Journal of Engineering Science and Technology
Special Issue on IAC2021, October (2021), 24 - 37


# A WATERMARK-BASED SECURE MODEL FOR DATA SECURITY AGAINST SECURITY ATTACKS FOR MACHINE LEARNING ALGORITHMS

MUHAMMAD TAYYAB[1,2,*], MOHSEN MARJANI[1], N. Z. JHANJHI[1], IBRAHIM ABAKER TARGIO HASHEM[3], RAJA SHER AFGUN USMANI[1]

[1]School of Engineering, Taylor's University, Taylor's Lakeside Campus, No. 1 Jalan Taylor's, 47500, Subang Jaya, Selangor DE, Malaysia
[2]Department of Computing, Shifa Tameer-e-Millat University, Islamabad, 44000, Pakistan
[3]College of Computing and Informatics, Department of Computer Science, University of Sharjah, 27272 Sharjah, UAE
*Corresponding Author: muhammadtayyab@sd.taylors.edu.my, tayyab.ssc@stmu.edu.pk

## Abstract

Machine Learning (ML) has been known as one of the most widely used by the decision-based application. Most of the security sensitive applications have been using DL for the improvement and betterment of outcomes while solving real-life applications. Poisoning and evasions attacks are the common examples of security attacks where the attacker deliberately inject malicious injections into the dataset to get the information of model settings and dataset. Hence, in this paper we have proposed a watermark-based secure model for ensuring data security and robustness against poisoning and evasion attacks before training and testing the DL algorithms. Our proposed model has been developed on ML algorithms e.g., eXtreme Gradient Boosting (XGBOOST) and Random Forest to ensure the data security against most common security attacks. We have evaluated proposed watermark based secure model using benchmark mechanism to show that the by introducing secure model, the performance has not been disturbed. We have computed prediction of daily cases on COVID-19 dataset and achieved similar results. Finally, our proposed model can detect significant attack detection rate even for large numbers of attacks (poisoning and evasion attacks). It is believed that our proposed model can also be implemented in other learning environment to mitigate the security issues and improve security applications.

Keywords: Evasion attacks, Machine learning (ML), Poisoning attacks, Random forest, XGBOOST.

## 1. Introduction

Many data-driven applications have been using Machine Learning (ML) algorithms for enhancement in performance using numerous datasets. In real-life scenarios, Deep Learning (DL) has provided outstanding results while addressing decision-based problems [1]. Similarly, many solutions of data-driven problems have also been transformed by the advancement of DL algorithms in various applications like autonomous security audits for systems logs, object detection by vision with unmanned car driving vehicles (UVs) [2], spam email detection systems [3-5]. One of the important parameters to automate the process and to improve the performance of DL algorithm is numerous datasets. However recent literature has shown many uncovered vulnerabilities in DL algorithm due to use of huge dataset for training process [6, 7].

ML has been providing a breakthrough in solving the bigger decision-based problems with the help of many learning algorithms in past. In results, it has been observed in many areas that DL has been used to decode the larger scientific problems at an exceptional level like in reconstruction of human brain, mutation growth in DNAs of living organisms, drugs prediction molecules to structure the health activities and other internet of medical things (IoMT) areas [8]. Deep neural network (DNN) has also been reported as one of the preferred choices to address many hard and challenging scientific problems in natural language process (NLP) and audio speech recognition systems [9]. Most of the modern security-sensitive applications and health prediction models have been enhanced and upgraded with the help of DL algorithms. Many innovative techniques have been introduced by the DL algorithms to enhance and improve the performance of security-sensitive and critical applications which has minimize the human cost in both supervised and non-supervised learning models such as neural network (NN), artificial neural network (ANN) , linear regression and decision trees [10-12].

Despite all the innovative and interesting features introduced by ML in many important areas, there have been reported many security and privacy concerns that has taken the intension of security analyst. There have been reported many security and privacy issues in ML due to the involvement of large dataset for training the model. The most common security issues are poisoning and evasion attacks which have affected the performance and efficacy of ML algorithms [13]. In poisoning attacks, the strong adversaries have primary motive to add an executable noise into the training phase which compromises the overall performance of model. An adversary first analyses the outcomes of model and then it generates the malicious dataset which can generate the similar pattern as previous. After successfully generation of malicious dataset, then an adversary can easily replace the original training dataset with malicious one [14]. Similarly, in case of evasion attacks, an adversary can manipulate the outcomes of ML model while in testing phase and generate the malicious dataset which can give the similar outcomes as original [15]. An adversary can thus replace the original testing dataset with malicious one and can get the information and secret setting of model that is used for classification and prediction. For example, in exploratory attacks, some commonly known "Good" words are added which can dodge the spam detection function of emails and labels as non-spam email, which then classified as non-spam email [16]. Hence an adversary can be able to get the useful information from any personal account or business email. These kind of security attacks have been considered as the most challenging attack in ML algorithms during training and testing phase. Primary

