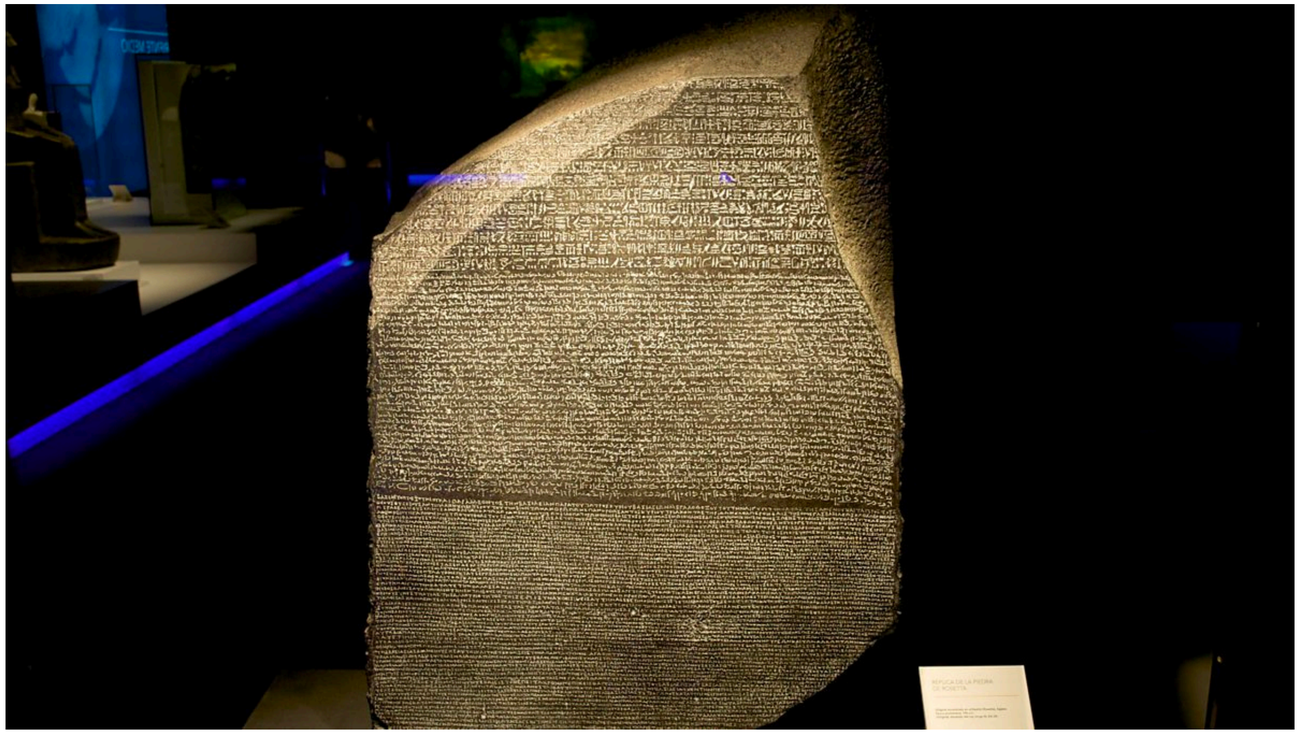
Attention Mechanism

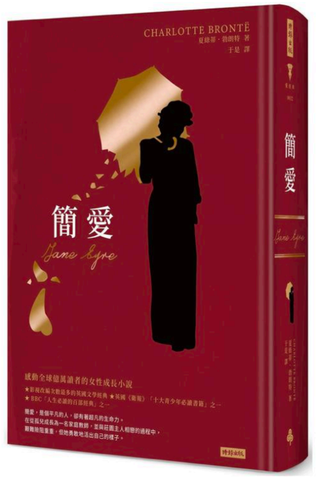
COMP3361 — Week 4

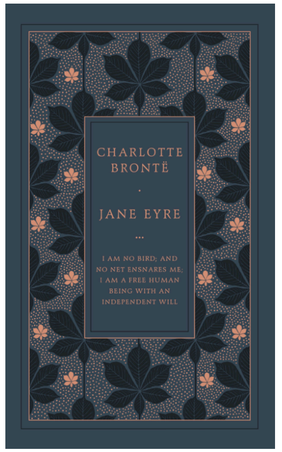
Lingpeng Kong

Department of Computer Science, The University of Hong Kong

Alignment in Machine Translation





