

Rotary position embeddings

Ten problems on rotary position embeddings (RoPE): the 2-D rotation matrix and its algebra, why the score of a rotated query and key depends only on their relative position, the score as a sinusoid in the offset, the block-diagonal form and shift invariance, the complex-number form, the frequencies and wavelengths at $d = 128$, the backward pass through the rotation, the derivative with respect to the offset and the frequencies, RoPE inside attention backward, and position interpolation, with worked solutions and the mistakes that lose the position or flip its sign.

BEFORE YOU START

Rotary position embeddings put position into attention without adding anything to the token: each query and key is rotated, two coordinates at a time, by an angle proportional to its position, and the dot product of two rotated vectors then depends only on how far apart they are. That one fact is the whole design, and it follows from three lines of trigonometry. These ten problems derive it for one pair of coordinates, extend it to the full block-diagonal rotation, rewrite it with complex numbers, work out the frequencies and wavelengths used in practice, and then do what a framework does: backpropagate through the rotation, inside attention. The five mistakes are the ones that leave the model running: both vectors rotated by the same angle, a transposed rotation with the sign of the offset flipped, values rotated too, a backward pass that rotates forwards, and a pairing of coordinates that differs between queries and keys.

- The conventions are those of the attention page: vectors are columns, a gradient has the shape of its variable, and a token at position m has query $q \in \mathbb{R}^d$ and key $k \in \mathbb{R}^d$ with d even.
- The 2-D rotation by angle α is $R(\alpha) = \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix}$; it turns a vector anticlockwise by α . The trig-identities page's addition formulas are used freely.
- The coordinates of q are grouped into $d/2$ pairs: pair p is $q^{(p)} = (q_{2p-1}, q_{2p})^\top \in \mathbb{R}^2$ for $p = 1, \dots, d/2$. Each pair has its own frequency $\theta_p = 10000^{-2(p-1)/d}$, so $\theta_1 = 1$ and the frequencies decrease geometrically. The ratio θ_{p+1}/θ_p is $10000^{-2/d}$.
- RoPE at position m rotates pair p by the angle $m\theta_p$: $R_m = \text{blockdiag}(R(m\theta_1), \dots, R(m\theta_{d/2}))$, a $d \times d$ matrix, and the rotated query is $\tilde{q} = R_m q$. A key at position n becomes $\tilde{k} = R_n k$. The score is $s = \tilde{q}^\top \tilde{k}$ (the attention page's S_{mn} before the division by $\sqrt{d}$). Values are not rotated.
- Implementations that pair coordinate j with $j + d/2$ instead ("rotate half") are the same construction after one fixed permutation of the coordinates, applied to every q and k alike.

- Complex form: a pair $(x, y)^\top$ is the complex number $x + iy$ with $i^2 = -1$; $\bar{z}$ is the conjugate and Re the real part. Euler's formula is $e^{i\alpha} = \cos \alpha + i \sin \alpha$.
- L is a scalar loss and $g = \partial L / \partial s$ the upstream gradient of a score; inside attention, G , $\tilde{G}$ and $\nabla_s L$ are the attention page's matrices.

Builds on: [Attention backward](#), [Trig identities](#)

Problems

1. Show that $R(\alpha)^\top R(\alpha) = I$, $\det R(\alpha) = 1$, $R(\alpha)R(\beta) = R(\alpha + \beta)$ and $R(\alpha)^\top = R(-\alpha)$. difficulty ·

2. One pair with frequency θ : the query $q \in \mathbb{R}^2$ at position m and the key $k \in \mathbb{R}^2$ at position n . Show that $(R(m\theta)q)^\top (R(n\theta)k) = q^\top R((n - m)\theta)k$. difficulty ·

3. For one pair, write $s(\varphi) = q^\top R(\varphi)k$ in the form $a \cos \varphi + b \sin \varphi$, giving a and b in terms of the entries of q and k . Show that $a^2 + b^2 = \|q\|^2 \|k\|^2$, and hence that $s(\varphi) = \|q\| \|k\| \cos(\varphi - \psi)$ for some angle ψ . What is ψ ? difficulty · ·

4. Full dimension d . Show that $R_m^\top R_n = R_{n-m}$, that the score $(R_m q)^\top (R_n k)$ is a sum over pairs that depends on m and n only through $n - m$ (so it is unchanged if both positions shift by the same s), and that $\|R_m q\| = \|q\|$. difficulty · ·

5. Complex form. Write pair p of q as $z_p = q_{2p-1} + iq_{2p}$ and of k as $w_p = k_{2p-1} + ik_{2p}$. Show that rotating a pair by α is multiplying its complex number by $e^{i\alpha}$, that $q^{(p)\top} k^{(p)} = \text{Re}(z_p \bar{w}_p)$, and that the RoPE score is $\text{Re} \sum_p z_p \bar{w}_p e^{i(m-n)\theta_p}$. difficulty · ·