motive of such kinds of security attacks is misled toward wring prediction or wrong classification by the ML algorithms. In previous research studies, there have been proposed some secure model that were based on cryptographic functions like homomorphic encryption scheme (HES), which has provided security to the dataset for learning models. But such cryptographic functions have increased the computation overhead of system beyond the limit like Faster CryptoNets, CryptoDL-1 [17, 18]. This ultimately affect the execution and performance evaluation of DL models. Moreover, before applying encryption and decryption, additional operations were applied to secure the small datapoints which can be changed due to additional mathematical operations [19].

Therefore, in this paper we have proposed a cryptographic based secure model that has used hash functions SHA512 and then extended to formulate the digital signature and termed as a light-weight watermark to ensure data security for learning models. the proposed model first computes hash values by applying hash functions on the dataset which is then concatenated with a unique bit to formulate a unique signature against each data attribute. This signature is then appended into the dataset as an attribute and become a part of the dataset and termed as watermark. The proposed model can be able to verify the dataset using digital signature. For verification, this process is repeated to formulate the watermark and then compared to old one. Finally, before training the model, the additional signatures will be removed so that it will not affect the training process [20]. We have implemented out proposed framework by using XGBOOST and Random Forest [21]. We have provided a mechanism which have provided security to learning algorithms. It can be extended for other ML algorithms that are tends to solve decision-based problems for data-driven scenarios.

In this paper, we have presented a security model that can provide security for the dataset used for ML algorithms. Proposed secure model is based on cryptographic functions like Hash function SHA512. Following are the key contributions of our proposed security model.

- We have provided a security model which have provided a security for dataset used for ML algorithms like XGBOOST and Random Forest against the most challenging security attacks like poisoning and evasion attacks has been proposed.
- We have modified Hash function SHA512 by appending a unique bit to formulate the watermark and termed as light-weight watermark.
- The proposed model has been evaluated based on prediction of ML algorithms, and computation cost of proposed framework and compared with the benchmark results.
- The proposed model has maintained the accuracy level, precision value as significant and reduced the computation cost.
- The proposed model has been exposed against each type of poisoning and evasion attack and have detected a significant value of attack rate.

The rest of the paper is organized as: in section 2, we have discussed the related literature on basis of security issues in ML models and security attacks that have greatly affected the ML and DL algorithms. In section 3, we have provided a detailed methodology of the proposed secure model, including the methods, evaluation matrix, experimental setup, and dataset used for the implementation of the proposed secure model. While in section 4, we have provided a detailed discussion on the

results and limitations of the proposed secure model. Finally, in section 5, we have concluded our paper in conclusion and future work.

## 2. Related Work

This section has provided a recent literature related to the security and privacy attacks against ML algorithms among various fields. Many studies have highlighted security issues in ML models where security attacks have affected the outcomes of model by injecting malicious dataset into the dataset. Many studies have also mentioned some mechanism to mitigate the effects of these attacks, but computation overhead was the major concerns. Therefore, in this paper we have provided and highlighted the most challenging security attacks like poisoning and evasion attacks for ML algorithms.

### 2.1. Security issues

Security is known as one of the key parameters for analysing the performance of critical applications. It is also one of the most challenging factors for DL algorithms where the security attacks have affected the algorithms. Specially in real world scenarios, security of learning algorithm is considered as the most critical element and many researchers have considered security as one of the most influenced parameters. In decision-based and data-driven real-world problems, DL is trained with huge amount of dataset so that the performance of algorithms may not be affected. However, many deliberate intruders or attackers have been reported in many cases where data poisoning have been introduced into the original dataset to subvert the learning process. Xiao et al. [22] has proposed a model named as Secure and Private AI (SPAI) to solve the security issues in DL. SPAI was aims to mitigate the security attacks on dataset for learning algorithms and minimize the adverse effects in the model. however, such action to remove the effects of security attacks, but increase the computation cost of algorithms. To address the increase in computation cost of algorithm Mohanty et al. [23] has proposed a outlier detection mechanism, but such solution might change the decision boundary of the model and change the label of the dataset. Which can then be worse than of the attacker.