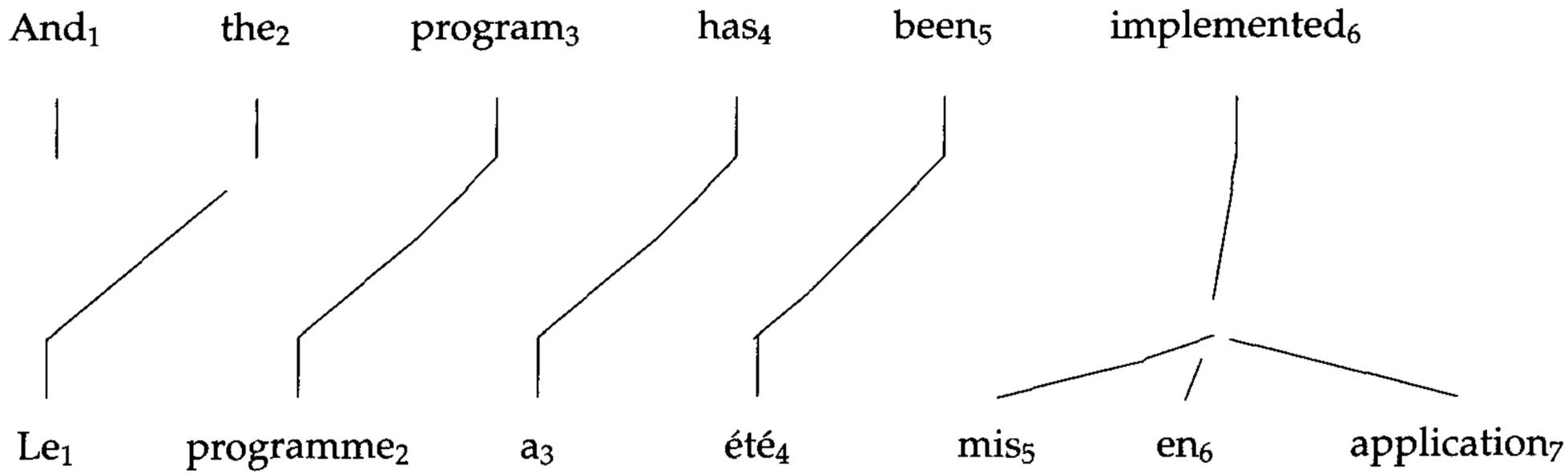


Parallel Corpus

$$p(\boldsymbol{y} \mid \boldsymbol{x})$$

target source

Alignment in Machine Translation

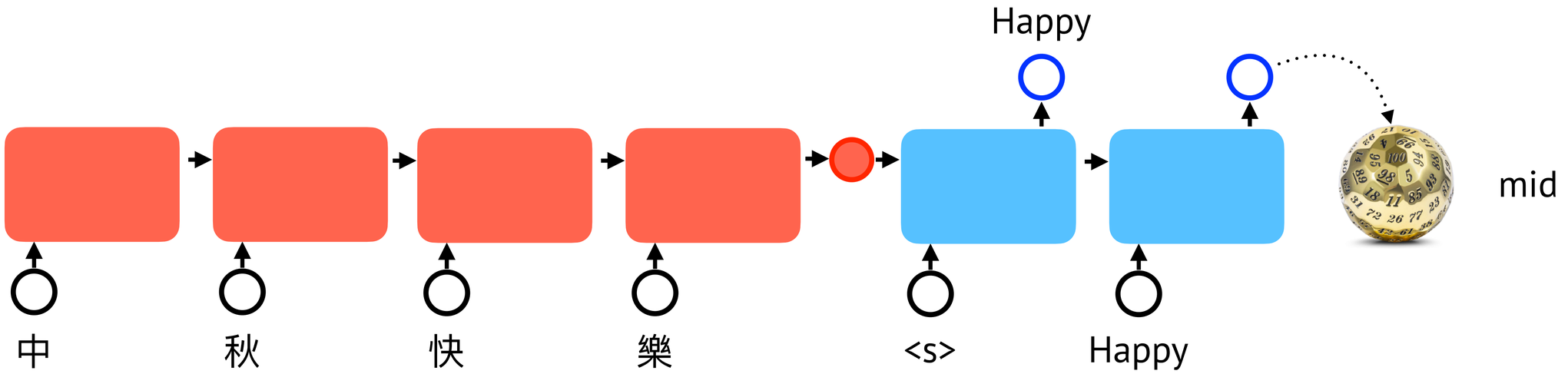


Some words might have no “counter-part”.

Alignment can be many-to-one (or one-to-many).

