EX 2 – RNN

Due date: 9/25/24, 11:59 PM

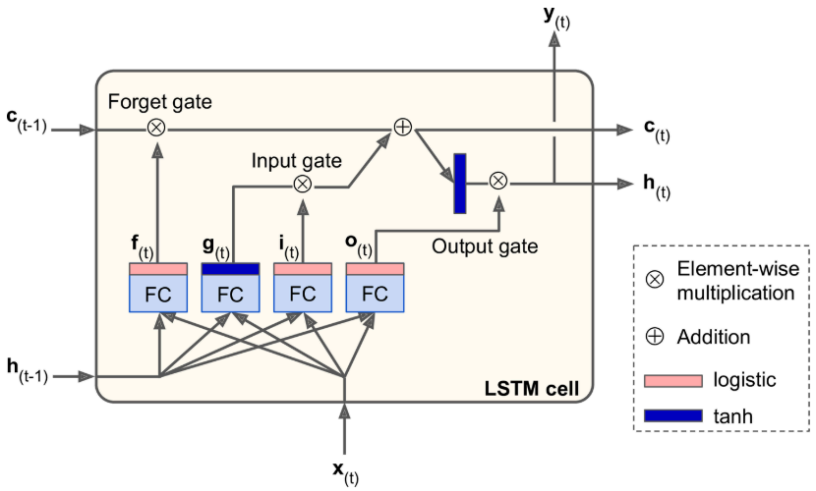
Please read the submission guidelines before you start.

Theory [50 pts]

1. [10 pts] One problem when working on sequence to sequence tasks, regardless of whether the model is an RNN or a Transformer, is varying sequence lengths. Assume you are given a minibatch with variable-length input and output sequences.
    a. Describe one way to deal with variable-length **input** sequences.
    b. Similarly describe one way to deal with variable-length **output** sequences. Does your loss computation change at all in this case?
2. [10 pts] Name two advantages of GRUs over LSTMs.
3. [10 pts] The LSTM cell equations are as follows:

$$
\begin{aligned}
\mathbf{i}_{(t)} &= \sigma(\mathbf{W}_{xi}{}^{T} \cdot \mathbf{x}_{(t)} + \mathbf{W}_{hi}{}^{T} \cdot \mathbf{h}_{(t-1)} + \mathbf{b}_{i}) \\
\mathbf{f}_{(t)} &= \sigma(\mathbf{W}_{xf}{}^{T} \cdot \mathbf{x}_{(t)} + \mathbf{W}_{hf}{}^{T} \cdot \mathbf{h}_{(t-1)} + \mathbf{b}_{f}) \\
\mathbf{o}_{(t)} &= \sigma(\mathbf{W}_{xo}{}^{T} \cdot \mathbf{x}_{(t)} + \mathbf{W}_{ho}{}^{T} \cdot \mathbf{h}_{(t-1)} + \mathbf{b}_{o}) \\
\mathbf{g}_{(t)} &= \tanh(\mathbf{W}_{xg}{}^{T} \cdot \mathbf{x}_{(t)} + \mathbf{W}_{hg}{}^{T} \cdot \mathbf{h}_{(t-1)} + \mathbf{b}_{g}) \\
\mathbf{c}_{(t)} &= \mathbf{f}_{(t)} \otimes \mathbf{c}_{(t-1)} + \mathbf{i}_{(t)} \otimes \mathbf{g}_{(t)} \\
\mathbf{y}_{(t)} &= \mathbf{h}_{(t)} = \mathbf{o}_{(t)} \otimes \tanh(\mathbf{c}_{(t)})
\end{aligned}
$$

An illustration of an LSTM cell:



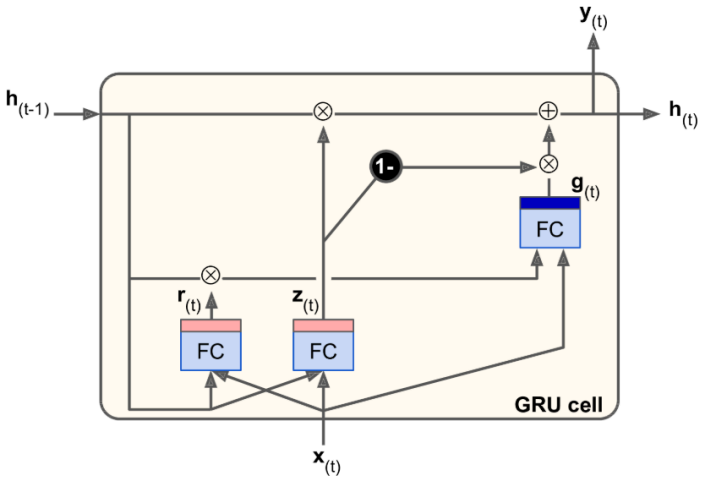
Assume that we have a network based on a single LSTM cell that uses a vector of size 200 to describe the current state. Also assume that its inputs are vectors of size 200. Considering only parameters related to the cell, how many parameters does it have?

