Research Track | Security

Advancing Web 3.0: Making Smart Contracts Smarter on Blockchain

Junqin Huang*, Linghe Kong*, Guanjie Cheng†, Qiao Xiang‡, Guihai Chen*, Gang Huang†, Xue Liu¶
*Shanghai Jiao Tong University, †Zhejiang University, ‡Xiamen University, ¶McGill University

Introduction

As the digital age accelerates, the evolution from Web2 to Web3 promises a new era of interconnected, autonomous services. Blockchain and smart contracts are one of the key infrastructures promoting Web3, but they are not ready to support intelligent Web3 applications.

One of the key issues hindering the promotion of Web3 applications is *the huge gap between the growing intelligence needs of Web3 applications and the limited feature provided by smart contracts*. To bridge the gap, we aim to make smart contracts smarter by supporting model inference functions. Unfortunately, existing smart contracts cannot support AI model inference for two challenges: *high complexity* and *non-determinism*.

High Complexity. Current blockchains have explicit complexity limits for smart contracts due to system throughput and security concerns. Ethereum only allows up to 15-30 million gas for each block. But the gas cost of all evaluated models far exceeds the block gas limit (Figure 1). It indicates that Ethereum smart contracts are far from supporting model inference.

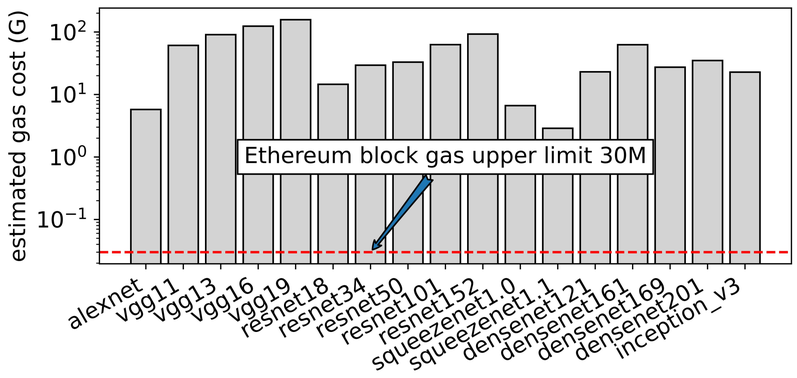


Figure 1 Estimated gas cost for on-chain model inference.

Non-determinism. Moreover, smart contracts should accept deterministic code for achieving consensus. However, AI model inference outputs are typically non-deterministic due to multiple factors (Figure 2). Thus, the inconsistent inference results cannot reach a consensus among blockchain nodes.

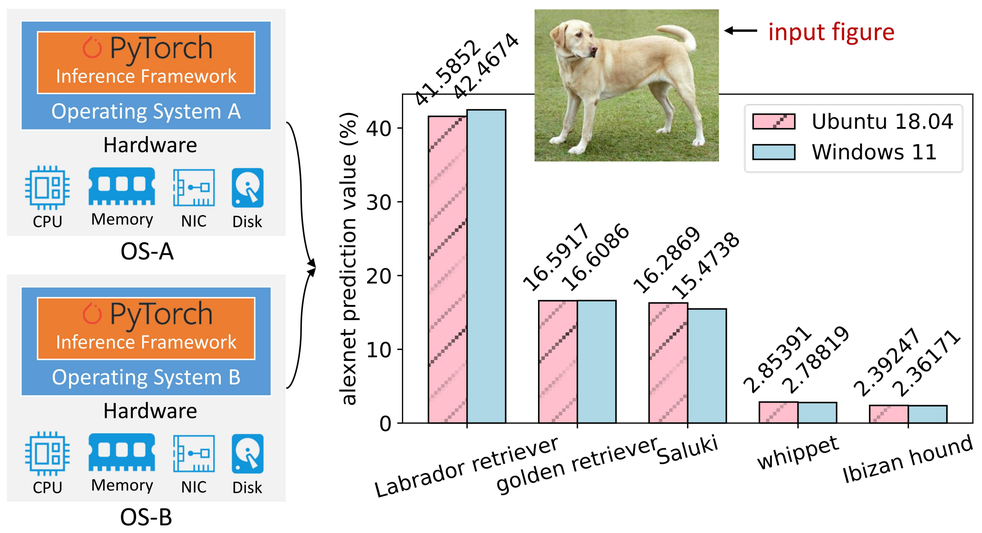


Figure 2 Non-deterministic model inference computation.

