Homework 4

Stats 315A, Winter 2026

March 1, 2026

Due date: Monday, March 9th at 11:59pm on Canvas.

Submission instructions: Upload a `.pdf` file with your responses to question 1 part II and questions 2-3. Submit a `.pdf` of the notebook from question 1 part I, and code from question 1 part II an additional separate `.pdf` or `.txt` file.

Question 1 (Transformer Language Models, 40pts)**:** This problem asks you to walk through some key ideas of autogressive language modeling, and to build out an implementation of a transformer language model in pytorch.

You are encouraged (but not required) to work in groups of two or three.

We recommend you to complete this part within a Google Colab notebook. The link in part I provides a starting point. It will be helpful (but not necessary) to have a Colab PRO membership, which you can get for free as a student.

For this problem you should submit:

- For part I, an exported `.pdf` of the `.ipynb` with your solutions.
- For part II, a write-up of your experiment (roughly ~ 500 words should be enough) **and** your code (e.g. as a pdf of Colab notebook). The code for Part 2 will not be graded, but will be checked for completeness.

(I) **Large language models, 25pts**

Open https://colab.research.google.com/drive/1QM-eqUyGy262lhMkJFDhUda1y4_LnRDT, create a copy, and respond to the questions and fill in the missing pieces therein.

(II) **Experimenting with and improving the transformer language model, 15 pts**

The goal for this part is to empirically explore some area (or two, if you prefer) of possible "improvement" to the tranformer language model you complete in part I. This part is open-ended. It is up to you to choose

(1) precisely what sort of improvement you are aiming for,

(2) what approach to condider to achieve obtain it, and

(3) what experiments and analyses to run to evaluate if there is an improvement.

Examples of possible innovations to consider include (but are not limited to) changes to architecture, losses, or optimization. You can receive full credit even if the approach you examine worsens performance, or if your analysis is inconclusive. The important part is that you make clear what you have done, and that your rationale and analysis are rigorous.

In your write-up you should

(a) describe the choices you make for (1-3) above,

(b) include a table or figure with the empirical results of your experiments, and

(c) describe any conclusions you can draw and their scope. For example, do you expect your findings are robust to different random seeds, hyper-parameter choices, or training datasets?

Recommendation on compute consumption. Keep computational expense and compute time in mind when choosing and planning your experiments. For example, if you were choose to explore increasing the width of depth of the transformer it will be important to be cautious of both runtime and the compute units provided in the free Colab PRO subscription. These will quickly run-out if, for example, you immediately switch to an A100 GPU runtime and do not turn it off when you are not using it.

If submitting as a group: If you completed this problem as a group you must (1) clearly list the other group members and (2) describe the division of labor. Both (or all three) group members must have been involved in implementation of code, running experiments, or running computational analyses. Submissions from members of the same group may be identical for this part (but not for part 1).

Question 2 (Residual Connections and Layer Norm, 6 pts)**:** In this question, we will explore the role of residual connections and layer norms in stabilizing neural network training.

(a) Consider a multi-layer perceptron (MLP) with L layers, defined by

$$h_0 = x, \qquad h_{\ell+1} = \phi(W_\ell h_\ell), \qquad \ell = 0, \dots, L-1, \tag{1}$$

where $\phi(h)$ is the ReLU activation applied element-wise to $h \in \mathbb{R}^d$ and $W_\ell \in \mathbb{R}^{d\times d}$ is the weight matrix at layer ℓ. Assume also that $\|W_\ell\| = c$, where $c > 1$ and $\|\cdot\|$ denotes the operator norm. Explain why, in the worst case, the activations and backpropagated gradients can grow on the order of c^L.
Hint: You should consider the setting

$$W_\ell = USV^\top, \qquad UU^\top = VV^\top = \mathbb{I}_d, \quad S \text{ diagonal.}$$

(b) Instead of eq. (1), suppose we parameterize the neural network as

$$h_{\ell+1} = h_\ell + f_\ell(h_\ell) \tag{2}$$

where each f_ℓ satisfies $\|J_{f_\ell}(h)\|_2 \le \varepsilon$ for all $h \in \mathbb{R}^d$, where $J_{f_\ell}(h)$ represents the Jacobian of f_ℓ at h. The term h_ℓ is called a **residual connection**. Show that the layer-to-layer Jacobian is

$$\frac{dh_{\ell+1}}{dh_\ell} = \mathbb{I}_d + J_{f_\ell}(h_\ell).$$

Deduce that the full network Jacobian $\frac{dh_L}{dh_0}$ is a product of matrices of the form $\mathbb{I}_d + E_\ell$. Compared to part (a), explain why this leads to improved stability when ε is small.

