STM JOURNALS

International Journal of
Information Security Engineering

ISSN: 3049-1150
Volume 4, Issue 1, 2026
January–June
DOI (Journal): 10.37591/IJISE

https://journals.stmjournals.com/ijise

Research Research

Securing Web Applications: A Machine Learning Approach for SQL Injection Threats

Nilima D. Bobade[1,*], Swati S. Shereker[2]

Abstract

The rapid evolution and widespread adoption of the internet have significantly transformed the world, leading to an increased number of cyberattacks. Cybersecurity has become one of the most critical challenges for society, incurring substantial financial losses annually. This research focuses on SQL injection attacks, the specific threat to web applications, aiming to detect malicious queries designed to exploit vulnerabilities and access sensitive data. In recent years, the frequency of SQLi attacks has surged, posing severe risks to web application security. Attackers use SQLi to execute arbitrary SQL code, potentially gaining unauthorized access to databases, exfiltrating data, and compromising the integrity of web services. Despite various efforts to mitigate SQLi attacks, a comprehensive and effective detection mechanism remains elusive. This paper highlights the machine learning approach to identify and prevent SQLi attacks, providing a comparative analysis of various classifiers and their performance. The study underscores the potential of machine learning in enhancing web application security, offering adaptive and dynamic solutions to combat the evolving threat of SQLi attacks.

Keywords: Cyber security, machine learning, SQL injection, threat detection, web security

INTRODUCTION

The field of cybersecurity has experienced tremendous growth, addressing a variety of threats, including SQL injection (SQLi) attacks, which are among the most significant dangers in the realm of web applications. SQLi attacks represent a persistent threat to database-driven applications that exploit vulnerabilities to malicious SQL code. These attacks are particularly challenging to prevent because they can be easily executed by inserting malicious queries into the input fields, causing immediate and severe damage.

SQL injection attacks allow cybercriminals to compromise web applications, leading to unauthorized access to sensitive data, identity theft, and other malicious activities. Attackers manipulate SQL queries to bypass security measures, access backend databases, and perform actions, such as data theft, deletion, or modification.

Specially crafted SQL queries are used by attackers to exploit vulnerabilities, gain control over the database, and potentially the entire application. Cybercriminals target web applications to create extensive networks of compromised systems, which can be managed and exploited remotely.

Traditional SQLi detection techniques, such as signature-based and rule-based methods, may not be sufficient for detecting modern and sophisticated SQLi threats. These conventional methods often struggle to keep pace with evolving tactics employed by attackers. Therefore, there is a critical need for advanced detection mechanisms that can adapt to new attack patterns [1].

***Author for Correspondence**
Nilima D. Bobade
E-mail: ndbobade@mitra.ac.in

[1,2]Assistant Professor, Department, Computer Science, Sant Gadge Baba Amravati University, Maharashtra, India

Received Date: May 12, 2025
Accepted Date: July 08, 2025
Published Date: March 24, 2026

Citation: Nilima D. Bobade, Swati S. Shereker. Securing Web Applications: A Machine Learning Approach for SQL Injection Threats. International Journal of Information Security Engineering. 2026; 4(1): 16–22p.

This study investigates and explores a machine learning based approach for recognizing and mitigating SQL injection threats, focusing on developing a robust detection system capable of identifying various attack vectors with high accuracy. By leveraging natural language processing and advanced classification algorithms, this study aims to provide an adaptive and dynamic method to defend against SQL injection threats in web applications.

MACHINE LEARNING FRAMEWORK FOR SQL INJECTION DETECTION

In many studies, machine learning approaches to detect SQL injection threats have been extensively discussed. Various researchers have introduced diverse approaches to enhance the accuracy and performance of SQL injection-threat detection. The proposed SQL injection detection technique uses a supervised machine learning algorithm and is intended to detect SQL injections, as shown in Figure 1. The first SQL injection dataset SQLiV3 from Kaggle contains a total of 30864 query. The dataset contains 11347 malicious and 19517 benign query. The SQLiV3 dataset contains the categorization of different types of SQL injection attack queries, as shown in Table 1.