6. With $d = 128$ and base 10000: give θ_1 and θ_{64} , the wavelength $2\pi/\theta_p$ of the first and last pairs in positions, and the number of pairs whose wavelength is at most a context length of 4096 (the pairs that complete at least one full turn inside the context). difficulty ·

7. Backward through the rotation. For $s = (R_m q)^\top (R_n k)$ with upstream gradient $g = \partial L / \partial s$, compute $\nabla_q L$ and $\nabla_k L$. Then, for a general vector $u = \nabla_{\tilde{q}} L$ of upstream gradient at the rotated query $\tilde{q} = R_m q$, give $\nabla_q L$. difficulty · · ·

8. Show that $\frac{d}{d\varphi} R(\varphi) = R(\varphi + \pi/2)$. Treating the offset $\Delta = n - m$ as a difficulty ... continuous variable, compute $\partial s / \partial \Delta$ for the full score $s(\Delta) = \sum_p q^{(p)\top} R(\Delta \theta_p) k^{(p)}$, and compute $\partial s / \partial \theta_p$ for learned or rescaled frequencies.

9. Attention with RoPE: $\tilde{Q}$ and $\tilde{K}$ have rows $(R_m q_m)^\top$ and $(R_n k_n)^\top$ for difficulty ... tokens at positions $m, n = 1, \dots, N$, $S = \tilde{Q} \tilde{K}^\top / \sqrt{d}$, $A = \text{softmax}(S)$ row-wise, $O = AV$. Using the attention page's results for $\nabla_{\tilde{Q}} L$ and $\nabla_{\tilde{K}} L$, write $\nabla_{q_m} L$ and $\nabla_{k_n} L$. Then show that shifting every position by the same s leaves A and O unchanged.

10. Position interpolation extends a model trained at context L_0 to difficulty ... context sL_0 by using positions m/s in place of m . Show that this equals keeping the positions and dividing every frequency by s , that every wavelength is multiplied by s , and compute, for $d = 128$ and $s = 4$, how many pairs now complete a turn within 4096 positions.

Answers

1. $R(\alpha)^\top R(\alpha) = I$, $\det R(\alpha) = 1$, $R(\alpha)R(\beta) = R(\alpha + \beta)$, $R(\alpha)^\top = R(-\alpha)$
2. $(R(m\theta)q)^\top (R(n\theta)k) = q^\top R((n - m)\theta)k$
3. $s(\varphi) = (q^\top k) \cos \varphi + (q_2 k_1 - q_1 k_2) \sin \varphi = \|q\| \|k\| \cos(\varphi - \psi)$, ψ the angle from k to q
4. $R_m^\top R_n = R_{n-m}$; $(R_m q)^\top (R_n k) = \sum_p q^{(p)\top} R((n - m)\theta_p) k^{(p)}$, a function of $n - m$ alone;
 $\|R_m q\| = \|q\|$
5. Rotation by α is multiplication by $e^{i\alpha}$; $q^{(p)\top} k^{(p)} = \text{Re}(z_p \bar{w}_p)$; score = $\text{Re} \sum_p z_p \bar{w}_p e^{i(m-n)\theta_p}$
6. $\theta_1 = 1$, $\theta_{64} \approx 1.15 \times 10^{-4}$; wavelengths ≈ 6.3 and $\approx 54,400$ positions; 46 of the 64 pairs complete a turn within 4096 positions
7. $\nabla_q L = g R_{n-m} k$, $\nabla_k L = g R_{m-n} q$; in general $\nabla_q L = R_{-m} \nabla_{\tilde{q}} L$: rotate the upstream gradient back by the token's own angle
8. $\frac{d}{d\varphi} R(\varphi) = R(\varphi + \pi/2)$; $\partial s / \partial \Delta = \sum_p \theta_p q^{(p)\top} R(\Delta \theta_p + \pi/2) k^{(p)}$; $\partial s / \partial \theta_p = \Delta q^{(p)\top} R(\Delta \theta_p + \pi/2) k^{(p)}$
9. $\nabla_{q_m} L = R_{-m} \left(\frac{1}{\sqrt{d}} (\nabla_S L) \tilde{K} \right)_{m,:}^\top$, $\nabla_{k_n} L = R_{-n} \left(\frac{1}{\sqrt{d}} (\nabla_S L)^\top \tilde{Q} \right)_{n,:}^\top$: each row of the attention page's gradient rotated back by its own position; A and O are invariant to a common shift of all positions
10. Positions m/s with frequencies θ_p equal positions m with frequencies θ_p/s ; wavelengths scale by s ; at $d = 128$, $s = 4$, 36 pairs complete a turn within 4096 positions, down from 46

