Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI-23)
Journal Track

SAMBA: A Generic Framework for Secure Federated Multi-Armed Bandits (Extended Abstract)*

Radu Ciucanu[1], **Pascal Lafourcade**[2], **Gael Marcadet**[2] and **Marta Soare**[3]
[1]INSA Centre Val de Loire, LIFO EA 4022, France
[2]Université Clermont Auvergne, LIMOS UMR 6158, France
[3]Université d'Orléans, Université Grenoble Alpes, LIG UMR 5217, France
radu.ciucanu@insa-cvl.fr, {pascal.lafourcade, gael.marcadet}@uca.fr, marta.soare@univ-orleans.fr

Abstract

We tackle the problem of secure cumulative reward maximization in multi-armed bandits in a cross-silo federated learning setting. Under the orchestration of a central server, each data owner participating at the cumulative reward computation has the guarantee that its raw data is not seen by some other participant. We rely on cryptographic schemes and propose SAMBA, a generic framework for Secure federAted Multi-armed BAndits. We show that SAMBA returns the same cumulative reward as the non-secure versions of bandit algorithms, while satisfying formally proven security properties. We also show that the overhead due to cryptographic primitives is linear in the size of the input, which is confirmed by our implementation.

1 Introduction

Federated learning is a machine learning paradigm where multiple data owners collaborate in solving a learning problem, under the coordination of a central orchestration server [Kairouz and et al., 2021]. Each data owner's raw data is stored locally and not exchanged or transferred. The development of machine learning algorithms in federated learning settings is a timely topic, which touches several communities: "*a longstanding goal pursued by many research communities (including cryptography, databases, and machine learning) is to analyze and learn from data distributed among many owners without exposing that data*" [Kairouz and et al., 2021]. We tackle this goal by relying on cryptographic techniques to develop a secure framework for learning on distributed data.

In particular, we focus on multi-armed bandits, a reinforcement learning model where a learning agent needs to sequentially decide which "arm" to choose among several options (with unknown reward distributions) available in the environment. After each arm selection, the environment responds with a stochastic reward drawn from the reward distribution associated to the chosen arm. To maximize the cumulative reward, the learning agent has to continuously face the so-called exploration-exploitation dilemma and decide whether to *explore* by choosing arms with more uncertain associated values, or to *exploit* the information already acquired by choosing the arm with the seemingly largest associated value. Bandits have practical applications such as Web recommender systems, where the arms are the recommended items and the rewards are given by the user ratings. More specifically, we tackle the problem of *secure cumulative reward maximization in federated multi-armed bandits*, a problem that to the best of our knowledge has not been previously studied in the literature. Our goal is to propose a generic federated framework that is guaranteed to return exactly the same cumulative reward as standard bandit algorithms [Sutton and Barto, 2018, Chapter 2], while guaranteeing formally proven security properties.

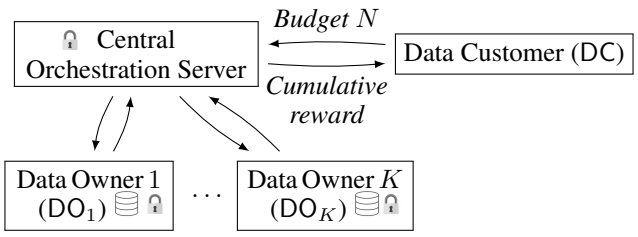

Figure 1: Federated cumulative reward maximization in multi-armed bandits.

2 Related Work on Federated Secure Bandits

Federated multi-armed bandits is an emerging topic, with few recent works that consider the federated learning paradigm for sequential decision making problems, where data is observed in response to interactions with an unknown environment. At each time step, the learner has only limited feedback about the arm that is pulled and this makes the setting more challenging compared to the typical supervised learning scenarios, where all training data is available from the beginning of the learning process. The recent works tackling federated bandits, consider different models: standard stochastic [Shi and Shen, 2021; Li *et al.*, 2020], bandits with graph structure [Zhu *et al.*, 2021], and linear bandits [Dubey and Pentland, 2020; Huang *et al.*, 2021]. For all these works, the main focus is on adapting bandit algorithms to the federated set-

*This extended abstract summarizes our full paper published in the Journal of Artificial Intelligence Research (JAIR), 73:737–765, 2022. https://doi.org/10.1613/jair.1.13163.

ting, and some of them additionally rely on differential privacy [Dwork and Roth, 2014] to protect the data.

