Deep Learning / Spring 2025

Homework 2

Please upload your assignments on or before April 3, 2025.

- You are encouraged to discuss ideas with each other, or to consult online tools (such as LLMs). But you **must acknowledge** who you collaborated with or which tools you used, and you **must compose your own** writeup and code independently. Answers should be self-contained and must not appear to be generated by an LLM. We will not grade generic responses.

- We **require** answers to theory questions typeset. Handwritten homework submissions will not be graded.
- We **require** answers to coding questions in the form of a Jupyter notebook. Within each notebook, it is **necessary** to include brief, coherent explanations of both your code and your results to show us your understanding. Use the text block feature of Jupyter notebooks to include explanations.
- Upload both your theory and coding answers in the form of a **single PDF** on **Gradescope**.

---

1. **(3 points)** *Recurrences using RNNs.* Consider the recurrent network below in Figure 1 where we have explicitly specified the weights. All inputs are integers, hidden states are scalars, all biases are zero, and all weights are indicated by the numbers on the edges. The output unit performs binary classification. Assume that the input sequence is of length 1000. What property of the input sequence is computed by the output unit at the final time step? Be precise in your answer. for example you can say something like "the output is measuring whether the sum of inputs is an even number" or "the output is the cumulative sum of all the inputs" or such. Explain your reasoning. If you are stuck, it may help to write out the recurrence clearly for a few steps to understand what is going on.

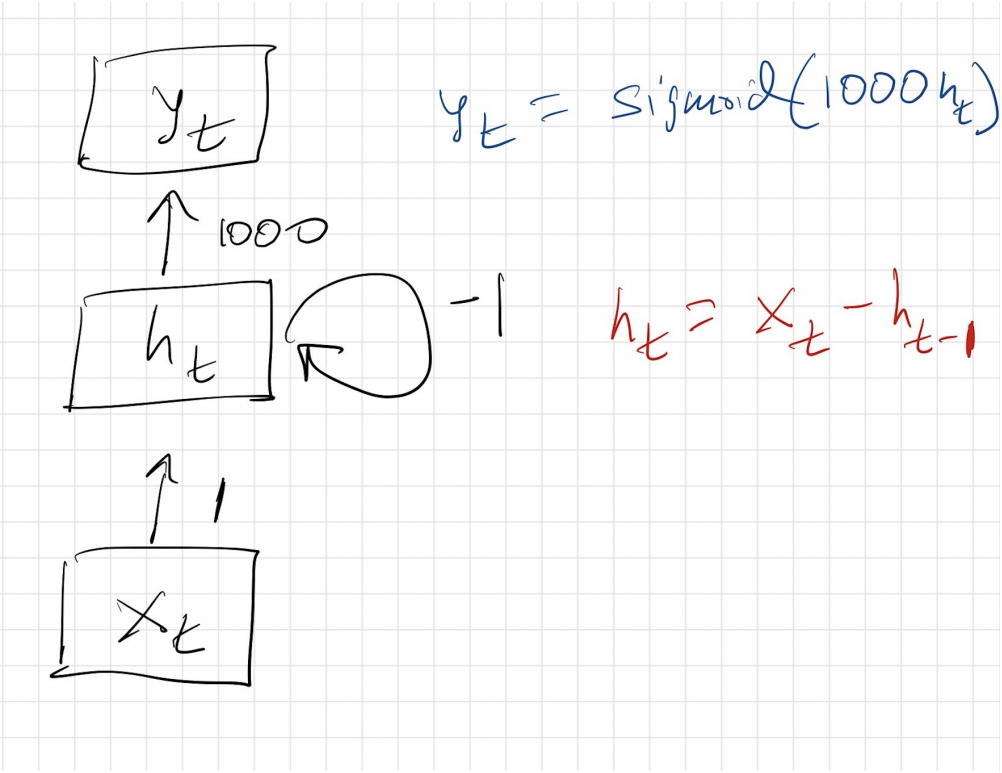

Figure 1: RNNs

2. **(2 points)** *Attention! My code takes too long.* In class, we showed that a computing a regular self-attention layer takes $O(T^2)$ running time for an input with T tokens. One alternative is to use "linear self-attention". In the simplest form, this is identical to the standard dot-product self-attention discussed in the class and lecture notes, except that the exponentials in the rowwise-softmax operation softmax(QK) are dropped; we just pretend all dot-products are positive and normalize as usual. Show that such this type of attention mechanism avoids the quadratic dependence on T and in fact can be computed in $O(T)$ time. Explain your reasoning.

3. **(5 points)** *Vision Transformers.* In HW1 you trained a dense neural network which can classify images from the FashionMNIST dataset. In this problem, you are tasked to achieve the same objective, but using Vision Transformers. Experiment with a patch size of 4x4, between 4-8 ViT layers, and between 2-4 heads and report your test accuracies for each. You can adapt the demo Jupyter notebook provided on Brightspace to train ViTs.

4. **(5 points)** *Sentiment analysis using Transformer models*. Open the (incomplete) Jupyter notebook provided as an attachment to this homework and complete the missing items. Save your finished notebook in PDF format and upload along with your answers to the above questions in a single PDF.