SAIaaS: A Blockchain-based solution for secure artificial intelligence as-a-Service

Nicolas Six, Andrea Perrichon-Chrétien, Nicolas Herbaut

SAIaaS: A Blockchain-based solution for secure artificial intelligence as-a-Service

Nicolas Six[1], Andrea Perrichon Chrétien[1], Nicolas Herbaut[1]

Université Paris 1 Panthéon-Sorbonne
Centre de Recherche en Informatique
F-75013 Paris, France

Abstract. Artificial Intelligence models are crucial elements to support many sectors in the current global economy. Training those models requires 3 main assets: data, machine learning algorithms, and processing capabilities. Given the growing concerns regarding data privacy, algorithm intellectual property, and server security, combining all 3 resources to build a model is challenging. In this paper, we propose a solution allowing providers to share their data and run their algorithms in secured cloud training environments. To provide trust for both clients and asset providers in the system, a blockchain is introduced to support the negotiation, monitoring, and conclusion of model production. Through a preliminary evaluation, we validate the feasibility of the approach and present a road map to a more secure Artificial Intelligence as-a-service.

Keywords: blockchain, artificial intelligence, security, privacy, TEE

1 Introduction

With the increasing number of companies that started to investigate AI solutions addressing business issues, the popularity of AIaaS (Artificial Intelligence as-a-service) has been rising throughout the years. It is expected that the market, valued at USD 2.68 billion in 2019, might reach USD 28.58 billion by 2025[1]. Indeed, AIaaS facilitates access to machine learning algorithms and learning infrastructure, two of the three assets required to compute AI models, along with datasets. However, its growing adoption raises issues, notably with the centralization of those services over a handful of actors, such as Googles Prediction API and Amazon ML. Also, even with the disposal of infrastructure, getting high-quality datasets and innovative algorithms is a difficult task. Owners of sensitive or valuable datasets as well as state-of-the-art algorithms might be reluctant to share their assets. They expect the client to pay a premium for getting an asset, or guarantees of confidentiality that are difficult to provide in regular cloud environments. Confidentiality of the assets can also be threatened during the learning phase if the computation service is compromised or the provider malicious. Finally, it is difficult to find such providers, as many datasets and

[1] https://bit.ly/3wfEATY

algorithms exist. Those issues have been partially addressed by academic studies, whether by the construction of blockchain-enabled data marketplaces (e.g., [6]) or blockchain cloud monitoring for third-party computation [9]. However, there still is no end-to-end solution to allow clients to get a desired model in a decentralized way.

The core of our proposal is two-fold: first, we use blockchain to design a transparent and tamper-proof marketplace facilitating the auction-based pricing for immaterial (datasets and ML algorithms) and material assets (cloud computing resources). Then we propose using Trusted Execution Environments for ML tasks, to guarantee that code and data loaded inside the infrastructure being protected with respect to confidentiality and integrity.

The rest of the paper is organized as follows. We first present our model for a Secure Artificial Intelligence as-a-Service marketplace in Section 2. We conduct a preliminary evaluation of the cost of such a platform in 3, then we mention some related work and position our proposal in Section4. We finally conclude and discuss future work in Section 5

2 A Secure Marketplace for IA

This paper proposes SAIaaS (Secure AI as-a-service), a model for a blockchain-based marketplace for collaborative and rewarded computation of models. In this section, we describe our model for SAIaaS and provide insights on how the 4 main steps of our proposal can be implemented.

2.1 Actors and High-level Workflow

In SAIaaS, a *client* willing to obtain an AI model publishes an *auction* on a public blockchain for providers to bid on. Providers are classified into 3 categories: *Data providers* (DP) who provide datasets, *Algorithm providers* (AP) who provide innovative machine learning algorithms, and *Infrastructure providers* (IP) who provide cloud resources for the learning phase.

The *auction* contains a description of the client's needs, and associated requirements (e.g., model accuracy). Each provider will be allowed to bid for its own category. Thus, three winners will be selected for each category, constituting a triplet of winners. They will have to collaborate to generate the expected model. First, the AP will have to set up a computing environment. Then, the DP and the AP will send their assets to the IP, through a secured channel. Finally, the IP will compute the model, and return it to the client. Providers will be rewarded according to their bid. The next sections provide more details on each step of the workflow. We discuss the implementation in Section 3.