4. [20 pts] The GRU equations are as follows:

$$\mathbf{z}_{(t)} = \sigma(\mathbf{W}_{xz}{}^{T} \cdot \mathbf{x}_{(t)} + \mathbf{W}_{hz}{}^{T} \cdot \mathbf{h}_{(t-1)} + \mathbf{b}_z)$$
$$\mathbf{r}_{(t)} = \sigma(\mathbf{W}_{xr}{}^{T} \cdot \mathbf{x}_{(t)} + \mathbf{W}_{hr}{}^{T} \cdot \mathbf{h}_{(t-1)} + \mathbf{b}_r)$$
$$\mathbf{g}_{(t)} = \tanh(\mathbf{W}_{xg}{}^{T} \cdot \mathbf{x}_{(t)} + \mathbf{W}_{hg}{}^{T} \cdot (\mathbf{r}_{(t)} \otimes \mathbf{h}_{(t-1)}) + \mathbf{b}_g)$$
$$\mathbf{h}_{(t)} = \mathbf{z}_{(t)} \otimes \mathbf{h}_{(t-1)} + (1 - \mathbf{z}_{(t)}) \otimes \mathbf{g}_{(t)}$$

Where $\sigma(x) = \frac{1}{1+e^{-x}}$

An illustration of a GRU cell:



Consider a GRU network with two timestamps (i.e. two iterations of the GRU cell), with loss $\epsilon_{(t)}$ (e.g., the l_2 loss: $\frac{1}{2}\left(h_{(t)} - y_t\right)^2$).

Assume the gradient $\frac{\partial \epsilon_{(2)}}{\partial h_{(2)}}$ is given.

Calculate the gradients of GRU for back propagation. For simplicity, you may ignore the bias, treat all variables as scalars and calculate the gradients of the second time stamp only. Using the chain rule, Calculate:

a. $\frac{\partial \epsilon_{(2)}}{\partial W_{xz}}$

b. $\frac{\partial \epsilon_{(2)}}{\partial W_{hz}}$

c. $\frac{\partial \epsilon_{(2)}}{\partial W_{xg}}$

d. $\frac{\partial \epsilon_{(2)}}{\partial W_{hg}}$

e. $\frac{\partial \epsilon_{(2)}}{\partial W_{xr}}$

f. $\frac{\partial \epsilon_{(2)}}{\partial W_{hr}}$

Practical [50 pts]

Your task will be to perform language modeling with LSTMs and GRUs on the Penn Tree Bank dataset (i.e., train a network to predict the next token given past tokens). We will be using perplexity rather than accuracy as our metric for this task. The dataset is available for download on the Assignment page on Canvas.

For your architecture, you should implement the "small" model as described in "Recurrent Neural Network Regularization", by Zaremba et al. That is, your models should have 200 hidden units rather than the 650 and 1500 used in their "medium" and "large" models respectively.

a. For the four following settings, present a convergence graph (as described at ex1) for both the train and test Perplexity. You should present 4 plots in total, with train and test perplexities presented in the same plots. For each experimental setting, you should also write the learning rate and dropout probability (and can include other details you think are relevant)
   - LSTM based network without dropout.
   - LSTM based network with dropout.
   - GRU based network without dropout.
   - GRU based network with dropout.

b. Summarize the results by a table, with the resulting Perplexity on the train set, validation, and test. For the table, you should select model parameters to evaluate based on validation perplexity.

c. Make conclusions regarding the results.

Comments:

- Describe in the readme file how to train each setting, and how to test it with the saved weights.
- All graphs should be clear with a proper heading.
- You may need more than 13 epochs when using dropout. Change the learning rate appropriately in this case.
- No need to find the absolute best perplexity for each settings, but make sure that without dropout the **validation** perplexity is below 125, and with it is below 100.
- You should use word-level tokens

Good Luck!