Paper Review

The Description Length of Deep Learning Models

Léonard Blier, Yann Ollivier

Gavrilyuk Alexander

October 2023

Outline

1. Motivation & Problem statement
2. Theory
3. Experiment

Motivation & Problem statement

- How much do current deep models actually compress data? (Explicit measurement)

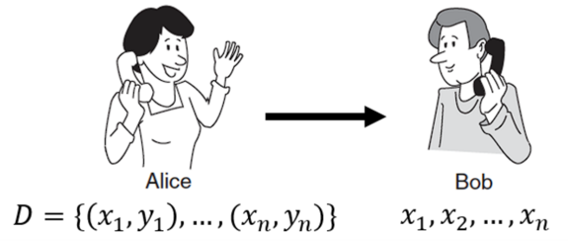

Figure: Supervised learning illustration when the input data x is public but predictions y are needed.

Definitions & Assumption on neural network

Definition (Notation)

Let X be the input space and $Y = \{1, ..., K\}$ the output space. The dataset is $D := \{(x_1, y_1), (x_2, y_2)\}$ A model for supervised learning is defined as a conditional probability distribution $p(y|x)$, such as for each $x \in X, \sum_{y \in Y} p(y|x) = 1$
A model class is a set of models depending one some parameter θ:
$M = \{p_\theta, \theta \in \Theta\}$
The Kullback–Leibler divergence between two distributions is

$$KL(\mu||\nu) = E_{X\sim\mu}[\log_2 \frac{\mu(x)}{\nu(x)}]$$

Definition (Shannon–Huffman code)

Suppose that Alice and Bob have agreed in advance on a model p, and both know the inputs $x_{1:n}$. Then there exists a code to transmit the labels $y_{1:n}$ losslessly with codelength

$$L_p(y_{1:n}|x_{1:n}) = -\sum_{i=1}^{n} \log_2 p(y_i|x_i)$$

Studied encodings

Definition (Uniform encoding)

The uniform distribution $p_{unif}(y|x) = \frac{1}{K}$ over K classes does not require any learning from the data, thus no additional information has to be transmitted. Using $p_{unif}(y|x)$ yields a codelength

$$L^{unif}(y_{1:n}|x_{1:n}) = n \log_2 K$$

Definition (Two-Part Encodings)

Assume that Alice and Bob have first agreed on a model class $(p_\theta)_{\theta \in \Theta}$. Let $L_{param}(\theta)$ be any encoding scheme for parameters $\theta \in \Theta$. Let θ^* be any parameter. The corresponding *two-part codelength* is

$$L_{\theta^*}^{2-part}(y_{1:n}|x_{1:n}) := L_{param}(\theta^*) + L_{p_{\theta^*}}(y_{1:n}|x_{1:n}) = L_{param}(\theta^*) - \sum_{i=0}^{n} \log_2 p_{\theta^*}(y_i|x_i)$$

Studied encodings

Definition (Variational and Bayesian Codes)

Assume that Alice and Bob have agreed on a model class $(p_\theta)_{\theta \in \Theta}$ and a prior α over Θ. Then for any distribution β over Θ, there exists an encoding with codelength

$$L_\beta^{var}(y|x) = KL(\beta||\alpha) + E_{\theta \sim \beta}[L_{p_\theta}(y_{1:n}|x_{1:n})] = KL(\beta||\alpha) - E_{\theta \sim \beta}[\sum_{i=0}^{n} \log_2 p_\theta(y_i|x_i)]$$

The variational bound L_β^{var} is an upper bound for the Bayesian description length bound of the Bayesian model p_θ with parameter θ and a prior α. Considering the Basyesian distribution of y, an associated code with model p_θ : $L^{Bayes}(y_{1:n}|x_{1:n}) = -\log_2 p_{Bayes}(y_{1:n}|x_{1:n})$ is provided:

$$L_\beta^{var}(y_{1:n}|x_{1:n}) \geq L^{Bayes}(y_{1:n}|x_{1:n})$$

Studied encodings

Definition (Prequential or Online Code)

