Conference Name: EnSci Singapore 2025 – International Conference on Engineering & Sciences, 04-05 March
Conference Dates: 04-Mar- 2025 to 05-Mar- 2025
Conference Venue: The National University of Singapore Society (NUSS), The Kent Ridge Guild House, 9 Kent Ridge Drive, Singapore
Appears in: MATTER: International Journal of Science and Technology (ISSN 2454-5880)
Publication year: 2025

Park and Park, 2025

Volume 2025, pp.10-24

DOI- https://doi.org/10.20319/stra.2025.1024

This paper can be cited as Park, J., Park, J.(2025). Web Attack Detection Comparative Analysis of LSTM and DNN-based Defense Models with Correspondence Analysis. 2025 SoRes Singapore – International Conference on Interdisciplinary Research in Social Sciences, 03-04 March, Proceedings of Scientific and Technical Research Association (STRA), 10-24

WEB ATTACK DETECTION COMPARATIVE ANALYSIS OF LSTM AND DNN-BASED DEFENSE MODELS WITH CORRESPONDENCE ANALYSIS

Jaehyung Park
Division of Smart Cities, Korea University, Sejong, South Korea
jaehyung101@korea.ac.kr

Junghee Park
Cognitive Engineering Lab, Yonsei University, Seoul, South Korea
shoutjoy@hanmail.net

Abstract

Recently since there exists more companies using web, web attacks to hijacking or manipulate the privacy information have increased. Among web vulnerability OWASP has introduced, SQL injection, XSS, File Inclusion have constantly occurred through more than a decade. It concludes that web servers have trouble with blocking old-fashioned web vulnerabilities. This paper is going to skim through web attack defending methods and compares existing web attack detection machine learning models and new ensemble model DPL with ANOVA, chi-square analysis, correspondence analysis to find out relativity between model and web attack. As result of correspondence analysis, brand new model DPL excels existing models but even DPL model have low relativity on XSS. It is expected that post research must introduce more XSS relevant model.

Keywords:
XSS, Deep learning, SQL injection, File Inclusion, Web Attack

1. Introduction

In recent years, more companies use web to decrease the cost for information, product, service marketing so that security vulnerabilities increase (A. K. Baranwal, 2012). This is not

only because of company itself, but also increasing novel attacks finding new vulnerability about various web services (Hong, Sunghyuck, 2013).

The Open Web Application Security Project (OWASP) has introduced 10 most frequently used security vulnerabilities through 2013 OWASP Top10. Injection, and Identification and Authentication Failures are the examples of 10 most frequently used from 2013 OWASP Top10 through 2021 OWASP Top10. So that research about classic vulnerabilities have been appeared until present days and these vulnerabilities are critical to the system have appeared (Jiho Bang & Rhan Ha, 2013; Joonseon Ahn et al., 2015).

However, there are less research about detection models for these web attacks. Each prior studies suggests a model for web attacks with improved evaluation metric, so comparing each model to find out which model detects which specific attack better need to be proved. The four important attacks, SQL injection (CWE-89, SQLi), Cross-Site Scripting (CWE-79, XSS), Password Attack (CWE-1216, Brute Force), File Inclusion (CWE-98) detecting models have been chosen for this research. models used for research are P-LSTM, which trained with HTTP logs payload, H-LSTM, which trained with HTTP request logs and modified from P-LSTM, DNN model trained with tcp packet, and DPL, which is suggesting from this research and is ensemble model.

2. Prior Studies

2. 1. Web attack vulnerabilities classification

OWASP TOP10, announced from 2010, through 2013, 2017, to 2021, has been announced about web attack vulnerabilities as displayed on **Table 1.1**. So we classify the attacks into groups that are lasting for more than a decade and still be a serious vulnerability and others. SQL injection is always on the OWASP TOP10 and CWE/SANS Top 25 which steal important data from database (Joonseon Ahn et al., 2015), and manipulates database to work in unintended manner (Begum et al., 2016). SQL injection is easy to use but have low chance to success, however if it once success, then there will be serious problem with data leak (Sinha et al. 2018; Okesola et al., 2023). SQL injection could bring results of database searching without appropriate account having permission (A. K. Baranwal, 2012; Jeom goo Kim et al., 2012) and privilege hijacking related to data exposure (Song-ha-Min et al., 2022).

