, enabling rapid identification and mitigation of security breaches.

To validate the efficacy of the proposed S_FE algorithm, extensive comparisons are conducted against widely used Feature Engineering (FE) algorithms. Performance evaluations encompass key metrics including accuracy, precision, recall, and F1-score, providing a comprehensive assessment of algorithmic effectiveness. The experimental results reveal that the proposed S_FE algorithm consistently outperforms traditional FE algorithms across various performance metrics, underscoring its superiority in anomaly detection and network security applications.

Overall, the findings of this study underscore the significance of feature engineering, particularly the innovative application of stochastic-based techniques, in enhancing anomaly detection capabilities within network security frameworks. By leveraging advanced algorithms and comprehensive feature extraction methodologies, the proposed S_FE algorithm offers a robust and efficient solution for safeguarding network infrastructures against emerging cyber threats.

6. Conclusion

In conclusion, this research has demonstrated the efficacy of the proposed Stochastic-based Feature Engineering (S_FE) algorithm in enhancing anomaly detection for network security applications. Through meticulous feature extraction and selection, the S_FE algorithm effectively captures the nuanced characteristics of network traffic, enabling accurate identification and mitigation of security threats in both manual packet capturing and real-time payload datasets. The comprehensive evaluation of the S_FE algorithm against traditional Feature Engineering (FE) methods has consistently highlighted its superior performance across key metrics, including accuracy, precision, recall, and F1-score. These findings underscore the significance of innovative feature engineering techniques, particularly stochastic-based approaches, in bolstering network security measures and mitigating the ever-evolving threat landscape.

6.1. Future Enhancements

While the proposed S_FE algorithm represents a significant advancement in anomaly detection for network security, several avenues for future enhancement and research warrant consideration. Firstly, the scalability and adaptability of the algorithm could be further investigated to accommodate increasingly complex network environments and larger datasets. Additionally, the incorporation of machine learning techniques, such as deep learning and reinforcement learning, could enhance the algorithm's ability to detect subtle and emerging security threats. Moreover, the integration of advanced anomaly detection mechanisms, such as anomaly ensembles and hybrid models, may offer synergistic benefits in terms of detection accuracy and robustness. Furthermore, exploring the potential for real-time adaptation and self-learning capabilities within the S_FE algorithm could enable proactive threat detection and response in dynamic network environments. Lastly, collaboration with industry partners and stakeholders to validate the algorithm in real-world settings and address practical implementation challenges would be instrumental in enhancing its effectiveness and adoption. By pursuing these avenues for future enhancement, the proposed S_FE algorithm holds promise for advancing anomaly detection capabilities and fortifying network security frameworks against evolving cyber threats.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Zhang, Y.; Wang, Z. Feature Engineering and Model Optimization Based Classification Method for Network Intrusion Detection. Appl. Sci. 2023, 13, 9363. https://doi.org/10.3390/app13169363

[2] Sapna Sadhwani, Asmi Sriwastawa, Raja Muthalagu et al. Intelligent Feature Engineering Based Intrusion Detection System for IoT Network Security, 20 February 2024, PREPRINT (Version 1) available at Research Square [https://doi.org/10.21203/rs.3.rs-3961151/v1]W.-K. Chen, Linear Networks and Systems. Belmont, CA, USA: Wadsworth, 1993, pp. 123–135.

[3] M. Panda, A. A. A. Mousa and A. E. Hassanien, "Developing an Efficient Feature Engineering and Machine Learning Model for Detecting IoT-Botnet Cyber Attacks," in IEEE Access, vol. 9, pp. 91038-91052, 2021, doi: 10.1109/ACCESS.2021.3092054.

[4] Ruizhe Yao, Ning Wang, Zhihui Liu, Peng Chen, Di Ma, Xianjun Sheng, Intrusion detection system in the Smart Distribution Network: A feature engineering based AE-LightGBM approach, Energy Reports,Volume 7, Supplement 7,2021,Pages 353-361,ISSN 2352-4847,https://doi.org/10.1016/j.egyr.2021.10.024.

