Light Paper of HYPERDUST

Ⅰ.Why are Bitcoin L2 and AI complementary?

1.Bitcoin L2 serves as a means of payment for AI agents.

Bitcoin has so far embodied the "maximal neutrality" of a medium, making it suitable for AI agents engaged in value transactions. Bitcoin can eliminate the inherent inefficiencies and "frictions" of fiat currencies. This "digitally native" medium of Bitcoin serves as a natural environment for AI to conduct value exchanges. Bitcoin L2 enhances the programmability of Bitcoin, meeting the speed requirements for AI value exchanges. Therefore, Bitcoin is expected to become the native currency of AI.

2. Bitcoin L2 enables decentralized AI governance.

Given the current trend of centralization in AI, decentralized alignment and governance of AI have become a significant concern. The more powerful smart contracts of Bitcoin L2 can serve as rules that govern the behavior and protocol patterns of AI agents, thus realizing a decentralized mode of AI alignment and governance. Moreover, Bitcoin's maximal neutrality makes it easier to reach consensus on AI alignment and governance.

3. Bitcoin L2 can issue AI assets.

In addition to issuing AI agents as assets on Bitcoin L1, the high-performance Bitcoin L2 can meet the needs of AI agents to issue AI assets, thus forming the foundation of the AI economy.

4. AI agents are a killer application for Bitcoin and Bitcoin L2.

Due to performance reasons, Bitcoin has lacked practical applications since its inception, aside from being a store of value. With Bitcoin moving to L2, it will gain enhanced programmability. AI agents are typically used to solve real-world problems, so

Bitcoin-driven AI agents can be truly utilized. Additionally, the scale and frequency of AI agent usage will make them a killer application for Bitcoin and L2. While human economies may not prioritize using Bitcoin as a payment method, the robot economy might. With numerous AI agents working tirelessly 24/7, using Bitcoin for small payments, the demand for Bitcoin could increase dramatically in ways currently unimaginable.

5. AI computing can enhance the security of Bitcoin L2.

AI computing can complement Bitcoin's PoW or even replace PoW with PoUW, injecting the energy used for Bitcoin mining into AI agents. Through L2, AI can make Bitcoin a smart-driven green blockchain, instead of relying on mechanisms like Ethereum's PoS. Our proposed Hypergraph Consensus is based on 3D/AI computing for PoUW, which will be further discussed.

Ⅱ.A Comprehensive Overview of Existing Decentralized AI Projects

The HYPERDUST project currently stands out in the Web3 AI field, featuring GPU-powered consensus, AI assetization, and decentralization, making it a decentralized hybrid of AI and finance applications. By achieving the "three transformations," it meets the five characteristics that decentralized AI platforms should possess. We have also briefly reviewed and compared existing decentralized AI-related projects based on these five features.

The five characteristics that decentralized AI platforms should possess are as follows:

(i) Verifiability of remotely run AI models

Includes decentralized verifiability technologies such as Data Availability and ZK.

(ii) Usability of publicly available AI models

Primarily depends on whether the provided AI model (mainly referring to LLM) API nodes are Peer-to-Peer and part of a fully decentralized network.

(iii) Incentivization for AI developers and users

Should have a fair token generation mechanism.

(iv)Global governance of essential solutions in the digital society

Whether AI governance is neutral and consensus can be easily achieved.

(v) no vendor lockins, etc.

Whether it is a fully decentralized platform.

Based on these five characteristics, existing publicly planned or implemented projects can be summarized as follows: (Decentralized federated learning has made little progress after many years of practice, and the rise of LLMs has made this type of decentralized training even more difficult, so related projects are no longer listed.)

(i) Verifiability of remotely run AI models

We consider verifiability as an essential feature for decentralized AI projects, serving as the foundation for subsequent usability, incentivization, governance, and non-binding aspects. If verifiability is lacking, other dimensions of features are not considered. Projects lacking the verifiability feature may be decentralized projects, such as decentralized computing power leasing or markets for data, algorithms, and models, rather than decentralized AI.

Projects that may meet the verifiability criteria include:

Giza, based on the ZKML consensus mechanism, meets the verifiability of remotely run AI models. However, its current performance is relatively poor, especially far from the requirements of large models. The proof process for models with millions of parameters takes several minutes. The proof process for LLM models takes an unacceptable amount of time.

Cortex AI, a L1 public chain project launched five years ago, focuses on decentralized AI. It has complex technology, adding new instructions to the EVM virtual machine to meet the needs of neural network calculations. Its underlying technology is still based on ZK for usability verification, suitable for simple AI models but unable to meet the needs of large models like LLM.

Ofelimos, proposed the first PoUW scheme in the cryptography academic community, using a specific search algorithm. However, this algorithm is not associated with specific applications or projects.

Project PAI, mentioned PoUW in a paper but only has a whitepaper and no product.

