FUN AI: A Peer-to-Peer AIGC Task Computation System

Formlessness Unfeelingness Namelessness

2024-11-26

Abstract: A purely peer-to-peer AIGC computation system enables inference requests to be initiated by one party and processed directly by another, without reliance on centralized AI giants, while ensuring efficient service delivery. Although privacy-preserving computation offers partial solutions, its extended verification times and high participation thresholds significantly limit its practical application.FUN AI introduces a peer-to-peer inference task trading network designed to support the deployment of any open-source large model. This network operates on a two-layer architecture. The second layer employs a proof of inference mechanism, comprising inference nodes and verification nodes responsible for executing and validating inference computations. The first layer leverages a Bitcoin-style Proof-of-Work (PoW) consensus mechanism, with transaction confirmation nodes ensuring network security and the inclusion of transactions on the blockchain. When an inference task is requested, inference nodes bid for eligibility by sending a specified amount of $FAI tokens to a group of verification nodes.A verifiable random function selects one node, among those sending the highest amount of tokens, to undertake the computation task. Once the computation is complete, verification nodes evaluate the result using the judgment of the large model, determining rewards or penalties for the inference node. Subsequently, the first-layer consensus nodes confirm the transaction.Verification nodes are composed of the group with the highest $FAI token stakes and are responsible for the consensus of the second-layer chain. The network maintains fairness as long as the majority of verification nodes and PoW nodes are controlled by non-colluding participants. This network supports parallel computation and can remain efficient as long as a sufficient number of inference nodes participate. All types of nodes can freely join or leave the network. The architecture supports any open-source large model and adopts a Bitcoin-style zero pre-mining issuance mechanism, providing the community with long-term security and sustainability.

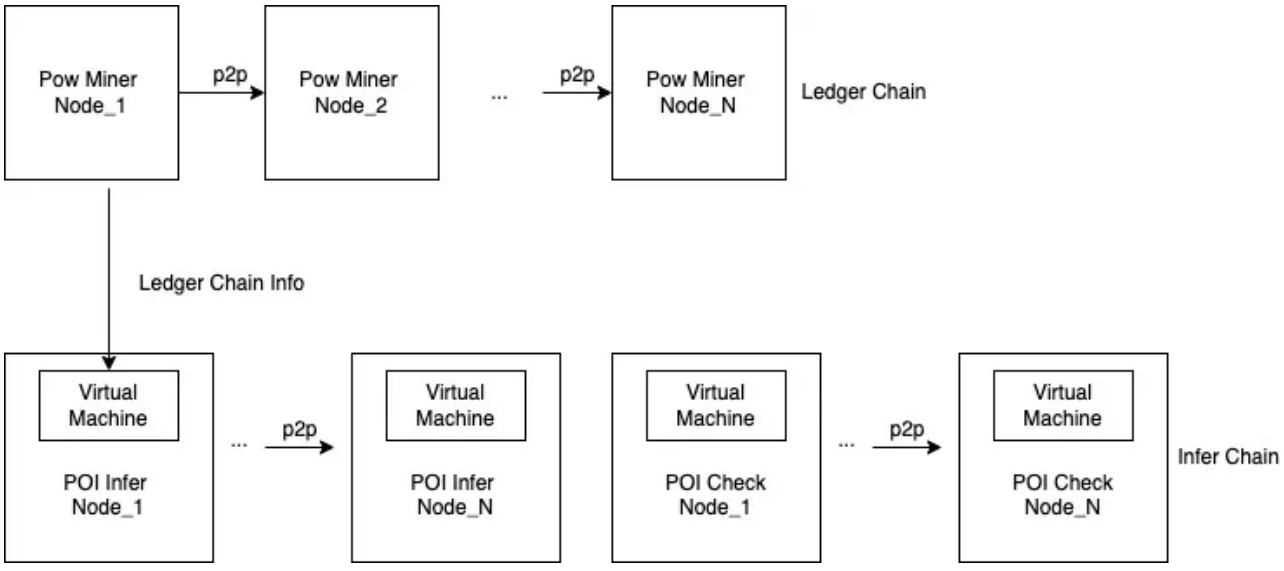
1. Introduction

AI has become almost entirely reliant on tech giants as trusted third parties to deliver all services. While the systems operated by these giants are sufficient for most AI needs, they come with the inherent flaw of monopolization. AI services controlled by internet giants inevitably lead to issues such as privacy breaches, high costs, result manipulation, and the inability of the broader community to share in the industry's growth dividends. Conversations with AI, along with personal information, may be permanently stored in the memory of large models, potentially becoming

