Department of Computer Science
University of Hong Kong
Final Year Project

Improvement of Blockchain Consensus Algorithm by Integrating Distributed Machine Learning

Sangwon Park (3035556060)
Chungha Yu (3035553135)
Supervisor: Dr. Chow, Kam Pui

Abstract

With the rise of the blockchain industry, concerns regarding high energy consumption in Proof of Work (PoW) blockchains have become a hotly debated topic. Various algorithmic solutions have been suggested to address this issue, but they have often failed to maintain the competitive market nature of PoW, which is one of its key benefits. This project's main objective is to tackle high energy consumption by converting wasted energy into a resource for machine learning training. Currently, the project involves a novel algorithm called Proof of Machine Learning (PoML), which operates alongside virtual machines. These servers serve as node providers, facilitating the distributed handling of machine learning tasks. However, beyond the scope of this project, further development is required to ensure widespread network efficiency among nodes and to deliver a seamless front-end user experience.

Acknowledgment

We would like to express our gratitude to our supervisor, Dr. Chow, Kam Pui who gave us valuable insights and guidance. The project could not have been completed without the help and development of Sangwon Park and Chungha Yu who led such project to be implemented.

Contents

List of Figures

List of Tables

Abbreviations

PoW - Proof of Work

PoS - Proof of Stake

PoML - Proof of Machine Learning

P2P - Peer-to-Peer

SGD - Stochastic Gradient Descent

GCP - Google Cloud Platform

HTTP - Hypertext Transfer Protocol

HTTPS - Hypertext Transfer Protocol Secure

FTP - File Transfer Protocol

TCP - Transmission Control Protocol

UDP - User Datagram Protocol

API - Application Programming Interface

MNIST - Modified National Institute of Standards and Technology database

1. Introduction

Section 1.1 explains the background of the topic and problem of the current platform. Section 1.2 depicts the motivation of the project. Section 1.3 illustrates the objectives of the project. Section 1.4 describes the scope and deliverables, and Section 1.5 outlines the research gap and significance followed by Section 1.6 with the outline of the report.

1.1 Background

Since the introduction of Bitcoin by Satoshi Nakamoto in 2008, blockchain has evolved to be applicable not only in the finance, government, and education industries but also as the pillar of the new web 3.0 protocol. As blockchain network does not have a central authority but instead adapt shared database method on peer-to-peer network, it increases trust, security, transparency, and traceability of data shared across a network; hence, favored by industries such as financial services, government, and even insurance. [1,2]

However, there is one major downside of the whole system. High computational power results in high energy consumption as the blockchain tries to establish a competitive market. According to the University of Cambridge Electricity consumption index, it is estimated that blockchain consumes electricity at an annualized rate of 127 terawatt-hours, which contributes 0.3% of global annual carbon emission [3,4]. The reason for such high consumption of energy is due to the nature of bitcoin's Proof of Work (PoW) consensus algorithm.

With the rise of AI and machine learning, having access to powerful computing power is important for efficiency and speed. While machine learning that uses a centralized server has been the conventional approach, experiments conducted show that a distributed approach to machine learning accelerates the training of modest to large sized models [5]. Popular models such as GPT and BERT are large machine learning models. Therefore, an algorithm that utilizes the high computational power for machine learning can potentially solve both downsides of blockchain and increase machine learning access in society.

1.2 Motivation

There is a clear need to explore alternative consensus algorithms that can achieve decentralization in an energy-efficient manner. PoW possess and advantage of formulating a competitive market but in return it wastes energy. If the project can leverage computing power for training machine learning datasets, then such energy is converted to be used in useful work

[6]. The project seeks to implement and evaluate a more environmentally sustainable layer 1 consensus protocol.

1.3 Objective

The objective of this project is to create a layer 1 blockchain network with a novel consensus algorithm known as Proof of Machine learning (PoML). The new blockchain network tackles the problem of high energy consumption by utilizing computational power for training datasets, whereas PoW wastes energy by solving mathematical problems solely for the creation of new hashes. The project aims to achieve this by completing 3 main objectives listed below:

Objective 1: Developing a blockchain network with a new algorithm

Objective 2: Supporting peer-to-peer (P2P) data transfer

Objective 3: Enabling distributed machine learning on blockchain nodes

Further details are presented throughout the report.

