CSE 493 G1 / 599 G1
Deep Learning
Spring 2025 Exam

May 20, 2025

Full Name: _____

UW Net ID: _____

Question		Score
Multiple Choice	(30 pts)	
Backpropagation	(22 pts)	
Convolution & Pooling	(16 pts)	
RNN Diagnostics	(12 pts)	
Tokenization	(20 pts)	
Total	(100 pts)	

Welcome to the CSE 493 G1 / 599 G1 Exam!

- The exam is 80 min and is **double-sided**.
- No electronic devices are allowed.

I understand and agree to uphold the University of Washington Student Conduct Code during this exam.

Signature: _____ Date: _____

Good luck!

This page is left blank for scratch work only. DO NOT write your answers here.

This page is left blank for scratch work only. DO NOT write your answers here.

This page is left blank for scratch work only. DO NOT write your answers here.

1 Multiple Choices (30 points) - Recommended 20 Minutes

Fill in the circle next to the letter(s) of your choice (like this: ●). No explanations are required. **Choose ALL options that apply.**

Each question is worth 5 points and the answer may contain **exactly one or two** options. Selecting all of the correct options and none of the incorrect options will get full credit. For questions with multiple correct options, each incorrect or missing selection gets a 2.5-point deduction (up to 5 points).

1.1 (5 points) Given scores $z = [z_1, \ldots, z_d]$, let $\text{softmax}(z)_i = \frac{e^{z_i}}{\sum_j e^{z_j}}$. Which statements are true?

- ◯ A: Adding a constant α to all z_i leaves the softmax output scores unchanged.
- ◯ B: The softmax function is non-differentiable at points where two components tie.
- ◯ C: Cross-entropy loss for multi-class classification is $-\sum_{i=1}^{N} \log \frac{e^{z_{y_i}}}{\sum_j e^{z_j}}$ (N is the size of training set).
- ◯ D: After applying softmax, the sum of the scores will exceed 1.
- ◯ E: None of the above.

1.2 (5 points) When using batch gradient descent (full-batch) to minimize a differentiable loss function, selecting a learning rate that is too large will most likely result in:

- ◯ A: Very slow convergence toward the minimum.
- ◯ B: Divergence or oscillation around the minimum.
- ◯ C: Convergence to the global minimum.
- ◯ D: The gradient norm becoming exactly zero at each step.
- ◯ E: None of the above.

1.3 (5 points) Mark all valid statements:

- ◯ A: ReLU is defined as $f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}$
- ◯ B: Leaky ReLU is defined as $f(x) = \max(0.01, x)$.
- ◯ C: Layer Norm uses batch statistics during inference.
- ◯ D: During inference, Batch Norm uses the running averages of mean and variance computed in training.
- ◯ E: None of the above.

1.4 (5 points) Which of the following is a key reason to use convolutional layers instead of fully-connected layers when processing images?

- ◯ A: Convolutional layers require more parameters to learn complex patterns.
- ◯ B: Convolutional layers exploit spatial locality and share weights across locations.
- ◯ C: Convolutional layers enforce global connectivity between all pixels.
- ◯ D: Convolutional layers are non-differentiable.
- ◯ E: None of the above.

1.5 (5 points) Select all **incorrect** statements about RNNs and LSTMs:

- ◯ A: Standard RNNs suffer from vanishing gradients for long sequences.
- ◯ B: Gradient clipping can worsen exploding gradients in RNNs.
- ◯ C: LSTM forget-gate activations lie in $(0, 1)$ due to the sigmoid nonlinearity.
- ◯ D: Standard RNNs and LSTMs also need explicit positional encoding to help them "remember" position, like what Transformers do.
- ◯ E: None of the above.

1.6 (5 points) In the "scaled dot-product" attention $\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$, which are **not true**?

- ◯ A: Scaling by $\sqrt{d_k}$ prevents softmax from becoming too peaked.
- ◯ B: Keys and queries must have the same dimensionality.
- ◯ C: Positional encoding cannot be learnable due to its inherent complexity.
- ◯ D: Multi-head attention concatenates parallel attention outputs.
- ◯ E: None of the above.

2 Backpropagation (22 points) - Recommended 20 Minutes

Please make sure to write your answer only in the provided space.

Consider the following 2-layer network with scalar input x, weights w_1, w_2, w_3, ReLU nonlinearity, and squared-error loss.

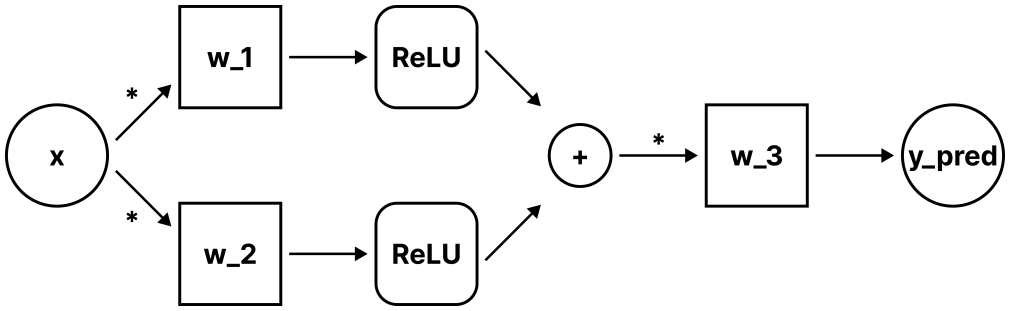

Figure 1: The model architectures of the 2-layer network.

