Scaling RCL Language Models

Dr. Morten Middelfart[1] **Sam Martin** **Ben Martin**

Abstract

In this paper, we demonstrate that RCL scales sublinearly with dataset size and linearly with compute. We also indicate how RCL both differs from transformer architecture and promises better performance. RCL does not employ neural networks; unlike deep learning, RCL training time is a function of only dataset size and compute, and RCL model size is a function of dataset size. Additionally, we demonstrate that RCL continues to scale linearly whether or not we control for entropy.[2] Finally, by employing a shared-nothing architecture and running on CPU, RCL scales without theoretical limit as 1) dataset size increases or 2) processes are distributed across more machines.

1. Introduction

In February 2022, we introduced a new approach to machine learning called Random Contrast Learning (RCL). We have focused primarily on applying RCL to language because the vast majority of reasoning tasks can be expressed and evaluated in language.[3] Our research continues to demonstrate its general applicability to fields in and beyond natural language processing. Our most recent tests compare a Keras-framework deep learning neural network to RCL. RCL trains 81,343x faster, runs inference 82x faster, and produces a model that is 150x smaller, and achieves 99% recall.

The following sections illustrate the behavior of RCL as 1) dataset size increases and entropy is constant, 2) dataset size increases and entropy is dynamic, and 3) dataset size is constant while thread count increases. We then compare neural network scaling behavior with RCL.

1.2 Dataset

OpenAI's GPT-J training dataset (*The Pile*) is the current unofficial standard for training open source large language models. Thus far, we have used the 278MB EU Document English subset[4] to demonstrate RCL scaling behavior.

1.3 CPU Hardware

RCL runs faster on CPU than GPU. The following RCL tests took place on a physical machine with 2x32 Cores 2.3 GHz and 128 GB RAM Non-GPU enabled.

1.4 Method

In each experiment, models were built at least 5 times. Maximum and minimum times were removed from each set to avoid skew by outliers. The graphs below exhibit the average metrics of the remaining models. Additionally, we used 1MB to 278MB datasets to demonstrate scaling behavior across two orders of magnitude.

2. Experiments

Increasing Dataset Size with Constant Entropy

Given a domain in which most distinct tokens are known (e.g. words in a language), data from that

[1] Lumina, Tampa, FL, USA. Correspondence to: RCL@Lumina247.com

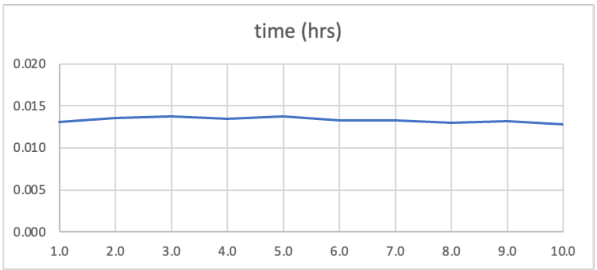
[2] RCL has theoretical implications for entropy in general. Given a domain has a finite number of distinct tokens, RCL scales linearly to sublinearly at scale.

[3] Scaling Neural Language Models https://arxiv.org/pdf/2001.08361.pdf

[4] EU Documents Dataset

domain will approach constant entropy as dataset size increases. RCL treats distinctive word combinations as tokens. In the training of RCL models, as entropy understood this way approaches constant, training time also approaches a constant value.

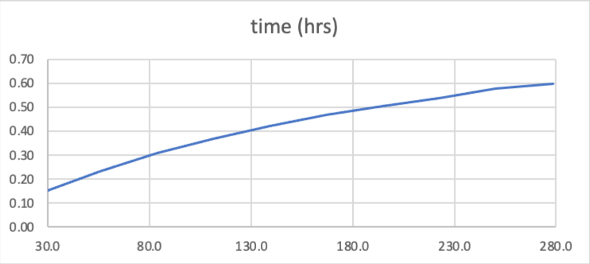
Figure 1



In the experiment illustrated in Figure 1, we began with a 1MB sample of the EU Document dataset, duplicated it up to 10x, and trained a new model at each 1MB interval. The training time remained constant as dataset size increased.