In particular, the first works on cumulative reward maximization in (private) federated multi-armed bandits [Shi and Shen, 2021; Li *et al.*, 2020; Zhu *et al.*, 2021; Shi *et al.*, 2021] focus on the analysis of the gain in sharing data coming from multiple DOs for obtaining better *local* (DO-specific) and respectively *global* cumulative rewards (for all participants in the federated learning process). The typical assumption is that all DOs have access to the same subset of arms, which corresponds to an horizontal data partition. Another typical assumption from all these works is that the DOs *exchange information about the rewards they observe and about the indices of their selected arms* with their neighbors [Li *et al.*, 2020; Zhu *et al.*, 2021], respectively with the central orchestration server [Shi and Shen, 2021; Zhu *et al.*, 2021]. Before sharing these pieces of information, DOs apply differential privacy mechanisms to inject noise in their local data to keep it private from the other participants. For the next time steps, the bandit algorithm will continue to select arms based on the differentially-private information that is transmitted between participants.

A differentially-private bandit algorithm takes roughly the same computation time as the standard algorithm, but because of the noise that is injected in the data to ensure differential privacy, the arm selection strategy is altered. Thus, the modified selection strategy leads to a different output and a reduced performance (increased regret) compared to that of the standard bandit algorithm. On the other hand, in a cryptographic approach, the local data of each DO (concerning e.g., their observed rewards) is never exchanged in clear: encryption techniques are used to guarantee that local data maintained by each DO is hidden from the other participants. By relying on a carefully chosen set of primitives (AES-GCM and Paillier in the case of SAMBA), cryptographic approaches do not change the arm selection and output the same result as the standard algorithm, at the price of an increased computation time due to the use of cryptographic primitives. Although we share the common goal of data protection in federated bandits, the use of different techniques (differential privacy in the related works vs cryptography in our work) leads to complementary systems, whose different architecture and trade-offs are not comparable. In addition, in contrast with previous federated multi-armed bandit frameworks for cumulative reward maximization, we focus on a vertical data partition and our secure framework guarantees that local data maintained by each DO is hidden from the other participants.

There exist only a few cryptography-based secure protocols for bandits, in settings where all data is outsourced to the honest-but-curious cloud [Ciucanu *et al.*, 2019; Ciucanu *et al.*, 2020a; Ciucanu *et al.*, 2020b] and no other work proposing cryptography-based secure protocols for federated bandits. The protocol that is the closest to SAMBA also considers the problem of secure cumulative reward maximization for standard stochastic bandits [Ciucanu *et al.*, 2020b]. There are two main differences between the protocol in [Ciucanu *et al.*, 2020b] and SAMBA. (i) The data distribution assumptions are different: in [Ciucanu *et al.*, 2020b] it is assumed that all data is outsourced to the cloud, whereas SAMBA focuses on a federated learning setting where data is stored locally by each owner and never exchanged. Consequently, the respective distributed architectures are intrinsically different. (ii) The protocol in [Ciucanu *et al.*, 2020b] is catered for securing the UCB algorithm, whereas SAMBA is a generic framework where multiple bandit algorithms can be easily plugged in. Among the algorithms supported in SAMBA, we have UCB and similar argmax-based algorithms, as well as algorithms where arms are pulled based on probability matching.