Cross-Site Scripting, also known as XSS, is an attack that injects malicious scripts without filtering to target computer and make them target to download and execute the attacking

script (A. K. Baranwal, 2012; Seung-pyo Huh et al., 2009; Rathore et al. 2017). Once the malicious script is injected, phishing attack, injecting malicious contents, stealing user private information can be occur (Hong, Sunghyuck, 2013; Choi, Eun-Jung et al., 2015).

Password Attack, also known as Brute Force, is an attack that includes checking every password available on the system to find out the right password for login. If the time is unlimited to try password attack, this attack will always eventually success. Moreover, known password or predicted password by user's name, birthday, keyboard layout, and other information could reduce time for attack until success (Sinha et al., 2018; S. Sarkar et al., 2022).

File Inclusion attack, classified as Insecure Direct Object References in OWASP TOP10, is attack that allows attack target to access insecure unallowed file through dynamic file inclusion mechanism so that information searching, or remote command execution could happen (Begum et al., 2016; Hassan et al., 2018; M. S. Tajbakhsh and J. Bagherzadeh). If file for attack is not existing on the server, execution through direct uploading the file to steal the server's information (Huang et al., 2019).

Table 1.1 *Annual changes of OWASP TOP10*

	2010	**2013**	**2017**	**2021**
A1	Injection	Injection	Injection	BASM
A2	XSS	BASM	BASM	Cryptographic Failures
A3	BASM	XSS	Sensitive Data Exposure	Injection
A4	Insecure Direct Object References	Insecure Direct Object References	XML External Entities	Insecure Design
A5	Cross-Site Request Forgery (CSRF)	Security Misconfiguration	Broken Access Control	Security Misconfiguration

BASM: Broken Authentication and Session Management

The commonalities among SQL injection, XSS, File Inclusion is that every attack are classified as injection attack and aims to steal or manipulate the information (Zhang et. al., 2016; Akbar, Et al., 2018). Password Attack and XSS are used to steal the information of user. SQL injection and Password Attack can steal information from database or privilege of authentication at the time of attack. However, XSS and File Inclusion cannot ensure the information leak or manipulation could occur until target user access at the malicious file or script.

2.2 Detection and Defense Model

From the prior research, SQL injection could be detected many ways such as open-source project ELK Stack (Song-ha-Min et al., 2022) and using machine learning including LSTM and Graph Convolutional Network (Yeonsu Kim et al., 2020; Valeur, F. & D., Vigna, G., 2005). There are several ways to defend XSS such as blocking HTML tag input by using HTML encoding, not containing unnecessary information at cookie, not allowing javascript to work at web browser, using proxy, using regular expressions to filter XSS attacks, and using ML to defense the attack (Baranwal, 2012; Eun-jung et al., 2015; Hong, 2013; Huh et al., 2009; Kim et al., 2020; Liang et al., 2017; Rathore et al., 2017). SQL injection and XSS could be detected by using snort rule(Alnabulsi et al., 2014; Mahoney & Chan, 2002; Roesch, 1999; Syaifuddin et al., 2018). File Inclusion could be defended by making static analyzer, using SAISAN, using Uchecker, and using AntiLFIer to check credibility of file (Ahn et al., 2015; Hassan et al., 2018; Huang et al., 2019; Tajbakhsh & Bagherzadeh, 2015). Several methods to defend Password Attack are creating a blacklist to block automated attack and to create a strong password enough to take very long time for attack (Sarkar & Nandan, 2022; 염태균, 2016).

2.3 Classification performance evaluation index

To evaluate the models' prediction performance, using confusion matrix is one way to figure out the performance and characteristics of each classification model. Accuracy, Precision, Recall, and F1-score is used for evaluation (Roh et al., 2022; Yacouby & Axman, 2020).

Table 2.1 *Confusion Matrix*

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

Accuracy is ratio of the number of accurately predicted attacks over the number of total attacks including true and false attacks. Recall is ratio of the number of accurately predicted attacks over the number of true attacks. Precision is ratio of true attacks over predicted attacks. F1-score is harmonic mean of recall and precision. It is known as evaluating unbalanced data's performance. Also, f1-score is used for calculating the mean from data containing various category. Macro

average and micro average is the method of it. Macro average is good at calculating balanced mean and distinguishing and evaluating categories (Pak et al., 2020; Roh et al., 2022). Micro average is good at distinguishing individual figures and evaluating it (Schütze et al., 2008; Zhang et al., 2015).