Table 1. Categorization of SQLi.

Benign	19517
Other	10326
Tautology	637
Union-based	366
Piggybacked	18
Name: Attack_type, dtype: int64	

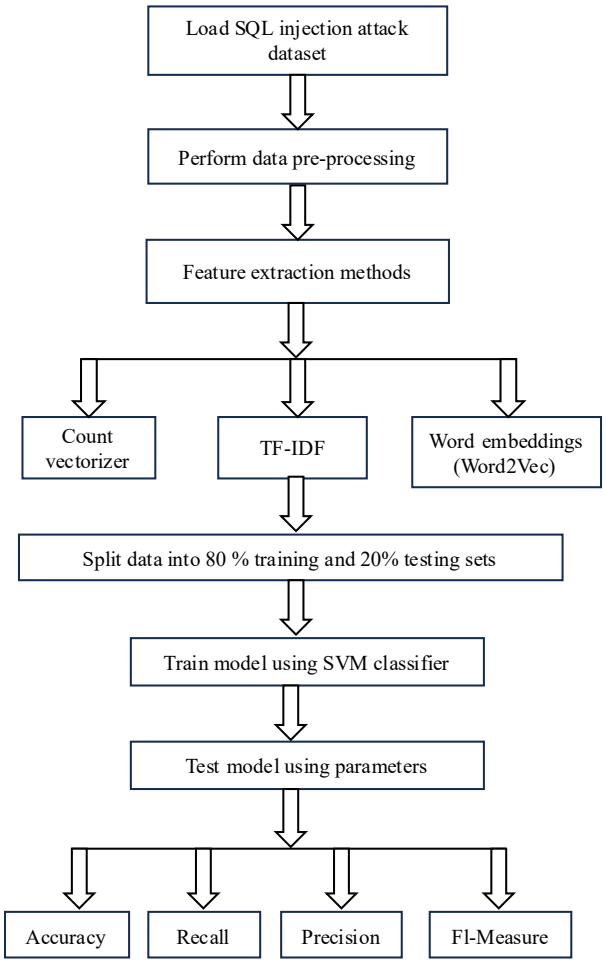

Figure 1. Proposed approach for SQL injection attack detection.

Data preprocessing was performed on the dataset to clean the data. Features were extracted from the dataset using three different methods: (1) CountVectorizer, (2) Term Frequency – Inverse Document Frequency (TF-IDF), and (3) Word2Vec from Word Embeddings. The data from the dataset was split into an 80% training set and a 20% testing set. Support vector machines (SVM) provides the highest accuracy compared with other models such as logistic regression, random forest, Naïve Bayes, and AdaBoost classifiers. The model is then trained using the SVM classifier consistently initialized with hyperparameters, such as a linear kernel, enabled probability estimates, and a fixed random state for reproducibility. The proposed model was tested and evaluated based on four different evaluation parameters for three different methods of feature extraction and compared.

EVALUATION TERM DESCRIPTION

Table 2 summarizes the key evaluation metrics for the SQL injection detection models [8].

THE RESULTS AND DISCUSSION

The proposed technique is evaluated based on four parameters and shown in Table 3.

THE RESULTS AND DISCUSSION

The proposed technique was evaluated based on the accuracy, precision, recall, and F1-measure parameters, as listed in Table 4.