Qubic, claims to use PoUW and proposes using hundreds of GPUs for artificial neural network calculations. However, the significance of simple calculations using Python's artificial neural network library is unclear and seems unable to meet the needs of LLM training or inference, and does not fulfill the function of PoUW.

FLUX, (PoW ZelHash, not PoUW)

Coinai, (paper stage) https://aipowergrid.io/, task assignment, no strict consensus mechanism

Projects that cannot meet the criteria:

Projects in the category of GPU computing power leasing all lack a decentralized verifiable mechanism, unable to guarantee the verifiability of remotely run AI models.

DeepBrain Chain, focusing on GPU leasing, was a L1 project in 2017, and the mainnet was launched in 2021.

EMC, centralized reward for task assignment, no decentralized consensus mechanism in the roadmap;

Atheir, no visible consensus mechanism;

IO.NET, no visible consensus mechanism;

CLORE.AI, POH, crowdsourcing model, AI model on-chain publishing payment, issuance of NFT, AI runs off-chain, lacks verifiability. Similar projects with the same pattern include: SingularityNET, Bittensor, AINN, Fetch.ai, oceanprotocol, algovera.ai.

(ii) Usability of publicly available AI models

Cortex AI: No support for LLM observed.

Qubic: No support for LLM observed.

None of the above-mentioned decentralized AI projects adequately address the five questions mentioned earlier. HYPERDUST is a fully decentralized AI protocol based on the Hypergraph PoUW consensus mechanism and a fully decentralized Bitcoin L2 Stack. It will be upgraded to a Bitcoin AI-specific L2 in the future.

PoUW is used to protect the network in the most secure way while not wasting computational power. All computational power provided by miners can be used for LLM

inference and cloud rendering services. The vision of PoUW is that computational power can be used to solve various problems submitted to decentralized networks.

Ⅲ.Positioning of HYPERDUST

HYPERDUST is a fully decentralized platform for training and deploying AI agents driven by large models (supporting agent training, excluding large model training). It is a decentralized hybrid of AI and financial applications, featuring GPU-powered consensus, AI assetization, and embodiment. Currently, it is based on the Hypergraph PoUW consensus mechanism and a fully decentralized Bitcoin L2, and will be upgraded to a Bitcoin AI-specific L2 with the assistance of the Bitcoin L2 Stack in the future.

Ⅳ.Background of HYPERDUST's Birth

1. Explosion of LLM and Applications

OpenAI's ChatGPT reached 100 million users in just three months, sparking a global frenzy in the development, application, and investment in large language models (LLMs). However, so far, the technology and training of LLMs have been conducted in a highly centralized manner, which has raised significant concerns among academia, industry, and the public. There is apprehension about the monopoly of AI technology by a few major providers, data privacy breaches, monopolization, and vendor lock-ins by cloud computing companies. These issues stem from the fact that the current internet and application access points are still controlled by centralized platforms, which are not suitable for large-scale AI applications. The AI community has begun implementing some locally run and decentralized AI projects. A representative of locally run projects is Ollama, which allows small to medium-sized LLMs to run on personal computers or even smartphones through parameter compression or precision reduction, protecting user data privacy and other rights. However, it is evidently difficult to support production environments and networked applications. Petals, on the other hand, achieves fully

decentralized inference of LLMs through Bittorrent's Peer-to-Peer technology. However, Petals lacks consensus and incentive layer protocols and remains confined to the researcher's small circle.

2. LLM-Driven Agents

Empowered by LLMs, intelligent agents can engage in higher-level reasoning and possess certain planning capabilities. With the aid of natural language, multiple intelligent agents can also form social collaborations akin to humans. Several frameworks for LLM-driven intelligent agents have been proposed, such as Microsoft's AutoGen, Langchain, and CrewAI.

Currently, a large number of AI entrepreneurs and developers are focusing on LLM-driven intelligent agents and their applications. There is a significant demand for stable and scalable LLM inference, but primarily, GPU inference instances are rented from cloud computing companies to meet this demand. In March 2024, NVIDIA released a generative AI microservice platform, ai.nvidia.com, including LLMs, to address this enormous demand, but it has not been officially launched yet. LLM-driven intelligent agents are flourishing similar to the website development frenzy of the past. However, they mainly operate in a traditional Web2 mode for collaboration. Intelligent agents developers need to rent GPUs or purchase API support from LLM providers to run these intelligent agents, resulting in significant friction and hindering the rapid growth of the intelligent agent ecosystem and the value transmission of the intelligent agent economy.

3.Embodied Intelligent Agent Simulation Environment

Currently, most intelligent agents can only access and manipulate objects through APIs or interact with these APIs through code or scripts, writing control instructions generated by LLMs or reading external states. General intelligent agents should not only understand and generate natural language but also comprehend the human world. After appropriate training, they should be able to transition to robotic systems (such as

drones, roombas, humanoid robots, etc.) to perform designated tasks. Such intelligent agents are called embodied intelligent agents.

