# A Robust NFT Assisted Knowledge Distillation Framework for Edge Computing

Nai Wang(✉), Atul Sajjanhar, Yong Xiang, and Longxiang Gao

School of Information Technology, Deakin University, Melbourne 3125, VIC, Australia
{wangnai,atul.sajjanhar,yong.xiang}@deakin.edu.au, Gaolx@sdas.org
https://www.deakin.edu.au

**Abstract.** With the development and improvement in chip manufacturing and network communication, Internet of Things (IoT) have been addressing more and more popularity around these days. Due to the fact that the end devices in an IoT system can perform higher computational tasks, there are more and more IoT applications requiring on-device local training procedures. Hence, the concept of Knowledge Distillation is introduced to solve the on-device machine learning problem–each end device will receive a distilled light-weight student model from the comprehensive central teaching model. However, several security concerns need to be resolved before KD being put into industrial environments, including data integrity and robustness over external attacks. In this paper, we propose an NFT assisted KD framework, aiming at leveraging the blockchain features on data security to solve the intrinsic robustness defects in a naive KD architecture. Our major contributions can be concluded as following 1) the first NFT assisted KD framework (KD-NFT) which initializes the chance of NFT usages in scientific fields; 2) providing a two-dimension (vertical and horizontal) security over KD data vulnerability under attacks; and 3) a fail-over scheme when external poisoning happened, to recovering KD-NFT training process back to last-best status, by using NFT history full-traceable feature and providing automatic system robustness.

**Keywords:** Knowledge Distillation · Federated Learning · Blockchain · NFT · Robustness · Poisoning attack

## 1 Introduction

As the concept of the Internet of Things (IoT) has gained popularity around these days, more and more research and system implementations have become realized with actual people's daily usages. Based on which, one of the high-level concept named Edge Computing has been brought to the front end, leveraging the features of high computational performance and extremely low network latency from modern IoT system [1,2].


S. Yu et al. (Eds.): TridentCom 2022, LNICST 489, pp. 20–31, 2023.
https://doi.org/10.1007/978-3-031-33458-0_2

Moreover, to perform a comprehensive machine learning task throughout an IoT system, the divide-and-conquer method has been taken into account with an edge computing framework: a large task $T$ could be evenly divided into numbers of sub-tasks $T_1, T_2, ..., T_n, n = 1, 2, 3...$, and then assign each sub-task to an end device to perform local training process, and finally a central server collects the sub-training results from each end device, and aggregates them to a comprehensive, global training result as the output from the IoT system [3]. The most famous framework adopts this idea is called Federated Learning [4].

However, a group of defects and weak-points can be identified to the above discuss execution pattern: 1) for a typical machine learning task, a large amount of training dataset is required [5], 2) the centralized architecture makes the central server and end devices are weak to external attacks [6], and 3) the network connectivity is massive between central server and end devices, which will result in tremendous data transmitting latency and increase the chance being attacked [7,8]. Several schemes for Federated Learning framework have been conducted, such as applying GAN framework inside a traditional federated learning framework, to reversely protect the system from attacking. However, the computational and timely consumption is excessively expensive, and not possible to directly impose to an IoT system [6].

Hence, researchers start to explore other ways to protect the Federated Learning, or similar IoT framework, and one of the choices is the Blockchain architecture [9] (Fig. 1).

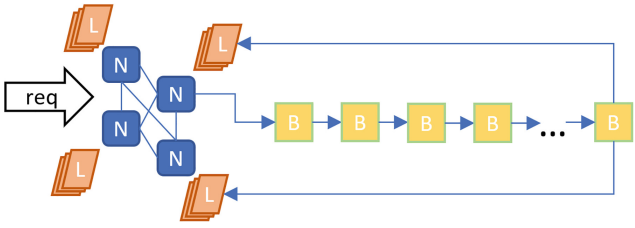


**Fig. 1.** Architecture of blockchain

The core mechanism for a blockchain network is the implementation of ledger–a file that locates on every blockchain network node, recording the copies of blockchain status in real-time and constantly synchronizing the value from one ledger to another node's ledger [10]. Therefore, theoretically it is impossible for an attacker to change every blockchain nodes ledgers with the same poisoning data in the same time, then by the next sync period's end, the blockchain nodes will notice the mistakes in their ledgers recording, then triggering a block reverting process to recover from the last confirmed ledger status [11].