Worked solutions

Problem 1

Show that $R(\alpha)^\top R(\alpha) = I$, $\det R(\alpha) = 1$, $R(\alpha)R(\beta) = R(\alpha + \beta)$ and $R(\alpha)^\top = R(-\alpha)$.

$$R(\alpha)^\top R(\alpha) = \begin{pmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} = \begin{pmatrix} \cos^2 \alpha + \sin^2 \alpha & 0 \\ 0 & \sin^2 \alpha + \cos^2 \alpha \end{pmatrix} = I.$$

Multiply out; the off-diagonal entries are $-\cos \alpha \sin \alpha + \sin \alpha \cos \alpha = 0$, and $\cos^2 + \sin^2 = 1$.

$$\det R(\alpha) = \cos^2 \alpha + \sin^2 \alpha = 1. \quad ad - bc \text{ with } b = -\sin \alpha.$$

$$R(\alpha)R(\beta) = \begin{pmatrix} \cos \alpha \cos \beta - \sin \alpha \sin \beta & -\cos \alpha \sin \beta - \sin \alpha \cos \beta \\ \sin \alpha \cos \beta + \cos \alpha \sin \beta & -\sin \alpha \sin \beta + \cos \alpha \cos \beta \end{pmatrix} = \begin{pmatrix} \cos(\alpha + \beta) & -\sin(\alpha + \beta) \\ \sin(\alpha + \beta) & \cos(\alpha + \beta) \end{pmatrix} = R(\alpha + \beta).$$

The entries are the addition formulas:
 $\cos(\alpha + \beta) = \cos \alpha \cos \beta - \sin \alpha \sin \beta$ and
 $\sin(\alpha + \beta) = \sin \alpha \cos \beta + \cos \alpha \sin \beta$.

$$R(\alpha)^\top = \begin{pmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{pmatrix} = \begin{pmatrix} \cos(-\alpha) & -\sin(-\alpha) \\ \sin(-\alpha) & \cos(-\alpha) \end{pmatrix} = R(-\alpha).$$

$\cos$ is even and $\sin$ is odd.

$$R(\alpha)^\top R(\alpha) = I, \det R(\alpha) = 1, \\ R(\alpha)R(\beta) = R(\alpha + \beta), R(\alpha)^\top = R(-\alpha)$$

A rotation preserves lengths and angles (orthogonal, determinant +1 so no reflection), rotations compose by adding angles, and the inverse of a rotation is its transpose, which is the rotation back. Step 3 also shows rotations in the plane commute: $R(\alpha)R(\beta) = R(\beta)R(\alpha)$.

Problem 2

One pair with frequency θ : the query $q \in \mathbb{R}^2$ at position m and the key $k \in \mathbb{R}^2$ at position n . Show that $(R(m\theta)q)^\top (R(n\theta)k) = q^\top R((n - m)\theta)k$.

$$(R(m\theta)q)^\top (R(n\theta)k) = q^\top R(m\theta)^\top R(n\theta)k. \quad (Ab)^\top = b^\top A^\top.$$

$R(m\theta)^\top = R(-m\theta).$	Problem 1, step 4.
$R(-m\theta)R(n\theta) = R(n\theta - m\theta) = R((n - m)\theta).$	Problem 1, step 3: angles add.
$(R(m\theta)q)^\top (R(n\theta)k) = q^\top R((n - m)\theta)k$	The absolute positions m and n have cancelled: the score is the unrotated q against k rotated by the <i>relative</i> offset $n - m$ times the frequency. Nothing was assumed about q or k , so this holds for whatever the projections produce.

Problem 3

For one pair, write $s(\varphi) = q^\top R(\varphi)k$ in the form $a \cos \varphi + b \sin \varphi$, giving a and b in terms of the entries of q and k . Show that $a^2 + b^2 = \|q\|^2 \|k\|^2$, and hence that $s(\varphi) = \|q\| \|k\| \cos(\varphi - \psi)$ for some angle ψ . What is ψ ?

$R(\varphi)k = (k_1 \cos \varphi - k_2 \sin \varphi, k_1 \sin \varphi + k_2 \cos \varphi)^\top.$	Matrix times vector.
$s(\varphi) = q_1 k_1 \cos \varphi - q_1 k_2 \sin \varphi + q_2 k_1 \sin \varphi + q_2 k_2 \cos \varphi = (q_1 k_1 + q_2 k_2) \cos \varphi + (q_2 k_1 - q_1 k_2) \sin \varphi.$	Dot with q and collect the cos and sin terms.
$a = q_1 k_1 + q_2 k_2 = q^\top k$ and $b = q_2 k_1 - q_1 k_2.$	Read off step 2. a is the dot product; b is the 2-D cross product of k and q (the signed area of their parallelogram).
$a^2 + b^2 = q_1^2 k_1^2 + 2q_1 k_1 q_2 k_2 + q_2^2 k_2^2 + q_2^2 k_1^2 - 2q_2 k_1 q_1 k_2 + q_1^2 k_2^2 = (q_1^2 + q_2^2)(k_1^2 + k_2^2).$	Expand both squares; the cross terms cancel and the remaining four terms factor.