(c) Now suppose that each residual block is preceded by **layer normalization**:

$$h_{\ell+1} = h_\ell + f_\ell(\mathrm{LN}(h_\ell)),$$

where for $h \in \mathbb{R}^d$,

$$\mathrm{LN}(h) = \frac{h - \mu(h)\mathbb{1}}{\sigma(h)}, \qquad \mu(h) = \frac{1}{d}\sum_{i=1}^{d} h_i, \qquad \sigma(h)^2 = \frac{1}{d}\sum_{i=1}^{d}(h_i - \mu(h))^2.$$

Show that the coordinates of $\mathrm{LN}(h)$ have mean 0 and variance 1 for every vector $h \in \mathbb{R}^d$. Why is it less likely that the Jacobians J_{f_ℓ} become very large, compared to an unnormalized network?

Question 3 (Learning Transitions in a HMM, 12 pts)**:** In this question, we will explore how the expectation-maximization (EM) algorithm can be used to learn the transition parameters in a hidden Markov model (HMM).

Let $Z_i \in \{1, \dots, K\}$ and $Y_i \in \mathbb{R}^k$ represent the latent variable and observation, respectively, at time $i \in \{1, \dots, n\}$. The joint distribution of $(Z_i, Y_i)_{i=1}^n$ is described by the probabilistic graphical model

$$p_\theta((Z_i, Y_i)_{i=1}^n) = p_\theta(Z_1)\left(\prod_{i=1}^{n-1} p_\theta(Z_{i+1}|Z_i)p_\theta(Y_i|Z_i)\right)p_\theta(Y_n|Z_n).$$

The initial distribution is $p_\theta(Z_1 = z_1) = \boldsymbol{\pi}_{z_1}$ and the transition distribution is $p_\theta(Z_{i+1} = z_{i+1}|Z_i = z_i) = \boldsymbol{P}_{z_i, z_{i+1}}$, where $\boldsymbol{\pi} \in \mathbb{R}^K$ and $\boldsymbol{P} \in \mathbb{R}^{K \times K}$. The model parameters θ consist of $\boldsymbol{\pi}$, $\boldsymbol{P}$, and the parameters of the likelihood model $p_\theta(Y_i|Z_i)$.

(a) Define the forward and backward messages

$$\alpha_i(z_i) = p_\theta(Z_i = z_i, Y_1 = y_1, \dots Y_{i-1} = y_{i-1}), \quad \beta_i(z_i) = p_\theta(Y_{i+1} = y_{i+1}, \dots, Y_n = y_n|Z_i = z_i)$$

as well as the K-dimensional vectors

$$\boldsymbol{\alpha}_i = [\alpha_i(1), \dots, \alpha_i(K)]^\top, \ \boldsymbol{\beta}_i = [\beta_i(1), \dots, \beta_i(K)]^\top, \ \ell_i = [\ell_i(1), \dots, \ell_i(K)]^\top,$$

where $\ell_i(z_i) = p_\theta(Y_i = y_i|Z_i = z_i)$. We take $\alpha_1(z_1) = \boldsymbol{\pi}$ and $\beta_n(z_n) = 1$ for all $z_1, z_n \in \{1, \dots, K\}$.