[5] Saif, S., Yasmin, N. & Biswas, S. Feature engineering based performance analysis of ML and DL algorithms for Botnet attack detection in IoMT. Int J Syst Assur Eng Manag 14 (Suppl 1), 512–522 (2023). https://doi.org/10.1007/s13198-023-01883-7

[6] Xinwei Zhang, Yaoci Han, Wei Xu, Qili Wang, HOBA: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture,Information Sciences, Volume 557, 2021, Pages 302-316 https://doi.org/10.1016/j.ins.2019.05.023.

[7] et. al., A. N. , . (2021). Feature Engineering based on Hybrid Features for Malware Detection over Android Framework. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(10), 2856–2864.

[8] Liu Z, Wang Y, Feng F, Liu Y, Li Z, Shan Y. A DDoS Detection Method Based on Feature Engineering and Machine Learning in Software-Defined Networks. Sensors. 2023; 23(13):6176. https://doi.org/10.3390/s23136176

[9] A. Ghubaish, Z. Yang, A. Erbad and R. Jain, "LEMDA: A Novel Feature Engineering Method for Intrusion Detection in IoT Systems," in IEEE Internet of Things Journal, vol. 11, no. 8, pp. 13247-13256, 15 April15, 2024, doi: 10.1109/JIOT.2023.3328795.

[10] M. S. Akter, H. Shahriar, J. R. Cardenas, S. Iqbal Ahamed and A. Cuzzocrea, "Feature Engineering-Based Detection of Buffer Overflow Vulnerability in Source Code Using Neural Networks," 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC), Torino, Italy, 2023, pp. 765-776, doi: 10.1109/COMPSAC57700.2023.00106.

[11] Hazman, C., Benkirane, S., Guezzaz, A., Azrour, M., Abdedaime, M. (2023). Intrusion Detection Framework for IoT-Based Smart Environments Security. In: Farhaoui, Y., Rocha, A., Brahmia, Z., Bhushab, B. (eds) Artificial Intelligence and Smart Environment. ICAISE 2022. Lecture Notes in Networks and Systems, vol 635. Springer, Cham. https://doi.org/10.1007/978-3-031-26254-8_79

[12] Abbasi, N., Soltanaghaei, M. & Zamani Boroujeni, F. Anomaly detection in IOT edge computing using deep learning and instance-level horizontal reduction. J Supercomput 80, 8988–9018 (2024). https://doi.org/10.1007/s11227-023-05771-6

[13] Mahmoud Ragab, Maha Farouk S. Sabir, Outlier detection with optimal hybrid deep learning enabled intrusion detection system for ubiquitous and smart environment, Sustainable Energy Technologies and Assessments, Volume 52, Part D, 2022, 102311, ISSN 2213-1388, https://doi.org/10.1016/j.seta.2022.102311.

[14] P. K. Reddy Shabad, A. Alrashide and O. Mohammed, "Anomaly Detection in Smart Grids using Machine Learning," IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society, Toronto, ON, Canada, 2021, pp. 1-8, doi: 10.1109/IECON48115.2021.9589851.

[15] M. Ravinder and V. Kulkarni, "A Review on Cyber Security and Anomaly Detection Perspectives of Smart Grid," 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2023, pp. 692-697, doi: 10.1109/ICSSIT55814.2023.10060871.