Let $r = \ q\ \ k\$ and choose ψ with $r \cos \psi = a$ and $r \sin \psi = b$. Then $s(\varphi) = r(\cos \psi \cos \varphi + \sin \psi \sin \varphi) = r \cos(\varphi - \psi)$.	Step 4 says (a, b) lies on the circle of radius r, so such a ψ exists (when $r > 0$); then the difference formula for cosine (the trig-identities page).
ψ is the angle from k to q.	With k along the first axis, $k = (\ k\ , 0)^\top$ and $q = \ q\ (\cos \psi', \sin \psi')^\top$ at angle ψ': then $a = \ q\ \ k\ \cos \psi'$ and $b = \ q\ \ k\ \sin \psi'$, so $\psi = \psi'$. The dot product is unchanged by rotating both vectors, so this holds in general.
$s(\varphi) = (q^\top k) \cos \varphi + (q_2 k_1 - q_1 k_2) \sin \varphi = \ q\ \ k\ \cos(\varphi - \psi)$, ψ the angle from k to q	As the offset $n - m$ grows, the score of one pair oscillates between $\pm \ q\ \ k\$ with period $2\pi/\theta$, and is largest when the rotation by $\varphi = (n - m)\theta$ brings k into line with q. The full score (Problem 4) is a sum of $d/2$ such sinusoids with different periods, which is what lets it single out particular offsets.

Problem 4

Full dimension d . Show that $R_m^\top R_n = R_{n-m}$, that the score $(R_m q)^\top (R_n k)$ is a sum over pairs that depends on m and n only through $n - m$ (so it is unchanged if both positions shift by the same s), and that $\|R_m q\| = \|q\|$.

$R_m^\top R_n = \text{blockdiag} \left(R(m\theta_p)^\top R(n\theta_p) \right)_p = \text{blockdiag} \left(R((n - m)\theta_p) \right)_p = R_{n-m}.$	Block-diagonal matrices multiply block by block, and the transpose of a block-diagonal matrix is block-diagonal with transposed blocks; each block is Problem 2 with $\theta = \theta_p$.
$(R_m q)^\top (R_n k) = q^\top R_m^\top R_n k = q^\top R_{n-m} k.$	$(Ab)^\top = b^\top A^\top$ and step 1.
$q^\top R_{n-m} k = \sum_{p=1}^{d/2} q^{(p)\top} R((n - m)\theta_p) k^{(p)}.$	A block-diagonal matrix acts on each pair separately, and the dot product adds up the pairs.

Replacing (m, n) by $(m + s, n + s)$ leaves $n - m$ unchanged, so the score is unchanged.	Step 3 contains m and n only in the combination $n - m$.
$\ R_m q\ ^2 = q^\top R_m^\top R_m q = q^\top R_0 q = q^\top q$.	Step 1 with $n = m$, and $R_0 = I$ since $R(0) = I$.
$R_m^\top R_n = R_{n-m}; (R_m q)^\top (R_n k) =$ $\sum_p q^{(p)\top} R((n - m)\theta_p) k^{(p)},$ a function of $n - m$ alone; $\ R_m q\ = \ q\ $	The score depends on relative position, and so do the attention weights built from it (Problem 9): a sequence shifted by s positions attends the same way. Each pair contributes $\ q^{(p)}\ \ k^{(p)}\ \cos((n - m)\theta_p - \psi_p)$ by Problem 3, and the rotation never changes the size of a query or key, so the $1/\sqrt{d}$ scaling argument of the attention page is untouched.

Problem 5

Complex form. Write pair p of q as $z_p = q_{2p-1} + i q_{2p}$ and of k as $w_p = k_{2p-1} + i k_{2p}$. Show that rotating a pair by α is multiplying its complex number by $e^{i\alpha}$, that $q^{(p)\top} k^{(p)} = \text{Re}(z_p \bar{w}_p)$, and that the RoPE score is $\text{Re} \sum_p z_p \bar{w}_p e^{i(m-n)\theta_p}$.