$$\boldsymbol{accuracy} = \frac{\boldsymbol{TP} + \boldsymbol{TN}}{\boldsymbol{TP} + \ \boldsymbol{TN} + \boldsymbol{FN} + \boldsymbol{TN}} \times \mathbf{100}$$

$$\boldsymbol{precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP} + \boldsymbol{FP}} \times \mathbf{100}$$

$$\boldsymbol{recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP} + \boldsymbol{FN}} \times \mathbf{100}$$

$$\boldsymbol{f1 - score} = \mathbf{2} \times \frac{\boldsymbol{precision} + \boldsymbol{recall}}{\boldsymbol{precision} \times \boldsymbol{recall}} \times \mathbf{100}$$

2.4. Correspondence Analysis

Correspondence Analysis (CA) is exploratory data analysis that finds out the relationship between the points on low dimensional space having column and row from the contingency table (Greenacre, 2017). Geometric distance between each points is meaningless. Two points heading same direction means that those points have corresponding relationship. It is preferred to draw two-dimensional coordinates to describe if explanatory power is greater than 70%.

III. Research Design and Procedures

3.1. Research Execution and Data Collection

Data collection was performed using website model called DVWA, collecting web attack logs by Wireshark.

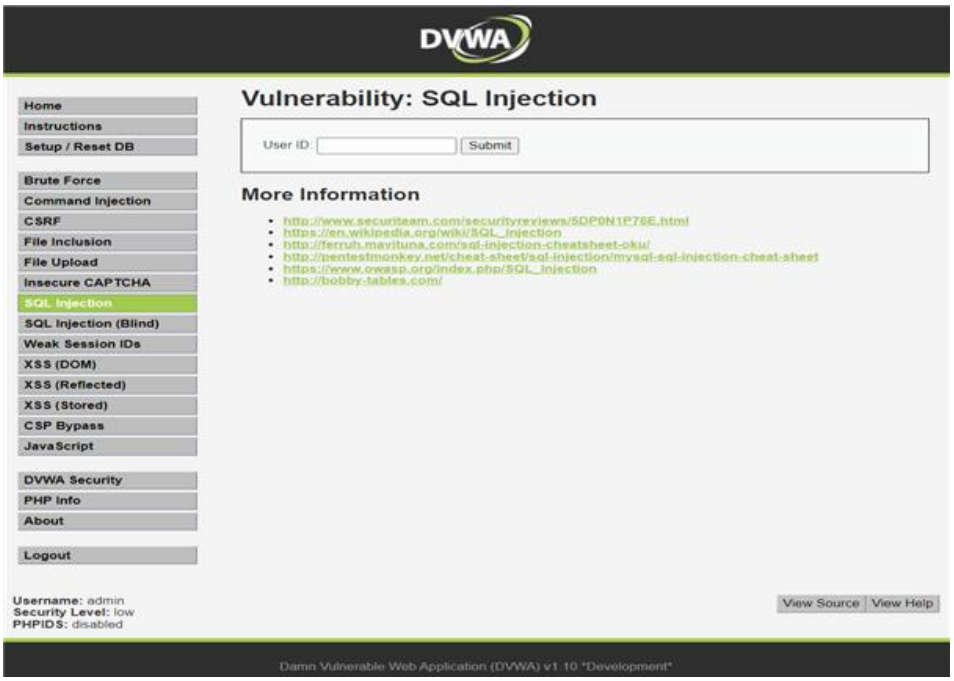

Figure 1: *DVWA SQL injection Page*

3.2. Research models for defense

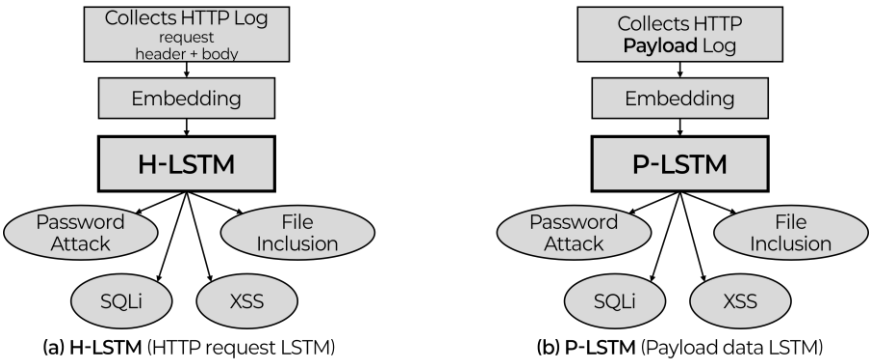

Figure 2: Defense Model (H-LSTM, P-LSTM)