1.4 Scope and Deliverables

The project makes three contributions. First, it explains the development of a blockchain network with PoML. This network allows tracking the training process, rewarding node providers with tokens, and rating the computational power efficiency of the nodes. Second, the project introduces a peer-to-peer (P2P) network for transferring training datasets. As large data such as training data cannot be sent in blockchain networks due to block size, the project implements a P2P network for high-speed file transfer. Lastly, participating nodes will be simulated in a cloud environment. Using Google Cloud Platform (GCP), servers will be set up to receive training data and models. The result will be a blockchain network that provides distributed machine learning services, and the network can be accessed through a command line or the terminal.

1.5 Research Gap and Significance

This project will allow existing node providers that are wasting computational power to provide machine learning services to society. These node providers will be given tokens for their services. The project focuses on creating a consensus algorithm which maintains the competitive market nature of Proof of Work, adopt deterministic random selection from Proof of Stake, and modify the token staking to that of rating based on computational power of the node provider.

1.6 Outline of the Report

The report consists of 5 sections. Section 2 depicts the methodology of the project, section 3 illustrates the current development progress and results, section 4 discusses future implementation, and section 5 concludes the overall report.

2. Methodology

The platform consists of 3 major aspects: a blockchain network, a peer-to-peer network, and a distributed machine learning algorithm. The combination of these three aspects allows users to train custom data using our platform and node providers to get tokens as a reward for providing computational power.

Section 2.1 summarizes our development platform. Sections 2.2 to 2.4 describe the procedures of how the blockchain network, peer-to-peer network, and distributed machine learning algorithm will be implemented. Section 2.5 shows how the platform will be tested to validate the effectiveness of the model.

2.1 Technologies and Platform

The development environment consists of two main parts: programming language and platform. For the development of the blockchain network and P2P network, JavaScript will be used as the programming language. Considering the requirements of the project, JavaScript is a suitable choice that holds solid documentation and libraries. Moreover, JavaScript is a language that the team was most familiar with. For the implementation of the distributed machine learning algorithm, Python will be used due to its strong management of large data and compatibility with TensorFlow. Google Cloud Platform (GCP) was selected to host virtual machines due to its well documented descriptions with the TensorFlow Federated library. The project will utilize Linux as the platform due to compatibility with Python, JavaScript, and GCP. Also, Linux holds strong community and documentation that will be helpful to solve further problems in the future.

2.2 Blockchain Network

To formulate a blockchain network that provides decentralized machine learning, the project will require a new form of consensus algorithm. The project will refer to the model of Proof of Work (PoW), a consensus algorithm used in Bitcoin, and Proof of Stake (PoS), a consensus algorithm used in Ethereum.

2.2.1 Proof of Work

Proof of Work (PoW) is heavily based on solving mathematical questions to generate new blocks. There is a repetitive process where miners continuously generate random hash until it matches the target value. This repetitive task results in high energy consumption and allows miners with higher computational power to earn rewards. The project plans to adapt the general flow of PoW yet plans to remove the repetitive process and adapt a random selection process.

2.2.2 Proof of Stake

Proof of Stake focuses on randomly selecting a validator according to how fresh and how many tokens one has staked. This design prevents high computational usage and provides fairness to the ecosystem. The project plans to adapt a deterministic random selection process based on the computational power rating of the stake node providers.

2.2.3 Proof of Machine Learning

Proof of Machine Learning (PoML) is a novel consensus algorithm that rewards tokens based on the usage of computational power for training dataset. PoML is a combination of PoW with Pos followed by a rating system which will measure how efficiently and effectively training has been conducted. Exact criteria to determine the rating has not been decided yet and is still under consideration.

2.3 Peer to Peer Network

Due to the size limit of blocks in blockchain, the project will implement another layer of a peer-to-peer network that is dedicated for transferring training datasets and returning the result transaction on the blockchain. Ideally, if there are a lot of node providers in the chain, it would be efficient to transfer large datasets using peer-to-peer network. BitTorrent is a great example in the existing market. BitTorrent is a peer-to-peer file sharing protocol that allows users to distribute large amounts of data and multimedia files over the internet [7]. The project will research deeper on the architecture of such protocol and implement the network for demo testing purposes.