Mathematically, the forward pass is defined as the following:

1. $z_1 = w_1 x$, $a_1 = \max(0, z_1)$
2. $z_2 = w_2 x$, $a_2 = \max(0, z_2)$
3. $y_{\text{pred}} = w_3 (a_1 + a_2)$
4. Loss $L = \frac{1}{2}(y_{\text{pred}} - y_{\text{true}})^2$

Given $x = 2$, $w_1 = 0.5$, $w_2 = -1.0$, $w_3 = 2.0$, and $y_{\text{true}} = 1.0$:

Given this setup, answer the following questions:

1. (16 points) Compute all forward-pass activations $(z_1, a_1, z_2, a_2, y_{\text{pred}})$ and then backpropagate to find gradients $\frac{\partial L}{\partial w_1}$, $\frac{\partial L}{\partial w_2}$, $\frac{\partial L}{\partial w_3}$. **Write your final answers on the labeled lines at the bottom.**

   Forward activations:

z_1: _____ a_1: _____ z_2: _____ a_2: _____ y_{pred}: _____

Backward pass:

$\frac{\partial L}{\partial w_1}$: _____ $\frac{\partial L}{\partial w_2}$: _____ $\frac{\partial L}{\partial w_3}$: _____

2. (6 points) Using learning rate $\alpha = 0.1$, perform one SGD update on (w_1, w_2, w_3) and report the new weight values. **Remember to fill your final answers on the labeled lines at the bottom.**

w_1: _____ w_2: _____ w_3: _____

3 Convolution & Pooling (16 points) - Recommended 15 Minutes

Please make sure to write your answer only in the provided space.

Let

$$I = \begin{bmatrix} 1 & 0 & 2 & 1 \\ 2 & 1 & 0 & 0 \\ 1 & 2 & 1 & 2 \\ 0 & 1 & 2 & 1 \end{bmatrix}, \quad F = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}.$$

Given this setup, answer the following questions:

1. (4 points) What is the shape of the feature-map produced by convolving I with F using **no padding** and **stride = 1**?

2. (4 points) Explain how setting **padding = 1** and **stride = 2** would change the output dimensions. Why might a model designer choose those settings?

3. (4 points) Apply a 2×2 max-pool with stride $= 2$, no padding, to the original I. What is the resulting 2×2 matrix?

4. (4 points) In practice, one can downsample a feature-map either via **stride $>$ 1** in a convolution or via a separate pooling operation. Give one advantage and one disadvantage of each approach.

4 RNN Diagnostics (12 points) - Recommended 10 Minutes

Please make sure to write your answer only in the provided space.

1. (6 points) During training, you observe that the training loss steadily decreases, but the validation accuracy on sequences longer than 50 time steps collapses to random chance. List two potential reasons related to RNN behavior and outline specific debugging experiments or changes you would perform to identify and mitigate each cause.

2. (6 points) You log the gradient norms of the hidden-state parameters at each time step and find that gradients for early time steps are nearly zero, while those for recent steps are large and unstable. Explain what this indicates about your model's training dynamics, and propose two modifications (architectural or optimization) to alleviate the problem, explaining how each modification addresses the issue.

5 Tokenization (20 points) - Recommended 15 Minutes

Please make sure to write your answer only in the provided space.

Consider a greedy subword tokenizer processing the sentence:

```
"The dog that chased the dog was tired."
```

Then, after tokenization, a standard Transformer encoder will receive them as input.

Assume we first run self-attention *without* any positional encodings.

Moreover, assume the sentence is tokenized with a greedy longest-match (maximal-munch) algorithm over a fixed subword vocabulary. At each step, among all vocabulary entries that match the current prefix of the remaining string, choose the longest one (break ties arbitrarily or by smaller ID), emit its ID, remove it from the front, and repeat until the string is consumed.

Example: Suppose the vocabulary contains the tokens {`"he"`:1, `"hello"`:2, `"llo"`:3}. Then the input `"hello"` is tokenized as [2] ("hello") rather than [1,3] ("he","llo"), because the tokenizer greedily matches the longest possible subword.

Given this setup, answer the following questions:

1. (4 points) Suppose we use a greedy subword tokenizer (BPE) with the following dictionary (token → index):

   {`" "` : 1, `"The "` : 2, `"dog "` : 3, `"dog"` : 4, `"that"` : 5, , `"chased"` : 6, `"the"` : 7,

   `"dog."` : 8, `"was"` : 9, `"tired"` : 10, `"."` : 11}.

   Describe step by step how this sentence will be tokenized using the longest-match rule, and list the resulting token indices.

2. (4 points) Given the tokenization you came up with from the last question as input, how will the model differentiate between the two occurrences of the token "dog" when no positional encodings are used? Provide a brief conceptual explanation.

3. (4 points) Suppose we use a greedy subword tokenizer (BPE) with a new dictionary (token → index), note that only index 4 is changed from "dog" to " dog" compared to the previous dictionary:

$$\{\texttt{" "} : 1,\ \texttt{"The "} : 2,\ \texttt{"dog "} : 3,\ \texttt{" dog"} : 4,\ \texttt{"that"} : 5, ,\ \texttt{"chased"} : 6, \texttt{"the"} : 7,$$

$$\texttt{"dog."} : 8,\ \texttt{"was"} : 9, \texttt{"tired"} : 10,\ \texttt{"."} : 11\}.$$

Describe step by step how this sentence will be tokenized using the longest-match rule, and list the resulting token indices.

4. (4 points) Given the tokenization you came up with from the last question as input, how will the model differentiate between the two occurrences of the token "dog" when no positional encodings are used? Provide a brief conceptual explanation.

5. (4 points) Explain why positional encodings remain crucial for capturing sequence order and structural relationships in a Transformer, even when token-level embeddings differ between occurrences.