Hence, instead of implementing a complicated and expensive adversary mechanism within an IoT system, the combination of blockchain and IoT concept will be more promising to solve the robustness problem.

To solve the problems in existing federated learning techniques in IoT system usages, in this work, we propose a new edge computing framework that using Knowledge Distillation (KD) concept in contrast to a traditional Federated Learning framework, which addressing the security features from blockchain architecture and leveraging the KD features for edge computing at the same time. Moreover, we focus on one of the blockchain implementations–Non-fungible Token (NFT), as the security method that fixing the intrinsic KD security drawbacks, majorly at the student model distribution processes.

The knowledge Distillation is a new architecture that solves the underperformance issue for edge devices in an IoT system. Typically an IoT system's end device will not have sufficient computational power, which it not possible to fully perform machine learning tasks locally. Hence, the major improvement for KD is, instead of distributing the training data to each of the end device in the IoT systems, the core point for KD is to derive student models from the centralized teacher model, where the teacher model is to be trained in a powerful centralized server with greater computational power [12,13]. However, the student models distributing process from central server to end devices are under external attacks–on receiving a maliciously modified student model, the end devices will incorrectly perform and inject malfunctions to the IoT equipment [14]. Hence, we leverage one of the blockchain implementations-NFT as the method to secure the students distribution process, which secures the student model integrity itself and provides fail-over techniques on attacks happening.

To prove our work, we conduct experiments over MNIST and CIFAR10 datasets showing that our proposed B-FL outperforms the state-of-the-art research works. The contribution of this paper can be summarised as follows:

- We propose a framework that creatively combining NFT security features with Knowledge Distillations (KD-NFT) to solve the security concerns.
- We provide a solution over teacher and student model attacks in a knowledge distillation framework, securing and guaranteeing the end device performance in central and local training processes.
- The KD-NFT supports for a fail-over mechanism when an attack happened throughout models dispatching processes, to secure the end device adopting a non-worse student model, which significantly improve the overall IoT system performing accuracy.
- The experimental results prove that our propose KD-NFT model address the robustness over both communication attacks and data poisoning attacks, when comparing to a naive KD framework.

The rest of this paper is organized as follows. Section 2 introduces the related works on recent blockchain federated learning, knowledge distillation approaches and poisoning attack threats on machine learning. The proposed framework is discussed in Sect. 3 and the experimental results are evaluated in 4. Section 5 gives the conclusion of the paper.

## 2 Related Works

This section provides the background of blockchained federated learning approaches and poisoning attacks on federated learning.

### 2.1 Blockchain and Non-Fungible Token (NFT)

The concept of Non-Fungible Token (NFT) is a high-level of blockchain application, originated from Ethereum blockchain ecosystem from 2014 [15]. By leveraging the blockchain features as its baselines, an NFT extends the usability from its counterpart–Fungible Token (FT), guaranteeing the **uniqueness** of the data an NFT secured on blockchain environment [16,17]. In theory, there are not two same NFTs across the world [18]. Programmably, the NFT complies with ERC-721 and ERC-1155 standards–ERC-1155 allows dividend (a fraction) of an NFT whereas the ERC-721 does not allows [19,20]; and FT is defined within standard ERC-20.

### 2.2 Blockchained Federated Learning

To protect external attacks, there are numbers of works which implement their blockchain assisted federated learning frameworks. The work [10] attach each client device to a blockchain node to achieve sufficient distribution. However, the time consumption is hard to accept. The work [21] proposed a BC-based PPFL framework with five blockchain node, each time a client producing a local training model, it will trigger the generation of a new blockchain block. The work [22] proposed a hybrid chain named PermiDAG and the work [23] proposed a blockchained federated learning conceptual framework to be used for Industry 4.0.

However, none of the above work and framework shows their robustness over data poisoning and external attacks.