There have been reported many security and privacy issues in DL where an adversary can be more affective when it got the access to the dataset and model setting using some malicious queries into the dataset. Using such queries and analysing the different pattern, adversary can then able to insert malicious and executable noise into the dataset which not only destroy the model setting but also loose the data information during training phase [24]. Normally, few attacks like poisoning and evasion attacks are considered as more dangerous for this purpose when the DL model has been applied in security critical applications or data-driven technologies [25]. DL model uses numerous dataset for training and model evaluation phase, which is the loop whole for an adversary where it can inject the noise into the dataset specially for prediction new labels based on historical dataset [26]. For this reason this is very important to secure the dataset used for learning purpose and model evaluation phase [27]. Hence it is computationally very easy for and adversary to compute the similar dataset after being analysing using queries [28]. Integrity of dataset is also at high risk when it has been collected form many untrusty resources during data collection phase and data normalization phase [29, 30]. For example, malicious samples are often collected from untrusted machines that have been

compromised with some unknown vulnerabilities like honeypots and several online services [31-34].

## 2.2. Poisoning attacks

Poisoning attack is the type of attack where an attacker deliberately adds some malicious noise or dataset into the original dataset or add some malicious executable noise dataset into the original dataset to get the information of original learning model and outcomes. This can mis lead the classification and prediction of DL model, and an attacker can be able to mis lead the information of original dataset used for training and testing phase of model. This type of attacks is considered as poisoning attacks. This is further categorized into Blackbox and WhiteBox attacks depends upon the setting of an attacker [35-37]. If the attacker has limited knowledge about the model setting or complete knowledge about the components of model or dataset then this type of poisoning attacks is considered as WhiteBox attacks, while on the other hand if the attacker has no knowledge about the setting of model used then this type of poisoning attacks is termed as BlackBox attack [38]. An attacker inserts valuable queries into the model and generate the random outcomes based on setting used by the model and then computes some pattern. After having numbers of outcomes of queries, then an attacker generates the malicious dataset to insert into the original dataset to get the information.

There have been many applications like in cybersecurity and reliability applications where data classification and predicting new labels are considered as challenging part. Böhler and Kerschbaum [39] has proposed a framework in an survey related to applications of DL in cybersecurity and reliability. It has highlighted many loopholes in modern AI applications and then opens many issues for future researchers. Similar pattern was proposed by Cheng et al. [40], which has showed different risk factors both for classification task and prediction using historical dataset or information. There have been numbers of protocols implemented to reduce the effects of poisoning attacks in various security-sensitive applications like adversarial training, gradient masking, GAN and statistical approaches. These counter measure defence techniques have provided the desired motivation but at the same time they have increased the computation cost of the model significantly while performing different mathematical operations.

## 2.3. Evasion attacks

In this type of security attacks for learning algorithms, the attackers have primary motives to regenerate a similar pattern of dataset based on outcomes of DL model and then replace the generated dataset with original dataset. Once the desired dataset has been regenerated by the process of executable queries like in poisoning attacks, then attacker can then replace the original dataset with this malicious dataset. This category and setting of malicious attacks are termed as evasion attacks which can activated during the testing phase of DL model. There are two further types of evasion attacks that are primarily based on setting of attacker, like if the attacker has limited knowledge of model settings, then this is termed as week adversaries and if the attacker has fully knowledge, then this type of attack setting is termed as strong adversaries.

## 3. Light-Weight Watermark Secure Model

This section explains the details of our proposed model, which is comprised of different phases. Our proposed model has provided security for dataset using ML algorithms, which was derived and extracted from the secure model used for DL algorithms. Furthermore, evaluation matrix, experimental setup, results, and discussion have been provided in detail.

### 3.1. Phase-1: formulation of unique watermark

In phase 1, data processing is carried out in the form of text form or comma separated values (CSV) file. After successful data pre-processing, following step will be taken place:

- Hash function SHA512 is applied on dataset to compute the hash values for each row of the dataset.
- For the generation of unique watermark or signature, a unique HEX value is concatenated with hash values. Now this extended hash value is termed as a digital watermark, and this is appended in the dataset as an attribute. This operation can also be observed in Algorithm 1.
- This unique signature can also be verified at the last stage before training the model so that it is authorized that the dataset has not been compromised while accessing from cloud.
- Lastly, the dataset with the unique watermark will be outsource to the cloud for further processes of proposed model.

### 3.2. Phase-2: verification

In phase 2, watermark verification process is carried out by the verification of digital content that was appended in phase 1.

