Attention, 40 pts

In this problem, we will explore the expressiveness of the attention mechanism. Recall that we have n keys $k_1, \ldots, k_n \in \mathbb{R}^d$ and values $v_1, \ldots, v_n \in \mathbb{R}^d$. On a query $q \in \mathbb{R}^d$, the attention score is $s_i = \langle k_i, q \rangle$, the weight is $\alpha_i = \frac{\exp(s_i)}{\sum_{j=1}^n \exp(s_j)}$, and the output is $\sum_{i=1}^n \alpha_i v_i$.

(a) (10 pts) **Copying a distinct value**. Assume $\|k_i\| = 1 \; \forall i$ and $\|k_1 - k_i\| \geq 1 \; \forall i \neq 1$. Fix $\epsilon \in (0, 1)$. Give a query q of length at most $2\ln(n/\epsilon)$ such that $\alpha_1 \geq 1/(1+\epsilon)$.

(b) (10 pts) **Averaging two values**. Assume $\|k_i\| = 1 \; \forall i$ and keys are orthogonal $\langle k_i, k_j \rangle = 0 \; \forall i \neq j$. Give a query q of length at most $2\sqrt{2}\ln(n/\epsilon)$ such that $\alpha_1, \alpha_2 \geq \frac{1}{2+\epsilon}$.

(c) (10 pts) **Averaging with noise**. Assume keys are orthogonal and $\|k_i\| = 1 \; \forall i \neq 1$. The first key is random: $k_1 = c \cdot u$ where $\|u\| = 1$ and $c \in \{1/2, 3/2\}$ uniformly. Describe α_1, α_2 for $q = \ln(n/\epsilon)(u + k_2)$.

(d) (10 pts) **Multi-headed attention**. Using two heads with queries q_1, q_2, show how to obtain an output close to $(v_1 + v_2)/2$ in the noisy setting of (c).

Solution:

(a) **Copying a distinct value (Hard Attention)**.

Intuition: In ML terms, "copying a distinct value" is equivalent to Hard Attention or simple retrieval. We want the probability distribution α to approximate an 'argmax' function, concentrating all mass on index 1. To achieve this with a Softmax, we need to make the logit (score) for index 1 significantly larger than all other logits. This is often done by scaling the query vector, which acts like lowering the temperature of the Softmax, making the distribution "peaky."

Step 1: Geometric setup. We know the keys are normalized vectors on the unit sphere. The condition $\|k_1 - k_i\| \geq 1$ implies that k_1 is sufficiently far from other keys in vector space. We convert distance to similarity (dot product):

$$\|k_1 - k_i\|^2 = \underbrace{\|k_1\|^2}_{1} + \underbrace{\|k_i\|^2}_{1} - 2\langle k_1, k_i \rangle = 2 - 2\langle k_1, k_i \rangle.$$

Since $\|k_1 - k_i\|^2 \geq 1$:

$$2 - 2\langle k_1, k_i \rangle \geq 1 \implies 2\langle k_1, k_i \rangle \leq 1 \implies \boxed{\langle k_1, k_i \rangle \leq 0.5}.$$

This means the cosine similarity between the target key and distractors is low.

Step 2: Designing the Query. To pick k_1, we set the query q in the direction of k_1. We scale it by a large factor t to widen the gap between scores. Let $q = tk_1$ where $t = 2\ln(n/\epsilon)$.

- Target score (logit): $s_1 = \langle k_1, tk_1 \rangle = t\|k_1\|^2 = t$.

- Distractor scores: $s_i = \langle k_i, tk_1 \rangle = t\langle k_i, k_1 \rangle \leq 0.5t$.

The **logit gap** is $\Delta s = s_1 - s_i \geq 0.5t$.

Step 3: Bounding the Softmax.

$$\alpha_1 = \frac{e^{s_1}}{e^{s_1} + \sum_{i \neq 1} e^{s_i}} = \frac{1}{1 + \sum_{i \neq 1} e^{-(s_1 - s_i)}}.$$

Substituting the gap $0.5t$:

$$\sum_{i \neq 1} e^{-(s_1 - s_i)} \leq (n-1)e^{-0.5t}.$$

With $t = 2\ln(n/\epsilon)$, we have $e^{-0.5t} = e^{-\ln(n/\epsilon)} = \frac{\epsilon}{n}$.

$$\text{Error term} \leq (n-1)\frac{\epsilon}{n} < \epsilon.$$

Thus, $\alpha_1 \geq \frac{1}{1+\epsilon}$. The model successfully retrieves v_1.

(b) **Averaging two values (Soft Attention)**.

Intuition: Here we want **Soft Attention**. The goal is to attend equally to two items. Geometrically, the query vector q must be the exact bisector of the angle between key 1 and key 2.

Step 1: Designing the Query. We choose q to be the sum of the two target keys: $q = t(k_1 + k_2)$. Because keys are orthogonal (perpendicular), the norm is:

$$\|k_1 + k_2\|^2 = \|k_1\|^2 + \|k_2\|^2 = 1 + 1 = 2 \implies \|q\| = t\sqrt{2}.$$

Setting $t = \ln(n/\epsilon)$ keeps the norm within the required bound.