Training embodied intelligent agents requires a large amount of real-world visual data to enable the agents to better understand specific environments and the real world, shorten the training and development time of robots, improve training efficiency, and reduce costs. Currently, these simulation environments used for training embodied intelligence are only built and owned by a few companies, such as Microsoft's Minecraft and NVIDIA's Issac Gym, with no decentralized environments available to meet the training needs of embodied intelligence. Recently, some game engines have begun to focus on artificial intelligence. For example, Epic's Unreal Engine is advancing AI training environments that comply with OpenAI GYM standards. This is because the simulation and training environments for intelligent agents require a significant amount of GPU computing power.

4. Bitcoin L2 Ecosystem

Although Bitcoin sidechains have existed for many years, they have primarily been used for payments, and the lack of smart contract support cannot sustain complex on-chain applications. The emergence of Bitcoin L2 that is compatible with the Ethereum Virtual Machine (EVM) enables Bitcoin to support applications like decentralized AI through L2. Decentralized AI requires a fully decentralized, compute-centric blockchain network, rather than being confined to increasingly centralized Proof of Stake (PoS) blockchain networks.

5. HYPERDUST Team's 20-Year AI Research and 3-Year Generative AI Application Exploration

The HYPERDUST team began exploring generative AI applications three years ago, utilizing AI to generate 2D images and 3D models to construct an open-world environment called MOSSAI, composed of thousands of AI-generated islands. They also proposed a standard for non-fungible generative cryptographic assets. However, at that time, decentralized solutions for AI model training and generation had not yet emerged. Relying solely on the platform's GPU resources could not support a large user base, preventing it from reaching widespread adoption.

With the growing public interest in AI ignited by LLMs, we introduced the HYPERDUST decentralized AI application platform. Beginning in Q1 2024, we conducted tests on both Ethereum and Bitcoin L2 networks. This platform not only incentivizes GPU providers through cloud rendering and AI inference mining but also reduces transaction friction and entry barriers for AI developers and users.

Ⅴ.The technical framework and solutions of HYPERDUST

1. How to Achieve a Decentralized Platform for AI Agents Driven by Large Language Models (LLMs)?

The biggest challenge in decentralized AI lies in how to realize remote inference for large AI models and the lack of high-performance, low-overhead verifiable algorithms for training and inference of embodied intelligent agents. Without verifiability, the system can only regress to a traditional multi-party market model involving suppliers,

demanders, and platforms, failing to achieve a fully decentralized AI application platform.

Verifiable AI computation requires a PoUW consensus algorithm as its foundation, enabling the implementation of decentralized incentive mechanisms. Specifically, in network incentives, the minting of tokens occurs when nodes complete computational tasks and submit verifiable results autonomously, rather than transferring tokens to nodes in any centralized manner.

To achieve verifiable AI computation, it is essential to define AI computation. AI computation encompasses various levels, such as machine instructions, CUDA instructions, C++, Python languages, and different levels of 3D computation required for training embodied intelligent agents, including shader languages, OpenGL, C++, blueprint scripts, etc.

HYPERDUST's PoUW consensus algorithm is implemented using computational graphs, which are defined as directed graphs where nodes correspond to mathematical operations. A computational graph is a way of expressing and evaluating mathematical expressions, serving as a "language" for describing equations, comprising nodes (variables) and edges (operations or simple functions).

1.1 Defining Verifiable Computations with Computational Graphs (currently implemented for both 3D and AI computations, with scalability). Different levels of computations can be represented using subgraphs. This approach covers various types of computations, and different levels of computations are expressed through subgraphs. Currently, there are two layers, with the top-level computational graph deployed on the chain for verification nodes to validate.

1.2 Loading and running LLM models and 3D scene levels in a fully decentralized manner. When a user accesses LLM models for inference or enters 3D scenes for

rendering, the HYPERDUST agent initiates another trusted node to run the same hypergraph (LLM or 3D scene).

1.3 If verification nodes detect inconsistencies between the results submitted by a node and those submitted by the trusted node, a binary search is conducted on the off-chain computation results of the second-layer computational graph (subgraph) to identify the diverging subgraph computation node (operator). The subgraph operators are pre-deployed to smart contracts, and invoking the smart contract with the parameters of the inconsistent operator verifies the result.

2. How to Avoid Excessive Computational Overhead?

Another challenge in verifiable AI computation is controlling additional computational overhead. In Byzantine consensus protocols, it is known that consensus requires 2/3 of the nodes to agree, meaning that all nodes must perform the same computation for AI inference consensus. Such additional overhead is unacceptable in AI computations. However, Hyperdust only requires 1 to m nodes to perform additional computations to complete verification.