2.4 Distributed Machine Learning platform

In order to replicate a real-life environment, the network will consist of multiple virtual machines hosted on Google Cloud Platform (GCP) that act as real-life node providers. Each of the virtual machines will represent a node provider and will receive tokens for providing machine learning services. Along with GCP, TensorFlow, an open-source machine learning library, will be used to process data for machine learning.

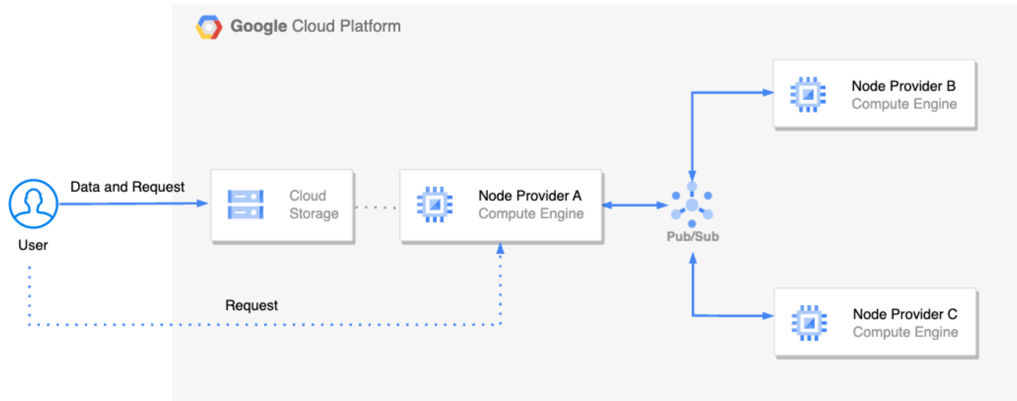

Figure 2. 1: Architecture of underlying distributed machine learning platform. Google Compute Engine, Cloud Storage, and Pub/Sub are used to distribute data and aggregate machine learning parameters.

The platform will work as follows. Once a user decides to request machine learning services, the user will have to either provide the data or request that machine learning be run on public data. If private data is used, a storage device will be provided. If public data is used, only the request will be sent to the compute engine. A server attached to the storage device will act as the central node and establish connection with the participating nodes on the network. Participating nodes that have TensorFlow's library of tools will be allocated customer data or public data and an initial model. The node will use the local data to train the initial model. The updated parameters will then be sent back to the server for consolidation. Once the parameters are received, the central server aggregates the models and forms a global model through a technique known as federated averaging.

2.4.1 Federated Averaging

While there have been numerous successful applications of deep learning that use stochastic gradient descent (SGD) for optimization, when SGD is applied to federated learning optimization, large rounds of training are required to produce good models on limited data [8]. To tackle this problem, Chen et al. [9] suggests that using large-batch synchronous SGD, where parameters of models are commonly initialized before distribution, outperforms asynchronous approaches where the parameters are independently initialized. When this approach is applied in a federated learning environment, studies show that combining the tuned parameters and averaging the models results in a significant decrease in loss [10].

Therefore, this project will use Federated Averaging (FA) to combine models received from participating nodes to form a global model. The system architecture design for FA is shown below (Figure 2.1). The training is conducted in three key steps:

1. The central server chooses an initial global model and broadcasts the model to participating client nodes.
2. Nodes receive a model that has common initialization across all nodes but are trained using local data that results in an updated model.
3. The updated models are sent back to the central server to create an aggregated global model for the next round of iteration.

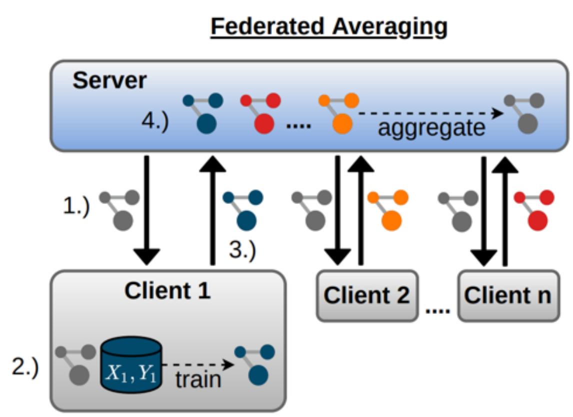

Figure 2. 2: The system architecture and data flow for Federated Averaging. Federated Averaging sends updated models. Figure adapted from [11].

