Instructions. Show intermediate calculations and explain your reasoning. Work by hand; a calculator may be used. Write answers on separate sheets and label each part clearly. All logarithms are natural. Vectors are columns unless stated otherwise. In Problems 6–13, each token or query occupies one row of a matrix, and softmax is applied separately to each row. Decimal answers are requested only in Problems 7 and 13.

1. Convolution output dimensions

Each input has shape $H \times W \times C_{\text{in}}$. Padding is applied equally to both sides of each spatial axis. Dilation is 1.

Input	Kernel	Stride	Padding	Output channels
$32 \times 32 \times 3$	3×3	1	1	16
$32 \times 32 \times 3$	5×5	2	2	8
$15 \times 19 \times 8$	3×5	2	0	4

For each row, compute the output shape $H_{\text{out}} \times W_{\text{out}} \times C_{\text{out}}$.

2. Two-dimensional convolution as a sparse matrix

Let $X \in \mathbb{R}^{4\times 4}$ and $K = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$. Use no padding, no bias, and dilation 1. Apply the kernel **without flipping it**:

$$Y_{r,c} = \sum_{i=0}^{1} \sum_{j=0}^{1} K_{i,j} X_{sr+i,\, sc+j},$$

where indices start at 0 and s is the stride. Flatten both X and Y row by row into column vectors; for example, $\text{vec}(X) = (X_{0,0}, X_{0,1}, \ldots, X_{3,3})^{\top}$.

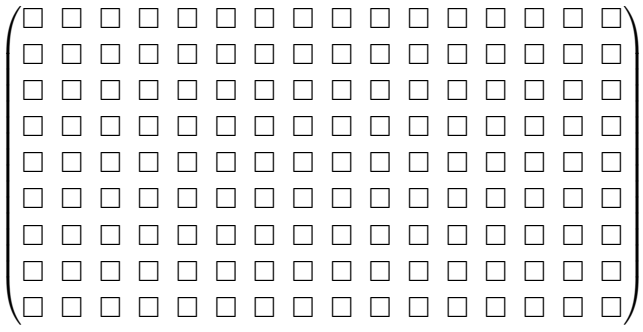
(a) For $s = 1$, write the matrix A_1 such that $\text{vec}(Y) = A_1 \,\text{vec}(X)$.

(b) For $s = 2$, write the matrix A_2 such that $\text{vec}(Y) = A_2 \,\text{vec}(X)$.

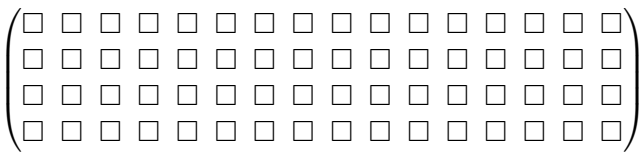
(c) For each matrix, count all entries and nonzero entries, and compute the percentage of entries that are zero.

Use the following layouts for (a) and (b); each square represents one entry.

$A_1 =$



$A_2 =$



3. Translation equivariance

Consider an input and the same input shifted one position to the right:

$$x = (0, 0, 1, 3, 2, 0, 0), \qquad x_{\text{shift}} = (0, 0, 0, 1, 3, 2, 0).$$

Use stride 1 and no padding for both operations below. Each output has six entries.

(a) Apply the convolution kernel $(1, 2)$ without flipping it: each adjacent pair (a, b) produces $a+2b$. Compute the outputs for x and x_{shift}.

(b) Apply max pooling with window size 2: each adjacent pair (a, b) produces $\max(a, b)$. Compute the outputs for x and x_{shift}.

(c) For each operation, compare the two outputs. Is the second output unchanged, or is it the first output shifted one position to the right? Here a right shift inserts 0 at the start and drops the last entry.

An output that shifts along with the input illustrates *translation equivariance*. No proof is required.

4. Residual connections

Compare a plain block and a residual block:

$$y_{\text{plain}} = Wx, \qquad y_{\text{res}} = x + Wx.$$

There is no activation after either expression. Use

$$x = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \qquad W = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/4 \end{pmatrix}.$$

For each block, the incoming column gradient is $g = \partial L/\partial y = (1, 1)^\top$.

(a) Compute y_{plain} and y_{res}.

(b) Compute $\partial L/\partial x$ for each block.

(c) Set $W = 0$ and keep x, g unchanged. Recompute both outputs and both input gradients.

5. RNN forward computation and input order

Let

$$h_t = \mathrm{ReLU}(Wh_{t-1} + ux_t), \qquad W = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad u = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad h_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Each x_t is a scalar, each $h_t \in \mathbb{R}^2$, and ReLU acts elementwise. The same W, u are used at every step.

(a) For $(x_1, x_2, x_3) = (1, 2, 3)$, compute h_1, h_2, h_3.

(b) Restart from $h_0 = 0$. For $(x_1, x_2, x_3) = (2, 1, 3)$, compute h_1, h_2, h_3.

(c) Both sequences end with $x_3 = 3$. Explain why their final hidden states differ.

6. Why query, key, and value?

For one query $q \in \mathbb{R}^{d_k}$ and key–value pairs (k_j, v_j), attention computes

$$a_j = \frac{\exp(q^\top k_j/\sqrt{d_k})}{\sum_\ell \exp(q^\top k_\ell/\sqrt{d_k})}, \qquad o = \sum_j a_j v_j.$$

(a) Match q, k_j, and v_j to these roles: the information being sought; the information used to assess a match; the information returned in the weighted sum.

(b) Keep q and all keys fixed, and change only the values. Which must remain unchanged: the weights a_j or the output o? Which can change?

(c) Keep the keys and values fixed, and change only q. Can the weights change? Can the output change? Explain using the formulas.