2.2 Semantic Matchmaking Phase

The client's request contains requested *asset specifications* for data, algorithms and infrastructure on the blockchain system. Since we aim to build a use case-agnostic system, we must support a wide and dynamic range of application

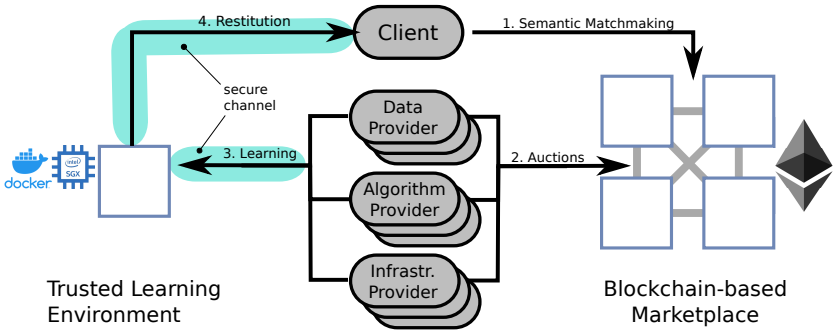

Fig. 1. *High-level architecture*

domains. To this end, we rely on an ontology-based resource retrieval and allocation system, which have already been proposed in the literature for cloud services provisioning [5] or dataset [4] discovery.

With a common ontology used to describe both client requirements and providers assets, matchmaking can be done on-chain, through the emission of a specific event targeted at providers that can fulfill client requests or off-chain, through the continuous monitoring of new clients requests publicly available on the blockchain. Based on the asset ontology, each provider can analyze the *asset specifications* published by the client and know if it owns a matching asset, in which case it will take part in the auction phase through *asset bids*.

2.3 Auction Phase

The auction phase is the sequence of actions after the client's request is made public. It involves providers interacting with the auction smart contract by placing *asset bids*. Along with its *asset specifications*, the client locks a cryptocurrency amount, the reserve price, that will be used to pay providers for their assets. The auction ends when the client adjudicates the auction contract or when a predetermined time elapses.

Each provider is able to participate in the auction and propose a price for an asset that semantically matches the client's request. The first bid proposed by a provider determine the base price for a particular *asset specification*, it is updated when a new *asset bid* is placed at a lower price for the same asset request. To foster competition between providers, when the base price is updated, competing providers are able to reduce the initial price of their *asset bid*. At the end of the auction phase, the winning triplet comprised of the semantically matching lowest-price bids is made public through an *Auction Service Level Agreement* (SLA) on the blockchain.

Since the platform assures both data privacy and algorithm confidentiality, it is not possible to assess the quality of immaterial assets (datasets and algorithms) before actually performing the machine learning. This can slow down the adoption of the system, since clients are reluctant to pay without knowing if

the results will be satisfactory. To circumvent this issue, each *asset bids* from a provider can contain references to previous auctions with semantically matching bids from the same provider. For example, if a particular dataset was proposed by another client in a previous auction, the provider can reference the *Auction* instance address in its bid to provide evidence of its suitability. Clients can specify how many references each bid must have in their auction specifications. Allow new providers to enter the system, it is expected that bids without a reference to a previous *Auction* SLA to be significantly cheaper than referenced ones, so compensating for the risk on the client side.

2.4 Secure Learning Phase

Once all the providers are identified, the immaterial assets from the DP and the AP need to be transmitted onto the IP infrastructure for machine learning computation. To prevent any data or intellectual property leaks from a malicious or compromised infrastructure provider, a secured learning environment is required.

For dataset security, trusted execution environments (TEE), such as Intel SGX enclaves, have been proposed to perform both machine learning and model creation in a secure way [3]. This technology brings trust in the learning process since the infrastructure provider cannot access the data stored in TEEs, and the data provider receives an attestation proving that the environment is secured and up to date[2], and a secure communication channel is created for data upload.

For algorithm security, TEE and Linux containers can also be leveraged to make sure that the intellectual property of the AP is not compromised. Through the previously mentioned secure channel, AP uploads its algorithms from a secure registry. It can be executed as Linux containers [1] which has the additional benefit of preventing the algorithm from being compromised through containers image signature, for example using Docker Content Trust[3].

2.5 Restitution Phase

Once the learning phase is over, the model is provided through the TEE secure channel to the client. The client then assesses the result of the model off-chain and publishes an acknowledgment in the auction SLA contract to close the process, unlock the payments to the providers and retrieve the potential unspent resources from its reserve price. The next section presents a preliminary implementation of the marketplace, the secured learning environment being left for future work.

3 Proof of Concept

This section presents a proof of concept of the main component of our proposal, the blockchain-based marketplace. First, a Solidity implementation of the

[2] SGX Remote attestation https://intel.ly/3ry4UoU
[3] https://dockr.ly/3m3ESss

blockchain marketplace is proposed. Then, a cost analysis is performed to evaluate the overall cost of the solution.