Step 2: Calculating Logits.

- $s_1 = \langle k_1, t(k_1 + k_2) \rangle = t(1 + 0) = t$.
- $s_2 = \langle k_2, t(k_1 + k_2) \rangle = t(0 + 1) = t$.
- $s_i = 0$ for $i > 2$ (due to orthogonality).

Step 3: Resulting Weights. The denominator partition function Z is dominated by the two large exponents:

$$\alpha_1 = \frac{e^t}{e^t + e^t + (n-2)e^0} = \frac{1}{2 + (n-2)e^{-t}}.$$

Using $e^{-t} = \epsilon/n$, the noise term $(n-2)\epsilon/n$ is negligible ($< \epsilon$).

$$\alpha_1 \approx \frac{1}{2}.$$

This achieves perfect averaging.

(c) **Averaging with noise (Single Head Failure)**.

Intuition: This part demonstrates the brittleness of single-head attention. The key k_1 has a random magnitude scaling factor c. Because the dot product is linear, this scaling factor c scales the logit directly. However, Softmax is exponential. A linear change in logits results in a multiplicative explosion in probabilities. The attention mechanism will "collapse" to whichever key happens to be slightly larger due to noise.

Step 1: Computing Logits with Noise. We use the bisecting query $q = t(u + k_2)$.

- $s_1 = \langle k_1, q \rangle = \langle cu, t(u + k_2) \rangle = ct$.
- $s_2 = \langle k_2, q \rangle = \langle k_2, t(u + k_2) \rangle = t$.

The difference in logits is $(c - 1)t$.

Step 2: The Ratio of Probabilities.

$$\frac{\alpha_1}{\alpha_2} = \frac{e^{ct}}{e^t} = e^{(c-1)t}.$$

Since $t = \ln(n/\epsilon)$ is a large number:

- **Case** $c = 1.5$ **(Signal boosting):** $c - 1 = 0.5$. The ratio is $e^{0.5 \ln(n/\epsilon)} = \sqrt{n/\epsilon}$. For large n, $\alpha_1 \gg \alpha_2$. The model ignores v_2.
- **Case** $c = 0.5$ **(Signal fading):** $c - 1 = -0.5$. The ratio is $e^{-0.5 \ln(n/\epsilon)} = \sqrt{\epsilon/n}$. $\alpha_1 \ll \alpha_2$. The model ignores v_1.

In both cases, the output is not an average. It is a random selection of either v_1 or v_2.

(d) **Multi-headed attention (Ensembling)**.

Intuition: This justifies why Transformers use Multi-Head Attention. By using two independent heads, we can create separate "subspaces." Head 1 can learn to be robust to the magnitude noise of key 1, while Head 2 focuses cleanly on key 2. Averaging the outputs of the heads (linear projection) acts like an ensemble, smoothing out the noise.

Step 1: Head 1 Configuration. We set Query 1 to focus only on the direction u (ignoring k_2). $q_1 = tu$.

$$s_1^{(1)} = \langle cu, tu \rangle = ct, \quad s_2^{(1)} = 0.$$

Even in the worst case where noise $c = 0.5$, the score is $0.5t$. Since t is very large, $e^{0.5t}$ is still massively larger than the noise from other keys. $\implies \alpha_1^{(1)} \approx 1$. Head 1 reliably retrieves v_1.

Step 2: Head 2 Configuration. We set Query 2 to focus only on k_2. $q_2 = tk_2$.

$$s_1^{(2)} = 0, \quad s_2^{(2)} = t.$$

$\implies \alpha_2^{(2)} \approx 1$. Head 2 reliably retrieves v_2.

Step 3: Final Output. The layer combines the heads:

$$\text{Output} = \frac{O_1 + O_2}{2} \approx \frac{v_1 + v_2}{2}.$$

The multi-head mechanism successfully averages the values despite the noise in k_1.

Position encoding, 30 pts

Consider input X, transformed to Q, K, V, attention output O, and feed-forward output Z.

(a) (25 pts) Show that if input is permuted $X^{perm} = PX$, output is $Z^{perm} = PZ$.

(b) (5 pts) Show that sinusoidal position encodings Φ have distinct rows for $t \neq t'$.

Solution:

(a) **Permutation Equivariance**.

Intuition: We are proving that the Transformer (without position encoding) is Permutation Equivariant. This means it treats the input as a "Bag of Words" (a set) rather than a sequence. If you shuffle the input words, the output vectors get shuffled in the exact same way, but their values don't change based on where they sit in the sentence. This is why we *need* position embeddings (part b)—otherwise, the model can't tell the difference between "The dog bit the man" and "The man bit the dog."

Proof Steps: 1. **Linear Layers are Equivariant:** Matrix multiplication acts on each row independently. If you shuffle rows of X (PX), the rows of $Q = XW_Q$ get shuffled exactly the same way (PQ).