Lets call p a *prediction strategy* for predicting the labels in Y knowing the inputs in X if for all k, $p(y_{k+1}|x_{1:k+1}, y_{1:k})$ is a conditional model. Any prediction strategy p defines a model on the whole dataset:

$$p^{preq}(y_{1:n}|x_{1:n}) = p(y_1|x_1) * p(y_2|x_{1:2}, y_1) * ... * p(y_n|x_{1:n}, y_{1:n-1})$$

Let $(p_\theta)_{\theta \in \Theta}$ be a DL model. We assume that we have a learning algorithm which computes, from any number of data samples $(y_{1:k}|x_{1:k})$, a trained parameter vector $\hat{\theta}(x_{1:k}, y_{1:k})$. This yields the following description length:

$$L^{preq}(y_{1:n}|x_{1:n}) = t_1 \log_2 K + \sum_{s=0}^{S-1} -\log_2 p_{\hat{\theta}_{t_s}}(y_{t_s+1:t_{s+1}}|x_{t_s+1:t_{s+1}})$$

where for each $s, \hat{\theta}_{t_s} = \hat{\theta}(x_{1:t_s}, y_{1:t_s})$ is the parameter learned on data samples 1 to t_s.

Experiment

Code	MNIST			CIFAR10		
	Codelength (kbits)	Comp. Ratio	Test Acc	Codelength (kbits)	Comp. Ratio	Test Acc
Uniform	199	1.	10%	166	1.	10%
float32 2-part	> 8.6Mb	$> 45.$	98.4%	> 428Mb	$> 2500.$	**92.9%**
Network compr.	> 400	$> 2.$	98.4%	> 14Mb	$> 83.$	**93.3%**
Intrinsic dim.	> 9.28	> 0.05	90%	$> 92,8$	> 0.56	70%
Variational	22.2	0.11	98.2%	89.0	0.54	66,5%
Prequential	**4.10**	**0.02**	**99.5%**	**45.3**	0.27	**93.3%**

Figure: Compression bounds via Deep Learning. The *codelength* is the number of bits necessary to send the labels to someone who already has the inputs. This codelength includes the description length of the model. The *compression ratio* for a given code is the ratio between its codelength and the codelength of the uniform code. The *test accuracy* of a model is the accuracy of its predictions on the test set.

Experiment

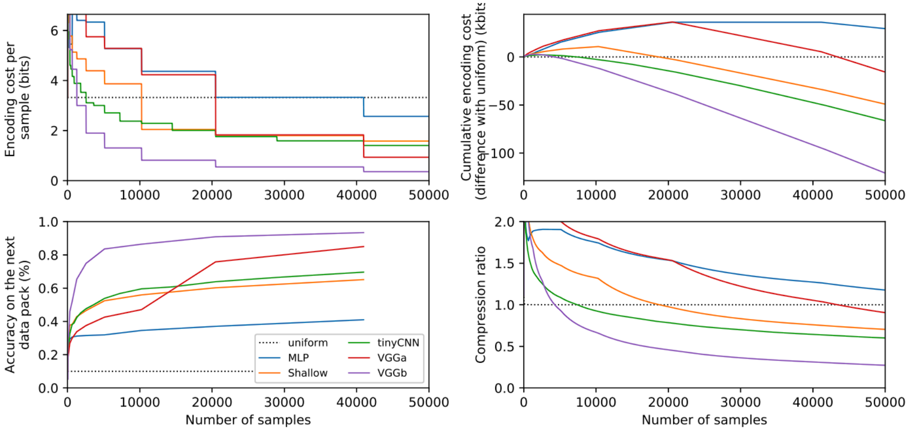

Figure: Prequential code results on CIFAR. Results of prequential encoding on CIFAR with 5 different models: a small Multilayer Perceptron (MLP), a shallow network, a small convolutional layer (tinyCNN), a VGG-like network without data augmentation and batch normalization (VGGa) and the same VGG-like architecture with data augmentation and batch normalization (VGGb).