3.1 Implementation

To evaluate the contribution, this paper introduces a concrete blockchain implementation of the marketplace on Ethereum. The code is available on Github[4]. The architecture of the marketplace is designed as shown in Figure 2.

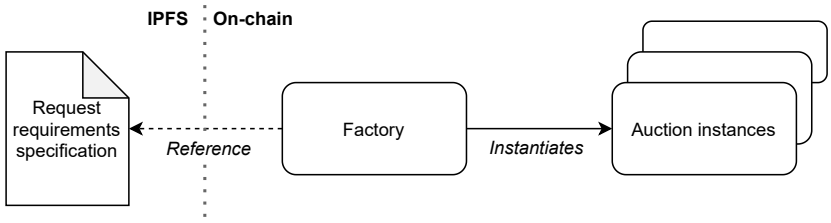

Fig. 2. *Marketplace on-chain architecture.*

This marketplace is based on two solidity smart contracts: *Factory*, and *Auction*. *Factory* is a contract dedicated to creating *Auction* instances. Once deployed on-chain, it can be called by an external party to create an auction, providing an adequate description of its requirements. Expected requirements are specified in an ontology file, stored off-chain on IPFS (Inter-Planetary File System), a decentralized storage platform[5]. The party willing to create an auction must provide its requirements following this file.

When an *Auction* instance is created, its requirements, the client address, and auction modalities are set as state variables. A reference to the *Factory* contract responsible for its instantiating is also kept. Thus, the *Factory* contract, as well as requirement metadata, can easily be updated, without losing the link between old metadata and older *Auction* instances. The main auction modality is the auction duration. Before the execution of each contract method, verification is done to check if the defined duration did not elapsed since auction creation. If so, the contract automatically adjudicates the winners of the auction.

Providers can bid on the auction if they can effectively provide the desired assets and if their assets are semantically compliant with the client's initial request. Providers can submit as many bids as they want, as long as the auction phased is not complete and if the new bid amount is below the previous one. A maximum bid value is also set in the contract, forbidding providers to bid above this value. After the adjudication, a triplet of winners is determined. Each member of the triplet is the winning provider for a category (data, algorithm, or infrastructure).

[4] https://github.com/nicoSix/solidity-data-marketplace
[5] https://docs.ipfs.io/

3.2 Cost Estimation

As Ethereum has been selected for this implementation, each operation performed on contracts that alter their states (e.g., deployment, bid, ...) has a defined cost in gas. The price of performing an operation in Ether (the main cryptocurrency of Ethereum) is the product between the total cost in gas and the current blockchain gas price (Ether per gas). By extension, the cost in $USD can be deducted from the cost in Ether. To get an accurate estimation of those costs, a scenario will guide the measurements.

Scenario A party wants to obtain a model, but he doesn't have any data, algorithm, or infrastructure. He decides to use the SAIaaS marketplace to find providers that could perform this task for him. He creates an auction on-chain through the *Factory* instance (already deployed) that acts as a gateway, by specifying its requirements and its maximal price. 6 providers are willing to bid, two per type of asset provided (data, infrastructure, algorithms). They don't know and don't trust each other. First, one provider per asset bid to provide their asset. The other providers then outbid the first three, who will also bid again. Finally, the auction stops, and the first three providers are the winners of this auction. Thus, 9 bids are placed for the auction.

Results Table 1 lists all possible operations and associated costs for each operation, and the sum of all the costs when running the scenario described before.

Table 1. *Cost per operation (in gas/in Ether/in $USD). When the experiment was conducted (28/03/2021), the gas price was 124 Gwei, and an Ether was valued $1,683.*

Operation	Gas cost	Price(Ether)	Price ($USD)
Factory deployment	2,344,498	0.2907178	489.28
Auction creation	1,760,105	0.218253	367.32
First bid of contract	237,518	0.0294522	49,57
First bid of user	207,829	0.0257708	43,37
Modify existing bid	94,747	0.0117486	19,77
Adjudication	198,295	0.0245886	41.38
Total (scenario)	**3,180,058**	**0.3943272**	**663.65**

3.3 Discussion

The results show an important gap between the cost of deploying contracts and the cost of bidding and adjudicating on the auction. Indeed, contract deployment implies storing large amounts of data and defining many states. This is

considered a very expensive operation as nodes will have to store contracts and their states forever on-chain. The cost associated with the scenario execution is also expensive, even if the *Factory* contract is considered as already deployed. By extension, the cost in $USD is prohibitively high to be used in its current state.