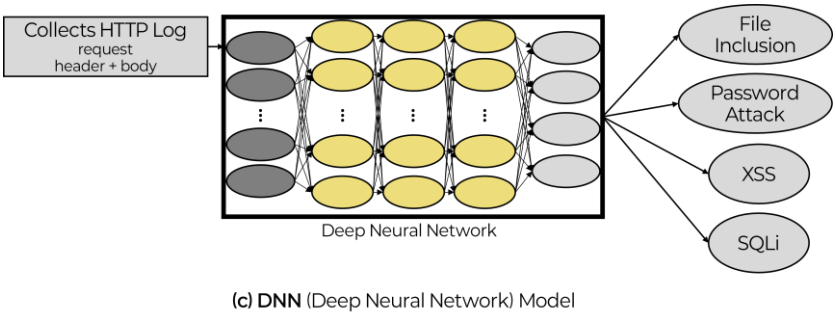

Figure 3: *Defense Model (DNN)*

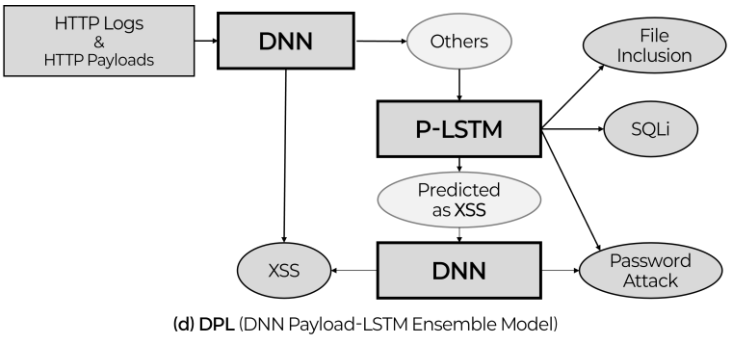

Figure 4: *Defense Model (DPL)*

There are 4 defense models, (1) H-LSTM, (2) P-LSTM, (3) DNN, (4) DPL. For (1) H-LSTM, HTTP request value has been used. (2) P-LSTM model used payload data. H-LSTM and P-LSTM model are both adequate for XSS and SQL injection detection. (3) DNN model is known as well detecting XSS (Juvonen et al., 2015; Kim et al., 2020; Mahoney & Chan, 2002).

DPL model (DNN Payload-LSTM Ensemble Model), suggested in this study, is a ensemble model. It classifies XSS by DNN and classifies others by P-LSTM and DNN again to classify all other attacks.

3.3. Execution Environment

Using 13th Gen Intel® Core™ i9-13980HX(RAM: 32G) as CPU and NVIDIA GeForce RTX 4060 Laptop as GPU. In this environment, every model have done learning 10 times each, using accuracy, precision, recall, and f1-score as performance index (Fawcett, 2006). 76.2% of the dataset used for learning is Password Attack, indicating that the dataset is unbalanced. So f1-score has more impact than accuracy (Pak et al., 2020).

IV. Research Results

4.1. Result Analysis of Performance Evaluation for each model

Figure 5 is a graph that compares every model's classification performance evaluation index. By comparing each model's evaluation index, proposed model DPL has outperformed for every index. The ANOVA result and Post-analysis results are illustrated on **Table 4**.