public information.Most proprietary large models controlled by giants require the most expensive GPUs, such as the H100, for inference tasks. In contrast, open-source models like Llama 3 can provide comparable AIGC services using consumer-grade GPUs like the 4090, significantly lowering the entry barrier. Centralized systems also frequently generate unrelated, misleading, or biased outputs due to politically correct constraints or centralized authority controls. AI has become the next frontier of technological dividends, yet outside of the tech giants, the broader community has received little to no share of these dividends. There is a pressing need for a decentralized AI computation network that allows the community to participate in and benefit from AI advancements. Such a system should enable the deployment of any high-performing open-source large model while allowing individuals to contribute as inference nodes using personal GPUs like the 4090 without incurring heavy costs. A peer-to-peer mechanism and proof of inference ensure that outputs are directly derived from large models, free from tampering. A disk recycling mechanism for deleted data, combined with a blockchain-based public key identity system, ensures the network does not expose any personal privacy. In this paper, we propose a peer-to-peer AIGC inference task trading system that leverages proof of inference and proof-of-work mechanisms to address the fairness and cost issues of centralized services, as well as the efficiency challenges of decentralized systems. The system is efficient as long as there are sufficient inference nodes and secure as long as the collective computation power of honest inference, verification, and PoW transaction confirmation nodes exceeds that of any colluding adversarial group.

2. Network Topology

The FUN AI system is composed of a two-layer architecture: the ledger layer and the inference layer, each with distinct consensus mechanisms, and they communicate through the Clarity virtual machine.



1) **Ledger Layer**

   The ledger layer is responsible for recording transactions on the blockchain and ensuring the security of the inference network. It uses the classic Bitcoin Proof-of-Work (PoW) consensus mechanism[1], with security determined by the overall computational power. The ledger layer records the block hashes of the inference layer and strengthens the security of the inference layer through PoW.

2) **Inference Layer**

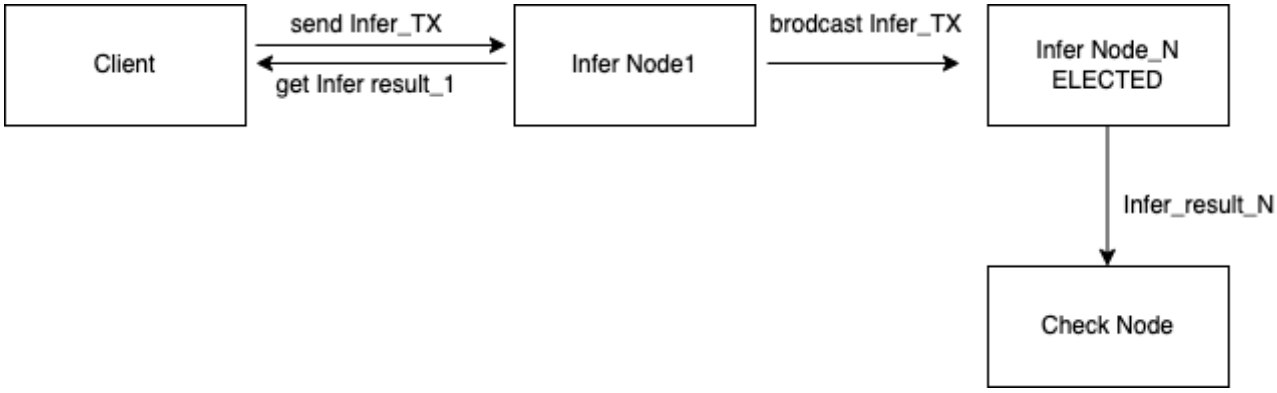
   The inference layer is responsible for executing inference transactions and uses a consensus mechanism based on proof of inference. The nodes in the inference layer are divided into inference nodes and verification nodes. All block hashes from the inference layer are written into the ledger chain via OP_RETURN.