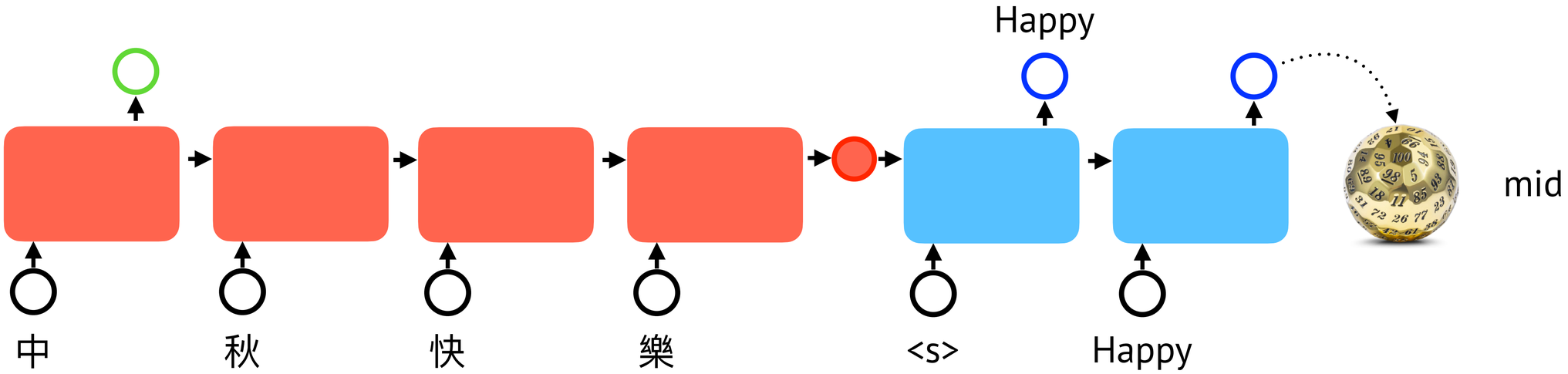
(Brown et al, 1993)

Sequence to Sequence Model

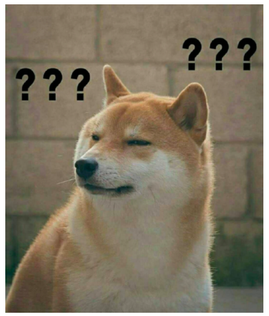


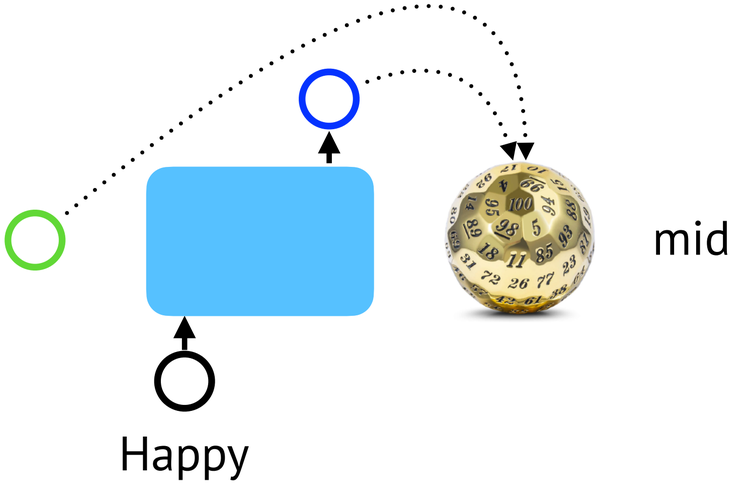
$p(y_t \mid \boldsymbol{y}_{<t}, \boldsymbol{x})$