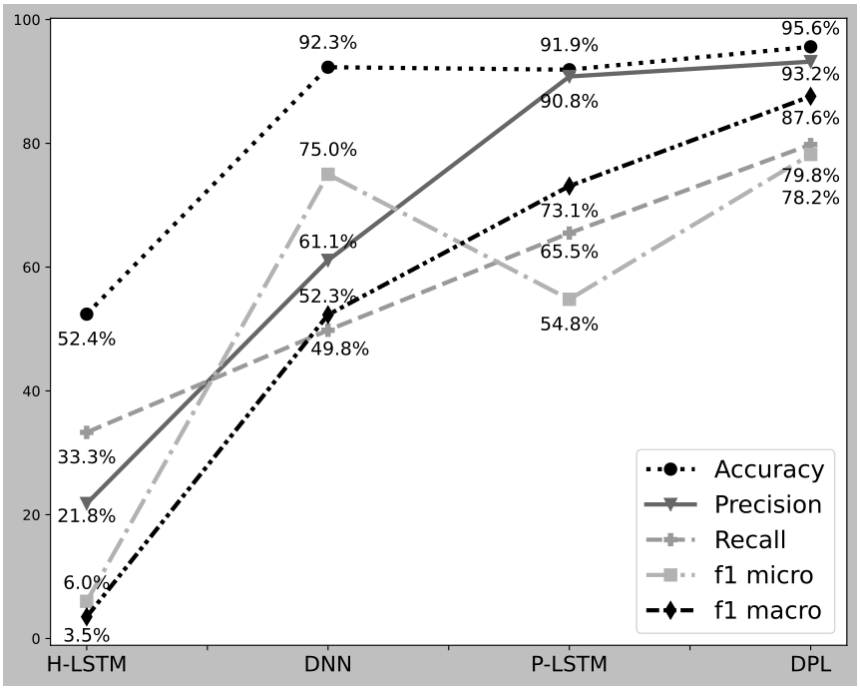

Figure 5: *Performance Index for each models*

To evaluate the classification performance of each models, significant difference has been shown in result of One-Way ANOVA. As result of Tukey HSD post-analysis, there exist significant differences between H-LSTM and DNN, and between P-LSTM and DPL. There is no significant differences between P-LSTM and DPL, but DPL model's performance evaluation index have been shown higher than P-LSTM. So it is expected that usage of suggested model DPL would have significant effect.

4.2. Correspondence Analysis by each Model

In CA, if result of chi-square test for row and column of confusion matrix is statistically significant, the coordinates from result of CA locates far from the center, showing correspondence about relationship placing on the opposite side respect to specific axis. CA result by performance evaluation is illustrated on Figure 6.

The test for independence between rows and columns about Accuracy is statistically significant, $\chi^2(9) = 94.09, p < 0.001$. DNN and DPL well detects Password Attack in Precision and have high relativity. The test for independence between rows and columns about

Recall is statistically significant, $\chi^2(9) = 431.04, p < 0.001$. P-LSTM and DPL well detects File Inclusion and have high relativity about Recall.

Table 4.1 *Analysis for Model Performance Evaluation Difference*

성능평가	model		score	F
Accuracy	H-LSTM	b	52.35	41.08 *
	DNN	a	92.30	
	P-LSTM	a	91.92	
	DPL	a	95.58	
Precision	H-LSTM	b	16.37	127.61 *
	DNN	c	45.80	
	P-LSTM	a	90.63	
	DPL	a	94.90	
Recall	H-LSTM	b	25.00	21.29 *
	DNN	b	37.32	
	P-LSTM	a	74.09	
	DPL	a	84.82	
f1 micro	H-LSTM	c	02.61	63.63 *
	DNN	b	39.21	
	P-LSTM	a	73.07	
	DPL	a	87.64	
f1 macro	H-LSTM	b	02.61	121.25 *
	DNN	a	39.21	
	P-LSTM	a	73.07	
	DPL	a	87.64	

The test for independence between rows and columns about f1-score(micro) is statistically significant, $\chi^2(9) = 240.84, p < 0.001$. The test for independence between rows and columns about f1-score(macro) is statistically significant, $\chi^2(9) = 247.55, p < 0.001$. Two types of f1-

score, macro and micro, have shown similar results. Analysis results show that DPL model has relativity with SQL injection and Password Attack, P-LSTM having relativity with File Inclusion. The test for independence between rows and columns about Precision is statistically significant, , $\chi^2(9) = 333.02, p < 0.001$.