3) **Communication Between the Ledger and Inference Layers**

   Communication between the ledger and inference layers is enabled through the Clarity virtual machine[2] running on the inference chain. The Clarity virtual machine is equipped with functions that read block information from the ledger layer, allowing it to retrieve the block header hashes, transactions within the blocks, and other related data. This facilitates communication from the ledger layer to the inference layer.

3. Inference Task Transactions

Inference task transactions are the native transactions of the FUN AI system, through which users access FUN AI's inference services. An inference transaction consists of fields such as the sender, GAS, input, output, context, system prompts, and others. Unlike regular transfer transactions, inference transactions have a high demand for immediate response to results. When any node receives an inference transaction in its memory pool, it immediately processes the transaction and returns the result to the client, after which it broadcasts the transaction to other nodes. The elected inference nodes then package their execution results into a block and submit it to the verification nodes for inspection. This means that:

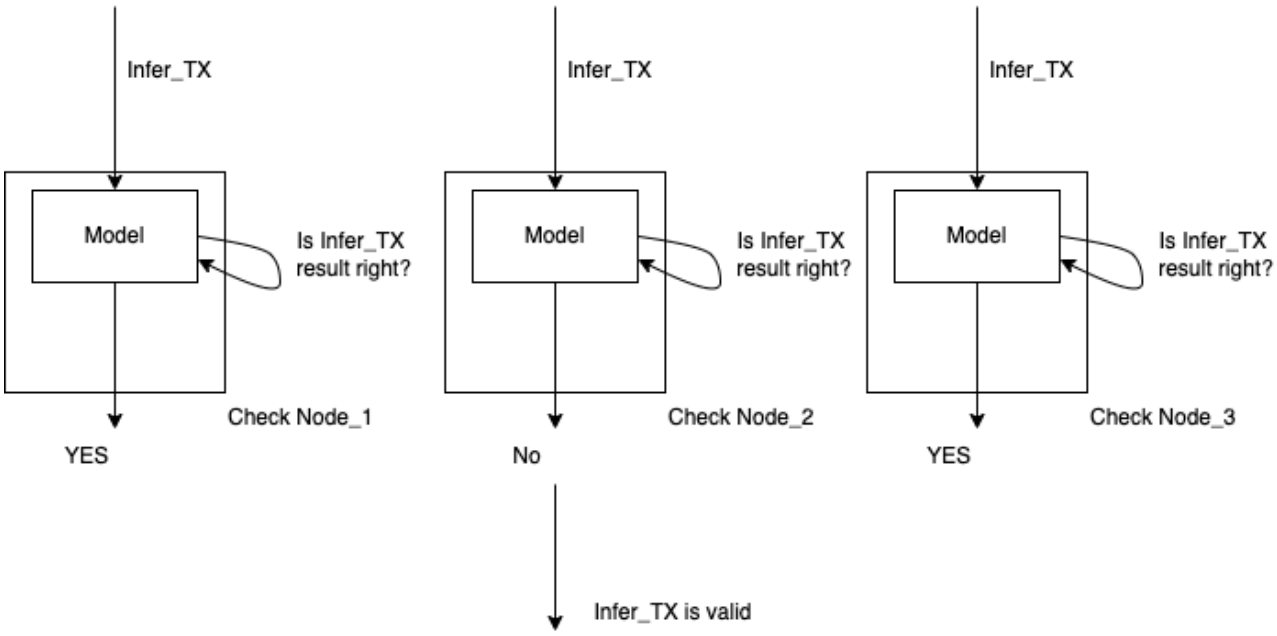


1) Clients connected to different nodes will receive different inference transaction results.