ANALYSIS AND RESULTS

In this work, Hasan et al. [2] evaluated a heuristic machine learning approach that combines static and dynamic analyses. The system was developed using MATLAB and utilized a well-established dataset and 23 machine learning classifiers for training and testing. Based on performance metrics, the top five classifiers were chosen, and a GUI application was developed to facilitate the easy testing of new SQL statements. Uwagbole et al. [3] worked on machine learning algorithms, specifically the support vector machine algorithm, for the detection and prevention of SQL injection threats. The SVM was trained with a dataset containing both benign and malicious SQL queries. This training enabled the SVM to generate a framework capable of distinguishing between benign and destructive queries. The system architecture includes a real-time monitoring component that applies an SVM model to incoming SQL queries to identify and block potential attacks. The author Valli Kumari et al. [4] used the “SQL1” and “SQL2” datasets from Kaggle, which contain labeled SQL queries. Linear discriminant analysis (LDA), logistic regression (LR), and K-nearest neighbor (KNN) supervised learning algorithms. Implemented a stacking technique where predictions from base models (LDA, LR, KNN) are used as inputs to a meta model (LR) and highlights the potential of machine learning algorithms combined with stacking to get better security. Gogoi et al. [5] developed an Apache web server module that intercepts HTTP/HTTPS requests and inspects them for SQLIA payload. The system parses HTTP requests into tokens and converts them into numerical data by using word embedding (word2vec). The Synthetic Minority Oversampling Technique was used to handle the imbalance of classes in the dataset (SMOTE). Trained classifiers use SVM with both linear and nonlinear models. The hyperparameters are optimized using Grid Search. Precision, F1-Score, and recall parameters were used for the evaluation of the system and achieved a high detection rate for all parameters to 99.9%.10-Fold cross validation was used to prevent overfitting. Roy et al. [6] tested machine learning classifiers to detect the payloads of SQL injection threats. The authors used a dataset from Kaggle for training and testing, and a variety of classifiers, including Naïve Bayes. Logistic regression, XGBoost, Naïve Bayes, AdaBoost, and random forest. Alkhathami et al. [7] suggested a model trained on a dataset of SQL queries, both normal and injection attacks, collected from publicly available repositories. Data preprocessing involves cleaning the dataset and converting text values into numerical features using the CountVectorizer class. Four algorithms were used for the training: KNN, MNB, DT, and SVM. The models were assessed using metrics such as accuracy, F-value, confusion matrix, and AUC-ROC curve. SVM achieved a maximum accuracy of 99.42%, followed by DT at 99.4%, MNB at 97.09%, and KNN at 92.45%. Author Ahmed Abadulla Ashlam et al. [8] proposed a methodology involving an arithmetical, systematic, and experimental approach to studying the harmful effects of SQLi attacks. It includes the collection of a

dataset, data preprocessing steps such as tokenization and stop word removal, feature extraction, and the development of a classification model using techniques such as CountVectorizer and support vector machine.

Muliono et al. [9] employed SVM to group and predict SQL injection attacks. Focuses on SQL query conditions and program fragments vulnerable to attacks. Zhumabekova et al. [10] worked on algorithms using machine learning, including Naïve Bayes, decision tree, support vector machine, XGBoost, random forest, and CastBoost. The dataset used for the experiments was sourced from Kaggle, containing 68,553 items split into 70% training set and 30% testing set. The first step was data preprocessing, involving collecting, cleaning, normalizing, scaling, and vectorizing the data with the help of the TF-IDF metric. The processed data was used to train and test a variety of machine learning models to evaluate the performance of SQL injections. The results were assessed using parameters such as accuracy, F1-Score, precision, and recall. Zhang et al. [11]. The proposed method, called the Feature Ratio Method, focuses on six main features. Regular expressions and fuzzy matching are used to extract these features and improve detection accuracy. Table 5 compares multiple authors' SQL injection detection models across key performance metrics.

GRAPHICAL REPRESENTATION OF VARIOUS PARAMETERS

Figures 2–6 illustrate a comparative analysis of SQL injection detection models reported in multiple studies. Each graph highlights variations in accuracy, precision, recall, and F-measure, enabling a clearer understanding of the model behavior and performance trends across different classification techniques.

Table 2. Description of evaluation terms.

S.N.	Evaluation term	Description
1	Accuracy	Overall performance of the model
2	Precision	Percentage of classified SQL injection that are actually SQL injection
3	Recall	Effectiveness of the model in detecting SQL injection
4	F-Measure	Blend of precision and recall

Table 3. Evaluation of proposed techniques.