However, as gas costs are inherent to Ethereum, selecting another blockchain might decrease costs a lot, even nulling them if using a private blockchain. Consequently, obtained results are only valid using the Ethereum mainnet, but this blockchain also comes with a lot of advantages, especially the decentralization factor. This is something to take into account if another blockchain was selected. Finally, the number of functions and their complexity might become higher as additional features are implemented (e.g., on-chain provider registry). By extension, costs of functions might increase in the future. However, applying smart-contract design patterns and refactoring code can help mitigate the increase.

4 Related Work

In the literature, several works propose a marketplace for IA using blockchain where authors propose a secure data oriented AI marketplace to guarantee privacy between users (such as [10], [8] and [7]) using blockchain. Our proposal goes further by also including ML Algorithms and Infrastructure as tradeable assets while considering security aspects of the learning process.

Other work has proposed solutions leveraging Trusted Execution Environments ([2]). Our proposals differ in the sense that it is more specialized and proposes ready-to-use solution targeted at AI needs with an auction-driver pricing scheme.

5 Conclusion and future works

This paper presents a blockchain-based marketplace to support secure artificial intelligence as-a-service (SAIaaS). From a requirements file shared on-chain, a client can request the computation of a specific model to solve an AI problem by creating an auction. Providers can then bid on the auction, meaning that they are willing to provide their assets (data, infrastructure, or algorithm) to compute the model. At the end, the best offers for each asset are retained, and the model is computed in the provided infrastructure using provided data and algorithms. A secure environment can be set up in the infrastructure to avoid any leakage of valuable or confidential data, using enclaves.

This paper paves the way for future progress in the proposed solution. First, by implementing a system capable of automatically bootstrapping the model computation on the *Infrastructure provider* dedicated services, with the dataset and the algorithm transferred from the other two providers. This includes the setup of a trusted learning environment, using enclaves. The designed system must also take into account the potential incompatibility between a provided

dataset and an algorithm. Second, with the implementation of a monitoring system connected to the blockchain that ensures on the infrastructure provider side that computations are performed following client requirements. This monitoring system could also help detect breaches of confidentiality of datasets provided by the *Data provider*.Third, by handling potential issues that could occur from the creation of an auction to the computation of the model. For example, handling potential client dissatisfaction towards the result through a litigation system. Also, we plan to integrate a reputation system into the solution to cope with potential malicious blockchain users that could try to bid maliciously on the auction.

Finally, current artifacts might be refined in the future. The costs incurred by the usage of Ethereum might be decreased with the usage of an Ethereum sidechain (e.g. Polygon[6]) rather than the mainnet, keeping the benefits of using a public blockchain while reducing the inconvenience of its drawbacks.

References

1. Arnautov, S., Trach, B., Gregor, F., Knauth, T., Martin, A., Priebe, C., Lind, J., Muthukumaran, D., O'keeffe, D., Stillwell, M.L., et al.: {SCONE}: Secure linux containers with intel {SGX}. In: 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16). (2016) 689–703
2. Fedak, G., Bendella, W., Alves, E.: Iexec whitepaper: Blockchain-based decentralized cloud computing (2017)
3. Hunt, T., Song, C., Shokri, R., Shmatikov, V., Witchel, E.: Chiron: Privacy-preserving machine learning as a service. arXiv preprint arXiv:1803.05961 (2018)
4. Kushiro, N.: A method for generating ontologies in requirements domain for searching data sets in marketplace. In: 2013 IEEE 13th International Conference on Data Mining Workshops, IEEE (2013) 688–693
5. Ma, Y.B., Jang, S.H., Lee, J.S.: Ontology-based resource management for cloud computing. In: Asian Conference on Intelligent Information and Database Systems, Springer (2011) 343–352
6. Nardini, M., Helmer, S., El Ioini, N., Pahl, C.: A blockchain-based decentralized electronic marketplace for computing resources. SN Computer Science **1**(5) (2020) 1–24
7. Özyilmaz, K.R., Doğan, M., Yurdakul, A.: Idmob: Iot data marketplace on blockchain. In: 2018 crypto valley conference on blockchain technology (CVCBT), IEEE (2018) 11–19
8. Sarpatwar, K., Sitaramagiridharganesh Ganapavarapu, V., Shanmugam, K., Rahman, A., Vaculin, R.: Blockchain enabled ai marketplace: The price you pay for trust. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. (2019) 0–0
9. Taghavi, M., Bentahar, J., Otrok, H., Bakhtiyari, K.: A blockchain-based model for cloud service quality monitoring. IEEE Transactions on Services Computing **13**(2) (2019) 276–288
10. Travizano, M., Minnoni, M., Ajzenman, G., Sarraute, C., Della Penna, N.: Wibson: A decentralized marketplace empowering individuals to safely monetize their personal data. White paper (2018)

[6] https://polygon.technology/