$e^{i\alpha}(x + iy) = (\cos \alpha + i \sin \alpha)(x + iy) =$ $(x \cos \alpha - y \sin \alpha) + i(x \sin \alpha + y \cos \alpha).$	Euler's formula, then multiply out using $i^2 = -1$.
The real and imaginary parts are the two entries of $R(\alpha)(x, y)^\top$.	Compare with $R(\alpha)$ applied to $(x, y)^\top$: first entry $x \cos \alpha - y \sin \alpha$, second $x \sin \alpha + y \cos \alpha$.
$z \bar{w} = (x_1 + iy_1)(x_2 - iy_2) = (x_1 x_2 +$ $y_1 y_2) + i(y_1 x_2 - x_1 y_2),$ so $\text{Re}(z \bar{w}) =$ $x_1 x_2 + y_1 y_2.$	Multiply out; the real part is the dot product of the two pairs, and the imaginary part is Problem 3's b .
$\tilde{z}_p = e^{im\theta_p} z_p$ and $\tilde{w}_p = e^{in\theta_p} w_p$ are the rotated pairs.	Step 2 with $\alpha = m\theta_p$ and $n\theta_p$.

$$\tilde{z}_p \overline{\tilde{w}_p} = e^{im\theta_p} z_p e^{-in\theta_p} \bar{w}_p = z_p \bar{w}_p e^{i(m-n)\theta_p}.$$

$$\overline{e^{i\beta} w} = e^{-i\beta} \bar{w}, \text{ and exponents add.}$$

Rotation by α is multiplication by

$$e^{i\alpha}; q^{(p)\top} k^{(p)} = \text{Re}(z_p \bar{w}_p); \text{score} =$$

$$\text{Re} \sum_p z_p \bar{w}_p e^{i(m-n)\theta_p}$$

Step 3 applied to the rotated pairs and summed over p . This is how RoPE is usually implemented: view the last axis as complex, multiply by a precomputed table of $e^{im\theta_p}$, view as real again. The relative-position property is the exponent rule $e^{im\theta} e^{-in\theta} = e^{i(m-n)\theta}$, which is Problem 1's $R(\alpha)R(\beta) = R(\alpha + \beta)$ in different clothes.

Problem 6

With $d = 128$ and base 10000: give θ_1 and θ_{64} , the wavelength $2\pi/\theta_p$ of the first and last pairs in positions, and the number of pairs whose wavelength is at most a context length of 4096 (the pairs that complete at least one full turn inside the context).

$$\theta_1 = 10000^0 = 1 \text{ and } \theta_{64} = 10000^{-126/128}.$$

$$p = 1 \text{ gives exponent } 0; p = 64 \text{ gives } -2 \cdot 63/128 = -126/128.$$

$$10000^{126/128} = e^{(126/128) \ln 10000} =$$

$$e^{0.984375 \cdot 9.2103} \approx e^{9.0664} \approx 8660, \text{ so } \theta_{64} \approx 1.15 \times 10^{-4}.$$

$$\ln 10000 = 4 \ln 10 \approx 9.2103.$$

Wavelength of pair 1: $2\pi/1 \approx 6.28$ positions; of pair 64: $2\pi \cdot 8660 \approx 54,400$ positions.

A pair rotates by θ_p per position, so it completes a turn every $2\pi/\theta_p$ positions.

Pair p completes a turn within 4096 positions iff $2\pi/\theta_p \leq 4096$ iff $10000^{2(p-1)/128} \leq 4096/(2\pi)$.

Substitute θ_p and invert; the inequality direction is preserved because $\theta_p > 0$.

$$\frac{2(p-1)}{128} \ln 10000 \leq \ln \frac{4096}{2\pi} \approx \ln 651.9 \approx 6.480, \text{ so } p-1 \leq \frac{128 \cdot 6.480}{2 \cdot 9.2103} \approx 45.03.$$

$$\text{Take logs; } 4096/(2\pi) \approx 651.9.$$

$\theta_1 = 1, \theta_{64} \approx 1.15 \times 10^{-4}$; **wavelengths**
 ≈ 6.3 **and** $\approx 54,400$ **positions; 46 of**
the 64 pairs complete a turn within
4096 positions

$p \leq 46.03$, so $p = 1, \dots, 46$. The remaining 18 pairs turn through less than a full circle over the whole context and act like a slowly varying, almost monotone position signal; the fast pairs resolve nearby offsets. Trained at a shorter context, those slow pairs have never seen the angles a longer context produces, which is the problem position interpolation addresses (Problem 10).

