Scientific Computing in Python / 2027 Lecture 6

Embeddings and scaled dot-product attention

Zan Ahmad

Proposed 2027; calendar dates pending

Embedding and projection matrices

$$Q = XW_Q \in \mathbb{R}^{n \times d_k}, \quad K = XW_K \in \mathbb{R}^{n \times d_k}, \quad V = XW_V \in \mathbb{R}^{n \times d_v}.$$

Embedding and projection matrices

$$Q = XW_Q \in \mathbb{R}^{n \times d_k}, \quad K = XW_K \in \mathbb{R}^{n \times d_k}, \quad V = XW_V \in \mathbb{R}^{n \times d_v}.$$

The score matrix QK^T has shape $n \times n$.

Row and column meanings

$$S_{ij} = \frac{q_i^\top k_j}{\sqrt{d_k}}.$$

Row and column meanings

$$S_{ij} = \frac{q_i^\top k_j}{\sqrt{d_k}}.$$

Row i fixes the query. Column j identifies a key whose associated value may contribute to that query's output.

Rowwise softmax

$$A_{ij} = \frac{e^{S_{ij}}}{\sum_{\ell} e^{S_{i\ell}}}, \qquad A_{ij} > 0, \quad \sum_{j} A_{ij} = 1.$$

Rowwise softmax

$$A_{ij} = \frac{e^{S_{ij}}}{\sum_{\ell} e^{S_{i\ell}}}, \qquad A_{ij} > 0, \quad \sum_{j} A_{ij} = 1.$$

Normalization is performed independently for each row.

Stable normalization

For any row constant c,

$$\frac{e^{s_j - c}}{\sum_\ell e^{s_\ell - c}} = \frac{e^{s_j}}{\sum_\ell e^{s_\ell}}.$$

Stable normalization

For any row constant c,

$$\frac{e^{s_j - c}}{\sum_\ell e^{s_\ell - c}} = \frac{e^{s_j}}{\sum_\ell e^{s_\ell}}.$$

Choose $c = \max_j s_j$ to avoid exponentiating large positive numbers.

Weighted values

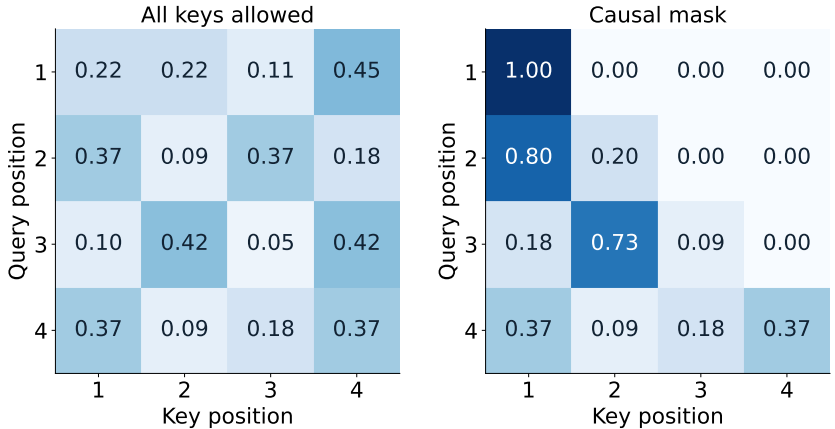
$$Z = AV, \qquad z_i = \sum_j A_{ij} v_j.$$

Weighted values

$$Z = AV, \qquad z_i = \sum_j A_{ij} v_j.$$

If every value is the same vector v_*, then $z_i = v_*$ for every row.

Attention weights



Scaling the values

With Q, K and the allowed-key set fixed,

$$A(\alpha V + \beta W) = \alpha AV + \beta AW.$$

Scaling the values

With Q, K and the allowed-key set fixed,

$$A(\alpha V + \beta W) = \alpha AV + \beta AW.$$

Doubling V doubles Z. Changing Q generally changes the weights themselves.

A two-key calculation

$$s = (1/\sqrt{2}, 0), \qquad A = \left(\frac{e^{1/\sqrt{2}}}{e^{1/\sqrt{2}} + 1}, \frac{1}{e^{1/\sqrt{2}} + 1} \right).$$

A two-key calculation

$$s = (1/\sqrt{2}, 0), \qquad A = \left(\frac{e^{1/\sqrt{2}}}{e^{1/\sqrt{2}} + 1}, \frac{1}{e^{1/\sqrt{2}} + 1} \right).$$

$$A \approx (0.66976, 0.33024).$$

The output vector

$$z = 0.66976(2, 0) + 0.33024(0, 4) \approx (1.33952, 1.32096).$$

The output vector

$$z = 0.66976(2, 0) + 0.33024(0, 4) \approx (1.33952, 1.32096).$$

The two output coordinates are value coordinates. They need not sum to one and are not token probabilities.

Causal masking

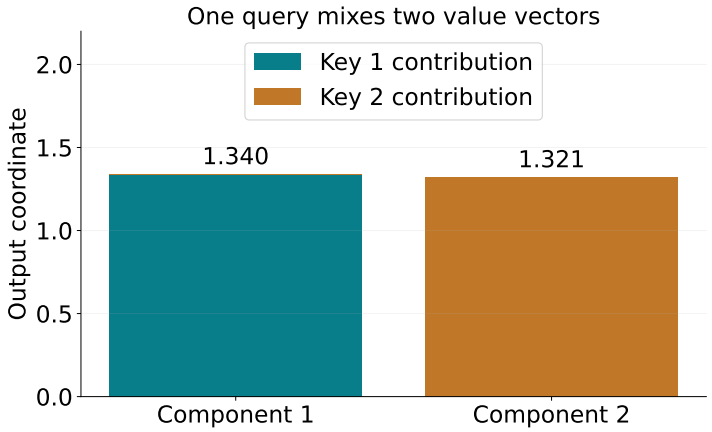
$$A_{ij} = 0 \quad (j > i), \qquad \sum_{j \le i} A_{ij} = 1.$$

Causal masking

$$A_{ij} = 0 \quad (j > i), \qquad \sum_{j \le i} A_{ij} = 1.$$

The first position has only one allowed key, hence $A_{11} = 1$.

Weighted values



A causal perturbation test

Change only the final embedding row while holding projection matrices fixed.

A causal perturbation test

Change only the final embedding row while holding projection matrices fixed. Earlier causal outputs remain unchanged. Earlier unmasked outputs can change because the final key and value are available to them.

Scaling and learned representations

Under independent, zero-mean, unit-variance coordinates,

$$\operatorname{Var}\left(\sum_{r=1}^{d_k} q_r k_r\right) = d_k.$$

Scaling and learned representations

Under independent, zero-mean, unit-variance coordinates,

$$\operatorname{Var}\left(\sum_{r=1}^{d_k} q_r k_r\right) = d_k.$$

Dividing by $\sqrt{d_k}$ keeps the variance at one.

Permutation and position

Without positional information, permuting input rows by P gives

$$Q' = PQ,\ K' = PK,\ V' = PV \quad \Longrightarrow \quad Z' = PZ$$

for unmasked attention.

Permutation and position

Without positional information, permuting input rows by P gives

$$Q' = PQ,\ K' = PK,\ V' = PV \quad \Longrightarrow \quad Z' = PZ$$

for unmasked attention. Position encodings or structured masks provide information not present in an unordered collection of embedding rows.