SMART: On-chain and Off-chain Contract

We propose **SMART, a plug-in contract framework that supports complex, non-deterministic model inference, while achieving good compatibility with existing blockchains**. We design an on-chain and off-chain joint execution model, which leaves the deterministic code executed on-chain while outsourcing the non-deterministic model inference to off-chain nodes. The goal of preserving on-chain execution is to be compatible with existing blockchains. The off-chain function handles the challenges of non-determinism and high complexity brought by model inference. Figure 3 shows the architecture and workflow of SMART.

We use TEE hardware to endorse the integrity and correctness of the off-chain model inference. To prove the validity of TEE hardware, we need to use the remote attestation service provided by the trusted TEE manufacturers. To avoid relying on the centralized remote attestation server provided by TEE manufacturers, we use blockchain nodes to design a distributed attestation service to mitigate the single point of failure.

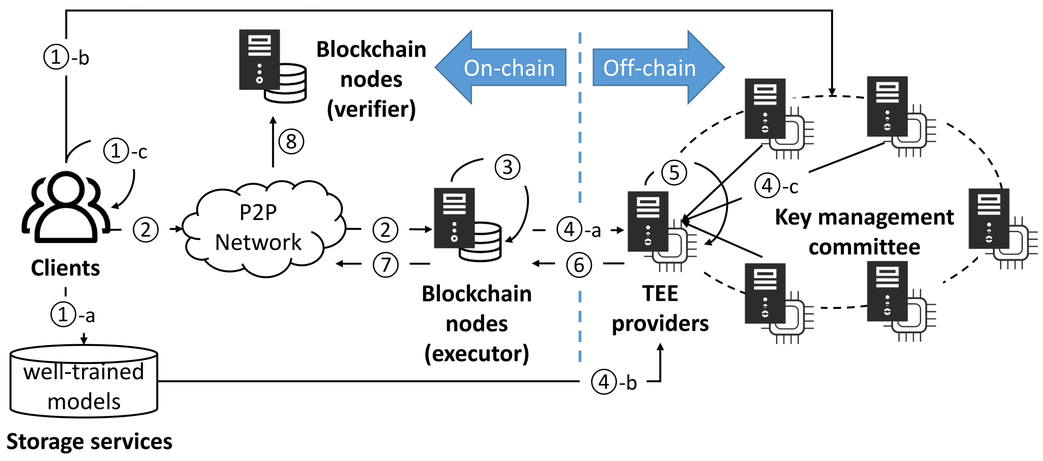


Figure 3 Overview of SMART system model and workflow.

Implementation and Evaluation

- Underlying blockchain system: FISCO BCOS[1], a EVM-compatible consortium blockchain
- TEE hardware: Intel SGXv1 with FLC
- Gramine (Graphene-SGX[2]), a TEE LibOS for simplifying SGX application development
- The codes and data are open-source at: https://github.com/imtypist/fisco-smart

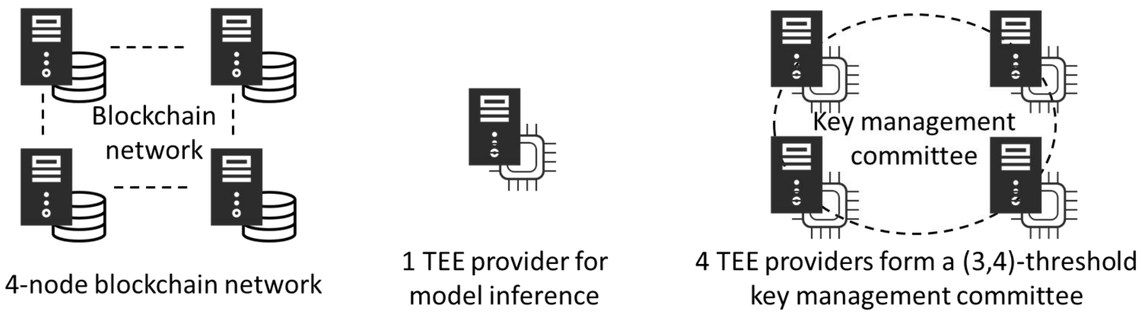


Figure 4 Experimental settings.