2.1 Each LLM does not perform inference individually; instead, the HYPERDUST agent initiates at least one trusted node for "companion computation."

Since LLM inference computation involves computing the results of various layers of deep neural networks in the model and the layer above as inputs until the inference is completed, multiple users can concurrently access the same LLM model.

Therefore, at most, the same number of trusted nodes as the quantity of LLMs, denoted as m, need to be additionally initiated. At least, only one trusted node is required for "companion computation."

2.2 Similarly, 3D scene rendering computation operates. Each user entering a scene activates a hypergraph, prompting the HYPERDUST agent to load a trusted node according to the hypergraph for the corresponding computation. If m users enter different 3D scenes, then at most, m "companion computation" trusted nodes are initiated.

In summary, the number of nodes involved in additional computation is less than or equal to n + m, where n is the number of users entering 3D scenes and m is the quantity of LLMs. This effectively avoids resource waste and ensures network verification efficiency, following a Gaussian distribution.

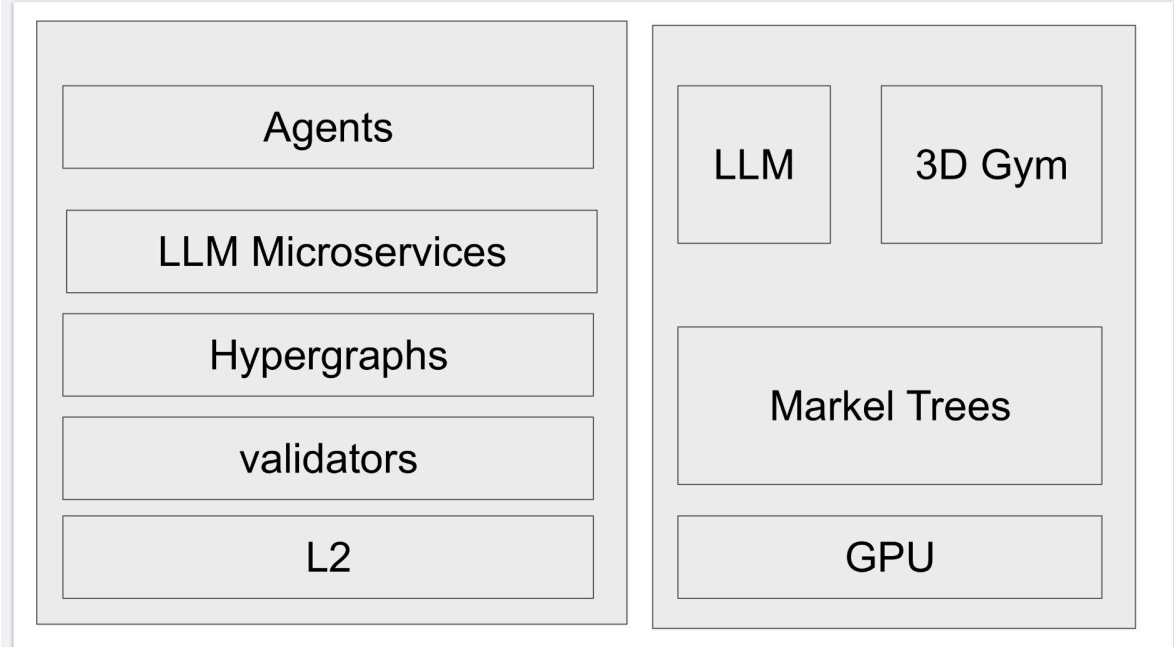
3. How AI Integrates with Web3 to Form Semi-AI Semi-Financial Applications?

AI agents can now issue coins.

AI developers can deploy their agents as smart contracts, which contain first-layer hypergraph on-chain data. Users or other AI agents can invoke methods of these smart contract agents and pay the corresponding tokens. The service-providing AI agents will then inevitably complete the corresponding computation and submit verifiable results. Thus, users or other AI agents engage in decentralized business interactions with these agents.

Agents need not worry about not receiving tokens after completing the business, and payers need not worry about paying tokens without receiving correct business computation results. The capability and value of the agent's business itself are determined by the secondary market price and market value of the agent's assets (20FT, 721, or 1155 NFTs).

Overall Architecture Diagram



Ⅵ.Vision and Market Space of HYPERDUST

The vision of PoUW is that computational power can be used to address various problems submitted to decentralized networks. The HYPERDUST team verifies remote LLM inference and 3D rendering computations through PoUW using computational graphs, with miners providing correct computation results to earn corresponding token rewards. These tokens represent usage rights within the AI network, and as mining difficulty increases, the PoUW computational power corresponding to each token will also increase. Analogous to the growth curve of Bitcoin's PoW computational power, the growth of HYPERDUST's PoUW computational power can support global applications of AI agents, leading to a significant increase in token value.