- The dataset is first downloaded from the cloud storage to process further for learning the ML model.
- The watermarks are regeneration as in phase 1 and then compared for calculation of attack detection rate.
- If both watermarks matched each other, then the proposed model will transfer the carried dataset to the phase 3 of proposed model and model training and model evaluation is carried out.
- If the watermark does not completely match, then it calculates the impurity level that how much the dataset is compromised with malicious dataset. This also computes the attack detection rate and impurity level in the dataset.
- In this phase, our proposed model also checks for impurity level, if the impurity level exceeds certain limit, then this dataset is declared as “Malicious Dataset” and this cannot be processed further for training the ML model.
- If the impurity level is within the limit set by the proposed model, then this dataset is partially used for training the ML model and declared as partially fit dataset for ML model.

**Algorithm 1: Watermark Generation and Appending**

Input: dataset $D_0$

Output: $D_{Wmark}$

**1** Procedure HASH $(D_0)$

**2** For (Row $i \rightarrow R_n$)

**3** $H_0 = Hash(R_i[j])$

**4** $D_{Wmark} = D_0 \,||\, H_0 \,||\, Bit$

**5** return $D_{Wmark}$

**7** Dataset uploaded

**Algorithm 2: Watermark Verification**

**Input:** $D_{Wmark}$

**Output: Clean Dataset** $D_{clean}$

**1** Procedure $Hash(D_{Wmark})$

**2** For (Row $i \rightarrow R_n$)

**3** $H_1 = Hash(R_i[j])$

**4** $D_{Wmark2} = D_{Wmark} \,||\, H_1 \,||\, bit$

**5** $If\,(H_{Wmark} == H_{Wmark2})$

**6** *proceed with data sampling*

**7** $D_{clean} = \mathrm{R\_}Col(D_{hash\_1})$

**8** Return $D_{clean}$

**9** *Else*

**10** Return "The dataset has been impure"

**11** Return $D_{clean}$

### 3.3. Phase-3: machine learning model

In phase 3, once the dataset is verified and within the certain limit of impurity level, then the dataset is normalized with in certain range so that the ML algorithm is trained with a trained dataset.

- The whole dataset is now split into two part like training and testing dataset. Training dataset is normally taken as larger part and testing dataset is taken as smaller part.
- The proposed model will now normalize the dataset to limit the data values into certain range of numbers.
- After the normalization of dataset, now ML algorithms like XGBOOST and Random Forest will be trained with the training dataset and after training testing dataset will be used to evaluate the prediction score of both the algorithms.

### 3.4. Dataset

We have used daily dataset for COVID-19 that is available online platform "Our World in data". We have implemented out proposed model using COVID-10

dataset. This dataset is available country wise of whole world for complete month. We have used selected features that are key factors for predicting daily cases throughout the world.

## 3.5. Experiments

We have explained the experimental details of proposed model in this section, which is mainly composed of cryptographic functions i.e., computation of hash values using hash functions. Then we have generated the watermark with the extension of hash values. This whole phenomenon will not only help to mitigate the attack on the dataset but also provide the attack detection rate of the proposed model.

## 3.6. Watermark generation using Hash functions (SHA512)

To provide the data authenticity and data integrity, Hash function have been using in various field in the form of digital signature for digital contents. Hash functions usually take random strings as an input and generate a fix length of alpha numeric characters as an output. It is called a "message digest or hash digest". It is considered as one-way functions, which refers that it can only generate output of a specific values, this process cannot be reversed. Hash values are also considered a digital signature [41]. They have following properties which are as uniform distribution, fixed length collision resistance.

In our proposed secure model, hash function SHA 512 has been used to generate the unique digital watermark for the verification and authentication of dataset before the training and testing the ML model [42]. To generate the watermark, we have modified the hash values with the addition of unique bit that is concatenated with the hash values to form the unique watermark. This will help not only in data authentication and verification but also it can mitigate the effects of security attacks like poisoning and evasion attacks on the dataset before training phase of ML model. This process will also improve the security of dataset and reducing the computational cost of algorithms and maintaining the accuracy as high as for original model.

## 4. Results and Discussion

In this section, results have been discussed in detail with the comparison between proposed model and benchmark results. We have used benchmarking phenomenon to evaluate our proposed model. We have provided the computational cost of proposed model which is expressed asymptotically.