2.5 Evaluation and Test Method

The project will conduct tests continuously based on three test methods: black box, white box, and gray box testing. During the development of each module such as consensus algorithm, blockchain network, and P2P network, continuous tests on internal structures will be conducted adapting white box testing. After the completion of the development, tests will focus on the functionality adapting black box testing.

To evaluate the effectiveness of the network, comparisons between locally run federated learning and cloud run federated learning will be tested. Identical data sets, optimizers, and parameters will be used across the tests. Once testing for errors is finished, model parameters will be adjusted and will be evaluated in terms of accuracy and time to completion.

Once all the modules are developed, the project will adapt gray box testing which is a combination of black and white testing to minimize bugs and potential error before deployment.

3. Current Progress and Preliminary Result

Section 3.1 illustrates the current development progress of blockchain network. Development of blockchain network consists of three major aspects: network layer, block design, and consensus algorithm. Section 3.2 depicts the test result on the performance of federated learning model in comparison to conventional machine learning.

3.1 Blockchain Network

3.1.1 Network Layer

The project managed to construct a blockchain network with a simple prototype of Proof of Stake. Currently the project stimulated an environment of blockchain when there are three devices connected via using different ports in one device. The project will be moving towards implementation of GCP where each device has unique public IP. Communication of node was implemented utilizing WebSocket protocol in peer-to-peer network. The project decided to utilize WebSocket protocol over conventional HTTPS as it supports a single persistent connection and bidirectional communication. Blockchain network requires continuous update of events such as new transactions or new block. If HTTPS protocol was used, the overall performance would be slow followed by multiple connection error as it does not support bidirectional communication.

However, for the communication between users and the blockchain, the project decided to use HTTPS request method. Hence, each node opens two ports where one is dedicated for blockchain network and the other for API connection. Users can access personal account details and perform transactions through API request. Details of further architecture design are depicted in figure 3.1.

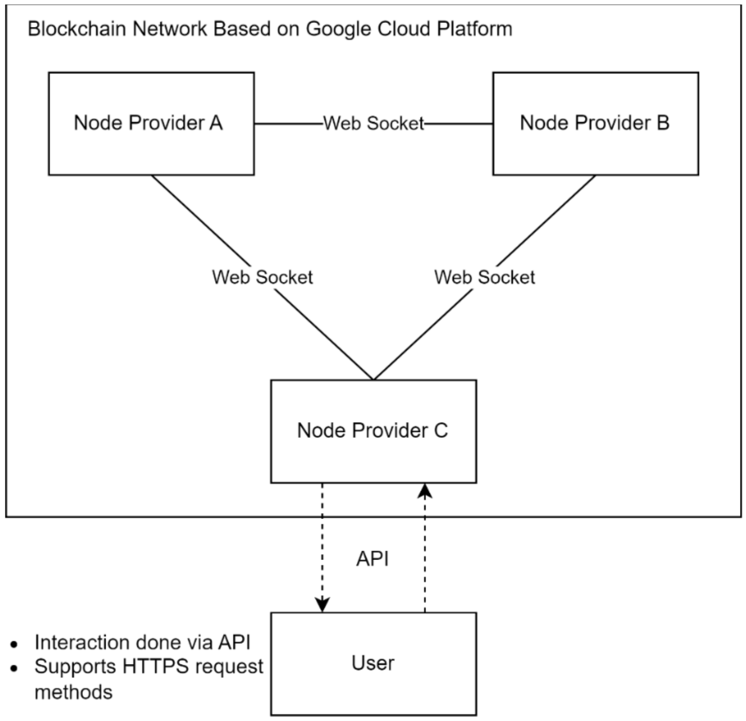

Figure 3. 1: Architecture design of PoML implemented Blockchain. Communication between nodes in blockchain are done through WebSocket and User interact blockchain with conventional API requests. Node Providers also holds communication with worker nodes via WebSocket.

3.1.2 Block Design

The project follows the standard format of block design by Satoshi Nakamoto. See left block design of Figure 3.2. The block consists of 5 variables in header: previous hash, timestamp, difficulty target, nonce, and merkle root. The combination of the values in 5 variables will determine the block's hash value. Blocks will now have a chain of links where the previous hash value is appended in the next block. Through this method, once a new block is added to the chain attacker is not able to change the content in new block and cannot append new block due to linkage in previous hashes [11].