3 Problem Formulation

As depicted in Figure 1, we assume that the *data* i.e., the reward functions associated to K bandit arms are stored locally by K *data owners* ($\mathsf{DO}_1, \ldots, \mathsf{DO}_K$). The data is potentially sensitive, hence it should remain stored locally and cannot be seen in clear by any participant other than its owner (this is why we depict locks near each DO_i). As typically done in federated learning, we assume that the learning algorithm is done by some *central orchestration server* (referred to as *server* in the sequel). The *data customer* (DC) sends a budget N to the server and receives the cumulative reward. Moreover, we assume that the participants in Figure 1 (data owners, server, and data customer) are *honest-but-curious* i.e., they correctly do the required computations, but try to gain as much information as possible based on the data that they see. In particular, we aim at minimizing the data leakage to the server (this is why we also depict a lock near the server) e.g., the server cannot see rewards produced by each data owner. Additionally, an external observer that has access to all messages exchanged between the aforementioned participants should not be able to learn any input, output, or intermediate data.

Our aim is to build a generic federated learning framework such that, given some standard bandit algorithm $\mathcal{A}$, we are able to plug $\mathcal{A}$ in our framework and obtain the same cumulative reward as $\mathcal{A}$, while guaranteeing data security. Our goal could be theoretically achieved by relying on a fully homomorphic encryption (FHE) scheme [Gentry, 2009], which allows to compute any function directly in the encrypted domain. Indeed, in theory it would suffice that each data owner encrypts its data with a FHE scheme; then, the server would do the computations needed for cumulative reward maximization directly in the encrypted domain. However, it remains an open question how to build a practical FHE system. Although state-of-the-art FHE systems (SEAL[1] and HElib[2]) have done remarkable progress, computations with real numbers are still limited because of the noise needed for FHE multiplications. Moreover, even simple functions such as comparisons needed in all bandit algorithms (e.g., compute an argmax or a probability matching) require complex and time-consuming computations in FHE systems, even for approximate results and even for recent state-of-the-art algorithms [Cheon *et al.*, 2020; Garcelon *et al.*, 2022]. Since FHE systems cannot be currently used off-the-shelf to propose secure federated bandit

[1]https://github.com/Microsoft/SEAL

[2]http://homenc.github.io/HElib/

Data \ Participant	DO_i	DC	Server		Ext
			Comp	Controller	
Cumulative reward		X		$\mathcal{E}$	$\mathcal{E}$
Sum of rewards and number of pulls for DO_i	X		α_t	$\mathcal{E}$	$\mathcal{E}$
Sum of rewards and number of pulls for $\mathsf{DO}_{j\neq i}$			α_t	$\mathcal{E}$	$\mathcal{E}$
Arm pulled at time step t	X*		σ_t	Enc	Enc
Reward at time step t	X*				

Figure 2: *Security properties of* SAMBA. The X means that the participant can see in clear the concerned piece of data, with $\star$ = only if DO_i is pulled at time step t. Ext means an external network observer having access to all messages exchanged between participants. In the cells without X or X*, we indicate the technique that we used to prevent the participant from seeing in clear the concerned data: Paillier encryption ($\mathcal{E}$), AES-GCM encryption (Enc), random masks (α_t), and random permutations (σ_t). A grayed cell means that the concerned participant does not see any message about the concerned piece of data.

algorithms, our approach is based on *simpler cryptographic schemes*, in conjunction with *secure multi-party computation*.

4 SAMBA in a Nutshell

We now present SAMBA, a generic framework for secure cumulative reward maximization for federated bandits. The key ingredients of SAMBA are:

- We distribute the server computations between two nodes: Controller (that sees only encrypted messages and distributes computation tasks among participants) and Comp (whose only goal is to compare numbers obtained after permuting and masking bandit arm scores). This distribution technique allows to perform comparisons, without revealing to the server neither the bandit arm scores nor the arm pulled at some time step.
- We exchange only encrypted messages such that an external network observer cannot learn any input, output, or intermediate data. Moreover, each data owner can see in clear the raw data pertaining to its bandit arm and nothing else. The data owners communicate only with Controller, with messages encrypted with *indistinguishable under chosen-plaintext attack (IND-CPA)* cryptographic schemes, namely symmetric AES-GCM [National Institute of Standards and Technology, 2001; National Institute of Standards and Technology, 2007] and asymmetric [Paillier, 1999].
- At the end of SAMBA, we compute the cumulative reward by summing up the rewards from each data owner directly in the encrypted domain, by relying on the additive homomorphic property of Paillier. Hence, neither the data owners nor the server nodes can see in clear the cumulative reward: only the data customer that invested a budget for computing the cumulative reward is able to decrypt it.