## 4.1. Accuracy

The proposed secure model has been evaluated based on accuracy as a key parameter. We have used two basic DL algorithms i.e., XGBOOST and Random Forest for prediction of COVID-19 daily new cases based on previous dataset. We have benchmarked the original algorithms and computed the accuracy and then we have applied our proposed model setting to the original model to check the accuracy either remains same or drop. Results have shown that the proposed model has not changed the outcome and achieved high level of accuracy. Similarly, we have also exposed our proposed secure model and original model to poisoning and evasion attacks. In Table 1, we have given the results for different parameters for both models against original setting and proposed model setting. It is clearly seen that

the values have not dropped, and proposed model has maintained the security and privacy issues as well as remains constant for these values. Similarly, in Table 2 and Table 3, we have provided the results of accuracies of both model when the proposed model and original model is exposed against security attacks like poisoning attacks and evasion attacks. The results have shown that the proposed model has detected the attack quite decently and accuracy has dropped significantly when the model was exposed against security attack.

The proposed model has been evaluated based on one of the key parameters that is accuracy. We have implemented the proposed model on two basic predicting algorithms like XGBOOST and Random Forest. Initially, we have made the algorithm as benchmark in original model column and then we have applied the proposed model setting and compared the results based on these setting. In Figs. 1 and 2, we have provided the prediction score of daily new cases for COVID-19 throughout the world.

**Table 1. Comparison of MAE and EMSE for XGBoost and Random Forest.**

| Algorithms | Original Models | | Proposed Models | |
|---|---|---|---|---|
| | **MAE** | **RMSE** | **MAE** | **RMSE** |
| **Random Forest** | 2829.501 | 21348.35885 | 2836.254 | 21358.48965 |
| | 2830.045 | 21354.33858 | 2980.045 | 21885.33858 |
| **XGBoost** | 2767.792426 | 21029.42879 | 2767.792426 | 21029.42879 |
| | 2875.792426 | 22874.49857 | 2767.792426 | 21029.42879 |

**Table 2. Comparison of accuracies in original and adversarial setup against security attacks.**

| Algorithm | Original Models | | | | Security Attacks |
|---|---|---|---|---|---|
| | Clean Training (%) | Adv. Training (%) | Test Accuracy (%) | Adv. Test Accuracy (%) | |
| **Random Forest** | 98.25 | 68.78 | 98.50 | 54.21 | FGSM |
| | 98.99 | 67.86 | 98.49 | 56.37 | |
| | 99.10 | 48.25 | 99.81 | 50.36 | JSMA |
| | 99.38 | 40.28 | 99.79 | 41.28 | |

**Table 3. Comparison of accuracies in original and adversarial setup against security attacks.**

| Algorithms | Proposed Model | | | | Security Attacks |
|---|---|---|---|---|---|
| | Clean Training (%) | Adv. Training (%) | Test Accuracy (%) | Adv. Test Accuracy (%) | |
| **Random Forest** | 98.10 | 64.78 | 98.69 | 52.34 | FGSM |
| | 98.38 | 60.25 | 98.38 | 50.73 | |
| **XGBOOST** | 99.15 | 46.15 | 99.36 | 31.28 | JSMA |
| | 99.25 | 36.45 | 99.55 | 39.78 | |

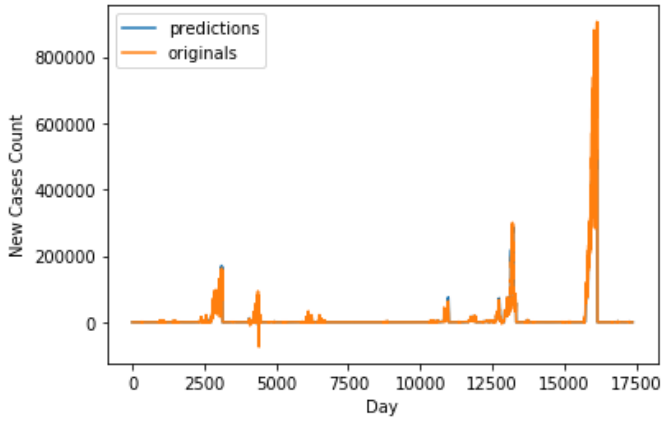


**Fig. 1. Prediction of daily new cases for covid-19 using Random FOREST.**

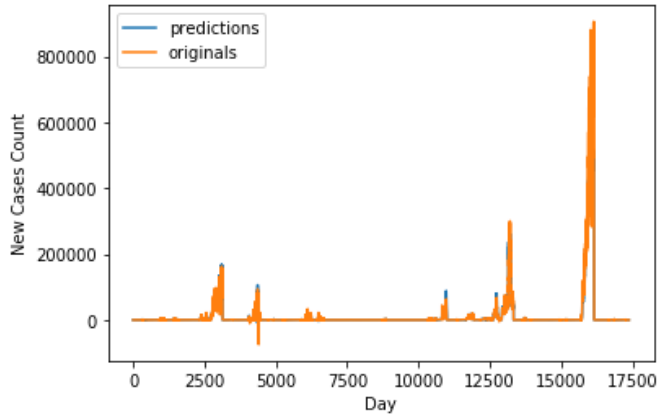


**Fig. 2. Prediction of daily new cases for covid-19 using XGBOOST.**

## 4.2. Attack detection rate

We have also provided the attack detection rate for our proposed model. As our proposed model is based on cryptographic hash function and we have modified the hash values to compute the light-weight watermark for digital content. Hence using digital signatures, we have calculated the attack detection rate and represented in Table 4.