### 2.3 Knowledge Distillation Security

Knowledge Distillation (KD) is the concept that addresses the model compressing and cost-balancing over large machine learning model to run on the small devices [24]. The goal is to use a comprehensive teacher model to generate a number of student models which are more light-weighted and retaining the most of the teacher model's features and effectiveness. However, there are a few identified security concerns that need to be considered: 1) teacher model training process can be attacked externally [25], which affects all following model distillation and student training processes [26], and 2) when the teacher model distilled in central server, there are risks that the student models are being poisoned along the channel dispatching to individual end devices [27].

### 2.4 Poisoning Attacks on Machine Learning

There are several identified poisoning schema to attack a machine learning framework.

The work [8] applied the constrain-and-scale technique to change the data and submit it to the server, which dramatically reduce the system overall effectiveness because of the change in global model.

The work [5] aimed to launch the poisoning attack without invading any clients in federated learning. Attackers act like the benign clients in the federated learning and deploy a GAN to reconstruct the data from the shared global model, and then flip the labels to initialize the poisoning attack.

The work [28] exploited the lack of transparency in federated learning and control a small number of attackers to perform a model poisoning attack.

The work [29] proposed a distributed backdoor attack (DBA), which decomposes a global trigger pattern into different local patterns, then embeds the local patterns into different adversarial parties.

All the works show the fact that attacks over the global model will result in more severe result than the attacks happened to end devices/ clients.

## 3 Framework of KD-NFT

This section presents the inter-structure of the proposed KD-NFT framework and unveils the inner connectivity regarding layers of the knowledge distillation and web3 NFT architectures.

### 3.1 KD-NFT Framework

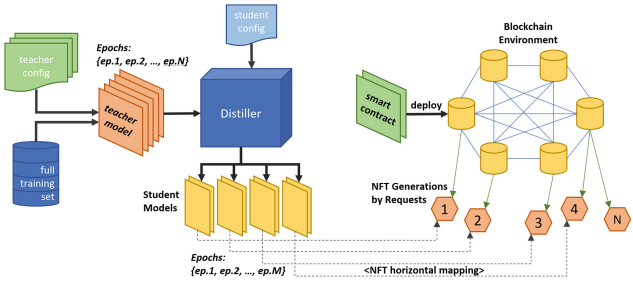


**Fig. 2.** The Architecture of KD-NFT

The Fig. 2 shows how the proposed framework looks like in general. There are two major parts 1) data training and knowledge distillation infrastructure located on the left part, and 2) blockchain and web3 NFT controllers are located on the right. In this paper, we do not explore much on the complication of KD

architecture, because we address more to leverage blockchain architecture to secure KD's data security and robustness over data poisoning.

For the model training and distillation part, we use a basic CNN model which is compatible with model distiller. We split the teacher and student training processes into a granularity of epochs–on hitting the number of training epoch, the framework will trigger a blockchain activity, namely create a unique NFT for the current training procedure before it goes under any potential external attacks. Moreover, the framework will test on if the training processes produce non-worse training result before creating the new NFT for each of them. The basic algorithm is described in algorithm below (Fig. 3).

**Algorithm: Student Failover Training Epoch**

***pre-requisite:***
StudentModel = {$M_{s1}$, $M_{s2}$, …, $M_{sm}$} for m student models, where
M = {$m_0$, $m_1$, $m_2$, …, $m_e$} for e number of training epochs and $m_0$ as init model,
M[j] = $m_j$.
EpochValidator = {$V_1$, $V_2$, …, $V_m$} to validate corresponding student model.
StudentData = {$D_1$, $D_2$, …, $D_m$} to be used for each student's training
ValidateData = {$D_{v1}$, $D_{v2}$, …, $D_{vm}$}to be used for student epochs validating.
Abstracted web3 smart contract service web3.

***algorithm:***
for teacher model training process with epoch k:
  lastBestStudentModel = LocalStorage.load('teacher_model',
web3.getLastBestTokenId(i, 'student'))

for each student (i) config and training process with epoch j:
  lastBestStudentModel = LocalStorage.load('student_model',
web3.getLastBestTokenId(i, 'student')), or $M_{si}$[j-1] if there was no local stored model.