Our evaluations show that SMART achieves its security goals while significantly enhancing inference efficiency, outperforming existing on-chain solutions[3-4] about 5 orders of magnitude (Figure 5).

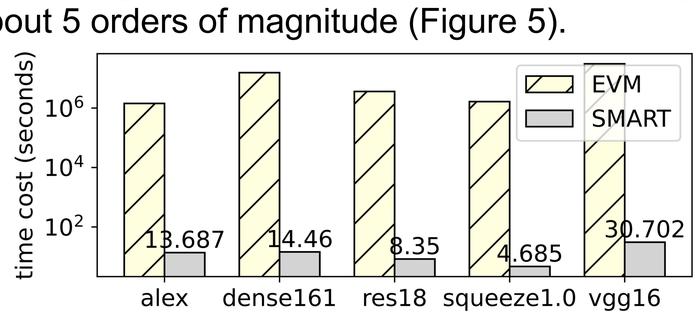
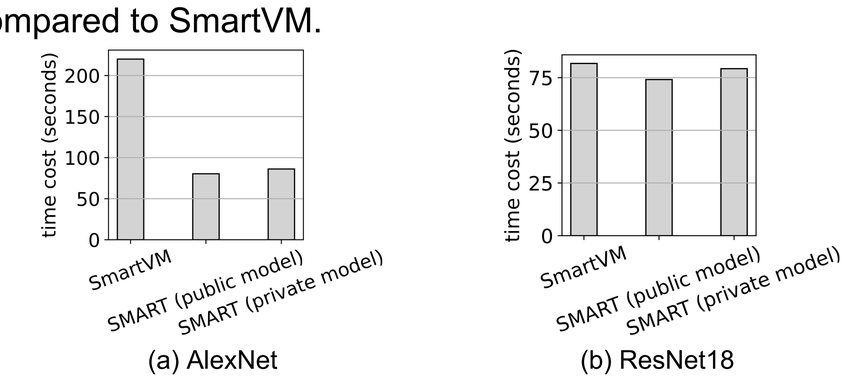


Figure 5 Model inference time comparison of EVM-based on-chain solutions[3-4] and SMART.

We evaluate the end-to-end latency of the SMART contract call under the public and private models, and compare it with SmartVM[5], an AI operators embedded smart contract VM, as shown in Figure 6. We can observe that the end-to-end latency of SMART is better than SmartVM, especially for AlexNet, there is about 2.7x improvement. Note that SmartVM does not support private model inference. Even for private model inference, SMART still has lower latency compared to SmartVM.



(a) AlexNet (b) ResNet18

Figure 6 End-to-end latency of SmartVM[5] and SMART.

Contact:

Junqin Huang
Shanghai Jiao Tong University
Email: junqin.huang@sjtu.edu.cn
Website: https://huangjunqin.com

References:

[1] Li Huizhong, Chen Yujie, Shi Xiang, Bai Xingqiang, Mo Nan, Li Wenlin, Guo Rui, Wang Zhang, and Sun Yi. 2023. FISCO-BCOS: An Enterprise-Grade Permissioned Blockchain System with High-Performance. In IEEE SC. 1–10.
[2] Chia-Che Tsai, Donald E Porter, and Mona Vij. 2017. Graphene-SGX: A Practical Library OS for Unmodified Applications on SGX. In USENIX ATC. 645–658.
[3] Syed Badruddoja, Ram Dantu, Yanyan He, Kritagya Upadhayay, and Mark Thompson. 2021. Making Smart Contracts Smarter. In IEEE ICBC. 1–3.
[4] Justin D Harris and Bo Waggoner. 2019. Decentralized and collaborative AI on blockchain. In IEEE Blockchain. 368–375.
[5] Tao Li, Yaozheng Fang, Ye Lu, Jinni Yang, Zhaolong Jian, Zhiguo Wan, and Yusen Li. 2022. SmartVM: A Smart Contract Virtual Machine for Fast On-Chain DNN Computations. IEEE Transactions on Parallel and Distributed Systems 33, 12 (2022), 4100–4116.