Slight adjustment has been made in the header section where nonce has been removed and few variables have been added to body section. Modification has been made from Satoshi's block design in the body section where rating point and transaction information of training data is added. See right side block design of Figure 3.2. Such information will be used to determine the validator of the next hash and reward token as a reward.

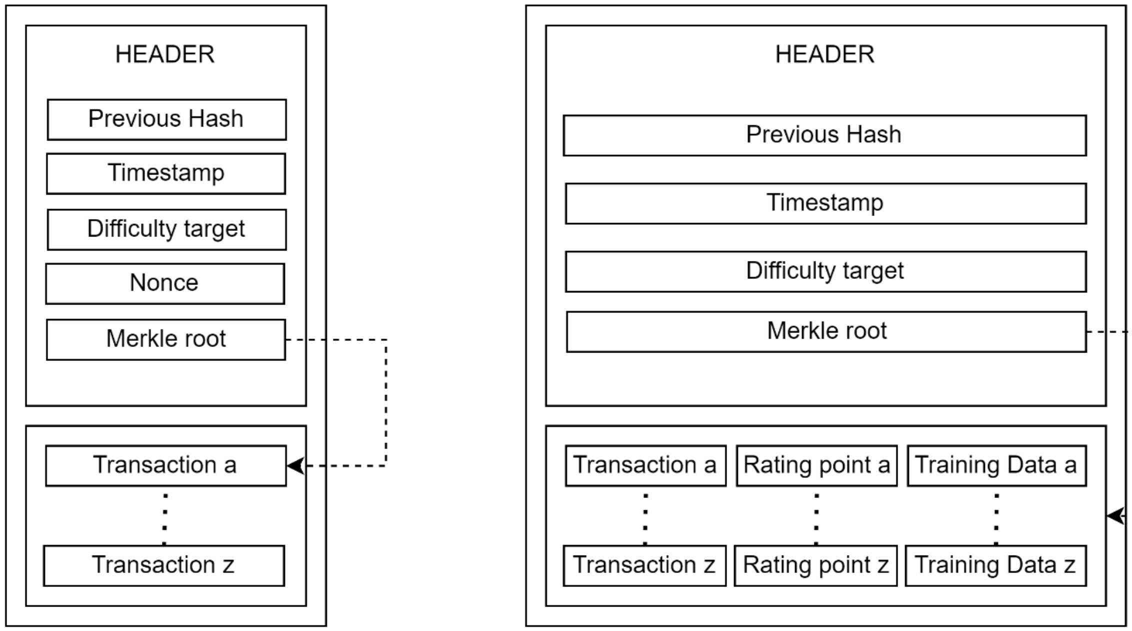

Figure 3. 2: Representation of Block design for both PoW and PoML. Left is design of PoW and right is design of PoML[11].

3.1.3 Consensus Algorithm

The current model holds a prototype Proof of Stake algorithm for validating the next block in the chain. It differs from actual Proof of Stake in Ethereum in three major ways. First, instead of calculating gas fee based on number of variables in code, the prototype imposes 0.1 for any transaction in the chain. Second, Ethereum can hold around 21,000 Gas in one block which is around 380 transactions [12]. However, the project's model holds 5 transactions per block. The block is designed to hold a small number of transactions so that unit tests can be conducted with more ease. Lastly, the prototype currently selects the validator with the most number of tokens instead of imposing deterministic random selection based on amount and token and age. As the project plans to change the selection process based on Proof of Machine Learning, exact replication of PoS was not needed. To fully implement Proof of Machine Learning consensus algorithm, the project has to decide what variables will be optimal to measure the efficiency and effectiveness on training a dataset. More research will be conducted on measuring the efficiency of training a task.

3.2 Distributed Machine Learning

Virtual machine instances using Linux were launched and libraries including NumPy, TensorFlow, and TensorFlow Federated were installed in each of the servers. Also, Pub/Sub, an asynchronous messaging service by Google, was used to transfer sample data between servers and ensure communication was reliable and scalable. Afterwards, to determine the initial performance of the federated learning model, the Sequential model in Keras with SGD was used as a base point. This model was placed into federated learning and conventional machine learning cloud environments. The MNIST dataset was used to train the models in both environments. The MNIST dataset is a collection of 70,000 images of handwritten numbers from 0 to 9 that includes 60,000 training images and 10,000 testing images [13].