7. Why divide by $\sqrt{d_k}$?

Assume all components $q_1, \ldots, q_{d_k}, k_1, \ldots, k_{d_k}$ are mutually independent random variables, each with mean 0 and variance 1. Let

$$s = q^\top k = \sum_{i=1}^{d_k} q_i k_i.$$

(a) Compute $\mathbb{E}[s]$ and $\mathrm{Var}(s)$.

(b) Compute $\mathrm{Var}(s/\sqrt{d_k})$.

(c) For $d_k = 64$, suppose the unscaled scores for two keys are $(8, 0)$. Compute

$$\mathrm{softmax}(8,0) \quad \text{and} \quad \mathrm{softmax}\left(\frac{(8,0)}{\sqrt{64}}\right).$$

Give probabilities to four decimal places. Which distribution is more concentrated on one key?

8. Cross-attention

Two query tokens attend to three source tokens. Their queries, keys, and values are

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad K = \sqrt{2}\log 2 \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \qquad V = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

The rows of Q belong to the query sequence; the corresponding rows of K, V belong to the source sequence. Here $d_k = 2$.

(a) Compute $S = QK^\top/\sqrt{2}$.

(b) Compute $A = \mathrm{softmax}_{\mathrm{row}}(S)$, where $A_{ij} = e^{S_{ij}} / \sum_{\ell=1}^{3} e^{S_{i\ell}}$.

(c) Compute $O = AV$ and state its shape. Explain why it has two rows rather than three.

9. Self-attention

The three rows of X are three input tokens:

$$X = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}, \qquad W_Q = \sqrt{2}\log 2\, I_2, \qquad W_K = W_V = I_2.$$

Here I_2 is the 2×2 identity matrix and $d_k = 2$.

(a) Compute $Q = XW_Q$, $K = XW_K$, and $V = XW_V$.

(b) Compute all three matrices

$$S = \frac{QK^\top}{\sqrt{2}}, \qquad A = \mathrm{softmax}_{\mathrm{row}}(S), \qquad O = AV.$$

Here $A_{ij} = e^{S_{ij}} / \sum_{\ell=1}^{3} e^{S_{i\ell}}$.

10. Masked self-attention

Use the score matrix S and value matrix V from Problem 9. Token i may attend only to itself and earlier tokens. With row and column indices $i, j \in \{1, 2, 3\}$, define

$$M_{ij} = \begin{cases} 0, & j \le i, \\ -\infty, & j > i. \end{cases} \qquad A^{\mathrm{mask}} = \mathrm{softmax}_{\mathrm{row}}(S + M).$$

Use $e^{-\infty} = 0$.

(a) Write the masked score matrix $S + M$.

(b) Compute A^{mask} and $O^{\mathrm{mask}} = A^{\mathrm{mask}}V$.

(c) Explain why the first output row equals the first value row $V_{1,:}$.

11. Self-attention computation and memory

Consider single-head attention with $Q, K, V \in \mathbb{R}^{N\times d}$ already given. The implementation explicitly forms and stores the dense attention matrix

$$A = \mathrm{softmax}_{\mathrm{row}}(QK^\top/\sqrt{d}), \qquad O = AV.$$

Use ordinary matrix multiplication. Count only multiplications in the two matrix products; exclude projections, scaling, and softmax. For memory, count only the entries of A, each stored in 4 bytes.

(a) Compute the number of scalar multiplications in $QK^\top$ and in AV, separately, in terms of N, d.

(b) Compute the number of entries in A and the bytes needed to store it.

(c) With d fixed, state the computation and storage complexity as functions of N. If N doubles, by what factor does each increase?

12. Layer normalization in a Transformer

Each row of X is one token with four features:

$$X = \begin{pmatrix} 1 & 1 & 3 & 3 \\ 2 & 2 & 6 & 6 \end{pmatrix}, \qquad \gamma = (1,2,1,2), \qquad \beta = (0,0,1,-1).$$

Normalize each token separately across its four features. For this hand calculation, set $\epsilon = 0$:

$$\mu_i = \frac{1}{4}\sum_{j=1}^{4} X_{ij}, \qquad \sigma_i^2 = \frac{1}{4}\sum_{j=1}^{4}(X_{ij}-\mu_i)^2,$$

$$\widehat{X}_{ij} = \frac{X_{ij}-\mu_i}{\sqrt{\sigma_i^2}}, \qquad Y_{ij} = \gamma_j \widehat{X}_{ij} + \beta_j.$$

(a) Compute $\mu_1, \mu_2, \sigma_1^2, \sigma_2^2$.

(b) Compute the normalized matrix $\widehat{X}$.

(c) Compute the final output matrix Y.

13. Sinusoidal positional encoding

Use $d_{\text{model}} = 4$ and the sinusoidal encoding

$$PE(p, 2i) = \sin\left(\frac{p}{10000^{2i/d_{\text{model}}}}\right), \qquad PE(p, 2i+1) = \cos\left(\frac{p}{10000^{2i/d_{\text{model}}}}\right),$$

where $i = 0, 1$. Positions and feature indices start at 0, and angles are in radians. Write $PE(p) = (PE(p, 0), PE(p, 1), PE(p, 2), PE(p, 3))$.

(a) Compute $PE(0)$, $PE(1)$, and $PE(2)$.

(b) The same token embedding $x = (1, 0, 1, 0)$ appears at positions 0 and 2. Compute $x + PE(0)$ and $x + PE(2)$.

(c) Of the two sine components $PE(p, 0)$ and $PE(p, 2)$, which varies more slowly as p changes? Justify your answer using their arguments.

Give the numerical vectors in (a) and (b) to four decimal places.