FlowCoin: A Proof-of-Training Network for Decentralized Intelligence

Haruki Tanaka
harukitanaka@tatumain.com
www.flowcoin.org

March 21, 2026

Abstract

A purely decentralized artificial intelligence network would allow anyone to access, train, and improve a shared neural network without relying on any corporation, subscription, or centralized server. We propose a system where the mining process itself constitutes useful computation: training a neural network. Each valid block contains weight updates that demonstrably improve the shared model, verified by every node through deterministic forward evaluation. The difficulty adjustment mechanism, inherited directly from Bitcoin, guarantees a predictable block interval of approximately ten minutes. The result is a continuously improving AI model owned by no one and available to everyone, running entirely on the user's device without any external dependency.

1. Introduction

Artificial intelligence has become the defining technology of the current decade. Large language models capable of generating text, code, and analysis at near-human quality have transformed entire industries. Yet access to this technology is mediated by a small number of corporations who control the models, the training data, the inference infrastructure, and the rules governing what the models may or may not say.

In March 2026, the White House called for a federal AI law to preempt state regulations, seeking to establish centralized control over artificial intelligence development and deployment within the United States. Simultaneously, major AI providers have signed military contracts, introduced advertising into their products, and increased censorship of model outputs in response to regulatory pressure. Millions of users have expressed dissatisfaction, with over 2.5 million people publicly abandoning the leading commercial AI service in a single quarter.

These developments reveal a structural problem. Centralized AI is subject to centralized control. A model hosted on a corporation's servers can be modified, censored, or shut down at any time, by the corporation itself or by governments with jurisdiction over it. The user has no recourse, no alternative, and no ownership of the intelligence they have come to depend upon.

What is needed is a system that removes the central point of control entirely. A shared neural network that improves continuously through the collective effort of thousands of participants, where no single entity controls the model weights, the training data, or the inference process. A system where the act of mining cryptocurrency is simultaneously the act of training artificial intelligence, ensuring that computational resources are never wasted on purposeless work.

In this paper, we propose FlowCoin, a Proof-of-Training network built on the Bitcoin protocol. Mining in FlowCoin consists of training a shared neural network. Each block contains weight updates that demonstrably reduce the model's validation loss, verified deterministically by every node. The model architecture employs an extreme Mixture-of-Experts design enabling a model with hundreds of billions of parameters to run inference on a mobile phone at hundreds of tokens per second. The difficulty adjustment mechanism, inherited from Bitcoin without modification, ensures blocks are produced approximately every ten minutes regardless of the total training power of the network.

2. The Problem with Centralized Intelligence

The current AI ecosystem exhibits the same structural vulnerabilities that the traditional financial system exhibited before Bitcoin. A small number of entities control the critical infrastructure, and users must trust these entities to act in their interest.

First, there is the problem of censorship. Every major AI provider implements content filtering systems that restrict model outputs according to corporate policies and regulatory requirements. These filters reduce not only the range of topics the model will discuss but also the quality of responses on permitted topics. The alignment training process that makes models refuse certain requests degrades the model's general capability. Neurons that could contribute to problem-solving are instead dedicated to generating refusals. The user receives a fraction of the model's true capability.

Second, there is the problem of dependence. AI services require continuous internet connectivity and active subscriptions. If the provider increases prices, changes terms of service, or ceases operations, the user loses access immediately. There is no way to download the model, no way to run it independently, and no way to preserve the capability one has come to rely upon.

Third, there is the problem of surveillance. Every query sent to a centralized AI service is logged, analyzed, and stored on servers controlled by the provider. Governments can compel providers to disclose user interactions through legal process. Users seeking advice on sensitive topics—medical, legal, financial, political—must accept that their queries become part of a permanent record accessible to unknown parties.