2) The inference results obtained by the client will differ from those verified by the verification nodes, which also determine and distribute rewards.

4. Legitimacy Check

Like any blockchain system, FUN AI faces the issue of validating the legitimacy of inference transactions. In blockchain systems, transaction validation is typically divided into two categories: Zero-Knowledge (ZK) proofs[3] and Optimistic (OP) schemes[4]. ZK is a cryptographic approach that generates a mathematical proof of the transaction's execution result, allowing verifiers to trust that the result was produced by correct logic. In the OP scheme, verifiers optimistically assume the transaction's validity and only check the result when discrepancies or disputes arise. However, neither of these approaches is suitable for validating inference transactions.



The ZK scheme is limited by hardware and speed, with the time required to generate proofs for large model inference transactions often calculated in days. In the OP scheme, even with the same large model and identical inputs, re-executing an inference transaction can produce different outputs due to issues related to natural language understanding.

FUN AI adopts a hybrid OP scheme for validating the legitimacy of inference transactions, combining the large model’s capabilities. When a verification node receives the result of an inference task, it takes both the task's input and output and uses them as input to the large model. By constructing prompts, the node asks the model whether the input and output match. The validity of the inference transaction is then determined based on the model’s response.

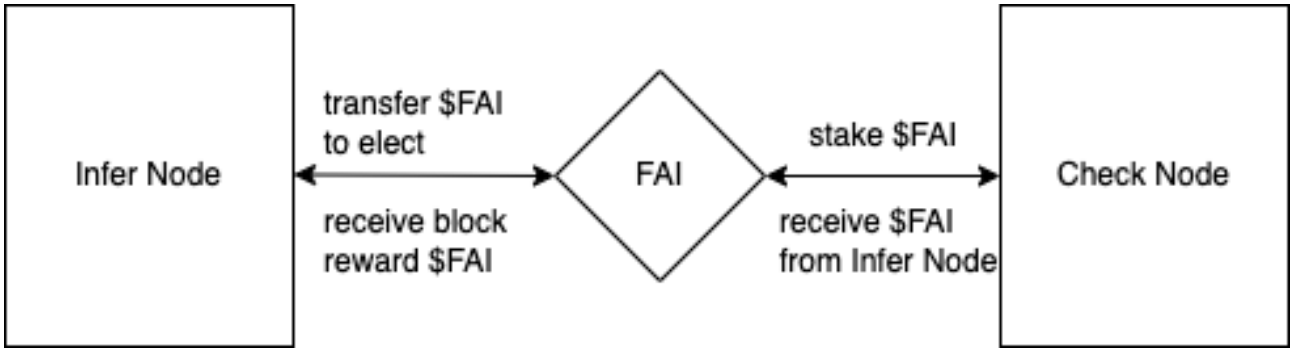
In FUN AI, verification nodes usually form a group. When the results from different verification nodes are inconsistent, the majority vote from the group is taken as the final result.

5. Proof of Inference

The inference chain in FUN AI employs a consensus mechanism called POI (Proof of Inference), which is a variation of POX (Proof of Transfer) consensus[5]. While traditional POX networks consist of miners and stakers, FUN AI's POI involves inference nodes and verification nodes.

- **Inference Nodes**

Any user capable of executing large model inference tasks can participate in the selection process for inference nodes. Users enter the election for inference nodes by sending a certain number of tokens to verification nodes on the ledger layer. The probability of being chosen as an inference node is calculated using a weighted random function, where the weight depends on the number of tokens sent. Once selected, the inference node executes inference transactions, packages blocks for the inference chain, and earns block rewards for the inference chain.



The tenure of an inference node consists of the following four stages:

1) **Registration**: The inference participant registers for future elections by sending consensus data to the main chain.
2) **Commitment**: The registered participant sends tokens to an address controlled by the verification node on the main chain to enter the election.
3) **Election**: The network selects an inference node using a verifiable random function.
4) **Block Assembly**: The elected inference node pulls transactions from the memory pool to write a new block and earns token rewards from the inference layer.

