Neural network knowledge distillation in tensor networks

Dereck Piché

March 9, 2023

Abstract

1 Introduction

void

2 Approches in knowledge distillation

2.1 Response-Based Knowledge Distillation

The first approach is to look exclusively at the outputs of the student and teachers. Each

3 Layer-by-layer approach

It has been common to each layer of a neural network as a certain abstracted representation of the previous information and their for generalisibility. Thus, we propose to change the cost, we use a tensor for multilinear mapping and train each mapping individualy according to the output of it's paired set of layers.

$$x \longrightarrow H_1 \longrightarrow H_2 \longrightarrow (\dots) \xrightarrow{g} \hat{y}$$

Each hidden layer is of the form;

$$a^l = \sigma^l(W^l a^{l-1})$$

In tensor layer form, it will be defined as

$$a^l = T^l \cdot \Phi\left(a^{l-1}\right)$$

b The main difference is that in the tensor approach, the "heavy" part is done by the non-linear transformation while a little work is done with the linear mapping. The opposite is true with neural networks. Here, $\Phi(X)$ (X begin the input vector) is a tensor product of several identical non-linear mappings of each element x_i. Thus, we have

$$\Phi(X) = \phi(x_1)\phi(x_2)\ldots\phi(x_n)$$

Were each $\phi : \mathbb{R} \to \mathbb{R}^d$, and each $d > 1$. Thus, our $\Phi(X) : \mathbb{R}^n \to \mathbb{R}^{(d\times)^{n-1}d}$. In other words, our Φ returns a tensor of order n, where each indices run from 1 to d.

4 Tensor networks

4.1 The *Matrix Product State* Tensor Network

4.2 Expressivity of MPS combinations with transformation $[1, x]^t$

Soit $f(x)R^d \mapsto R^{2^d}$ une fonction qui prend un vecteur de variables et qui renvoit un tenseur de dimensions quelconques dont les éléments contiennent les bases de la fonction multilinéaire des variables du vecteur x.

The vector given by $f(x)$ will return a tensor containing every combination of monomes of degree z, where z is the smallest repeating variable in the input vector.

Soit $m(f, \theta) :\ R^{2^d} \mapsto (R^d \mapsto R)$ une fonction qui effectue une combinaison linéaire des éléments de f and returns an *instance* of a multilinear function. Let this *instance* depend on θ.

Let S_ν be a set of n_ν variables. Let $\nu*$ be a vector where each variable in the set S_ν is repeated n_ν times.

Since the vector given by $f(x)$ will return a tensor containing every combination of monomes of degree z and $z = n_\nu$, the elements of the ouput of $f(x)$ will return the basis for the space of n_ν-degree polynomial over the set of variables S_ν.