Increasing Dataset Size with Dynamic Entropy

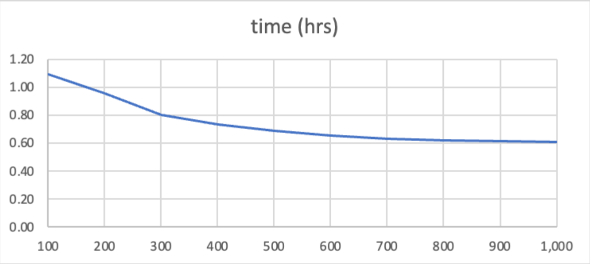
Figure 2



Next, we increased dataset size in 10% intervals of the EU Document dataset from 27.8MB to 278.3MB. Though entropy increased at each interval, training time increased sublinearly.

Increasing Thread Count

Figure 3



We then increased thread count in 100 thread increments. The graphs show that training speed improves as we distribute RCL processes across threads.

Neural Network Scaling Behavior

In deep learning, training time and inference time are independent variables and not functions of dataset size and entropy. Instead, neural scaling laws are far more complex. In Figures 5 and 6, we illustrate why one characteristic of neural networks — neurons per layer — poses problems at scale.

Figure 4

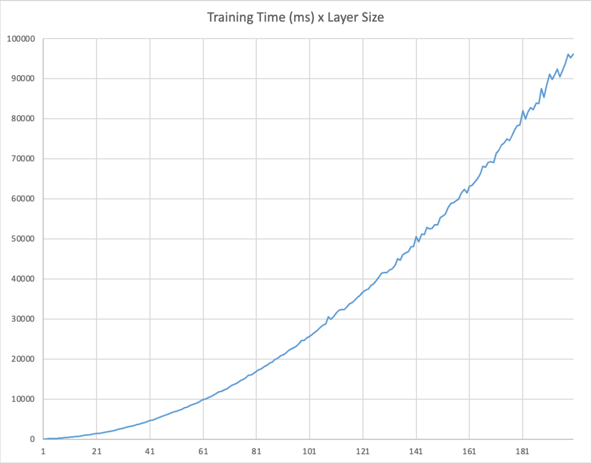
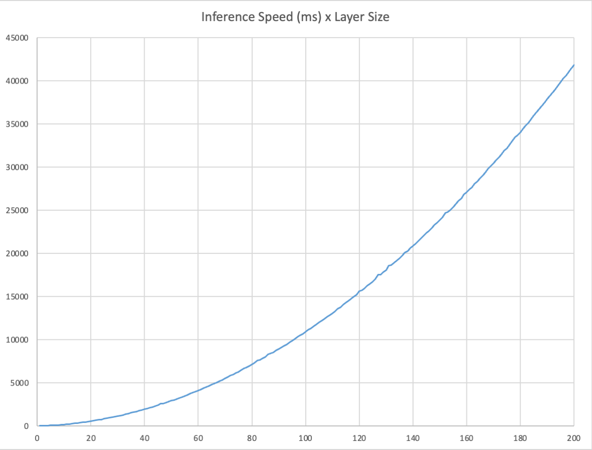


Figure 5



We increased the number of neurons per layer in the neural network. For each new neuron per layer, training time and inference time both increased exponentially.

3. Theoretical Implications

Inference time for a neural network increases exponentially as the number of neurons per layer increases. This is because most neurons of each layer connect to all neurons in the adjacent layers. In contrast, the nodes of an RCL model only ever have one parent. This means that inference time will scale sublinearly as model size increases. Subsequently, though our current experiments show 82x improved inference speed at small scales, RCL models likely run at orders of magnitude faster inference speeds when compared to neural models at larger scales.

4. Conclusion

RCL reduces the number of independent variables in the machine learning process: The approach employs a CPU-based shared-nothing architecture that scales sublinearly with dataset size and linearly with compute, without theoretical limits for either. Its unique approach minimizes the impact of entropy and maintains close to constant inference speed at scale. RCL outperforms neural networks in training speed, inference speed, model size, and recall and promises orders of magnitude improved performance for state-of-the-art machine learning systems.

Updated July 5, 2022