In the full paper [Ciucanu *et al.*, 2022], we instantiate SAMBA to secure five bandit algorithms: ε-greedy, UCB, Thompson Sampling, Softmax, and Pursuit. We also provide the theoretical analysis of SAMBA, as well as experiments[3] that support our theoretical findings. In a nutshell, we show that SAMBA enjoys the following features:

- *Genericity*: SAMBA can be instantiated with any bandit algorithm that satisfies the properties (i) computing the score of an arm does not depend on the other arms, and (ii) selecting the arm to be pulled at some round can be done in the presence of some random masks and permutations on which SAMBA relies to hide the real arm scores. In particular, the five aforementioned bandit algorithms satisfy these properties. We also include examples of bandit algorithms that cannot be instantiated in SAMBA: an existing algorithm (Reinforcement Comparison) that cannot be instantiated because of (i), and an hypothetical algorithm that cannot be instantiated because of (ii) as we are not aware of any off-the-shelf algorithm that does not satisfy (ii).
- *Correctness*: SAMBA returns exactly the same cumulative reward as the standard (non-secure and non-federated) bandit algorithms because the cryptographic primitives and distribution of tasks do not change the arm selection strategy w.r.t. the standard algorithms.
- *Security*: we summarize the *security properties* in Figure 2. We give a brief intuition for each participant:
  - DO_i can see data concerning arm i and nothing else about other arms, nor about the cumulative reward.
  - Only DC can see the cumulative reward for which she spends a budget. She can see only this piece of information for which she pays, and nothing else.
  - The server nodes (Controller and Comp) and external observers cannot learn any input, output, and intermediate data.
- *Complexity*: the number of cryptographic operations is linear in the input: SAMBA uses $O(NK)$ AES-GCM operations and $O(K)$ Paillier operations. It is a desirable feature that the number of Paillier operations does not depend on the budget N because N is typically larger than the number of arms K, and AES-GCM is much faster than Paillier.

In addition to the fundamental contributions presented in the full paper [Ciucanu *et al.*, 2022], we also implemented a complementary SAMBA system demonstration [Marcadet *et al.*, 2022] that is based on a Web interface simulating the SAMBA federated components. The user-friendly SAMBA Web interface is open source[4] and allows data scientists to configure the end-to-end workflow of deploying a federated bandit algorithm, by examining the interaction between three key dimensions of federated bandits: cumulative reward,

[3] https://github.com/gamarcad/paper-samba-code

[4] https://github.com/gamarcad/samba-demo

computation time, and security guarantees.

5 Conclusion and Future Work

We proposed SAMBA, a generic secure protocol that is able to easily transform multi-armed bandit algorithms in their secure federated version, while yielding the exact same cumulative reward as their standard (non-secure non-federated) version. To achieve SAMBA's security properties, we rely on secure multi-party computations and cryptographic schemes under the honest-but-curious threat model. Through theoretical analysis and experiments, we show that the cryptographic overhead implied by SAMBA is linear in the size of the input, and thus remains reasonable in practice.

We plan to extend SAMBA such that it provides security guarantees in more complex threat models and for more complex federated multi-armed bandit and federated reinforcement learning frameworks (such as the one considered in [Tzamaras *et al.*, 2022]). More in general, using cryptography to ensure data security for machine learning algorithms is a promising, timely direction. We plan to pursue this direction and to design secure protocols useful for other machine learning models and applications.

Acknowledgments

This work has been partially supported by MIAI@Grenoble Alpes (ANR-19-P3IA-0003), two projects funded by EU Horizon 2020 research and innovation programme (TAILOR under GA No 952215 and INODE under GA No 863410), and the French BPI project D4N.