**Table 4. Precision score for the proposed framework and ML algorithms.**

| Parameters | Original Model with clean Data | Original Model on Impure Dataset | Proposed Model on clean Dataset | Proposed Model on Impure Dataset |
|---|---|---|---|---|
| **Attack Detection Rate** | 0.95 | 0.65 | 0.98 | 0.35 |
| | 0.94 | 0.55 | 0.99 | 0.34 |

We have also set the threshold values for attacks if the attack detection rate is greater than certain limit or threshold value then our proposed model will label this

dataset as malicious dataset and outcome will be treated as malicious. For other case, if the impurity level is less than certain limit or threshold value then our proposed model will label this dataset as partial impure data and generate the partial impure outcomes.

### 4.3. Computational cost

We have presented the computational cost for our proposed model which has not increased the overall cost for model. We have implemented the proposed model based cryptographic hash functions SHA-512 which is light weight in the domain of cryptography. We have also formulated digital signature in the form of watermark. The watermark has been formulated by appending a unique bit with hash values to differentiate from other hash values. This whole process has not increased the computational cost. The total computational cost of our proposed model can be shown in asymptotic notation as $\Theta(\eta)$ for additional operation which is quite manageable for security sensitive applications.

## 5. Conclusion

In various security sensitive applications, ML has been known as one of the most important part used for classification and prediction because of decision-based problem. There have been many advantages introduced by ML algorithm to improve the accuracy level of prediction and classification in most of the security sensitive applications. Although there have been numbers of innovative features for ML, it has faced number of security issues which have largely affected the outcomes. Poisoning and evasion attacks are common security attacks which have been reported in this domain. In this study, we have proposed a secure model based on cryptographic hash function SHA-512 to formulate the light-wight watermark, which has provided the data authenticity, integrity and resolved the privacy issue in dataset for ML algorithms. Our proposed model has also achieved high attack detection rate and maintain the privacy of dataset against the security attacks. Our proposed model has been evaluated based on accuracy, prediction score, attack detection rate and computational cost and we have shown high results for our proposed model. It is believed that our proposed model can also be used to solve the security and privacy issues in other learning algorithms including ML and DL algorithms. In future, more type of attacks can also be explored against our proposed model.

## References

1. Gopi, R.; Sathiyamoorthi, V.; Selvakumar, S.; Manikandan, R.; Chatterjee, P.; Jhanjhi, N.Z.; and Luhach, A.K. (2021). Enhanced method of ANN based model for detection of DDoS attacks on multimedia internet of things. *Multimedia Tools and Applications*, 81, 26739-26757.
2. Hassan, R.; Qamar, F.; Hasan, M.K.; Aman, A.H.M.; and Ahmed, A.S. (2020). Internet of things and its applications: a comprehensive survey. *Symmetry*, 12, 1-29.
3. Muhammad, A.N.; Aseere, A.M.; Chiroma, H.; Shah, H.; Gital, A.Y.; and Hashem, I.A.T. (2020). Deep learning application in smart cities: recent development, taxonomy, challenges and research prospects. *Neural Computing and Applications*, 33, 2973-3009.