Show that the posterior pairwise marginal distributions can be written as

$$\begin{aligned} p_\theta(Z_i = z_i, Z_{i+1} = z_{i+1}|Y_1 = y_1, \dots, Y_n = y_n) &= \frac{\beta_{i+1}(z_{i+1})\ell_{i+1}(z_{i+1})\boldsymbol{P}_{z_i, z_{i+1}}\ell_i(z_i)\alpha_i(z_i)}{\sum_{z'_i, z'_{i+1}} \beta_{i+1}(z'_{i+1})\ell_{i+1}(z'_{i+1})\boldsymbol{P}_{z'_i, z'_{i+1}}\ell_i(z'_i)\alpha_i(z'_i)} \\ &= \frac{\beta_{i+1}(z_{i+1})\ell_{i+1}(z_{i+1})\boldsymbol{P}_{z_i, z_{i+1}}\ell_i(z_i)\alpha_i(z_i)}{(\ell_i \odot \boldsymbol{\alpha}_i)^\top \boldsymbol{P}(\boldsymbol{\beta}_{i+1} \odot \ell_{i+1})}, \end{aligned}$$

for $i \in \{1, \dots, n-1\}$, where $\odot$ represents the element-wise product between vectors.

(b) From part (a), deduce that the posterior transitions satisfy

$$\begin{aligned} p_\theta(Z_{i+1} = z_{i+1}|Z_i = z_i, Y_1 = y_1, \dots, Y_n = y_n) &= \frac{\boldsymbol{P}_{z_i, z_{i+1}}\beta_{i+1}(z_{i+1})\ell_{i+1}(z_{i+1})}{\sum_{z'_{i+1}} \boldsymbol{P}_{z_i, z'_{i+1}}\beta_{i+1}(z'_{i+1})\ell_{i+1}(z'_{i+1})} \\ &= \frac{\boldsymbol{P}_{z_i, z_{i+1}}\beta_{i+1}(z_{i+1})\ell_{i+1}(z_{i+1})}{\boldsymbol{P}_{z_i}^\top(\boldsymbol{\beta}_{i+1} \odot \ell_{i+1})}, \end{aligned}$$

where $\boldsymbol{P}_{z_i}$ is the z_ith row of $\boldsymbol{P}$. This is the probability of transitioning from state z_i at time i to state z_{i+1} at time $i+1$, under the Bayesian posterior.

(c) The prior transitions are **time homogeneous**, meaning that for each $j, k \in \{1, \ldots, K\}$ and for all $1 < i \leq n$,

$$p_\theta(Z_{i+1} = k | Z_i = j) = p_\theta(Z_i = k | Z_{i-1} = j) = \boldsymbol{P}_{jk}.$$

Do the posterior transitions $\boldsymbol{\Omega}^{(i)}_{j,k} = p_\theta(Z_{i+1} = k | Z_i = j, Y_1 = y_1, \ldots, Y_n = y_n)$, $\boldsymbol{\Omega}^{(i)} \in \mathbb{R}^{K \times K}$, $i \in \{1, \ldots, n-1\}$ satisfy this property?

(d) Let $q((Z_i)_{i=1}^n) = p_\theta((Z_i)_{i=1}^n | Y_1 = y_1, \ldots, Y_n = y_n)$ be the Bayesian posterior over latent states derived in parts (a) and (b).

Show that the expected complete data log likelihood is

$$\mathbb{E}_q[\log p_\theta((Z_i, Y_i)_{i=1}^n)] = \sum_{i=1}^{n-1} \sum_{j,k=1}^{K} \boldsymbol{\Omega}^{(i)}_{jk} \log \boldsymbol{P}_{jk} + c \tag{3}$$

where c is a constant independent of $\boldsymbol{P}$.

(e) Conclude that the transition matrix maximizing the expected complete data log-likelihood is

$$\boldsymbol{P}^\star = \frac{N_{jk}}{\sum_{k'=1}^K N_{jk'}}, \quad N_{jk} = \sum_{i=1}^{n-1} \boldsymbol{\Omega}^{(i)}_{jk}.$$

You may assume $\sum_{k'=1}^K N_{jk'} > 0$ for all $1 \leq j \leq K$.
Hint: Introduce K Lagrange multipliers to the expected complete data log likelihood (3).

Question 4 (Time and collaboration, **2 points**)**:**

(a) How many hours in total did you spend on this assignment? (This will help to calibrate future assignments.)

(b) With whom (if anyone) did you collaborate on this assignment?