Based on the test result, it is concluded that federated learning module has much lower accuracy than conventional machine learning modules. The MNIST dataset, a collection of 70,000 images of handwritten numbers from 0 to 9 that includes 60,000 training images and 10,000 testing images, was used to train the models.

Environment	Accuracy
Federated Learning	55.92%
Conventional Machine Learning	88.20%

Table 3. 1: federated learning and conventional machine learning accuracy comparison. Accuracy in [%], trained using the MNIST dataset. SGD with a learning rate of 0.01 and an epoch value of 10 is used in both environments.

There are significant differences in accuracy between the models. The conventional model has shown an accuracy of 88.2 % where federated learning is only 55.92%. See Table 3.1. However, there was some inaccurate setup leading to a false comparison. While conventional machine learning performed training 10 times on the entire dataset, the federated learning model split into 10 subsets and distributed to participating clients. Hence, the conventional method performed 9 more times of extra training compared to federated learning. After identifying the flaw in the test case, the project plans to conduct a fair comparison test to validate the performance in the future.

4. Discussion and Future Works

Section 4.1 outlines the project's schedule and plans. Section 4.2 discusses the improvements needed in Proof of Machine learning algorithm, peer to peer network.

4.1 Project Schedule

Objective	Deadline	Details and Learning Hours	Status

Preliminary Blockchain Network Setup	November 15th 2023	Set up blockchain network along with servers	Completed
Integration of Distributed Machine Learning system	Dec 31st 2023	Integrate Google Cloud Platform with TensorFlow	Completed
Peer to Peer Network Implementation	February 15th 2024	Develop a fully functional network layer that allows large data to set to be transferred at high-speed rate	In Progress
Proof of Machine Learning algorithm Implementation	Feb 29th 2023	Apply the consensus algorithm to the blockchain network	In Progress
Formulation of test unit	April 1st 2024	1. Initial and development stage 2. Final testing and running models	In Progress Pending
Deployment	April 15th 2024	Deploy the platform	Pending

Table 4. 1: Project Time Schedule. It outlines the objective, deadline, details and required hours, and status of the tasks required to finish the project.

Currently, we are on plan. The project is expected to face huge challenges in the development of Proof of Machine Learning as it is a novel approach without any prior examples. Therefore, 50 hours will be allocated for completion of Proof of Machine Learning algorithm. The project will heavily focus on completion of PoML as it is the key trait to our platform.

Moreover, the project aims to complete the integration of blockchain, peer-to-peer network, and distributed machine learning before 1st of April to allocate enough time for final testing.

4.2 Improvements Required

4.2.1 Proof of Machine Learning

A major challenge in the development of PoML is computational rating system: what are optimal criteria to rate how well dataset has been trained? How to measure and reward tokens based on the performance of machine learning? Simply weighing CPU and GPU performance is not a solution as for machine learning tasks, classification problems rely on accuracy, precision, F1 score and other metrics to measure performance. The project has realized such difficulty; hence, planning to implement a baseline rating system. Instead of giving rating every time according to the training done, node providers need to run the baseline test datasets and acquire the rating. Node providers who want to join the chain must run the baseline dataset too. Baseline dataset will be updated monthly to incorporate changes in node environment. As all the nodes in chain have done the testing with same datasets, the project believes comparison on duration, energy consumption, accuracy, precision, F1 score, and other metrics will be a fair method. The project requires further research on formulating exact criteria for evaluation of participating nodes. Through these criteria, nodes will be given tokens according to their performance. After the criteria are constructed, the project will further investigate if the given criteria are valid for such an evaluation through conducting tests on different computational environments.

4.2.2 Peer-to-Peer Network

Peer-to-peer network is not only applicable in blockchain construction but also data transfer. BitTorrent applies peer-to-peer network system so that users can download files not from one source but from various sources in the chained network. Such protocol allows faster download speed compared to HTTP and FTP [14]. The project will follow and implement data transfer layer based on [7]. Minor adjustment will be required for selection of network layer as [7] utilizes TCP and UDP whereas the project will use WebSocket. However, due to the limited time and scope of demonstration, the project might decide to simply transfer file from one end to another simply using file transfer protocol or SSH file transfer protocol.