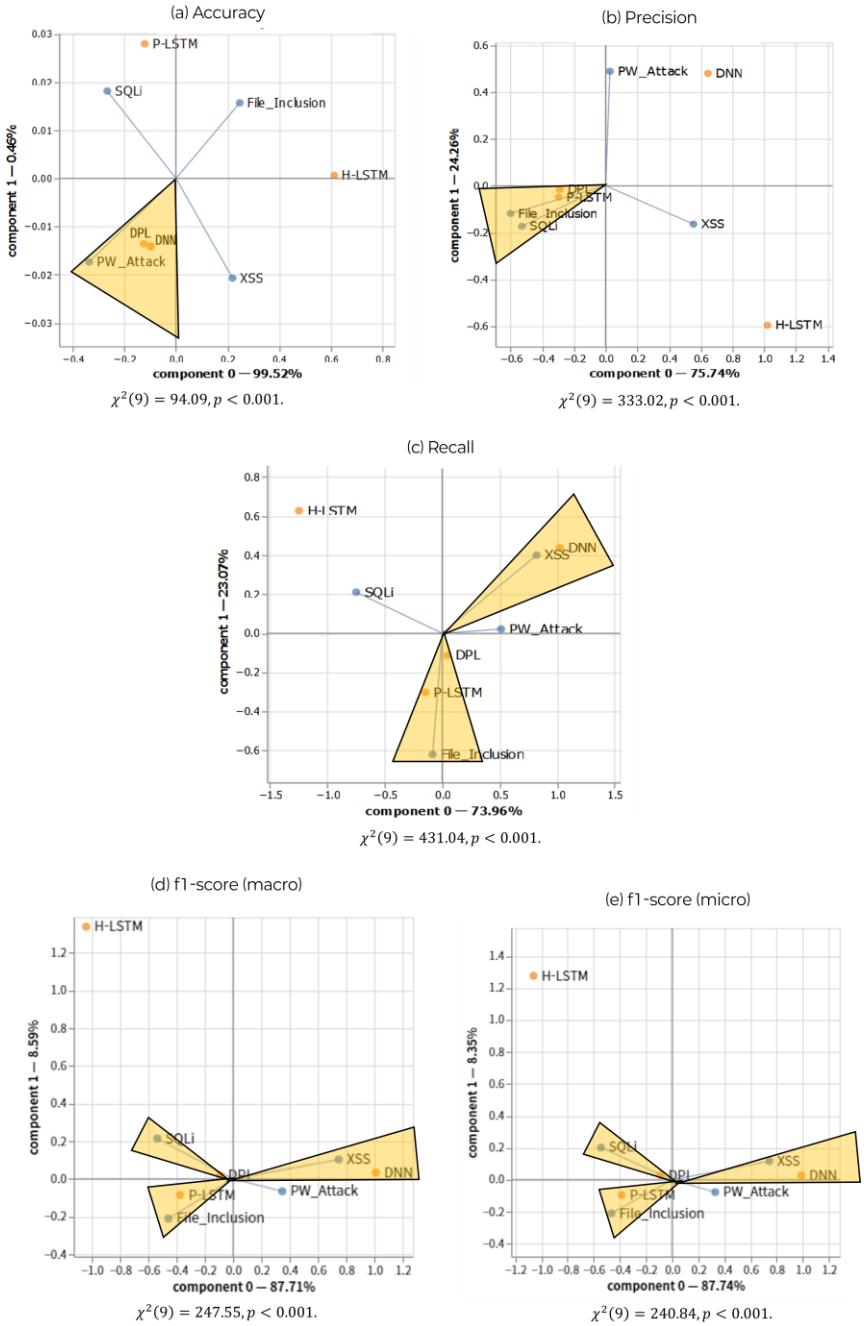

Figure 6: *CA result by performance evaluation*

V. Conclusion and Discussion

In this study, performance of existing P-LSTM, H-LSTM and DNN models have been compared and ensemble model DPL combining P-LSTM and DNN model have been suggested, evaluating the performance of it. The result of the performance evaluation showed that P-LSTM and DPL models' performance is higher than the other models, DPL showing higher performance than P-LSTM. DPL model's performance index shows better performance compared to DNN, H-LSTM, and P-LSTM. Result of ANOVA shows that DPL showed significant performance approvement. It is obvious that using the suggested DPL model in web attack detection field is considered better.

Using statistical methods including ANOVA and CA, this study has found the specific well detecting models corresponding to each of the attacks which are Password Attack, SQL injection, and File Inclusion. Also, the suggested DPL model has relationships with Password Attack, SQL injection, and File Inclusion. But there were no models that has relationship with XSS, constantly shown in OWASP TOP10. So it can be expected that XSS would not be omitted from the future OWASP TOP10 and efforts to build a model having high relationship with XSS need to be made.

It is expected that this study will help making cost-efficient strategy to detect and take a measurements to frequently occurring attack types. s

References

Ahn, J., Lee, E., & Chang, B.-M. (2015). SW 개발보안을 위한 보안약점 표준목록 연구. *Review of KIISC*, *25*(1), 7-17.