  $M_{si}$[j] = TrainingProcessi(lastBestStudentModel, $D_i$), j = 1, 2, …
  $Hash_{ij}$ = HashFunction($M_{si}$[j]);
  $Accuracy_{ij}$ = Validator$_i$($M_{si}$[j] , $D_{vi}$);

  if (web3.getLatestModelAccuracy(i, 'student') < $Accuracy_{ij}$) then {
    studentTokenId = currentTimeStamp
    web3.createNFT(studentTokenId, i, 'student')
    LocalStorage.save('student_model', studentTokenId, $M_{si}$[j])
  }

**Fig. 3.** The overall algorithm with NFT generating policies

The above algorithm provides an example that when the training threshold to create a new NFT is set as 1 epoch. In the following sections, we provide more info regarding the blockchain smart contract implementation and how the system guarantees proposed robustness.

### 3.2 Web3 NFT Implementation

In this work, we choose the Etherum (Eth) blockchain environment as the decentralized services provider. Although the Eth network would trigger latency issue

at the current settings, its property in public trust-worth and worldwide acceptance are without any argument, when comparing to other implemented private chains.

Fundamental logics are implemented in the smart contract (SC), which is written in solidity and directly deployed on public Polygon Mumbai testnet through Remix web IDE. After the success of the SC deployment, there will not be another chance to make changes into the SC, because the Mumbai testnet is one of the Etherum compatible networks and once SC deployed. The SC will be confirmed and lodged by all Etherum EVMs (virtual machines that running Etherum sync logics), where the SC state modification can only be performed through secure web3 calling and in-SC function calling. To trigger a secure SC function calling, it requires the wallet private key and the wallet has to pay for a small amount of gas fee to make sure the transaction can be confirmed by the blockchain.

**NFT vertical data structure**

```
{
    tokenId: uint256;
    entityId: uint256;
    timestamp: uint256;
    type: string;
    storageUrl: string;
    originatedTokenId: string;

    accuracy: string;
    modelHashValue: string;
}
```

**NFT horizontal data structure**

```
{
    tokenId: uint256;
    entityId: uint256;
    timestamp: uint256;
    type: string;
    storageUrl: string;

    associatedTokenId: string;
    accuracy: string;
    modelHashValue: string;
}
```

**NFT aggregated data structure**

```
{
    tokenId: uint256;
    entityId: uint256;
    timestamp: uint256;
    type: string;
    storageUrl: string;
    originatedTokenId: string;
    associatedTokenId: string;
    accuracy: string;
    modelHashValue: string;
    isValid: bool;
}
```

**Fig. 4.** NFT data structure on smart contract

Each time the SC receiving a request to create a new NFT for the training epoch, it will require the KD part to provide the current validated model accuracy, and only the non-worse model's creation request would be accepted by the blockchain, otherwise the SC will discard the requesting model. In the case of starting the training process, the KD part will query on the blockchain to gain the latest entities' (either teacher model or student model) model valid id, then the KD part will fetch the model specified by blockchain returned id to continue next training epoch.

Above Fig. 4 illustrates how each NFT are structured on SC. The vertical one specifies all attributes that required by teacher training process, and the horizontal one is for students training processes. And finally we aggregate the both attributes needs together to form up the general NFT data structure on SC.

### 3.3 Validation on KD-NFT's Robustness

By understanding the basic framework settings and how the NFT interacting with the training architecture, in this section we are going to discuss what kind of robustness can KD-NFT provide to actual use cases.

– **Attacks to teacher training process** There are two aspects that guarantees the robustness over teacher model training process: 1) while the trainer loading the stored previous best teacher model, it would first ask blockchain the model id to specify. Hence, always the best model is loaded to continuously train on.2) when the training epoch threshold is hit, the temporary latest teacher model is going to be uploaded to the blockchain. Only the non-worse teacher model is going to be accepted and the blockchain generates the unique NFT for the current teacher model. Hence, the non-worse policy in uploading and reloading procedures prevent the under attacking teacher model to affect the blockchained model's accuracy, and teacher model trainer can always train on the best stored model by far.