Problem 7

Backward through the rotation. For $s = (R_m q)^\top (R_n k)$ with upstream gradient $g = \partial L / \partial s$, compute $\nabla_q L$ and $\nabla_k L$. Then, for a general vector $u = \nabla_{\tilde{q}} L$ of upstream gradient at the rotated query $\tilde{q} = R_m q$, give $\nabla_q L$.

$$s = q^\top R_{n-m} k.$$

Problem 4, step 2.

$$\nabla_q s = R_{n-m} k.$$

$s = q^\top c$ with $c = R_{n-m} k$ not depending on q , and $\nabla_q (q^\top c) = c$ (the matrix-calculus page).

$$s = k^\top R_{n-m}^\top q = k^\top R_{m-n} q, \text{ so } \nabla_k s = R_{m-n} q.$$

A scalar equals its transpose; $R_{n-m}^\top = R_{m-n}$ by Problem 1, step 4, block by block.

$$\nabla_q L = g R_{n-m} k \text{ and } \nabla_k L = g R_{m-n} q.$$

L depends on q and k only through s , so the chain rule multiplies by g .

$$\tilde{q} = R_m q \text{ gives } \partial \tilde{q} / \partial q = R_m, \text{ so } \nabla_q L = R_m^\top \nabla_{\tilde{q}} L = R_m^\top u = R_{-m} u.$$

A linear map's Jacobian is its matrix, and for a scalar L , $\nabla_q L = (\partial \tilde{q} / \partial q)^\top \nabla_{\tilde{q}} L$ (the Jacobians page); then $R_m^\top = R_{-m}$.

$\nabla_q L = g R_{n-m} k$, $\nabla_k L = g R_{m-n} q$; **in general** $\nabla_q L = R_{-m} \nabla_{\tilde{q}} L$: **rotate the upstream gradient back by the token's own angle**

Step 4 is step 5 applied to $\nabla_{\tilde{q}} L = g \tilde{k} = g R_n k$: $R_{-m} R_n k = R_{n-m} k$. The backward pass of RoPE is RoPE with the sign of the position flipped, which in the complex form is multiplication by $e^{-im\theta_p}$, the conjugate table. It costs the same as the forward pass and needs no stored activations beyond the positions.

Problem 8

Show that $\frac{d}{d\varphi} R(\varphi) = R(\varphi + \pi/2)$. Treating the offset $\Delta = n - m$ as a continuous variable, compute $\partial s / \partial \Delta$ for the full score $s(\Delta) = \sum_p q^{(p)\top} R(\Delta \theta_p) k^{(p)}$, and compute $\partial s / \partial \theta_p$ for learned or rescaled frequencies.

$$\frac{d}{d\varphi} R(\varphi) = \begin{pmatrix} -\sin \varphi & -\cos \varphi \\ \cos \varphi & -\sin \varphi \end{pmatrix}.$$

Differentiate each entry.

$$R(\varphi + \pi/2) = \begin{pmatrix} \cos(\varphi + \pi/2) & -\sin(\varphi + \pi/2) \\ \sin(\varphi + \pi/2) & \cos(\varphi + \pi/2) \end{pmatrix} = \begin{pmatrix} -\sin \varphi & -\cos \varphi \\ \cos \varphi & -\sin \varphi \end{pmatrix}.$$

$\cos(\varphi + \pi/2) = -\sin \varphi$ and $\sin(\varphi + \pi/2) = \cos \varphi$ (the trig-identities page). The two matrices agree.

$$\frac{\partial}{\partial \Delta} q^{(p)\top} R(\Delta \theta_p) k^{(p)} = \theta_p q^{(p)\top} R(\Delta \theta_p + \pi/2) k^{(p)}.$$

Chain rule: the angle $\Delta \theta_p$ has derivative θ_p with respect to Δ , and step 2 gives the derivative of R with respect to its angle. $q^{(p)}$ and $k^{(p)}$ are constants here.

$$\partial s / \partial \Delta = \sum_p \theta_p q^{(p)\top} R(\Delta \theta_p + \pi/2) k^{(p)}.$$

Sum step 3 over the pairs.

θ_p appears only in pair p 's term, and the angle $\Delta \theta_p$ has derivative Δ with respect to θ_p .

Each frequency belongs to one block.

$$\partial s / \partial \theta_p = \Delta q^{(p)\top} R(\Delta \theta_p + \pi/2) k^{(p)}.$$

Chain rule as in step 3 with the roles of Δ and θ_p swapped.

$$\frac{d}{d\varphi} R(\varphi) = R(\varphi + \pi/2); \partial s / \partial \Delta = \sum_p \theta_p q^{(p)\top} R(\Delta \theta_p + \pi/2) k^{(p)}; \partial s / \partial \theta_p = \Delta q^{(p)\top} R(\Delta \theta_p + \pi/2) k^{(p)}$$

For one pair, by Problem 3 with $\varphi \rightarrow \varphi + \pi/2$, the derivative is $-a \sin \varphi + b \cos \varphi$, the derivative of $a \cos \varphi + b \sin \varphi$. The frequency gradient carries a factor Δ : distant pairs of tokens push the frequencies hardest, which is why learned or fine-tuned frequencies are sensitive to the longest offsets in the training data.

