Natural Language Processing (CO3086)
NLP 242 - Lab 11: Deep Neural Network

HO CHI MINH UNIVERSITY OF TECHNOLOGY

Vietnam National University Ho Chi Minh

Problem 1

Why are vanishing or exploding gradients an issue for RNNs?

Problem 2

GRUs. In class, we learned about RNNs and an extension — Gated Recurrent Units. GRUs can adaptively reset or update its "memory" of previous states. The feedforward computation for a GRU is given by

$$
\begin{aligned}
z_t &= \sigma(W_z x_t + U_z h_{t-1}) \\
r_t &= \sigma(W_r x_t + U_r h_{t-1}) \\
\hat{h}_t &= \tanh(W x_t + r_t \circ U h_{t-1}) \\
h_t &= (1 - z_t) \circ h_{t-1} + z_t \circ \hat{h}_t
\end{aligned}
$$

a) Show that for the sigmoid function $\sigma(x) = \frac{1}{1+\exp(-x)}$, $\sigma(-x) = 1 - \sigma(x)$

b) True/False. If the update gate z_t is close to 0, the net does not update its state significantly. (Explain)

c) True/False. If the update gate z_t is close to 1 and the reset gate r_t is close to 0, the net remembers the past state very well. (Explain)

Problem 3

Here are the defining equations for a LSTM cell.

$$
\begin{aligned}
i_t &= \sigma(W^{(i)} x_t + U^{(i)} h_{t-1}) \\
f_t &= \sigma(W^{(f)} x_t + U^{(f)} h_{t-1}) \\
o_t &= \sigma(W^{(o)} x_t + U^{(o)} h_{t-1}) \\
\tilde{c}_t &= \tanh\Big(W^{(c)} x_t + U^{(c)} h_{t-1}\Big) \\
c_t &= f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \\
h_t &= o_t \circ \tanh(c_t)
\end{aligned}
$$

Recall that $\circ$ denotes element-wise multiplication and that σ denotes the sigmoid function.

a) (True/False) If x_t is the 0 vector, then $h_t = h_{t-1}$. (Explain)

b) (True/False) If f_t is very small or zero, then error will not be back-propagated to earlier time steps. (Explain)

c) (True/False) The entries of f_t, i_t, o_t are non-negative. (Explain)

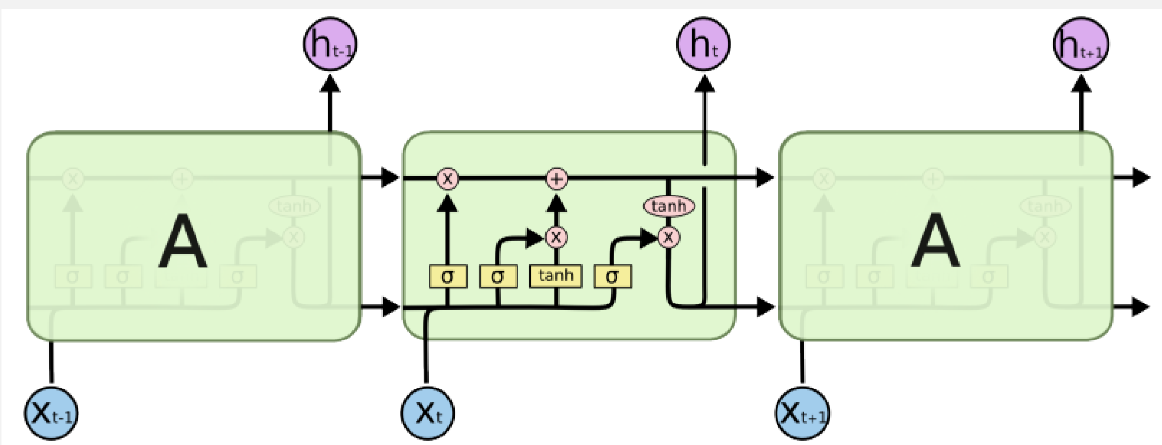
d) (2 points) (True/False) f_t, i_t, o_t can be viewed as probability distributions. (i.e., their entries are non-negative and their entries sum to 1.) (Explain)

Problem 4

To address the problem of vanishing and exploding gradients, we can use a different kind of recurrent cell – the LSTM cell (standing for "long short term memory"). The layout of the cell is shown in Figure 4. The LSTM has two states which are passed between timesteps: a "cell memory" C and the hidden state h. The LSTM update is given as follows:

$$\begin{aligned}
f_t &= \sigma(x_t W_f + h_{t-1} W'_f) \\
i_t &= \sigma(x_t W_i + h_{t-1} W'_i) \\
o_t &= \sigma(x_t W_o + h_{t-1} W'_o) \\
\tilde{C}_t &= \tanh\left(x_t W_g + h_{t-1} W'_g\right) \\
C_t &= f_t \circ C_{t-1} + i_t \circ \tilde{C}_t \\
h_t &= \tanh(C_t) \circ o_t
\end{aligned}$$

where $\circ$ represents the Hadamard Product (elementwise multiplication).



a) Denote the final cost function as J. Compute the gradient $\frac{\partial J}{\partial W_g}$ using a combination of the following gradients,

$$\frac{\partial h_t}{\partial h_{t-1}}, \frac{\partial h_{t-1}}{\partial W_g}, \frac{\partial J}{\partial h_t}, \frac{\partial C_t}{\partial W_g}, \frac{\partial C_{t-1}}{\partial W_g}, \frac{\partial C_t}{\partial C_{t-1}}, \frac{\partial C_t}{\partial \tilde{C}_t}, \frac{\partial h_t}{\partial o_t}$$

b) Using the previously derived gradient, which part of $\frac{\partial J}{\partial W_g}$ allows LSTMs to mitigate the vanishing gradient problem?

Problem 5

a) Explain how we incorporate self-attention into an RNN model at a high-level.

b) Consider a form of attention that matches query q to keys $k_1, \ldots, k_t$ in order to attend over associated values $v_1, \ldots, v_t$. If we have multiple queries $q_1, \ldots, q_n$, how can we write this version of attention in matrix notation?

Problem 6

In practice, Transformers use a Scaled Self-Attention. Suppose $q, k \in \mathbb{R}^d$ are two random vectors with $q, k \sim \mathcal{N}(\mu, \sigma^2 I)$, where $\mu \in \mathbb{R}^d$ and $\sigma \in \mathbb{R}^+$.

a) Define $\mathbb{E}[q^\top k]$ in terms of μ, σ, d

b) Define $\mathrm{Var}(q^\top k)$ in terms of μ, σ, d

c) Let s be the scaling factor on the dot product. We would like $\mathbb{E}[q^\top k/s]$ to scale linearly with d. What should s be in terms of μ, σ, d

d) Briefly explain what would happen to the variance of dot product if $s = 1$.

Problem 7

1. What is the reason for positional encoding? How is it typically implemented?

2. What is the advantage of multi-head attention? Give some examples of structures that can be found using multi-head attention.

3. For input sequences of length M and output sequences of length N, what are the complexities of (1) Encoder Self-Attention (2) Decoder-Encoder Attention (3) Decoder Self-Attention. Further let k be the hidden dimension of the network.

4. Do activation of the encoder depend on decoder activation? How much additional computation is needed to translate a source sequence into a different target language, in terms of M and N?