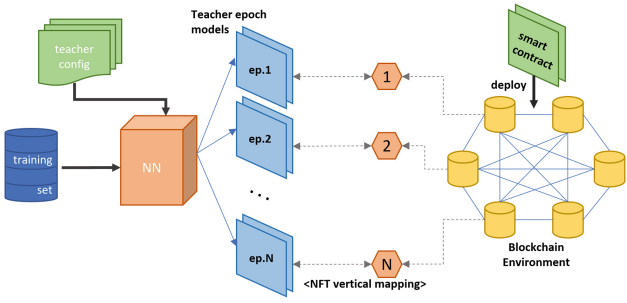


**Fig. 5.** The Architecture of KD-NFT

– **Attacks to student training process** The non-worse policy for teacher model training process can be totally adopted by the student training epochs. The difference would lie in the student models need to specify their unique student entity ids, to let blockchain distinguish a student model away from other student models teacher models.
– **Attacks to distilled teacher model and passing to end devices** The only pathway for an end device to receive a student model is from blockchain. By providing the entity id to calling the SC function, the student model can consequently downloaded. To activate the SC call function, wallet private key is needed, which is secured by the framework owner and strong enough from being hacked externally (Fig. 5).

Hence, the frameworks robustness is provided by the combination of the three identified aspects, which has a very low probability to be attacked and fail simultaneously.

# 4 Experiment Evaluation

## 4.1 Experiment Preparation

To evaluate the proposed framework's performance, we conduct our experiments and corresponding evaluations on two of the benchmark image recognition datasets MNIST and CIFAR10 datasets. All the experiments are running on a single machine with multiple GPU attached ($12 \times 3070$ and $1 \times 1070$ti). As comparison, we perform the similar evaluation as what has been done in work [30] with the same CNN settings, shown in the Fig. 6.

| MNIST Architecture | | CIFAR10 Architecture | |
|---|---|---|---|
| Relu Convolutional | 32 filters (3×3) | Relu Convolutional | 96 filters (3×3) |
| Relu Convolutional | 32 filters (3×3) | Relu Convolutional | 96 filters (3×3) |
| Max Pooling | 2×2 | Relu Convolutional | 96 filters (3×3) |
| Relu Convolutional | 64 filters (3×3) | Max Pooling | 2×2 |
| Relu Convolutional | 64 filters (3×3) | Relu Convolutional | 192 filters (3×3) |
| Max Pooling | 2×2 | Relu Convolutional | 192 filters (3×3) |
| Relu Convolutional | 200 units | Relu Convolutional | 192 filters (3×3) |
| Relu Convolutional | 200 units | Max Pooling | 2×2 |
| Softmax | 10 units | Relu Convolutional | 192 filters (3×3) |
| | | Relu Convolutional | 192 filters (1×1) |
| | | Relu Convolutional | 192 filters (1×1) |
| | | Global Avg. Pooling | |
| | | Softmax | 10 units |

**Fig. 6.** CNN experimental settings for Knowledge Distillation part

## 4.2 Accuracy

The work [30] conducted multiple attack schema, which were model dependent and not easy to be replicated. Instead, we calculate for their averaged accuracy over they proposed individual attacking scheme for comparison.

**Table 1.** Comparison of Accuracy Drops

| Model | MNIST | MNIST.atk | CIFAR10 | CIFAR10.atk |
|---|---|---|---|---|
| KD-NFT | 98.8 | 93.34 | 95.59 | 88.92 |
| Model [30] | 98.82 | 89.87 | 95.61 | 84.79 |

The experiments are both run for 500 epochs (400 teacher training epochs and 100 student training epochs) and arbitrarily impose 200 number of different attacks throughout the whole training process. Among all types of attacks, we mimic the most common two types 1) data removal and 2) data replacing in trained intermediate models. The Accuracy shown in the Table 1 are aggregated and averaged at the end devices ends. The table of accuracy shows the fact that over 500 number of all training epochs, our proposed model are resulted in better training overall accuracy than the compared model [30]. However, the table does

not show that the compared model are less of performance, because in their work they identified five different types of attacks under the two types of attacks that we mimic, and in this paper only the averaged value from their running results are considered to be compared with ours. As a result, the running results show that our proposed KD-NFT can produce a non-worse than results as its competitors.

