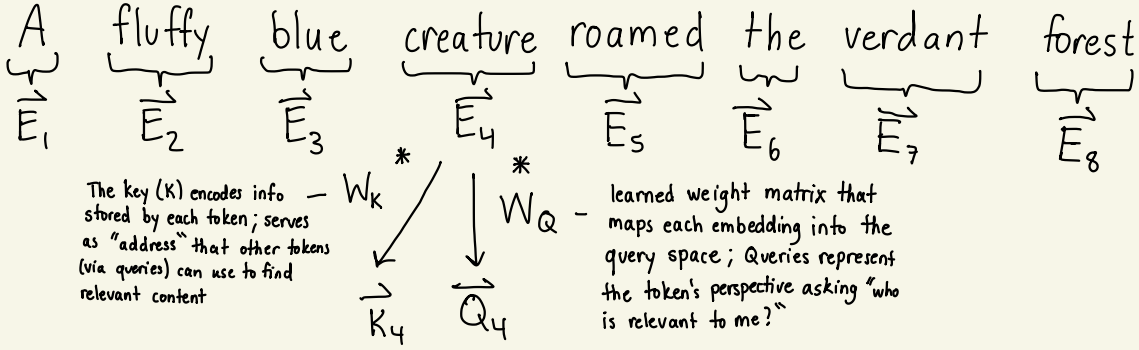
Attention Mechanism

Motivation: Words in the English language have multiple meanings, yet are represented as one embedding. The attention mechanism looks at the context of the rest of the sequence to modify its embedding to accurately represent the true meaning of the word.

High-level example



The key (K) encodes info stored by each token; serves as "address" that other tokens (via queries) can use to find relevant content

learned weight matrix that maps each embedding into the query space; Queries represent the token's perspective asking "who is relevant to me?"

Once Q and K is calculated for every token, the dot product of Q and K is taken to measure the similarity between the current token (Q) and every other token (K) to determine how much "attention" the current token should pay to every other token. A softmax is then applied to normalize all values in a column (each token) to add up to 1 (weight each key based off of relevance). For numerical stability, each dot product is divided by the sq. root of the dimensions of key-query space ($\sqrt{d_k}$). Finally, we use the soft max'd values to determine how much each Value (V) vector should contribute to the final output. V is there to provide the actual content that the attention mechanism will return, based on attention scores

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Multi-head Attention

Motivation: There are multiple ways in which context can affect the meaning of a word. Therefore, there are multiple attention heads for each different contextual meaning; a distinct key-query pair.