References

[Cheon *et al.*, 2020] J. H. Cheon, D. Kim, and D. Kim. Efficient Homomorphic Comparison Methods with Optimal Complexity. In *ASIACRYPT*, pages 221–256, 2020.

[Ciucanu *et al.*, 2019] R. Ciucanu, P. Lafourcade, M. Lombard-Platet, and M. Soare. Secure Best Arm Identification in Multi-Armed Bandits. In *International Conference on Information Security Practice and Experience (ISPEC)*, pages 152–171, 2019.

[Ciucanu *et al.*, 2020a] R. Ciucanu, A. Delabrouille, P. Lafourcade, and M. Soare. Secure Cumulative Reward Maximization in Linear Stochastic Bandits. In *International Conference on Provable and Practical Security (ProvSec)*, pages 257–277, 2020.

[Ciucanu *et al.*, 2020b] R. Ciucanu, P. Lafourcade, M. Lombard-Platet, and M. Soare. Secure Outsourcing of Multi-Armed Bandits. In *IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 202–209, 2020.

[Ciucanu *et al.*, 2022] R. Ciucanu, P. Lafourcade, G. Marcadet, and M. Soare. SAMBA: A Generic Framework for Secure Federated Multi-Armed Bandits. *Journal of Artificial Intelligence Research (JAIR)*, 73:737–765, 2022.

[Dubey and Pentland, 2020] A. Dubey and A. Pentland. Differentially-Private Federated Linear Bandits. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[Dwork and Roth, 2014] C. Dwork and A. Roth. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends in Theoretical Computer Science*, 2014.

[Garcelon *et al.*, 2022] E. Garcelon, V. Perchet, and M. Pirotta. Encrypted Linear Contextual Bandit. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2022.

[Gentry, 2009] C. Gentry. Fully Homomorphic Encryption Using Ideal Lattices. In *Symposium on Theory of Computing (STOC)*, pages 169–178, 2009.

[Huang *et al.*, 2021] R. Huang, W. Wu, J. Yang, and C. Shen. Federated Linear Contextual Bandits. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

[Kairouz and et al., 2021] P. Kairouz and et al. Advances and Open Problems in Federated Learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021.

[Li *et al.*, 2020] T. Li, L. Song, and C. Fragouli. Federated Recommendation System via Differential Privacy. In *International Symposium on Information Theory (ISIT)*, pages 2592–2597, 2020.

[Marcadet *et al.*, 2022] G. Marcadet, R. Ciucanu, P. Lafourcade, M. Soare, and S. Amer-Yahia. SAMBA: A System for Secure Federated Multi-Armed Bandits. In *IEEE International Conference on Data Engineering (ICDE)*, pages 3154–3157, 2022.

[National Institute of Standards and Technology, 2001] National Institute of Standards and Technology. Advanced Encryption Standard (AES). https://doi.org/10.6028/NIST.FIPS.197-upd1, 2001. Accessed: 2023-05-11.

[National Institute of Standards and Technology, 2007] National Institute of Standards and Technology. Recommendation for BlockCipher Modes of Operation:Galois/Counter Mode (GCM) and GMAC. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf, 2007. Accessed: 2023-05-11.

[Paillier, 1999] P. Paillier. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In *International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*, pages 223–238, 1999.

[Shi and Shen, 2021] C. Shi and C. Shen. Federated Multi-Armed Bandits. In *AAAI*, pages 9603–9611, 2021.

[Shi *et al.*, 2021] C. Shi, C. Shen, and J. Yang. Federated Multi-armed Bandits with Personalization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2917–2925, 2021.

[Sutton and Barto, 2018] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

[Tzamaras *et al.*, 2022] S. Tzamaras, R. Ciucanu, M. Soare, and S. Amer-Yahia. FeReD: Federated Reinforcement Learning in the DBMS. In *International Conference on Information and Knowledge Management (CIKM)*, pages 4989–4993, 2022.

[Zhu *et al.*, 2021] Z. Zhu, J. Zhu, J. Liu, and Y. Liu. Federated Bandit: A Gossiping Approach. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 5(1), 2021.