- **Verification Nodes**

Verification nodes are a group of nodes that participate in token staking on the inference layer. By default, the top five nodes with the highest staked tokens are selected. Verification nodes validate the legitimacy of inference chain blocks, distribute block rewards to inference nodes, and receive tokens sent to the ledger layer during inference node elections.

6. Incentive Mechanism

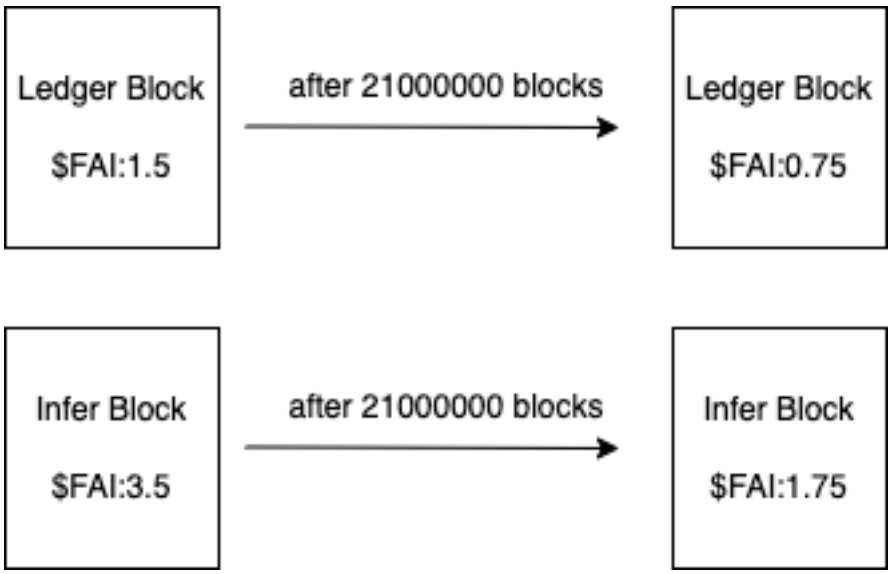
FUN AI uses $FAI as its native token to incentivize miners within the network. Similar to Bitcoin, $FAI is zero pre-mined, with a total supply capped at 21 million tokens.

The initial block rewards are distributed as follows:

- Ledger Layer: 1.5 $FAI per block

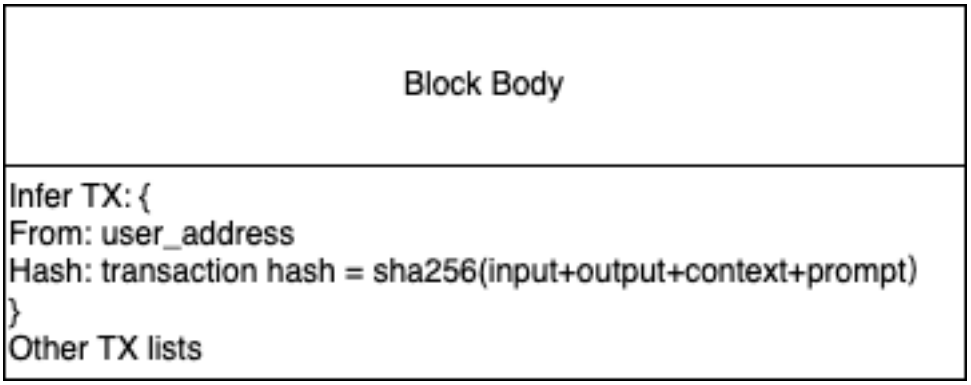
- Inference Layer: 3.5 $FAI per block

The block rewards are halved every 21 million blocks.



7. Disk Space Reclamation

FUN AI does not store the inputs and outputs of inference transactions within the blocks. Instead, only the hash of the inference transaction is included in the block. The inputs and outputs of inference transactions are temporarily stored on the nodes executing the transactions until the transaction is packaged into a block. Once the block is confirmed, the nodes delete the specific content of the inference transaction.