Sequence to Sequence Model



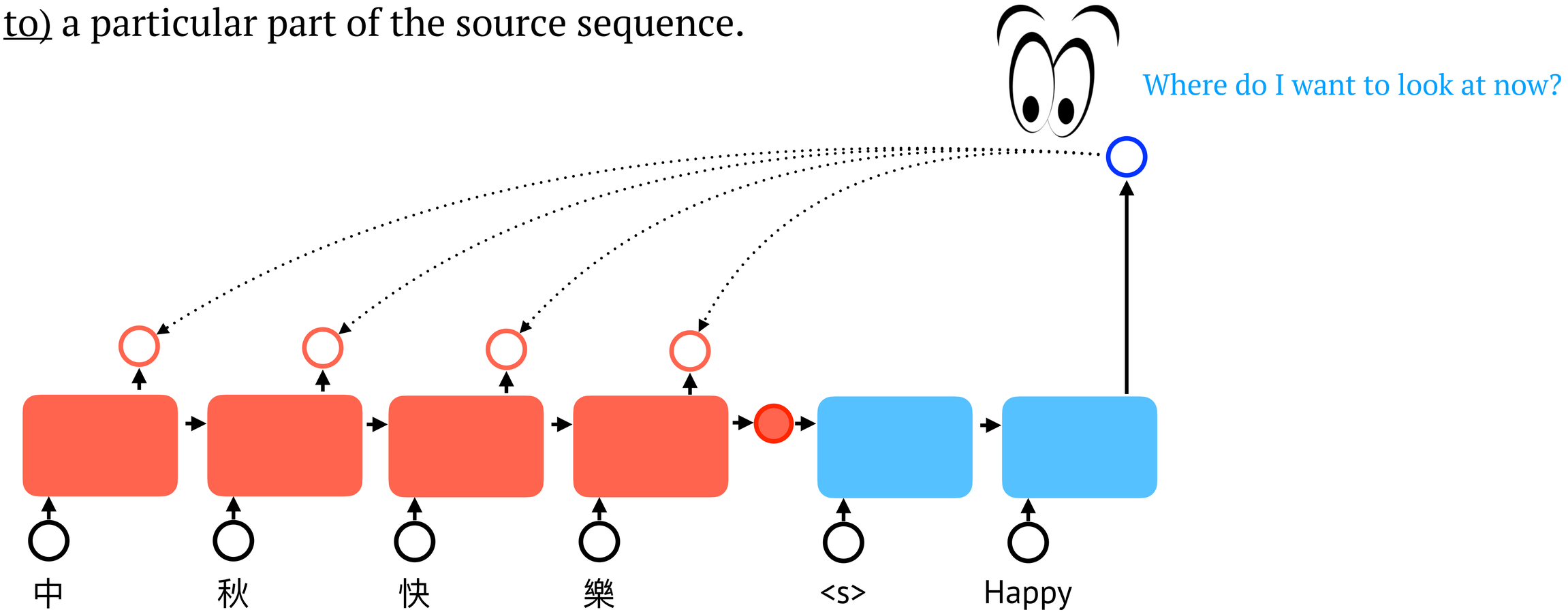
$p(y_t \mid \boldsymbol{y}_{<t}, \boldsymbol{x})$