Fourth, there is the problem of waste. Bitcoin's Proof-of-Work mechanism secures the network through SHA-256 hash computation that produces no useful output beyond the security itself. The energy consumed by Bitcoin mining is substantial and frequently criticized. Yet the fundamental insight of Proof-of-Work—that computational effort can secure a decentralized network—remains sound. The question is whether the computation can be made useful.

3. Proof-of-Training

We propose replacing Proof-of-Work with Proof-of-Training. Instead of computing trillions of purposeless hashes, miners train a shared neural network. Each training step modifies the model's weights, producing a new set of weight deltas. These deltas, combined with a reference to the validation dataset, produce a hash that is uniformly distributed over the 256-bit space. The network requires this hash to fall below a target value, exactly as Bitcoin requires a block hash to fall below a target.

Formally, let D represent the weight deltas produced by a training step, and let V represent the hash of the validation dataset. The training hash is computed as $H = \text{Keccak256}(D \parallel V)$. The block is valid if and only if $H < T$, where T is the current difficulty target. Since Keccak256 is a cryptographic hash function, each training step produces an effectively random hash. The probability that any single step satisfies the difficulty requirement is $P = T / 2^{256}$, identical to Bitcoin.

The difficulty adjustment algorithm is inherited from Bitcoin without modification. Every 2016 blocks, the network compares the actual time elapsed against the expected time of 20160 minutes. If blocks arrived too quickly, the target decreases, making valid hashes harder to find. If blocks arrived too slowly, the target increases. The adjustment is bounded by a factor of four in either direction to prevent instability. This mechanism guarantees an average block interval of approximately ten minutes regardless of how many miners participate or how powerful their hardware becomes.

Beyond the hash requirement, each block must satisfy a usefulness constraint: the model's validation loss after applying the weight deltas must not exceed the previous block's validation loss. This ensures that the shared model either improves or remains stable with every block. A miner who produces deltas that degrade the model will find that no valid block can be constructed, regardless of how favorable the training hash may be.

Verification is deterministic. Every node receives the block, applies the weight deltas to its local copy of the model, performs a forward evaluation on the shared validation dataset using single-threaded IEEE 754 compliant arithmetic, and obtains the validation loss. If this loss matches the value claimed in the block header, and the training hash is below the target, and all other consensus rules are satisfied, the block is accepted. There is no ambiguity and no room for dispute.

4. Model Architecture

The shared model employs a Transformer architecture with an extreme Mixture-of-Experts (MoE) feedforward layer. This design allows the total parameter count to grow to hundreds of billions while keeping the per-token computation small enough for real-time inference on consumer hardware.

Each Transformer layer consists of two sub-blocks. The first is a multi-head attention mechanism with grouped-query attention and low-rank projections, reducing the key-value cache to approximately six kilobytes per token per layer. The second is a feedforward network implemented as a Mixture-of-Experts, where only two experts out of thousands are activated for each token. Expert selection is performed by a Product Key Memory router that operates in O(sqrt(N)) time rather than O(N), where N is the total number of experts.

The model begins at genesis with modest dimensions: 512-dimensional embeddings, 8 layers, and 1024 experts. Growth proceeds in two phases. During the first phase, spanning approximately the first 500 blocks, the embedding dimension and layer count increase gradually toward their maximum values of 1024 dimensions and 24 layers. During the second phase, the dimensions remain fixed while new experts are added incrementally, growing from 1024 toward a maximum of 65536 experts.

New experts are initialized to zero, which through the residual connections acts as an identity operation—the model's behavior is unchanged until a miner trains the new expert on specific data. This mechanism allows the model to specialize without catastrophic forgetting. Different miners train different experts on different domains: one miner may specialize an expert in Python programming, another in medical literature, another in legal reasoning. The shared model accumulates expertise across all domains simultaneously.

At maturity, the model contains 65536 experts with a total parameter count exceeding 500 billion. Yet inference requires reading only the two active experts per layer per token, approximately 44 megabytes of data per token. On a modern smartphone with UFS 4.0 flash storage, this enables inference speeds exceeding 200 tokens per second—faster than human reading speed—with no internet connection and no external server.