S.N.	Feature Extraction Method	Evaluation parameters			
		Accuracy	*Precision*	*Recall*	*F-Measure*
1	Count Vectorizer	0.9917	0.9898	0.9876	0.9887
2	TF-IDF	0.9765	0.9907	0.9449	0.9673
3	Word2Vec	0.9872	0.9897	0.9753	0.9824

Table 4. Evaluation of proposed techniques.

S.N.	Feature extraction method	Evaluation parameters			
		Accuracy	*Precision*	*Recall*	*F-Measure*
1	Count Vectorizer	0.9917	0.9898	0.9876	0.9887
2	TF-IDF	0.9765	0.9907	0.9449	0.9673
3	Word2Vec	0.9872	0.9897	0.9753	0.9824

Table 5. Comparison of SQL injection detection parameters.

S.N.	Name of author	Classifier/Algorithm	Parameters			
			Accuracy	*Precision*	*Recall*	*F-measure*
1	M. Hasan et al., 2019 [2]	Ensemble Boosted Trees, Ensemble Bagged Trees, Linear Discriminant, Cubic SVM, Fine Gaussian SVM	0.938	-	-	-
2	Uwagbole SO et al., 2017 [3]	SVM	0.986	0.974	0.997	0.985
3	Valli Kumari	LR,KNN,LDA and Meta model	0.977	0.929	1.0	-

	et al., 2024 [4]					
4	B. Gogoi et al.,2021 [5]	Linear SVM, Non linear SVM	-	0.999	0.999	0.999
5	P. Roy et al.,2022 [6]	Logistic Regression,AdaBoost,Random Forest,XGBoost,Naïve Bayes	0.9833	0.941	-	0.970
6	J. M. Alkhathami et al.,2022 [7]	KNN,MNB,Decision Tree,SVM	0.99	-	-	0.99
7	Ahmed Abadulla Ashlam et al.,2022 [8]	SVM, Decision Tree, K-Nearest Neighbors, AdaBoost, Random Forest, SVM with Performance analysis and Iterative optimization of the SQLI Detection Model i.e PALOSDM	0.99	1	0.98	0.99
8	Yohan Muliono et al.,2022 [9]	SVM	0.94	-	-	-
9	A. Zhumabekova et al.,2023 [10]	Decision Tree, Random Forest, XGBoost, and CatBoost	0.99	0.99	0.98	0.99
10	Shengyuan Zhang et al.2023 [11]	LR,KNN,RF,SVM,NB,CNN	0.989	0.979	-	0.989

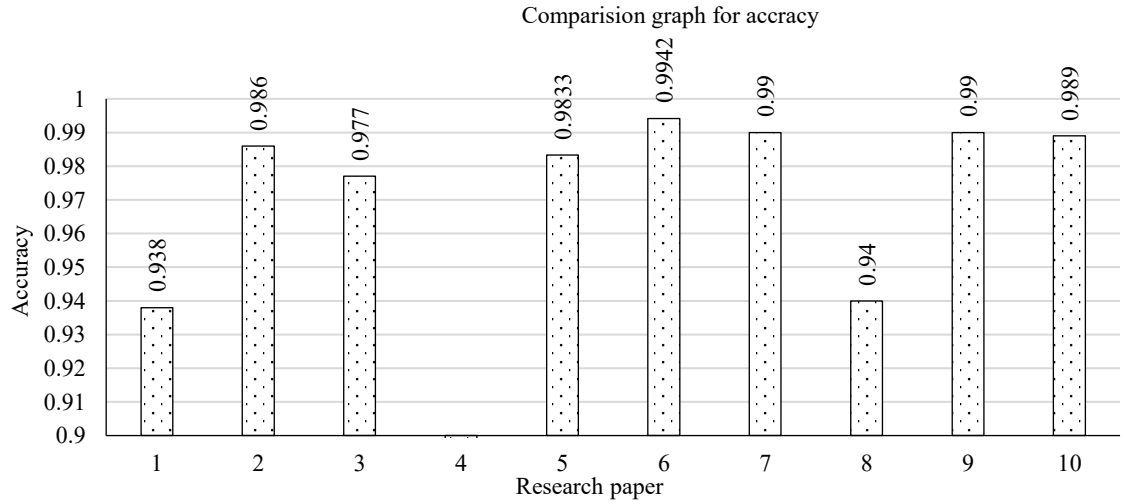

Figure 2. Comparison graph illustrating accuracy values across ten referenced research papers.

