EECE695D: Assignment#1

Due: `10.04.2022 (Tuesday) 23:59PM`

In this homework assignment, we count the FLOPs of transformers—a neural network architecture which seems to have a good chance in enabling the multi-modal learning of machines. We consider an overly simplified encoder-only transformer, which consists of a single-head self-attention layer and a feed-forward block. We ignore the nitty-gritty details, such as the embedding layer, residual connection, bias, GeLU, or layer normalization. We first consider the *inference phase*:

- Our model receives a length-ℓ sequence of tokens $\mathbf{x} = \mathbf{x}[1:\ell] = \mathbf{x}[1]\mathbf{x}[2]\cdots\mathbf{x}[\ell]$ as an input, where each token is a length-d_e vector, i.e., $\mathbf{x}[i] \in \mathbb{R}^{d_{\text{in}}}$.
- Given the sequence, the self-attention layer computes the following:
  - For each token $\mathbf{x}[i]$, we compute the *query*, *key*, and *value* vectors

    $$\mathbf{q}[i] \in \mathbb{R}^{d_{\text{attn}}}, \quad \mathbf{k}[i] \in \mathbb{R}^{d_{\text{attn}}}, \quad \mathbf{v}[i] \in \mathbb{R}^{d_{\text{out}}}$$

    by performing

    $$\mathbf{q}[i] = W_{\mathbf{q}}\mathbf{x}[i], \quad \mathbf{k}[i] = W_{\mathbf{k}}\mathbf{x}[i], \quad \mathbf{k}[i] = W_{\mathbf{v}}\mathbf{x}[i]$$

    for the parameter matrices $W_{\mathbf{q}}, W_{\mathbf{k}} \in \mathbb{R}^{d_{\text{attn}} \times d_{\text{in}}}, W_{\mathbf{v}} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$.
  - Compute the *attention score* of the i-th token on the j-th token as

    $$\alpha[i,j] = \frac{\exp\left(\mathbf{q}[i]^\top \mathbf{k}[j]/\sqrt{d_{\text{attn}}}\right)}{\sum_{u=1}^{\ell} \exp\left(\mathbf{q}[i]^\top \mathbf{k}[u]/\sqrt{d_{\text{attn}}}\right)}$$

  - Compute and return the output for each token as

    $$\tilde{\mathbf{v}}[i] = \sum_{j=1}^{\ell} \alpha[i,j] \cdot \mathbf{v}[j].$$

- The feed-forward block computes the following:
  - For each token, we apply a two-layer MLP to compute the output for each token as

    $$\hat{\mathbf{y}}[i] = w_2^\top \texttt{ReLU}\left(W_1 \tilde{\mathbf{v}}[i]\right), \qquad W_1 \in \mathbb{R}^{d_{\text{hidden}} \times d_{\text{out}}}, w_2 \in \mathbb{R}^{d_{\text{hidden}}}.$$

In other words, the simplified transformer model computes the output sequence $\hat{\mathbf{y}} = \hat{\mathbf{y}}[1]\hat{\mathbf{y}}[2]\cdots\hat{\mathbf{y}}[\ell]$ based on the input $\mathbf{x} = \mathbf{x}[1]\mathbf{x}[2]\cdots\mathbf{x}[\ell]$, using the parameters $W_{\mathbf{q}}, W_{\mathbf{k}}, W_{\mathbf{v}}, W_1, w_2$.

Question 1

What is the total number of weight parameters?

Question 2

How many FLOPs do we need to process through the self-attention layer? Assume that a dot product of two length-d vectors require $2d$ FLOPs, and we do not need any FLOP for computing the softmax.

Question 3

What is the size (in the number of parameters) of the activation $\tilde{\mathbf{v}}[1]\tilde{\mathbf{v}}[2]\cdots\tilde{\mathbf{v}}[\ell]$?

Question 4

How many FLOPs do we need to process through the feed-forward block?

Now, we consider a very simplified training of this transformer model. We use a single data instance (i.e., batch size 1) to compute the gradient of the parameters, where we have a 1-dimensional label for each token:

$$\mathbf{y} = \mathbf{y}[1]\mathbf{y}[2]\cdots\mathbf{y}[\ell], \qquad \mathbf{y}[i] \in \mathbb{R}$$

and the loss is computed as

$$l(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{i=1}^{\ell} (\mathbf{y}[i] - \hat{\mathbf{y}}[i])^2$$

Question 5

Write down the gradient of each parameter matrix.

Question 6

How many FLOPs do we need for the forward propagation, including the squared loss?

Question 7

How many FLOPs do we for the backward propagation? Assume that we have stored all possible intermediate computes (e.g., $\exp(\mathbf{q}[i]^\top \mathbf{k}[j]/\sqrt{d_{\text{attn}}})$) during the forward propagation.