8. Multi-Model Network

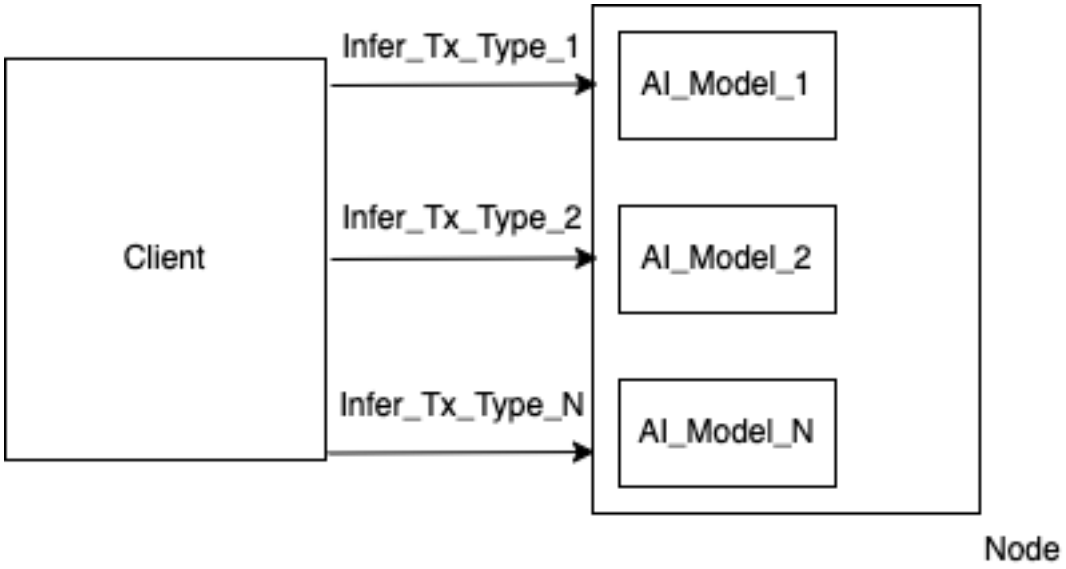
FUN AI is a network that supports the coexistence of multiple large models, and dynamically adding new models does not lead to network forks. When a new large model is introduced, FUN AI requires the addition of a new independent transaction type specific to that model, necessitating updates to both client and node software to handle this transaction type.

Verification nodes in the network must upgrade to support the new model, while inference nodes are not required to upgrade. However, inference nodes that do not upgrade will be unable to process the new transaction type and, consequently, will not be eligible to earn rewards for those inference transactions.

The block structure of FUN AI remains unchanged when new transaction types are added, ensuring that subsequent blocks are compatible with already confirmed blocks.

9. Smart Contracts and Cross-Layer Communication

FUN AI utilizes the Clarity programming language for its smart contracts. Clarity is designed with



principles of clarity and verifiability, offering a Turing-incomplete interpreted language that eliminates compiler uncertainties. This allows developers and users to directly read and verify the behavior of the code, ensuring alignment with its intended functionality.

Clarity smart contracts enable the inference layer to read the ledger chain's state by leveraging the immutable records of the ledger layer and the verification mechanisms of the inference layer. This establishes a secure and efficient communication bridge between the two layers.

Specifically, Clarity contracts support event reading and verification capabilities, enabling developers to write contracts that can monitor specific transactions or state changes on the ledger layer. The inference layer reflects these events in its own blockchain and ensures the authenticity of the data through its consensus mechanism. This mechanism relies on on-chain cryptographic proofs, avoiding dependence on external intermediaries.

10. Privacy and Fairness

Unlike centralized large models that differentiate users based on their IP address or geographic location, FUN AI is a fair and decentralized large-model network. Anyone can access the full capabilities of the large model by simply connecting to the FUN AI network through a client.

A user's wallet address serves as their sole identifier within the FUN AI network, ensuring that no other personal or private information is disclosed.