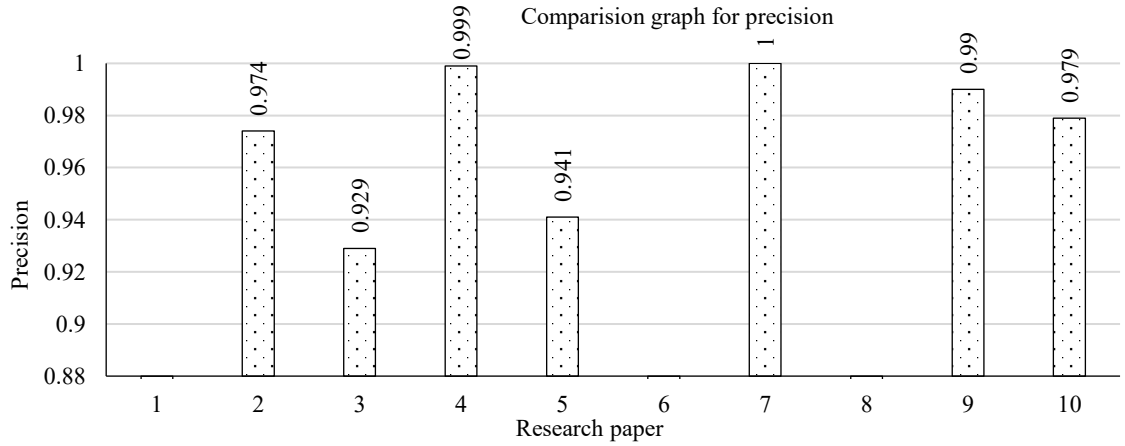

Figure 3. Comparison graph showing precision measurements for SQL injection detection models reported in the selected studies.

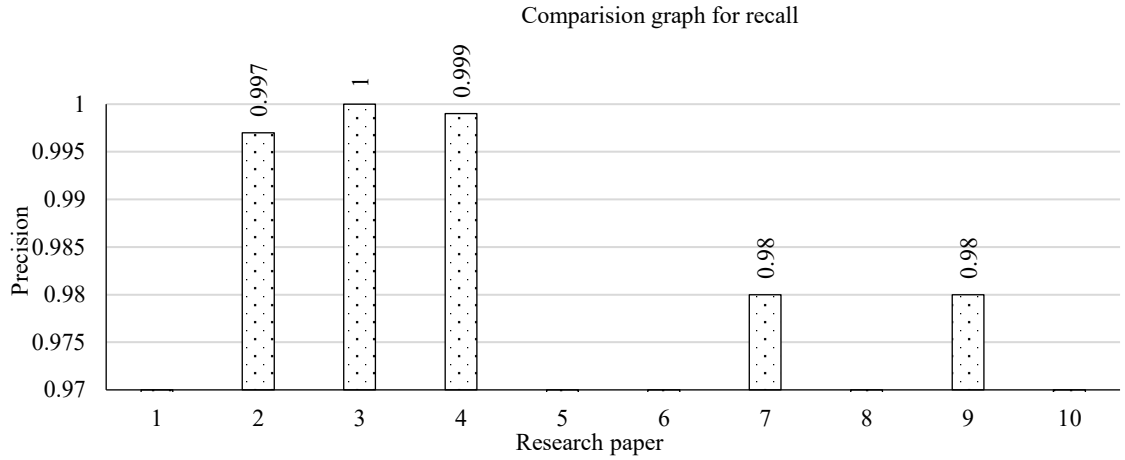

Figure 4. Comparison graph depicting recall performance for various machine learning approaches identified in the literature.