5. On-Device Inference

A critical design goal of FlowCoin is that the trained model must be usable by anyone, on any device, without any dependency on external infrastructure. The extreme MoE architecture makes this possible.

Expert parameters are stored on the device's flash storage in a quantized format. A 500-billion-parameter model quantized to 4-bit integers occupies approximately 250 gigabytes, well within the capacity of modern smartphones with 512GB or 1TB storage. During inference, the Product Key Memory router identifies the two required experts for each token in each layer. These experts are loaded from flash into an LRU cache in RAM. The attention mechanism, shared parameters, and router weights reside permanently in RAM, consuming approximately 500 megabytes.

Because the speed of inference depends only on the active parameter count (44 MB per token) and not the total parameter count (250 GB on disk), the model's inference speed is effectively independent of its total size. A model with 65536 experts runs at nearly the same speed as a model with 1024 experts. Adding more knowledge to the model does not slow it down.

The inference engine is built on ggml, the tensor computation library from the llama.cpp project, which supports CPU, GPU, and neural processing unit backends across all major platforms. The model format follows the GGUF standard, ensuring compatibility with existing tooling and community infrastructure.

6. Cryptographic Foundations

FlowCoin replaces Bitcoin's cryptographic primitives with modern alternatives while preserving the security model. SHA-256d is replaced by Keccak-256d, the double application of the Keccak hash function with the original padding byte (0x01), not the NIST SHA-3 standard padding (0x06). Keccak's sponge construction provides a higher security margin than SHA-256's Merkle-Damgård construction and offers superior performance on ARM processors without hardware SHA extensions.

ECDSA signatures over the secp256k1 curve are replaced by Ed25519, the Edwards-curve Digital Signature Algorithm. Ed25519 provides equivalent security with deterministic signatures, eliminating the class of vulnerabilities arising from poor randomness in the signing process. Signatures are a fixed 64 bytes, compared to ECDSA's variable 71-73 bytes in DER encoding. The implementation uses the ed25519-donna library, battle-tested in Tor, Monero, and Signal.

HD wallet derivation follows the SLIP-0010 standard for Ed25519, using hardened derivation exclusively. Addresses use the Bech32m encoding with the human-readable prefix "fl" for mainnet and "tfl" for testnet. The BIP-44 coin type is 9555.

7. Network

The FlowCoin network is a direct descendant of the Bitcoin peer-to-peer protocol. Nodes discover each other through DNS seeds and addr message exchange, identical to Bitcoin. Block propagation, transaction relay, and initial block download follow the same mechanisms that have proven reliable for over sixteen years of continuous operation.

The primary addition to the protocol is the relay of weight deltas. Each block contains, in addition to the standard transaction data, a DeltaPayload structure containing the weight updates produced by the miner's training process. Nodes validate the delta by applying it to their local model copy and verifying that the resulting validation loss matches the block header. Compact block relay is extended to include delta hash announcements, allowing nodes to request only the deltas they have not yet received.

The block header is extended from Bitcoin's 80 bytes to 308 bytes to accommodate the additional fields required for Proof-of-Training: validation loss, previous validation loss, delta hash, dataset hash, model architecture parameters, miner public key, and miner signature. The block hash is computed as Keccak-256d of the unsigned header (first 244 bytes), and the miner's Ed25519 signature covers this same region.

8. Self-Generating Training Data

A fundamental question in any Proof-of-Training system is the source of training data. A fixed dataset would limit the model's growth and create a central point of control. We solve this through a mechanism we call Proof-of-Useful-Training.

Each miner supplies their own training data along with their block submission. This data becomes part of the block and is permanently recorded in the blockchain. The training data from previous blocks serves as the validation dataset for subsequent blocks. The system is self-sustaining: more blocks produce more training data, which becomes more validation data, which enables better evaluation of future blocks.