### 4.3 Robustness over Data Poisoning Attacks

**Table 2.** Comparison of Accuracy Drops in 500 training epochs

| Attack number | MNIST | Diff | CIFAR10 | Diff |
|---|---|---|---|---|
| 0 | 98.8 | – | 95.59 | – |
| 10(t) | 98.01 | +0.08 | 95.07 | +0.05 |
| 10(s) | 98.22 | +0.29 | 95.31 | + 0.29 |
| 10(t,s) | 97.93 | 0 | 95.02 | 0 |
| 50(t) | 97.24 | +0.13 | 93.67 | +0.16 |
| 50(s) | 97.88 | + 0.77 | 94.12 | +0.61 |
| 50(t,s) | 97.11 | 0 | 93.51 | 0 |
| 100(t) | 95.92 | +0.19 | 91.32 | +0.65 |
| 100(s) | 96.69 | +0.96 | 92.13 | +1.46 |
| 100(t,s) | 95.73 | 0 | 90.67 | 0 |
| 200(t,s) | 93.34 | – | 88.92 | – |
| 300(t,s) | 91.07 | – | 86.98 | – |

Moreover, we conduct a set of experiments with different attacking schema applied to our framework, over 500 teacher and student training epochs (400 teacher epochs and 100 student epochs). In the Table 2 first column, (t) represents for only attacking on teacher model and (s) represents for student models, (t, s) represents both teacher and student models are under attacks. From the running result, we can conclude following facts:

- The number of attacks negatively affects the framework's overall accuracy. However the results for 300 attacks out of 500 training epochs are still acceptable over different datasets–above 0.91 accuracy for MNIST dataset and nearly 0.87 for CIFAR10 dataset.
- Within the same number of attacks, teacher model's attack affect more negatively than student models attacks to the final running accuracy, which indicates the guarantees on protection and robustness over teacher model are of more importance than student models–one attack on teacher model is equal to 3-4 attacks on student models in downgrading the overall framework accuracy.

## 5 Conclusion

This paper proposes a robust NFT secured Knowledge Distillation framework (KD-NFT). It is the first framework that addresses the security features over Non-Fungible Token into machine learning fields, and provides a solution in recover the training procedure by leveraging the blockchain features. The experiments show the proposed framework can produce decent effectiveness over both model accuracy and robustness when attacks happened.

## References

1. Sachs, J., et al.: Adaptive 5G low-latency communication for tactile internet services. Proc. IEEE **107**(2), 325–349 (2018)
2. Shalf, J.: The future of computing beyond Moore's law. Phil. Trans. Royal Soc. A **378**(2166), 20190061 (2020)
3. Chen, J., Ran, X.: Deep learning with edge computing: a review. Proc. IEEE **107**(8), 1655–1674 (2019)
4. Yang, Q., Liu, Y., Chen, T., Tong, Y.: Federated machine learning: concept and applications. ACM Trans. Intell. Syst. Technol. (TIST) **10**(2), 1–19 (2019)
5. Zhang, J., Chen, J., Wu, D., Chen, B., Yu, S.: Poisoning attack in federated learning using generative adversarial nets. In: 2019 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/13th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), pp. 374–380. IEEE (2019)
6. Zhao, Y., Chen, J., Zhang, J., Wu, D., Blumenstein, M., Yu, S.: Detecting and mitigating poisoning attacks in federated learning using generative adversarial networks. Concurr. Comput. Pract. Exp. **34**, e5906 (2020)
7. Konečnỳ, J., McMahan, H.B., Yu, F.X., Richtárik, P., Suresh, A.T., Bacon, D.: Federated learning: strategies for improving communication efficiency. arXiv preprint arXiv:1610.05492 (2016)
8. Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., Shmatikov, V.: How to backdoor federated learning. In: International Conference on Artificial Intelligence and Statistics, pp. 2938–2948 (2020)
9. Zheng, Z., Xie, S., Dai, H.-N., Chen, X., Wang, H.: Blockchain challenges and opportunities: a survey. Int. J. Web Grid Serv. **14**(4), 352–375 (2018)
10. Kim, H., Park, J., Bennis, M., Kim, S.-L.: Blockchained on-device federated learning. IEEE Commun. Lett. **24**(6), 1279–1283 (2019)
11. Lu, Y., Huang, X., Dai, Y., Maharjan, S., Zhang, Y.: Blockchain and federated learning for privacy-preserved data sharing in industrial IoT. IEEE Trans. Ind. Inf. **16**(6), 4177–4186 (2019)
12. Heo, B., Lee, M., Yun, S., Choi, J.Y.: Knowledge distillation with adversarial samples supporting decision boundary. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 3771–3778 (2019)
13. Shao, R., Yi, J., Chen, P.-Y., Hsieh, C.-J.: How and when adversarial robustness transfers in knowledge distillation? arXiv preprint arXiv:2110.12072 (2021)
14. Wang, H., Deng, Y., Yoo, S., Ling, H., Lin, Y.: AGKD-BML: defense against adversarial attack by attention guided knowledge distillation and bi-directional metric learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 7658–7667 (2021)