Problem 9

Attention with RoPE: $\tilde{Q}$ and $\tilde{K}$ have rows $(R_m q_m)^\top$ and $(R_n k_n)^\top$ for tokens at positions $m, n = 1, \dots, N$, $S = \tilde{Q} \tilde{K}^\top / \sqrt{d}$, $A = \text{softmax}(S)$ row-wise, $O = AV$. Using the attention page's results for $\nabla_{\tilde{Q}} L$ and $\nabla_{\tilde{K}} L$, write $\nabla_{q_m} L$ and $\nabla_{k_n} L$. Then show that shifting every position by the same s leaves A and O unchanged.

$$\nabla_{\tilde{Q}} L = \frac{1}{\sqrt{d}} (\nabla_S L) \tilde{K} \text{ and } \nabla_{\tilde{K}} L = \frac{1}{\sqrt{d}} (\nabla_S L)^\top \tilde{Q}.$$

The attention page's Problems 5 and 6, with $\tilde{Q}, \tilde{K}$ in place of Q, K : the attention computation is unchanged, it only receives rotated inputs.

Row m of $\tilde{Q}$ depends on q_m alone, through $\tilde{q}_m = R_m q_m$.

Each token is rotated by its own position; no other token's query enters its row.

$$\nabla_{q_m} L = R_m^\top (\text{row } m \text{ of } \nabla_{\tilde{Q}} L)^\top = R_{-m} (\nabla_{\tilde{Q}} L)_{m,:}^\top.$$

Problem 7, step 5, applied to row m : the upstream gradient at $\tilde{q}_m$ is row m of $\nabla_{\tilde{Q}} L$, written as a column.

$$\nabla_{k_n} L = R_{-n} (\nabla_{\tilde{K}} L)_{n,:}^\top.$$

The same argument for keys, with $\tilde{k}_n = R_n k_n$.

$S_{mn} = (R_m q_m)^\top (R_n k_n) / \sqrt{d}$ depends on m and n only through $n - m$.

Problem 4, step 4, for every entry.

Shifting all positions by s leaves every S_{mn} unchanged, hence A and $O = AV$.

$(n + s) - (m + s) = n - m$; A is a function of S and V is not rotated.

$\nabla_{q_m} L = R_{-m} \left(\frac{1}{\sqrt{d}} (\nabla_S L) \tilde{K} \right)_{m,:}^\top$, $\nabla_{k_n} L = R_{-n} \left(\frac{1}{\sqrt{d}} (\nabla_S L)^\top \tilde{Q} \right)_{n,:}^\top$: **each row of the attention page's gradient rotated back by its own position; A and O are invariant to a common shift of all positions**

So RoPE is a layer wrapped around attention: forward, rotate Q and K row by row; backward, rotate the gradients back row by row and hand the rest to the attention page. The shift invariance means a RoPE model sees no absolute position at all; the only asymmetry it has is the causal mask, which tells a token how many keys lie before it.

Problem 10

Position interpolation extends a model trained at context L_0 to context sL_0 by using positions m/s in place of m . Show that this equals keeping the positions and dividing every frequency by s , that every wavelength is multiplied by s , and compute, for $d = 128$ and $s = 4$, how many pairs now complete a turn within 4096 positions.

Pair p at position m/s is rotated by $(m/s)\theta_p = m(\theta_p/s)$.

The angle is a product, and the factor $1/s$ can be attached to either side.

So the score with positions m/s , n/s and frequencies θ_p equals the score with positions m , n and frequencies θ_p/s .

Step 1 applies to every pair of every query and key; the rotated vectors are identical.

The wavelength becomes $2\pi/(\theta_p/s) = s \cdot 2\pi/\theta_p$.

Problem 6's definition with the new frequency.

Pair p completes a turn within 4096 iff $s \cdot 2\pi/\theta_p \leq 4096$ iff $10000^{2(p-1)/128} \leq 4096/(2\pi s) = 4096/(8\pi) \approx 163.0$.

Problem 6, step 4, with $2\pi s$ in place of 2π .

$p - 1 \leq \frac{128 \ln 163.0}{2 \ln 10000} \approx \frac{128 \cdot 5.094}{18.42} \approx 35.4$,
 so $p \leq 36.4$.

Take logs as in Problem 6, step 5.

**Positions m/s with frequencies θ_p
equal positions m with frequencies
 θ_p/s ; wavelengths scale by s ; at $d =$
128, $s = 4$, 36 pairs complete a turn
within 4096 positions, down from 46**

The first pair's wavelength goes from 2π to 8π positions. Every angle the model sees at the long context is one it saw during training (its range is the same as at L_0), which is why interpolation needs little fine-tuning; the cost is that adjacent positions now differ by a quarter of the angle they used to, so the fast pairs resolve neighbours less sharply. Schemes such as NTK-aware scaling and YaRN divide the slow pairs' frequencies by more than the fast pairs', trading between the two effects, and Problems 6 and 10 are the arithmetic they rest on.