[16] Kumar, V., Patra, S.K. (2021). Feature Engineering for Machine Learning and Deep Learning Assisted Wireless Communication. In: Oliva, D., Houssein, E.H., Hinojosa, S. (eds) Metaheuristics in Machine Learning: Theory and Applications. Studies in Computational Intelligence, vol 967. Springer, Cham. https://doi.org/10.1007/978-3-030-70542-8_4

[17] Dongqi Han, Zhiliang Wang, Wenqi Chen, Ying Zhong, Su Wang, Han Zhang, Jiahai Yang, Xingang Shi, and Xia Yin. 2021. DeepAID: Interpreting and Improving Deep Learning-based Anomaly Detection in Security Applications. In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21). Association for Computing Machinery, New York, NY, USA, 3197–3217. https://doi.org/10.1145/3460120.3484589

[18] D. Upadhyay, J. Manero, M. Zaman and S. Sampalli, "Gradient Boosting Feature Selection With Machine Learning Classifiers for Intrusion Detection on Power Grids," in IEEE Transactions on Network and Service Management, vol. 18, no. 1, pp. 1104-1116, March 2021, doi: 10.1109/TNSM.2020.3032618.

[19] Shahhosseini, M., Mashayekhi, H. & Rezvani, M. A Deep Learning Approach for Botnet Detection Using Raw Network Traffic Data. J Netw Syst Manage 30, 44 (2022). https://doi.org/10.1007/s10922-022-09655-7

[20] B. Liu, Y. Zhao, Y. Kang, Y. Cao, P. Bai and Z. Xu, "A Feature Engineering-based Method for PCB Solder Paste Position Offset Prediction," 2023 6th International Symposium on Autonomous Systems (ISAS), Nanjing, China, 2023, pp. 1-6, doi: 10.1109/ISAS59543.2023.10164303.

[21] Berghout T, Benbouzid M, Amirat Y. Towards Resilient and Secure Smart Grids against PMU Adversarial Attacks: A Deep Learning-Based Robust Data Engineering Approach. Electronics. 2023; 12(12):2554. https://doi.org/10.3390/electronics12122554

[22] Liu H, Lang B. Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey.

Applied Sciences. 2019; 9(20):4396. https://doi.org/10.3390/app9204396

[23] V. T. Pham, T. V. Huu, M. T. Nguyen and H. -C. Le, "Advanced Feature Processing for IoT-Based Intrusion Detection System," 2023 RIVF International Conference on Computing and Communication Technologies (RIVF), Hanoi, Vietnam, 2023, pp. 37-42, doi: 10.1109/RIVF60135.2023.10471837.

[24] Lean Yu, Xiaoming Zhang, Hang Yin, An extreme learning machine based virtual sample generation method with feature engineering for credit risk assessment with data scarcity, Expert Systems with Applications, Volume 202, 2022, 117363, ISSN 0957-4174,https://doi.org/10.1016/j.eswa.2022.117363.

[25] Rajput, V., Mulay, P. and Mahajan, C.M. (2024), "Bio-inspired algorithms for feature engineering: analysis, applications and future research directions", Information Discovery and Delivery, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1108/IDD-11-2022-0118

[26] Ti, YW., Hsin, YY., Dai, TS. et al. Feature generation and contribution comparison for electronic fraud detection. Sci Rep 12, 18042 (2022). https://doi.org/10.1038/s41598-022-22130-2

[27] Althar, R.R., Samanta, D. The realist approach for evaluation of computational intelligence in software engineering. Innovations Syst Softw Eng 17, 17–27 (2021). https://doi.org/10.1007/s11334-020-00383-2

[28] C. Feng, T. Li and D. Chana, "Multi-level Anomaly Detection in Industrial Control Systems via Package Signatures and LSTM Networks," 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Denver, CO, USA, 2017, pp. 261-272, doi: 10.1109/DSN.2017.34.

[29] Zhijian Qu, Hanxin Liu, Zixiao Wang, Juan Xu, Pei Zhang, Han Zeng,

[30] A combined genetic optimization with AdaBoost ensemble model for anomaly detection in buildings electricity consumption, Energy and Buildings, Volume 248, 2021, 111193, ISSN 0378-7788, https://doi.org/10.1016/j.enbuild.2021.111193.

[31] P. Kopyt *et al.,* "Electric properties of graphene-based conductive layers from DC up to terahertz range," *IEEE THz Sci. Technol.*, to be published. DOI: 10.1109/TTHZ.2016.2544142.