15. Wang, Q., Li, R., Wang, Q., Chen, S.: Non-fungible token (NFT): overview, evaluation, opportunities and challenges. arXiv preprint arXiv:2105.07447 (2021)
16. Chohan, U.W.: Non-fungible tokens: blockchains, scarcity, and value. In: Critical Blockchain Research Initiative (CBRI) Working Papers (2021)
17. Dowling, M.: Is non-fungible token pricing driven by cryptocurrencies? Finan. Res. Lett. **44**, 102097 (2022)
18. Ante, L.: The non-fungible token (nft) market and its relationship with bitcoin and ethereum. FinTech **1**(3), 216–224 (2022)
19. Pirker, D., Fischer, T., Witschnig, H., Steger, C.: Velink-a blockchain-based shared mobility platform for private and commercial vehicles utilizing erc-721 tokens. In: 2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP), pp. 62–67. IEEE (2021)
20. Kim, M., Hilton, B., Burks, Z., Reyes, J.: Integrating blockchain, smart contract-tokens, and iot to design a food traceability solution. In: IEEE 9th annual information technology, electronics and mobile communication conference (IEMCON), pp. 335–340. IEEE (2018)
21. Awan, S., Li, F., Luo, B., Liu, M.: Poster: a reliable and accountable privacy-preserving federated learning framework using the blockchain. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, pp. 2561–2563 (2019)
22. Lu, Y., Huang, X., Zhang, K., Maharjan, S., Zhang, Y.: Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles. IEEE Trans. Veh. Technol. **69**(4), 4298–4311 (2020)
23. Qu, Y., Pokhrel, S.R., Garg, S., Gao, L., Xiang, Y.: A blockchained federated learning framework for cognitive computing in industry 4.0 networks. IEEE Trans. Ind. Inf. (2020)
24. Gou, J., Yu, B., Maybank, S.J., Tao, D.: Knowledge distillation: a survey. Int. J. Comput. Vision **129**(6), 1789–1819 (2021)
25. Zhang, Z., Wu, T.: Learning ordered top-k adversarial attacks via adversarial distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 776–777 (2020)
26. Maroto, J., Ortiz-Jiménez, G., Frossard, P.: On the benefits of knowledge distillation for adversarial robustness. arXiv preprint arXiv:2203.07159 (2022)
27. Yoshida, K., Fujino, T.: Disabling backdoor and identifying poison data by using knowledge distillation in backdoor attacks on deep neural networks. In: Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security, pp. 117–127 (2020)
28. Bhagoji, A.N., Chakraborty, S., Mittal, P., Calo, S.: Analyzing federated learning through an adversarial lens. In: International Conference on Machine Learning, pp. 634–643 (2019)
29. Xie, C., Huang, K., Chen, P.-Y., Li, B.: Dba: distributed backdoor attacks against federated learning. In: International Conference on Learning Representations (2019)
30. Mirzaeian, A., Kosecka, J., Homayoun, H., Mohsenin, T., Sasan, A.: Diverse knowledge distillation (dkd): a solution for improving the robustness of ensemble models against adversarial attacks. In: 22nd International Symposium on Quality Electronic Design (ISQED), PP. 319–324. IEEE (2021)