Alnabulsi, H., Islam, M. D. R., & Mamun, Q. (2014). Detecting SQL injection attacks using SNORT IDS. In *1st IEEE Asia-Pacific World Congress on Computer Science and Engineering* (pp. 1-7). United States: IEEE Xplore.

Baranwal, A. K. (2012). Approaches to detect SQL injection and XSS in web applications. *EECE 571b, Term Survey paper*.

Eun-jung, C., 정휘찬, & 김승엽. (2015). Attacks and Defenses for Vulnerability of Cross Site Scripting. *디지털융복합연구*, *13*(2), 177-183.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern recognition letters*, *27*(8), 861-874.

Greenacre, M. (2017). *Correspondence analysis in practice*. chapman and hall/crc.

Hassan, M. M., Bhuyian, T., Sohel, M. K., Sharif, M. H., & Biswas, S. (2018). SAISAN: an automated local file inclusion vulnerability detection model. *International Journal of Engineering & Technology*, *7*(2-3), 4.

Hong, S. (2013). XSS Attack and Countermeasure: Survey. *Journal of digital Convergence, 11*(12), 327-332.

Huang, J., Li, Y., Zhang, J., & Dai, R. (2019). UChecker: Automatically detecting php-based unrestricted file upload vulnerabilities. 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN),

Huh, S.-p., Lee, D.-s., & Kim, G.-n. (2009). A Study on XSS Attacks Characters, Sample of Using Efficient the Regular Expressions. *Proceedings of Korea Information Processing Society*, *16*(2), 663-664.

Juvonen, A., Sipola, T., & Hämäläinen, T. (2015). Online anomaly detection using dimensionality reduction techniques for HTTP log analysis. *Computer Networks*, *91*, 46-56.

Kim, Y., Ko, Y., Euom, I., & Kim, K. (2020). Web Attack Classification Model Based on Payload Embedding Pre-Training. *Journal of The Korea Institute of Information Security & Cryptology*, *30*(4), 669-677.

Liang, J., Zhao, W., & Ye, W. (2017). Anomaly-based web attack detection: a deep learning approach. Proceedings of the 2017 VI International Conference on Network, Communication and Computing,

Mahoney, M. V., & Chan, P. K. (2002). Learning nonstationary models of normal network traffic for detecting novel attacks. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining,

OWASP. (2024). *Top10*. Retrieved Jan 03 from https://github.com/OWASP/Top10

Pak, D., Hwang, M., MINJI, L., SUNG-IL, W., Hahn, S.-W., Jung, L. Y., & Hwang, J. (2020). Application of Text-Classification Based Machine Learning in Predicting Psychiatric Diagnosis. *Korean Journal of Biological Psychiatry*, *27*(1).

Rathore, S., Sharma, P. K., & Park, J. H. (2017). XSSClassifier: an efficient XSS attack detection approach based on machine learning classifier on SNSs. *Journal of Information Processing Systems*, *13*(4), 1014-1028.

Roesch, M. (1999). Snort: Lightweight intrusion detection for networks. Lisa,

Roh, J.-H., Min, S.-H., & Kong, M.-S. (2022). Analysis of Fire Prediction Performance of Image Classification Models based on Convolutional Neural Network. *Fire Science and Engineering*, *36*(6), 70-77.

Sarkar, S., & Nandan, M. (2022). Password Strength Analysis and its Classification by Applying Machine Learning Based Techniques. 2022 Second International Conference on Computer Science, Engineering and Applications (ICCSEA),

Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval* (Vol. 39). Cambridge University Press Cambridge.

Syaifuddin, S., Risqiwati, D., & Sidharta, H. A. (2018). Automation snort rule for XSS detection with honeypot. 2018 5th International conference on electrical engineering, computer science and informatics (EECSI),

Tajbakhsh, M. S., & Bagherzadeh, J. (2015). A sound framework for dynamic prevention of Local File Inclusion. 2015 7th Conference on Information and Knowledge Technology (IKT),

Yacouby, R., & Axman, D. (2020). Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models. Proceedings of the first workshop on evaluation and comparison of NLP systems,

Zhang, D., Wang, J., Zhao, X., & Wang, X. (2015). A Bayesian hierarchical model for comparing average F1 scores. 2015 IEEE International Conference on Data Mining,

염태균. (2016). *A Black list Extraction Method for Automated Web Attack Tools based on weighted intrusion events* (Publication Number Master's Thesis) Chonnam National University]. Gwangju.

https://www.riss.kr/link?id=T14053003