Where this goes wrong

1. Rotating the query and the key by the same angle

RoPE is applied to queries and keys with one function, and the position passed in has to be the right one for each.

$\tilde{q} = R_m q$ for the query at position m

Right so far.

“Apply the same rotation to the key.”

The slip that causes the mistake: the key belongs to position n , and its rotation must use n , not the query's m .

$$s = (R_m q)^\top (R_m k) = q^\top R_m^\top R_m k = q^\top k \quad \text{✗}$$

Problem 4, step 1 with $n = m$: $R_m^\top R_m = I$. Every score is the plain dot product, the positions have cancelled completely, and the model is a bag of tokens with no position information at all; the loss still goes down, so nothing fails loudly.

2. Transposing the wrong rotation

The relative-position step has a transpose in it, and which rotation carries it decides the sign of the offset.

$(R_m q)^\top (R_n k) = q^\top R_m^\top R_n k$	Right so far: Problem 2, step 1.
“ $R_m^\top R_n$ is the rotation by $m - n$.”	The shortcut that causes the mistake: the transposed factor is the one whose angle is negated, so $R_m^\top R_n = R_{-m} R_n = R_{n-m}$.
$s = q^\top R_{m-n} k$ ✗	The offset's sign is flipped. For one pair the score is $a \cos \varphi + b \sin \varphi$ (Problem 3), and $\varphi \rightarrow -\varphi$ flips the sign of the b term, so the two agree only when $b = q_2 k_1 - q_1 k_2 = 0$, that is, when q and k are parallel. In Problem 7 the same slip gives $\nabla_q L = g R_{m-n} k$ in place of $g R_{n-m} k$, which fails a gradient check for every pair of positions except $m = n$.

3. Rotating the values as well

The rotation is applied to two of the three projections, and the third looks like an omission.

$\tilde{q}_m = R_m q_m$ and $\tilde{k}_n = R_n k_n$ give S_{mn} depending on $n - m$ only	Right so far: Problem 4.
“For consistency, rotate V by position too.”	The analogy that causes the mistake: Q , K and V are produced alike, so they should be treated alike; but position enters attention through the scores, and V never touches a score.

$$O_m = \sum_n A_{mn} R_n v_n \quad \times$$

Shifting all positions by s leaves A unchanged (Problem 9) but replaces R_n by R_{n+s} , so the output of every token changes with the absolute position of the sequence, and the relative-position property is lost at the output. With the rotation on Q and K only, $O_m = \sum_n A_{mn} v_n$ is invariant to the shift.

4. Backward pass that rotates forwards

The backward pass of a layer reuses the forward code, and for RoPE the forward code rotates by $+m\theta_p$.

$$\tilde{q} = R_m q \text{ and the upstream gradient } u = \nabla_{\tilde{q}} L$$

Right so far.

“The gradient flows back through the same rotation.”

The shortcut that causes the mistake: the backward of a linear map is its transpose, and $R_m^\top = R_{-m}$ is the rotation the other way.

$$\nabla_q L = R_m u \quad \times$$

Problem 7 gives $R_{-m} u$. For the score $s = \tilde{q}^\top \tilde{k}$ this makes $\nabla_q L = g R_m R_n k = g R_{m+n} k$ instead of $g R_{n-m} k$: a gradient with the right norm and the wrong direction, by the angle $2m\theta_p$ in pair p . In the complex implementation the fix is one conjugate: multiply by $e^{-im\theta_p}$ on the way back.

5. Pairing the coordinates differently for queries and keys

The paper pairs coordinate $2p - 1$ with $2p$; many implementations pair j with $j + d/2$. Both are RoPE, until they are mixed.

Pair (q_{2p-1}, q_{2p}) rotated by $m\theta_p$, and the same pairing for k , gives a score depending on $n - m$

Right so far: Problem 4, whichever fixed pairing is used, provided it is the same for q and k .

“The pairing is a convention, so the query code can use one and the key code (or the pretrained weights) another.”

The assumption that causes the mistake: Problem 2 cancels $R_m^\top R_n$ block by block, which needs the blocks of R_m and R_n to sit on the same pairs of coordinates.

Rotate (q_{2p-1}, q_{2p}) by $m\theta_p$ but $(k_j, k_{j+d/2})$ by $n\theta_j$ ✗

$R_m^\top R_n$ is no longer block-diagonal on matching pairs, so the score is not a function of $n - m$: it changes when the sequence is shifted, and the model trained with one convention is garbage under the other. This is the bug behind permuting the query and key projection weights when converting checkpoints between the two layouts; the test is Problem 4, step 4: compute the score at (m, n) and at $(m + s, n + s)$ and check they agree.

eulerpad.app/practice/rotary-position-embeddings