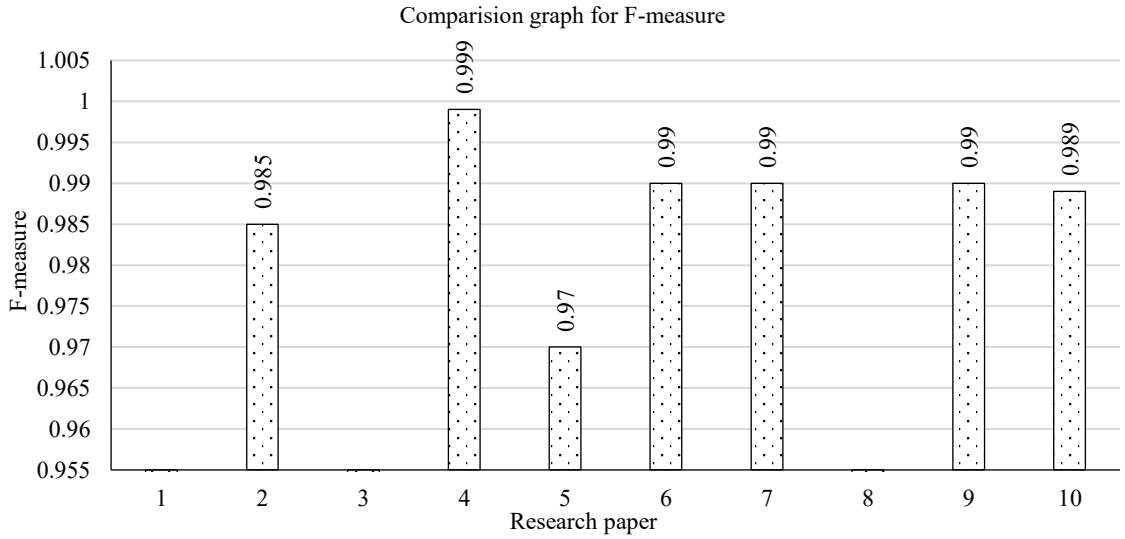

Figure 5. Comparison graph presenting F-measure values derived from different classification algorithms across research articles.

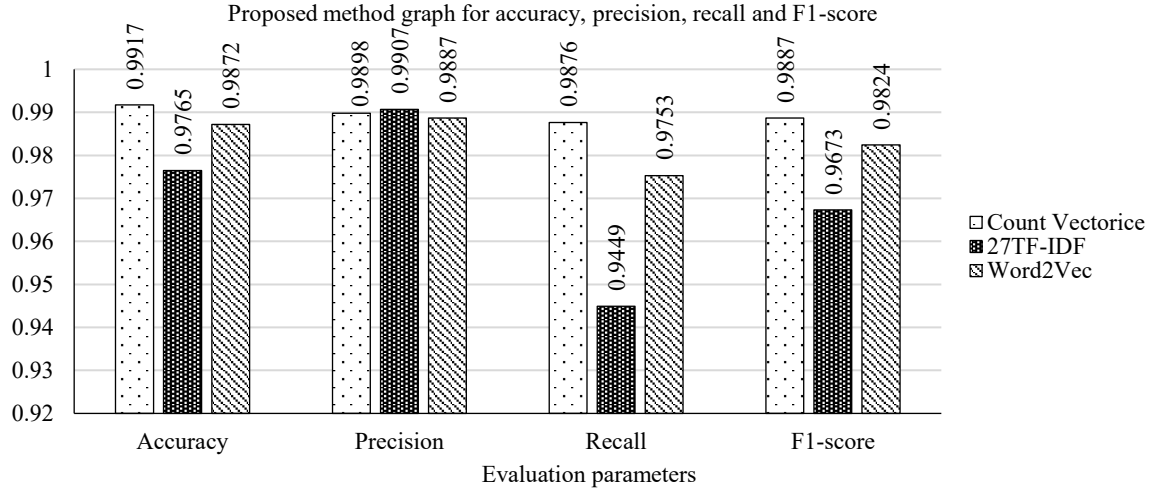

Figure 6. Proposed method evaluation graph comparing accuracy, precision, recall, and F1-score using Count Vectorizer, TF-IDF, and Word2Vec techniques.