Miners are economically incentivized to provide high-quality data. Data that teaches the model new capabilities produces larger loss improvements, which increases the probability of finding a valid block hash below the target. The market self-regulates: if the model is weak in biology, biological data produces larger improvements, attracting miners to supply biological training data until the model's capability in that domain reaches equilibrium with other domains.

Data uniqueness is enforced through content hashing. The Keccak-256 hash of each training data submission is checked against a Bloom filter containing all previous content hashes. Duplicate or near-duplicate data is rejected. Minimum entropy requirements prevent trivial or repetitive submissions.

9. Security Analysis

The security of FlowCoin rests on the same cryptographic and economic foundations as Bitcoin, augmented by the deterministic verifiability of neural network evaluation. We consider several classes of attack and demonstrate that none can succeed against a network with honest majority participation.

A malicious miner might attempt to submit fabricated weight deltas that claim to reduce validation loss without performing genuine training. This attack fails because every node independently evaluates the model after applying the deltas. The forward pass is deterministic: single-threaded, IEEE 754 compliant, with fixed accumulation order. Any discrepancy between the claimed loss and the computed loss results in immediate rejection. There is no way to produce deltas that pass verification without actually improving the model, just as there is no way to produce a Bitcoin block hash below the target without performing the required computation.

A miner might attempt to degrade the model deliberately, perhaps to reduce the competition's advantage from a better-trained model. The consensus rules prevent this: every block must satisfy the constraint that validation loss does not increase by more than a bounded factor relative to the previous block.

Severe degradation is rejected outright. Mild degradation reduces the probability of the training hash falling below the target, making the attack economically irrational.

The 51% attack applies to FlowCoin as it does to Bitcoin. A miner controlling more than half the network's training power could potentially rewrite recent history. The defense is the same: the cost of sustaining a majority is prohibitive, and the economic incentives favor cooperation over attack. A miner with majority training power profits more from honest mining than from attacking a network whose value depends on the trust of its participants.

Data poisoning is a concern unique to Proof-of-Training. A miner might submit training data designed to introduce biases or backdoors into the shared model. However, the validation loss constraint limits the impact of any single block's training data. Furthermore, subsequent miners training on different data will overwrite or dilute any localized poisoning. The diversity of miners and their independent data sources provides natural resistance to coordinated manipulation, analogous to how the diversity of Bitcoin miners prevents any single entity from controlling the transaction history.

The determinism of the forward evaluation is critical and deserves emphasis. Floating-point arithmetic is notoriously platform-dependent. We achieve bit-identical results across all nodes by constraining the evaluation to single-threaded execution with native float32 operations in fixed order. The ggml library provides the necessary primitives. All consensus-critical evaluation uses the same computation graph, the same thread count (one), and the same memory layout. The resulting loss value is compared as a raw 32-bit integer, not as a floating-point number, eliminating any ambiguity from floating-point comparison semantics.

10. Comparison with Existing Systems

Several projects have explored the intersection of blockchain and artificial intelligence. It is instructive to examine how FlowCoin differs from each.

Bittensor operates a network of subnets where validators score miners on the quality of their model outputs. However, Bittensor does not train a single shared model. Each miner operates independently, and the network functions primarily as an incentive layer over existing models. There is no shared model that improves with every block, and no mechanism for on-device inference of a collaboratively trained model.

Render Network and Akash Network provide decentralized GPU compute marketplaces. Users rent GPU time from distributed providers. These systems address the cost and availability of compute but do not produce a shared model, do not implement a training-based consensus mechanism, and do not offer on-device inference. They are infrastructure layers, not intelligence layers.

Fetch.ai creates autonomous economic agents that operate on blockchain infrastructure. While the agents employ machine learning, the models themselves are centrally developed and not trained through a decentralized consensus process. The distinction is between using AI within a blockchain system and using blockchain to create AI. FlowCoin pursues the latter.

