Proof of Learning: Towards a Practical Blockchain Consensus Mechanism Using Directed Guiding Gradients (Student Abstract)

Yongqi Wu, Xingjun Wang, Chen Chen, Guining Liu

SIGS, Tsinghua University
University Town of Shenzhen, Shenzhen, P.R. China
wuyq20@mails.tsinghua.edu.cn

Abstract

Since Bitcoin, blockchain has attracted the attention of researchers. The consensus mechanism at the center of blockchain is often criticized for wasting a large amount of computing power for meaningless hashing. At the same time, state-of-the-art models in deep learning require increasing computing power to be trained. Proof of Learning (PoL) is dedicated to using the originally wasted computing power to train neural networks. Most of the previous PoL consensus mechanisms are based on two methods, recomputation or performance metrics. However, in practical scenarios, these methods both do not satisfy all properties necessary to build a large-scale blockchain, such as certainty, constant verification, therefore are still far away from being practical. In this paper, we observe that the opacity of deep learning models is similar to the pre-image resistance of hash functions and can naturally be used to build PoL. Based on our observation, we propose a method called Directed Guiding Gradient. Using this method, our proposed PoL consensus mechanism has a similar structure to the widely used Proof of Work (PoW), allowing us to build practical blockchain on it and train neutral networks simultaneously. In experiments, we build a blockchain on top of our proposed PoL consensus mechanism and results show that our PoL works well.

Introduction

Researchers hope to modify the Proof of Work (PoW) (Nakamoto 2008) of digital currencies so that it can not only reach consensus, but also perform meaningful tasks. The most influential type of computing tasks that occupies the largest portion of computing power is deep learning. Previous works on designing PoL are mostly based on two ideas, Recomputation and Performance Metrics. However, both have problems which are hard to solve. In this work, we propose a consensus mechanism based on the observation of the opacity of most deep learning models. Thinking outside of Recomputation and Performance Metrics, our PoL simultaneously satisfies all of the properties that underpin large-scale blockchain applications. We also build blockchains on our PoL and demonstrate our conclusions through experiments.

Algorithm 1: PoL Proof Creation

Input: Trainset D, Initial Weights W,Batch size S, Requirements R
Output: Proof P, Weights W'

$W' \leftarrow W$
repeat
  $P \leftarrow$ Generate S different random numbers from $[1, |D|]$.
  $B \leftarrow \{D_{p_1}, ..., D_{p_S}\}$
  $W \leftarrow W'$
  $W' \leftarrow update\,(W, B)$
until CheckRequirements(W, W', R)==True
return P, W
Function CheckRequirements(W, W', R):
for all r_i in R **do**
  if $r_i \; mod \; 2 == 1$ **then**
    if $w'_{r_i/2} > w_{r_i/2}$ **then**
      return False
    end if
  else
    if $w'_{r_i/2} < w_{r_i/2}$ **then**
      return False
    end if
  end if
end for
return True

Proposed Mechanism

The cornerstone of PoW lies in the pre-image resistance of the hash function, i.e., for each leading 0 of the binary string N, there is no faster way to predict whether the result on this bit is 0 or 1, except to keep trying different inputs. We consider the opacity of neural networks can be the cornerstone of PoL. Formally speaking, in Stochastic Gradient Descent (SGD), for most deep learning models that satisfy opacity, we can safely assume that there is no faster way to predict how a randomly selected weight w_i will change other than a one-step training on a mini-batch. Just as PoW requires the hash values to satisfy the leading 0s, in PoL, we require a specific set of weights to satisfy the requirement of directed change. We call that requirement of Directed Guiding Gradients, denoted by R.