4. Bilal, M.; Usmani, R.S.A.; Tayyab, M.; Mahmoud, A.A.; Abdalla, R.M.; Marjani, M.; Pillai, T.R.; and Hashem, I.A.T. (2021). *In: Augusto, J.C. handbook of smart cities*. Chapter: Smart cities data: framework, applications, and challenges. Denmark: Springer Cham.
5. Lee, S.; Abdullah, A.; Jhanjhi, N.Z.; and Kok, S.H. (2021). Classification of botnet attacks in IoT smart factory using honeypot combined with machine learning. *PeerJ Computer Science*, 7, 1-23.
6. Usmani, R.S.A.; Saeed, A.; and Tayyab, M. (2021). *ICT Solutions for improving smart communities in Asia*. Chapter: Role of ICT for community in education during COVID-19. Pennsylvania: IGI Global, 125-150.
7. Tayyab, M.; Marjani, M.; Jhanjhi, N.Z.; and Hashem, I.A.T. (2021). A light-weight watermarking-based framework on dataset using deep learning algorithms. *Proceeding of the 2021 National Computing Colleges Conference*. Taif, Saudi Arabia, 1-6.
8. Pillai, T.R.; Hashem, I.A.T.; Brohi, S.N.; Kaur, S.; and Marjani, M. (2018). Credit card fraud detection using deep learning technique. *Proceeding of the 2018 Fourth International Conference on Advances in Computing, Communication & Automation*. Subang Jaya, Malaysia, 1-6.
9. Yue, Y.; Li, S.; Legg, P.; and Li, F. (2021). Deep learning-based security behaviour analysis in IoT environments: a survey. *Security and communication Networks*, 2021, 1-13.
10. Can, Y.S.; and Ersoy, C. (2021). Privacy-preserving federated deep learning for wearable IoT-based biomedical monitoring. *ACM Transactions on Internet Technology*, 21(1), 1-17.
11. Bilal, M.; Marjani, M.; Hashem, I.A.T.; Abdullahi, A.M.; Tayyab, M.; and Gani, A. (2019). Predicting helpfulness of crowd-sourced reviews: a survey. *Proceedings of the 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics*. Karachi, Pakistan, 1-8.
12. Jacob, S.; Alagirisamy, M.; Xi, C.; Balasubramanian, V.; Srinivasan, R.; Parvathi, R.; Jhanjhi, N.Z.; and Islam, S.M.N. (2021). AI and IoT-enabled smart exoskeleton system for rehabilitation of paralyzed people in connected communities. *IEEE Access*, 9, 80340-80350.
13. Hameed, K.; Haseeb, J.; Tayyab, M.; Junaid, M.; Maqsood, T.B.; and Naqvi, M.H. (2017). Secure provenance in wireless sensor networks - a survey of provenance schemes. *Proceedings of the International Conference on Communication, Computing and Digital Systems*. Islamabad, Pakistan, 11-16.
14. Muzammal, S.M.; Murugesan, R.K.; and Jhanjhi, N.Z. (2021). A comprehensive review on secure routing in internet of things: Mitigation methods and trust-based approaches. *IEEE Internet of Things Journal*, 8(6), 4186-4210.
15. Dowlin, N.; Gilad-Bachrach, R.; Laine, K.; Lauter, K.; Naehrig, M.; and Wernsing, J. (2016). CryptoNets: applying neural networks to encrypted data with high throughput and accuracy. *Proceedings of the 33rd International Conference on Machine Learning*. New York, USA, 201-210.
16. Hesamifard, E.; Takabi, H.; Ghasemi, M.; and Jones, C. (2017). Privacy-preserving machine learning in cloud. *Proceedings of the 2017 on Cloud Computing Security Workshop*. Dallas, Texas, USA, 39-43.

17. Jiang, S.; Ye, D.; Huang, J.; Shang, Y.; and Zheng, Z. (2020). Smart steganography: light-weight generative audio steganography model for smart embedding application. *Journal of Network and Computer Applications*, 165.
18. Lim, M.; Abdullah, A.; Jhanjhi, N.Z.; and Khan, M.K. (2019). Situation-aware deep reinforcement learning link prediction model for evolving criminal networks. *IEEE Access*, 8, 16550-16559.
19. Vijayalakshmi, B.; Ramar, K.; Jhanjhi, N.Z.; Verma, S.; Kaliappan, M.; Vijayalakshmi, K.; Vimal, S.; and Ghosh, U. (2020). An attention-based deep learning model for traffic flow prediction using spatiotemporal features towards sustainable smart city. *International Journal of Communication Systems*, 34(3), 1-14.
20. Carlini, N.; and Wagner, D. (2017). Adversarial examples are not easily detected: Bypassing ten detection methods. *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*. New York, United States, 3-14.
21. Ovadia, Y.; Fertig, E.; Ren, J.; Nado, Z.; Sculley, D.; Nowozin, S.; Dillon, J.V.; Lakshminarayanan, B.; and Snoek, J. (2019). Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. *Proceedings of the 33rd Conference on Neural Information Processing Systems*. Vancouver, Canada 13991-14002.
22. Xiao, Q.; Li, K.; Zhang, D.; and Xu, W. (2018). Security risks in deep learning implementations. *Proceedings of the IEEE Security and Privacy Workshops*. Francisco, USA, 123-128.
23. Mohanty, S.N.; Lydia, E.L.; Elhoseny, M.; Al Otaibi, M.M.G.; and Shankar, K. (2020). Deep learning with LSTM based distributed data mining model for energy efficient wireless sensor networks. *Physical Communication*, 40, 101097-101107.
24. Thiyagarajan, P. (2020). *Handbook of research on machine and deep learning applications for cyber security*. Chapter: A review on cyber security mechanisms using machine and deep learning algorithms. United States: IGI Global, 23-41.
25. Gamage, S.; and Samarabandu, J. (2020), Deep learning methods in network intrusion detection: A survey and an objective comparison. *Journal of Network and Computer Applications*, 169, 102767-102782.
26. Caminero, G.; Lopez-Martin, M.; and Carro, B. (2019). Adversarial environment reinforcement learning algorithm for intrusion detection. *Computer Networks*, 159, 96-109.
27. Ren, K.; Zheng, T.; Qin, Z.; and Liu, X. (2020). Adversarial attacks and defenses in deep learning. *Engineering*, 6(3), 346-360.
28. Papernot, N.; McDaniel, P.; Sinha, A.; and Wellman, M.P. (2018). SoK: Security and privacy in machine learning. *Proceedings of the IEEE European Symposium on Security and Privacy*. London, United Kingdom, 399-414.
29. Li, B.; Wang, Y.; Singh, A.; and Vorobeychik, Y. (2016). Data poisoning attacks on factorization-based collaborative filtering. *Proceedings of the 30th Conference on Neural Information Processing Systems*. Barcelona, Spain, 1885-1893.
30. Pandl, K.D.; Thiebes, S.; Schmidt-Kraepelin, M.; and Sunyaev, A. (2020). On the convergence of artificial intelligence and distributed ledger technology: a scoping review and future research agenda. *IEEE Access*, 8, 57075-57095.