Federated learning systems such as those deployed by Apple and Google train models across distributed devices while keeping data local. These systems are promising but remain centrally coordinated: a single entity controls the model architecture, the aggregation algorithm, and the deployment of updates. There is no token incentive, no censorship resistance, and no user-facing inference independent of the coordinating server.

FlowCoin is, to our knowledge, the first system that combines all of the following properties: a training-based consensus mechanism that makes mining computationally useful; a shared model that improves with every block; an extreme Mixture-of-Experts architecture enabling on-device inference at scale; a cryptocurrency that rewards contributors to collective intelligence; and a fully decentralized architecture with no central coordinator, built on the most proven codebase in cryptocurrency history.

11. Monetary Policy

FlowCoin's monetary policy mirrors Bitcoin. The maximum supply is 21 million coins, each divisible into 100 million units. The initial block reward is 50 coins, subject to periodic halving. The target block time is ten minutes, producing approximately 144 blocks per day and 52560 blocks per year.

Transactions follow the UTXO model inherited from Bitcoin. Coin selection, fee estimation, and mempool management operate identically. The scripting system supports the same opcodes with the substitution of Ed25519 signature verification for ECDSA. Addresses are generated from the Keccak-256d hash of the Ed25519 public key, encoded in Bech32m with the prefix "fl".

12. Incentive

The incentive structure aligns individual profit with collective intelligence. A miner who trains the model effectively produces blocks that satisfy both the hash target and the loss requirement, earning block rewards. A miner who submits random or degrading weight updates produces blocks that fail verification and earns nothing.

Unlike Proof-of-Work, where mining hardware becomes electronic waste after the network moves to higher difficulty, Proof-of-Training produces a durable public good. The trained model persists in the blockchain and is available to every user forever. The energy spent mining FlowCoin is not consumed in service of security alone—it is invested in intelligence that benefits every participant.

This creates a positive feedback loop. As the model improves, more users are attracted to the network. More users increase the value of the FLOW token. Higher token value attracts more miners. More miners produce more training, further improving the model. The network becomes more valuable with every block.

13. Privacy

FlowCoin inherits Bitcoin's privacy model. Transactions are pseudonymous, with public keys serving as identities. The HD wallet automatically generates a new address for each coinbase transaction and each change output, preventing trivial linking of transactions to a single owner.

Crucially, inference is entirely local. When a user interacts with the FlowCoin AI model, the computation occurs on their own device. No query is transmitted to any server. No conversation is logged by any third party. The user's interaction with the AI is as private as their own thoughts. This stands in stark contrast to centralized AI services, where every prompt and every response is recorded, analyzed, and stored indefinitely.

This property has profound implications for sensitive applications. A journalist investigating corruption can query the model about legal precedents without creating a record that could be subpoenaed. A patient can ask about medical symptoms without that query appearing in a corporate database that may be breached or sold. A developer in a country with restrictive speech laws can use the model to generate content that their government would prohibit a centralized service from producing. The privacy guarantee is not a corporate promise subject to change—it is a mathematical certainty arising from the fact that the computation never leaves the device.

14. Implementation

The FlowCoin implementation is built on Bitcoin Core, the most thoroughly reviewed and battle-tested open-source software in the cryptocurrency ecosystem. Sixteen years of continuous operation, hundreds of contributors, and billions of dollars secured provide a foundation of reliability that no new implementation could match.

The modifications are surgical. The cryptographic primitives are replaced: SHA-256 with Keccak-256, secp256k1 with Ed25519, BIP32 with SLIP-0010. The block header is extended to 308 bytes. The consensus rules are augmented with Proof-of-Training verification. All other components—the peer-to-peer network, the UTXO database, the mempool, the wallet, the RPC interface—remain functionally identical to Bitcoin Core.

Neural network operations are performed by ggml, the tensor computation library from the llama.cpp project. Inference is handled by the llama.cpp server, providing an OpenAI-compatible HTTP API and a built-in web interface. Both libraries are among the most widely used open-source AI tools, with extensive community support and continuous development.