Algorithm 2: PoL Verification

Input: Trainset D, Proof P, Weights W,Batch size S, Requirements R, Testset E
Output: $\{True, False\}$

$B \leftarrow \{D_{p_1}, ..., D_{p_S}\}$
$W' \leftarrow update\,(W, B)$
if CheckRequirements(W, W', R)==True **and** $accuracy\,(W', E) > accuracy\,(W, E)$ **then**
  return True
else
  return False
end if

In Function CheckRequirements of Algorithm 1, we check whether the change in weights from W to W' satisfies the requirements R. R contains a set of random numbers in the range of 0 to $2|W|-1$, where $|W|$ denotes the number of learnable parameters. If r_i in R is an odd number, it means that weight indicated by ordinal number r_i divided exactly by 2 must decrease to satisfy the requirement and an even r_i means parameter denoted by $r_i/2$ is required to increase.

An honest miner runs Algorithm 1 to generate a valid proof. A mini-batch is denoted by a set of ordinal numbers P of the training set. Then, P is used to generate a mini-batch B. W records the data before training, and then updates W to get W' and checks whether the change from W to W' satisfies the requirement R. If satisfied, return P, W as a proof of PoL. R is generated from the previous block.

In Algorithm 2, PoL Verification, the validator reproduces the last step of training by simply update W on mini-batch B and check if change of weights from W to W' satisfy Requirement R. An additional testset E and test function $accuracy()$ is introduced to prevent spoofing.

Results

We use proposed PoL to train Resnet18 (He et al. 2016) for 10-class classification task, using CIFAR-10 (Krizhevsky and Hinton 2009) as the dataset. Size of a mini-batch is set to 4. Learning rate is 0.001. The requirements are generated from the hash digest of the previous block, with a collision avoidance. We use an Nvidia Geforce GTX 970 GPU. Detailed setup can be found in the supplementary.

Figure 1 shows the the accuracy of the deep learning model and block interval as blockchain grows. It can be seen from the figure that as the block length grows, i.e., consensus is reached, the accuracy of the deep learning model grows. This indicates that a portion of the computing power in the process of reaching consensus is effectively used to train the neural network. The bar chart in Figure 1 represents the block interval. It can be seen that due to the random nature of PoL itself, longer block intervals may occasionally occur, but, in general, the block rate remains stable when the accuracy of the deep learning model increases.

Conclusion and Future Work

In this work, we propose a PoL consensus mechanism that can both support large-scale blockchain applications such as digital currencies and use the consensus reaching computing power to train neural networks. Our biggest contribution is to find similarities between the opacity of deep learning models and the pre-image resistance of hash functions and build a PoL consensus mechanism on top of that opacity. Our proposed PoL is the only one among all similar works so far that does not rely on both Recomputation and Performance Metrics to reach consensus, and is structurally most similar to the tried-and-true PoW consensus mechanism.

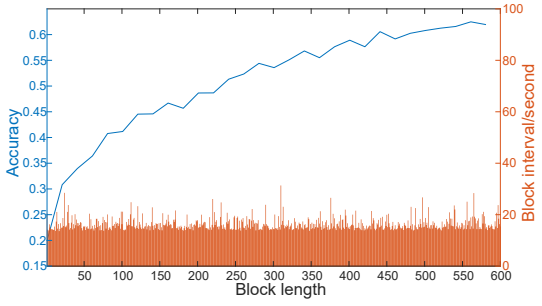

Figure 1: Block interval and accuracy as blockchain grows.

However, since we are still at the early stage of PoL, there is still much work to be done to bring PoL closer to the application. From the perspective of safety, whether there exists a strategic way to change the weights to successfully spoof and whether the latest results of deep learning interpretability will threaten the security of our proposed mechanism remain to be answered. In our proposed mechanism, we sacrifice training efficiency to satisfy all properties. However, whether there are alternatives to achieve the goal of PoL without sacrificing much efficiency, especially by modifying the structure of the neural network, will be one of our future research priorities. We also hope that future work can be extended to more areas, such as privacy protection, which is a pressing need in some application areas of deep learning.

Acknowledgements

This work was supported by Shenzhen STIC (WDZC 20200818121348001, KCXFZ202002011010487, SGDX20190918101201 69), Industrial Information Security Industry Application Support Platform and China National Key R&D Program(2017YFC0112500).

References

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. *Handbook of Systemic Autoimmune Diseases*, 1(4).

Nakamoto, S. 2008. Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf. Accessed: 2021-09-07.