11. Security Analysis

We consider a scenario where an attacker attempts to generate an alternative chain that is faster than the honest chain, with the objective of manipulating any consecutive blocks.

A. Basic Assumptions

- Total computational power of the ledger layer: H_L
- Proportion of honest miners in PoI: M_h
- Proportion of attacker miners in PoI: M_a
- Transaction cost on the ledger layer (transfer fee): C_{tx}
- Miner reward on the inference layer (such as inference layer tokens): R_{TXN}

B. Attack Costs

FUN AI ensures security by requiring inference layer miners to submit transactions (transfers) on the ledger layer to participate in consensus. The security depends on the following factors:

1) **Ledger Layer Cost for Attacks**: The attacker needs to transfer tokens on the ledger layer to their own controlled address during each cycle to gain control over the inference layer's state. The attack cost is:

$$C_{attack} = k \cdot C_{tx} \cdot M_a$$

where k is the number of attack cycles.

2) **Ledger Layer Security**: Since FUN AI relies on the security of the ledger layer, the attacker must also launch an attack on the ledger layer to manipulate transactions related to the inference layer. This requires the attacker to control the computational power of the ledger layer. The probability of success is modeled based on a random walk:

$$P_{success\_L} = \left(\frac{H_a}{H_L}\right)^z \cdot e^{-2z}$$

The attacker will bear the cost of performing PoW attacks on the ledger layer.

C. Fork Resistance

The security of the inference layer is directly tied to the ledger layer. When inference layer miners submit transactions on the ledger layer, the irreversibility of the ledger provides fork protection for the inference layer. Assuming the fork probability of the ledger layer is P_{fork_P}, the fork probability for the inference layer can be approximated as:

$$P_{fork\_P} = P_{fork\_L} \cdot (1 - M_h)$$

As long as the ledger layer remains stable with a high proportion of honest miners, the fork probability of the inference layer remains very low.

D. Censorship Resistance

In FUN AI, miners participate in the consensus of the inference layer by broadcasting transactions on the ledger layer. To execute a censorship attack on the inference layer, an attacker would need to meet the following conditions:

1) Control the mining power of the ledger layer to censor specific transactions.
2) Or directly control the proportion of miners on the inference layer such that $M_a > M_h$.

Therefore, FUN AI's resistance to censorship depends on the censorship resistance of the ledger layer. Due to the high computational power and decentralization of the ledger layer, the inference layer inherits strong censorship resistance

12. Conclusion

We have proposed an AI large model service system that does not rely on centralized giants. To address the issues of high costs, privacy breaches, unfair results, and the inability for communities to participate in AI infrastructure controlled by large corporations, we present a peer-to-peer network that uses proof of inference to provide AI computation services. As long as the majority of honest inference verification nodes and PoW ledger nodes are in control, any attacker's efforts will be costly and futile.

This network is powerful due to its structural simplicity, requiring minimal coordination between nodes for simultaneous operation. No type of node needs permission to participate, and the deployment of new open-source AI models can occur as long as the majority of nodes approve, especially through the consensus of delegated verification nodes. This approval can be determined via community on-chain voting.

References

[1] Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System.

[2] Marvin J. Clarity Book:https://book.clarity-lang.org/

[3] Stefanos Chaliasos, Itamar Reif, Adrià Torralba-Agell, Jens Ernstberger, Assimakis Kattis, Benjamin Livshits. Analyzing and Benchmarking ZK-Rollups. Cryptology ePrint Archive, Paper 2024/889. DOI: 10.4230/LIPIcs.AFT.2024.14

[4] Hemrajani, I., & Li, J. (2023). The Security of Arbitrum’s Optimistic Roll-Up Implementation. Journal of Student-Scientists' Research, Vol. 5.

[5] Ali, M., Nelson, J., Shea, R., & Nelson, M. (2019). Proof of Transfer: A Blockchain Consensus Mechanism Leveraging Bitcoin. Blockstack PBC. [Online] Available: https://blockstack.org/papers/pox.pdf.