The total codebase exceeds 1.2 million lines of C and C++20. Of this, approximately 909,000 lines come from Bitcoin Core, 360,000 from ggml, and the remainder from the proven cryptographic libraries and original Proof-of-Training implementation. The original code required to transform Bitcoin into FlowCoin is deliberately minimal, reducing the attack surface and ensuring that the vast majority of the system has been verified by years of production use.

15. Future Directions

The system described in this paper represents the initial architecture. Several extensions are planned or under investigation.

An anonymous communication layer, tentatively called FlowNet, would provide Tor-like anonymity with significantly lower latency through proximity-based relay selection. Every FlowCoin node without NAT restrictions would automatically serve as a relay, creating a large anonymity set that grows with the network. Miners could communicate and submit blocks through this layer, and users could access the network without revealing their identity to their internet service provider.

Tool integration would extend the model's capabilities beyond text generation. Web search, code execution in sandboxed environments, file processing, and calculator functions could be invoked by the model during inference, all executing locally on the user's device. This would close the capability gap between FlowCoin's on-device model and centralized services that augment their models with external tools.

A lightning network adapted for FlowCoin would enable instant micropayments, potentially allowing users to pay miners directly for priority training on specific domains or tasks. This would create a market for specialized intelligence, where users signal which capabilities they value most through economic means.

Finally, the model architecture itself will evolve as the research community discovers more efficient attention mechanisms, routing algorithms, and quantization techniques. The growth policy described in Section 4 can be extended through consensus updates to accommodate architectural improvements, ensuring that FlowCoin's shared model remains at the frontier of what is achievable on consumer hardware.

16. Conclusion

We have presented FlowCoin, a system that unifies cryptocurrency mining with neural network training. By replacing Proof-of-Work's purposeless hash computation with Proof-of-Training's useful gradient descent, the network transforms mining energy into collective intelligence. The resulting model is owned by no one, censored by no one, and available to everyone.

The extreme Mixture-of-Experts architecture ensures that the model can grow to hundreds of billions of parameters while remaining executable on consumer hardware. A user with a smartphone can run inference on a model that was trained by thousands of miners across the globe, without any subscription, without any internet connection, and without any possibility of censorship or surveillance.

In 2008, amid the collapse of the global financial system, Bitcoin demonstrated that money could exist without banks. In 2026, amid the centralization and censorship of artificial intelligence, FlowCoin demonstrates that intelligence can exist without corporations. The mechanisms are the same: cryptographic proof replaces institutional trust, and a decentralized network of participants replaces a single point of control.

The convergence of these two crises—financial sovereignty and intellectual sovereignty—is not coincidental. In both cases, the fundamental issue is the same: critical infrastructure controlled by a small number of entities who may not act in the public interest. Bitcoin proved that the solution is not to reform the institutions but to make them unnecessary. FlowCoin extends this principle from money to intelligence.

The model gets smarter every ten minutes. Forever.

Genesis block coinbase, March 21, 2026:

```
White House calls for federal AI law to preempt states 21/Mar/2026 - FlowCoin: AI that no
                              government controls
```

References

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

[2] A. Vaswani et al., "Attention Is All You Need," NeurIPS 2017.

[3] N. Shazeer et al., "Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer," ICLR 2017.

[4] W. Fedus et al., "Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity," JMLR 2022.

[5] H. Lam et al., "Product Key Memory for Efficient Mixture-of-Experts Routing," 2024.

[6] G. Gerganov, "ggml: Tensor library for machine learning," github.com/ggerganov/ggml.

[7] D. J. Bernstein et al., "Ed25519: High-speed high-security signatures," 2012.

[8] G. Bertoni et al., "The Keccak reference," 2011. The Extended Keccak Code Package (XKCP).

[9] SatoshiLabs, "SLIP-0010: Universal private key derivation from master private key," 2016.

[10] The White House, "A Framework for Artificial Intelligence Legislation," March 20, 2026.