2. **Attention Scores ($A \propto QK^T$):**

$$A' \propto (PQ)(PK)^T = P(QK^T)P^T = PAP^T.$$

The matrix PAP^T represents permuting both the rows and the columns of the adjacency matrix. The relationship between token i and token j is preserved, just moved to a new location in the matrix.

3. **Softmax Row-Independence:** Since softmax is applied per-row, permuting the order of rows (P on the left) and the order of columns (P^T on the right) results in:

$$\text{softmax}(PAP^T) = P\,\text{softmax}(A)P^T.$$

4. **Output:**

$$Z^{perm} = PZ.$$

This confirms that the geometry of the data is preserved, just re-ordered.

(b) **Uniqueness of Position Encodings**.

Intuition: Since the network is permutation equivariant (as proved above), we must "stamp" every token with a unique time signature. The sinusoidal encoding ensures uniqueness using different frequencies.

Proof: We only need to look at the first two dimensions ($i = 0$).

$$\Phi_{t,0} = \sin(t), \quad \Phi_{t,1} = \cos(t).$$

If two positions t and t' had identical encodings, they would have the same sine and cosine values. This can only happen if they differ by a full rotation:

$$t - t' = 2\pi k \quad \text{for some integer } k.$$

However, positions t, t' are integers. Their difference must be an integer. The number 2π is irrational. The only way an integer can equal an irrational multiple is if the multiple is zero ($k = 0$). Therefore, $t = t'$. The encoding is unique.

SVD, 30 pts

Let $A \in \mathbb{R}^{n \times d}$ ($n \geq d$) have singular values $\sigma_1 \geq \cdots \geq \sigma_d$.

(a) (12 pts) Show $\sum_{i=1}^{d} \sigma_i^2 = \|A\|_F^2$.

(b) (6 pts) Show $\sigma_k^2 \leq \|A\|_F^2 / k$.

(c) (12 pts) Show there exists rank $k-1$ matrix B such that $\|A - B\|_2 \leq \|A\|_F / \sqrt{k}$.

Solution:

(a) **Energy Conservation (Trace Identity)**.

Intuition: The squared Frobenius norm $\|A\|_F^2$ represents the total "energy" or information content of the matrix (sum of all squared elements). The Singular Value Decomposition (SVD) rotates the matrix into a coordinate system where the energy is concentrated along the diagonal axes (σ_i). This proof shows that the total energy is conserved during this rotation.

Proof: Using the definition $\|A\|_F^2 = \text{trace}(A^\top A)$ and SVD $A = U\Sigma V^\top$:

$$\text{trace}(A^\top A) = \text{trace}(V\Sigma U^\top U\Sigma V^\top) = \text{trace}(V\Sigma^2 V^\top).$$

Using the Cyclic Property of Trace ($\text{trace}(XYZ) = \text{trace}(ZXY)$):

$$\text{trace}(V\Sigma^2 V^\top) = \text{trace}(V^\top V\Sigma^2).$$

Since V is orthogonal ($V^\top V = I$):

$$= \text{trace}(I\Sigma^2) = \sum_{i=1}^{d} \sigma_i^2.$$

(b) **Bounding the k-th Component**.

Intuition: This is a "pigeonhole principle" argument for energy. If you have total energy E, and you sort the components from largest to smallest, the k-th largest component cannot be arbitrarily large. Specifically, it cannot be larger than the average energy of the top k components.

Proof: We split the total sum: $\|A\|_F^2 = \sum_{i=1}^{d} \sigma_i^2 \geq \sum_{i=1}^{k} \sigma_i^2$. Because singular values are sorted descending ($\sigma_1 \geq \sigma_2 \cdots \geq \sigma_k$):

$$\sigma_i^2 \geq \sigma_k^2 \quad \text{for all } i \leq k.$$

Replacing every term in the sum with the smallest one (σ_k^2):

$$\sum_{i=1}^{k} \sigma_i^2 \geq \sum_{i=1}^{k} \sigma_k^2 = k\sigma_k^2.$$

Therefore, $k\sigma_k^2 \leq \|A\|_F^2 \implies \sigma_k^2 \leq \frac{\|A\|_F^2}{k}$.

(c) **Low-Rank Approximation Error**.

Intuition: This relates to Principal Component Analysis (PCA) or compression. If we approximate matrix A by keeping only the top $k-1$ principal components (matrix B), we discard the "tail" of the spectrum starting at σ_k. The error of this approximation is determined by the largest singular value we threw away, which is σ_k.

Proof: We construct B as the truncated SVD of rank $k-1$:

$$B = \sum_{i=1}^{k-1} \sigma_i u_i v_i^\top.$$

The residual error matrix is the sum of the discarded components:

$$A - B = \sum_{i=k}^{d} \sigma_i u_i v_i^\top.$$

The Spectral Norm (operator norm) of a matrix is equal to its largest singular value. For the diagonal matrix $A - B$, the largest value is σ_k.

$$\|A - B\|_2 = \sigma_k.$$

From part (b), we know $\sigma_k \leq \frac{\|A\|_F}{\sqrt{k}}$. Thus, the bound holds.