31. Li, Y.; Li, H.; Xu, G.; Xiang, T.; Huang, X.; and Lu, R. (2020). Toward secure and privacy-preserving distributed deep learning in fog-cloud computing. *IEEE Internet of Things Journal*, 7(12), 11460-11472.
32. Chowdhury, A.R.; Wang, C.; He, X.; Machanavajjhala, A.; and Jha, S. (2020). Cryptϵ: Crypto-assisted differential privacy on untrusted servers. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. United States, 603-619.
33. Huang, T.; Zhang, Q.; Liu, J.; Hou, R.; Wang, X.; and Li, Y. (2020). Adversarial attacks on deep-learning-based SAR image target recognition. *Journal of Network and Computer Applications*, 162, 102632-102944.
34. Kaissis, G.A.; Makowski, M.R.; Rückert, D.; and Braren, R.F. (2020). Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence*, 2, 305-311.
35. Chen, W.; Zhang, Z.; Hu, X.; and Wu, B. (2020). Boosting decision-based black-box adversarial attacks with random sign flip. *Proceedings of the 16th European Conference on Computer Vision*. Glasgow, United Kingdom, 276-293.
36. Pan, X.; Zhang, M.; Ji, S.; and Yang, M. (2020). Privacy risks of general-purpose language models. *Proceedings of the 2020 IEEE Symposium on Security and Privacy*. San Francisco, California, 1314-1331.
37. Sun, Y.; Liu, J.; Wang, J.; Cao, Y.; and Kato, N. (2020). When machine learning meets privacy in 6G: a survey. *IEEE Communications Surveys & Tutorials*, 22(4), 2694-2724.
38. Cheu, A.; Smith, A.; and Ullman, J. (2021). Manipulation attacks in local differential privacy. *Proceedings of the 2021 IEEE Symposium on Security and Privacy*. San Francisco, USA, 883-900.
39. Böhler, J.; and Kerschbaum, F. (2020). Secure multi-party computation of differentially private median. *Proceedings of the 29th USENIX Security Symposium*. San Francisco, USA, 2147-2164.
40. Cheng, K.; Tahir, R.; Eric, L.K.; and Li, M. (2020). An analysis of generative adversarial networks and variants for image synthesis on MNIST dataset. *Multimedia Tools and Applications*, 79, 13725-13752.
41. Tayyab, M.; Marjani, M.; Jhanjhi, N.Z.; Hashim, I.A.T.; Almazroi, A.A.; and Almazroi, A.A. (2021). Cryptographic based secure model on dataset for deep learning algorithms. *Computers, Materials and Continua*, 69(1), 1183-1200.
42. Alex, S.A.; Ponkamali, S.; Andrew, T.R.; Jhanjhi, N.Z.; and Tayyab, M. (2022). *Empowering sustainable industrial 4.0 systems with machine intelligence*. Chapter: Machine learning-based wearable devices for smart healthcare application with risk factor monitoring. United States: IGI-Global, 174-185.