CONCLUSION

This paper explores SQL injection and presents a detection technique using the count vectorizer method for feature extraction with an SVM classifier, achieving an accuracy of 0.9917, which is better than the TF-IDF method of feature extraction with SVM and the Word2vec IDF method of feature extraction with SVM, and performance analysis of various parameters of the existing methodology. Additionally, this study provides a detailed analysis of SQL injection parameters, offering insights for researchers. The results were compared with those of a variety of methods based on performance metrics.

REFERENCES

1. Brindavathi B, Karrothu A, Anilkumar C. An analysis of AI-based SQL injection (SQLi) attack detection. 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India. 2023. p. 31–35. doi:10.1109/ICAISS58487.2023.10250505.
2. Hasan M, Balbahaith Z, Tarique M. Detection of SQL injection attacks: a machine learning approach. 2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA), Ras Al Khaimah, United Arab Emirates. 2019. p. 1–6. doi:10.1109/ICECTA48151.2019.8959617.
3. Uwagbole SO, Buchanan WJ, Fan L. Applied machine learning predictive analytics to SQL injection attack detection and prevention. 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM), Lisbon, Portugal. 2017. p. 1087–1090. doi:10.23919/INM.2017.7987433.
4. Valli Kumari V, Prasanna Kumar Y. SQL injection detection using recurrent neural networks (RNN). In: Mesnager S, Stănică P, Debnath SK, editors. Security and Privacy. ICSP 2024. Communications in Computer and Information Science. Cham: Springer; 2025. p. 154–167. doi:10.1007/978-3-031-90587-2_11.
5. Gogoi B, Ahmed T, Dutta A. Defending against SQL injection attacks in web applications using machine learning and natural language processing. 2021 IEEE 18th India Council International Conference (INDICON), Guwahati, India. 2021. p. 1–6. doi:10.1109/INDICON52576.2021.9691740.
6. Roy P, Kumar R, Rani P. SQL injection attack detection by machine learning classifier. 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India. 2022. p. 394–400. doi:10.1109/ICAAIC53929.2022.9792964.
7. Alkhathami JM, Alzahrani SM. Detection of SQL injection attacks using machine learning in cloud computing platform. J Theor Appl Inf Technol. 2022;100(15):1–4.
8. Ashlam AA, Badii A, Stahl F. A novel approach exploiting machine learning to detect SQLi attacks. 2022 5th International Conference on Advanced Systems and Emergent Technologies (IC_ASET), Hammamet, Tunisia. 2022. p. 513–517. doi:10.1109/IC_ASET53395.2022.9765948.
9. Muliono Y, Darus MY, Pardomuan CR, Ariffin MAM, Kurniawan A. Predicting confidentiality, integrity, and availability from SQL injection payload. 2022 International Conference on Information Management and Technology (ICIMTech), Semarang, Indonesia. 2022. p. 600–605. doi:10.1109/ICIMTech55957.2022.9915227.
10. Zhumabekova A, Matson ET, Karyukin V, Zhumabekova K, Zhuandykov B, Ussatova O, Telbayeva T. Determining web application vulnerabilities using machine learning methods. 2023 19th International Asian School-Seminar on Optimization Problems of Complex Systems (OPCS), Novosibirsk, Moscow, Russian Federation. 2023. p. 136–139. doi:10.1109/OPCS59592.2023.10275756.
11. Zhang S, Li Y, Jiang Q. Feature ratio method: a payload feature extraction and detection approach for SQL injection attacks. 2023 3rd Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS), Shenyang, China. 2023. p. 172–175. doi:10.1109/ACCTCS58815.2023.00019.

