Proving the Lottery Ticket Hypothesis: Pruning is All You Need

Written by: Mike Greenbaum

In this work, I review the paper "Proving the Lottery Ticket Hypothesis: Pruning is All You Need", written by Eran Malach, Gilad Yehudai, Shai Shalev-Shwartz, and Ohad Shamir, https://arxiv.org/abs/2002.00585.

The paper is divided into 3 distinct parts and each of them tries to show a new idea on pruning neural networks. In general, pruning a neural network means removing some of the information of the neural network. Regularly it's done by zeroing some of the weights, removing neurons, or using the hessian Matrix. In the paper, the authors call pruning a network via zeroing the weights by "*weight pruning*", and pruning via removing neurons as "*neuron pruning*". The network that the paper talks about is a fully connected network with ReLU activation and the same width for each layer.

The first part of the paper focuses on *weight pruning*. It was proved in the paper that every target network of depth l could be approximated by *weight pruning* of a random network of depth $2l$[1]. Thus, instead of learning weights by gradient descent of the original network, one can perform *weight pruning* of a random network with twice the depth. Note, that the random network width is polynomic in the original parameters. This presents an entirely new way of learning neural networks that might not suffer the drawbacks of gradient descent (such as vanishing gradient)! This is a huge breakthrough in the understanding of neural networks. As stated in the paper, pruning a neural network is computationally hard; however, in most cases, a good approximation/heuristic might be found as have been done in many other fields of studying computer science.

---

[1] Fix some $\varepsilon, \delta \in (0,1)$. Let F be some target network of depth l such that for every $i \in [l]$ we have $\left\|W^{F(i)}\right\|_2 \leq 1$, $\left\|W^{F(i)}\right\|_{max} \leq \frac{1}{\sqrt{n_{in}}}$ (where $n_{in} = d$ for $i = 1$ and $n_{in} = n$ for $i > 1$). Let G be a network of width $poly\left(d, n, l, \frac{1}{\varepsilon}, \log\frac{1}{\delta}\right)$ and depth $2l$, where we initialize $W^{G(i)}$ from $U([-1,1])$, Then with probability at least $1 - \delta$ there exists a weight-subnetwork $\overline{G}$ of G such that: $\sup_{x \in \mathcal{X}} \left|\overline{G}(x) - F(x)\right| \leq \varepsilon$

Furthermore, due to our knowledge, neural networks can approximate any target function to arbitrary precision (with probability at least $1-\delta$ the error is at most ε). Together with the claim mentioned above, one can conclude that the *weight pruning* method is also capable of that!

Moreover, the proof for statement 1 was rather elegant and constructive. The authors demonstrated that the *weight pruning* of a random 2-layer neural net can approximate the function $x \to \alpha \cdot x_i$ via the formula $\alpha = \sigma(\alpha) - \sigma(-\alpha)$. The result can be obtained using this claim inductively. Thus, the usage of *weight pruning* of a random 2-layer network for the approximation of a single layer with help of the described above formula is the main unique achievement of the article, in my vision.

The second part of the paper focuses on *neuron pruning*. In this part, the authors try to explore a connection between *neuron pruning* of a random 2-layer network and a linear random features model[2]. Lemma 2 states that each linear can be approximated via *neuron pruning* of a random 2-layer network with sufficient width up to a scaling factor and vice versa. Thus, one can conclude that the *neuron pruning* of a random 2-layer network behaves similarly to the training linear model on the last layer.

In the first part of the paper, it was stressed that *weight pruning* of a random 2-layer network is a computationally hard problem. However, the current section presents the *neuron pruning* as an approximately linear random model that is not computationally hard. Therefore, one can conclude that *weight pruning* is a sufficiently stronger method than *neuron pruning.*

---

[2] Let D be some distribution over $\mathcal{X} \times [-1,+1]$, and let $\sigma: \mathbb{R} \to \mathbb{R}$ be L-Lipschitz with $\sigma(0) \leq L$. Let $\varepsilon, \delta \in (0,1)$, $n \in \mathbb{N}$ and D^* a distribution over $\{w: \|w\| \leq 1\}$ such that for $w_1, \dots, w_n \in D^*$ with probability at least $1-\delta$ there exists $u_1, \dots, u_n \in \mathbb{R}$ such that $|u_i| \leq C$ and the function $f(x) = \sum_{i=1}^{n} u_i \cdot \sigma(\langle w_i, x \rangle)$ satisfies that $L_D(f) \leq \varepsilon$. Ley $k \geq poly\left(C, n, L, \frac{1}{\varepsilon}, \frac{1}{\delta}\right)$, and suppose we initialize a 2-layer neural network g with width k where $w_i \in D^*$ and $u_i \sim U([-1,1])$. Then there exists a neuron-subnetwork $\overline{g}$ of g and constant $c > 0$ such that $L_D(c \cdot \overline{g}) \leq \varepsilon$

Nevertheless, the last part of the paper is devoted to describing the power of the *neuron pruning* method. The paper shows that overfitting of a finite sample is equivalent to *neuron pruning* of a random 2-layer network[3]. The most important achievement of proof 3 is an efficient algorithm for solving *neuron pruning* of a 2-layer network. Thus, I suppose that by studying the subject deeper in detail it might be possible to find a good approximation to the *weight pruning* method as well.

In summary, I would say that the authors explored the complex problem of training a fully connected neural network by implementing a new idea of weight pruning of a random network. They showed that training a fully connected neural network can be done by weight pruning of a random network instead of the usual gradient descent method. Furthermore, the authors presented another way of pruning, neuron pruning, which is a less expressive method than *weight pruning*. Even though it's possible to find an efficient algorithm to solve the problem. This gives a strong confirmation to study pruning as a concept.

[3] Let $S = \{(x_1, y_1), \dots, (x_m, y_m)\} \in \mathcal{X} \times [-1, +1]$. Let H be the $m \times m$ matrix defined by $H_{i,j} = \mathbb{E}\left[\sigma(\langle w, x_i \rangle) \cdot \sigma\left(\langle w, x_j \rangle\right)\right]$ and assume that $\lambda_{min}(H) = \lambda > 0$. If $k \geq poly\left(m, \frac{1}{\lambda}, L, \log\frac{1}{\delta}, \frac{1}{\varepsilon}\right)$ then with probability at least $1 - \delta$ there exists a neuron subnetwork $\overline{g}$ and a constant $c > 0$ such that $\sup_{1 \leq i \leq m} |c\overline{g}(x_i) - y_i| \leq \varepsilon$