4.2.3 Distributed Machine Learning

Further development will be necessary to ensure reliable data transfer from the user to the machine learning network and proper partitioning of the dataset among the participating nodes to enable efficient and scalable machine learning. Additionally, protocols related to the appropriate responses from the central server when a node seeks to join the network or when a user wishes to utilize the machine learning services will be thoroughly researched and further developed.

5. Conclusion

To tackle the problem of high energy consumption in blockchain with Proof of Work consensus algorithm, the project aims to convert the wasted energy to perform machine learning tasks. The project is expected to manage and convert wasted energy to useful energy via implementing a blockchain network with new consensus algorithm, peer-to-peer network for data transfer, and distributed machine learning system.

The system introduces a new form of consensus algorithm known as Proof of Machine Learning. Formulated from the mixture of Proof of Work, Proof of Stake, and computational power rating system, the blockchain allows the user to train custom data using the system and node providers get token as a reward for providing computational power. The project also constructs peer-to-peer network for high-speed transfer of user training data to the node. Once the data is collected by the node provider, the node will utilize distributed machine learning system to train large data in fast manner.

Currently, the project is in the development stage where the prototype PoS model of blockchain network is developed, and virtual machines that represent participating nodes have been setup to run distributed machine learning. Further development will be done on peer-to-peer network and PoML implementation.

Reviewing the current development, the prototype model of the blockchain network can move towards being deployable network. Minor adjustment followed by implementation of PoML will result in a functional blockchain. Empirical results suggest that conventional machine learning performs better but research into models that are customized for distributed learning is further needed. However, the team believes that further investigation into large scale models and optimized algorithms could improve accuracy.

For future development, peer-to-peer network will be implemented based on architecture of BitTorrent, intense research will be conducted to formulate criteria for computational power for PoML completion, and multiple tests will be conducted to ensure reliable and accurate distributed machine learning across servers.

References

[1] Stanford Graduate School of Business, "Popular Blockchain Use Cases Across Industries", 2022.

[2] IBM, "Blockchain Consulting and Services – IBM Blockchain", 2022.

[3] J. W. Kirkwood, "From work to proof of work: Meaning and value after blockchain," Critical Inquiry, vol. 48, no. 2, pp. 360-380, 2022.

[4] "Cambridge Bitcoin Electricity Consumption Index (CBECI)," Cambridge Centre for Alternative Finance, Cambridge, UK, 2023.

[5] Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, Marc'Aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, and Andrew Y. Ng. 2012. Large scale distributed deep networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'12). Curran Associates Inc., Red Hook, NY, USA, 1223–1231.

[6] M. Ball, A. Rosen, M. Sabin, N. Prashant, and Vasudevan, "Proofs of Useful Work," 2021.

[7] B. Cohen, "Incentives build robustness in BitTorrent," in Workshop on Economics of Peer-to-Peer systems, 2003.

[8] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In ICML, 2015.

[9] Jianmin Chen, Rajat Monga, Samy Bengio, and Rafal Jozefowicz. Revisiting distributed synchronous sgd. In ICLR Workshop Track, 2016.

[10] M. H. Brendan, E. Moore, D. Ramage, S. Hampson, and Blaise, "Communication-Efficient Learning of Deep Networks from Decentralized Data," *arXiv (Cornell University)*, Feb. 2016, doi: https://doi.org/10.48550/arxiv.1602.05629.

[11] F. Sattler, A. Marbán, R. Rischke, and W. Samek, "Communication-Efficient Federated Distillation," *arXiv (Cornell University)*, Dec. 2020, doi: https://doi.org/10.48550/arxiv.2012.00632.6

[12] S. W. Lo, Y. Wang, and D. K. C. Lee, "Bitcoin Mining and Python Programming Demonstration," in Singapore University of Social Sciences - World Scientific Future Economy SeriesBlockchain and Smart Contracts, pp. 73-98, 2021.

[13] Y. LeCun and C. Cortes, “MNIST handwritten digit database,” 2010.

[14] A. Kim, “Decentralized data sharing protocol,” Github, 2016.

Appendices