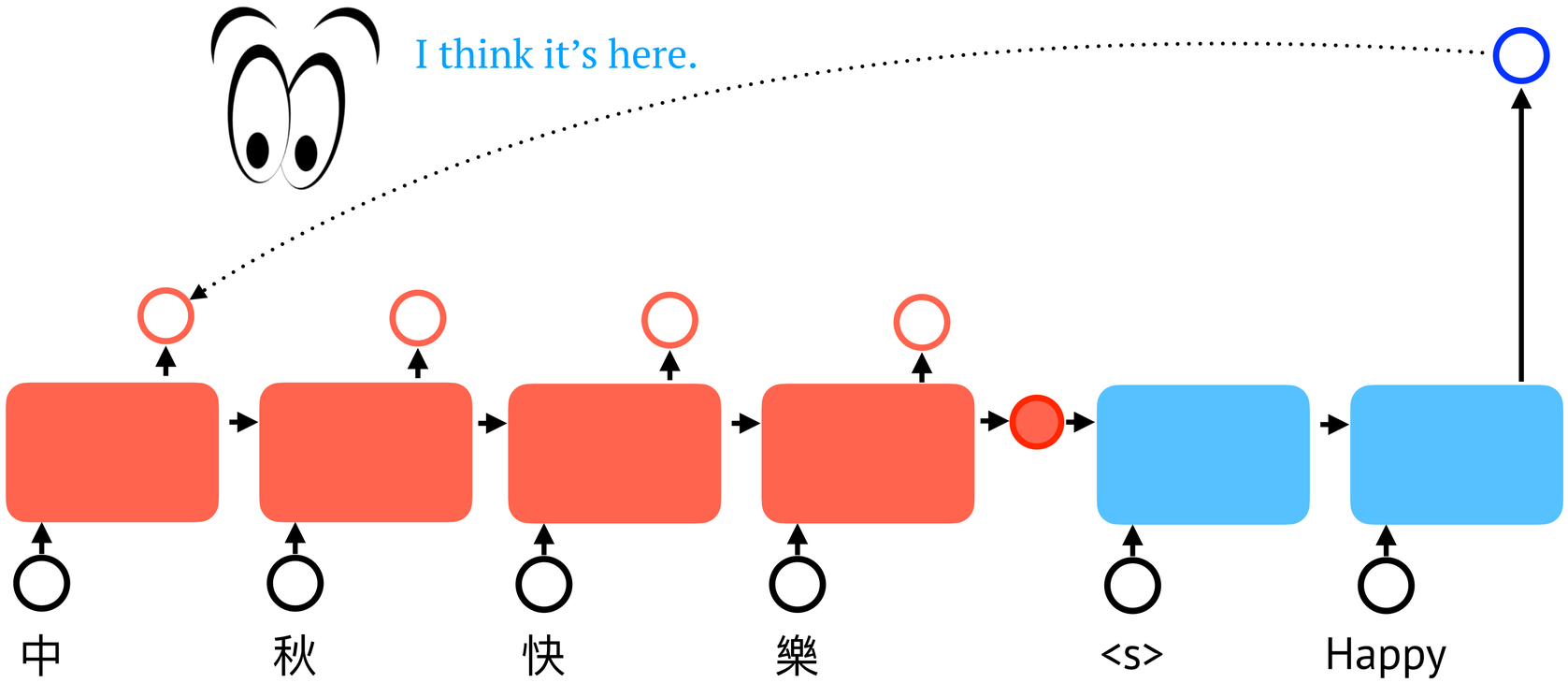
Attention Mechanism

Use direct connection to the encoder to focus on (attend to) a particular part of the source sequence.



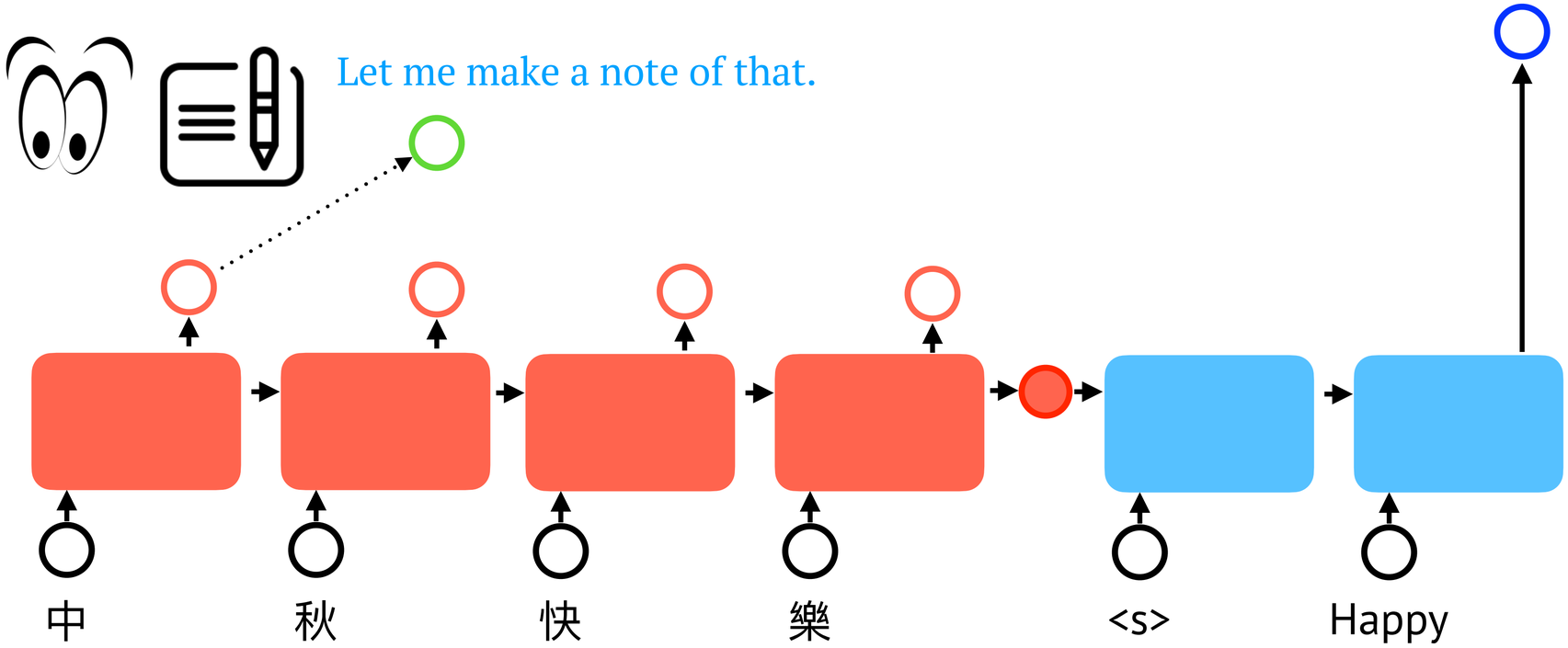
Attention Mechanism

Use direct connection to the encoder to focus on (attend to) a particular part of the source sequence.



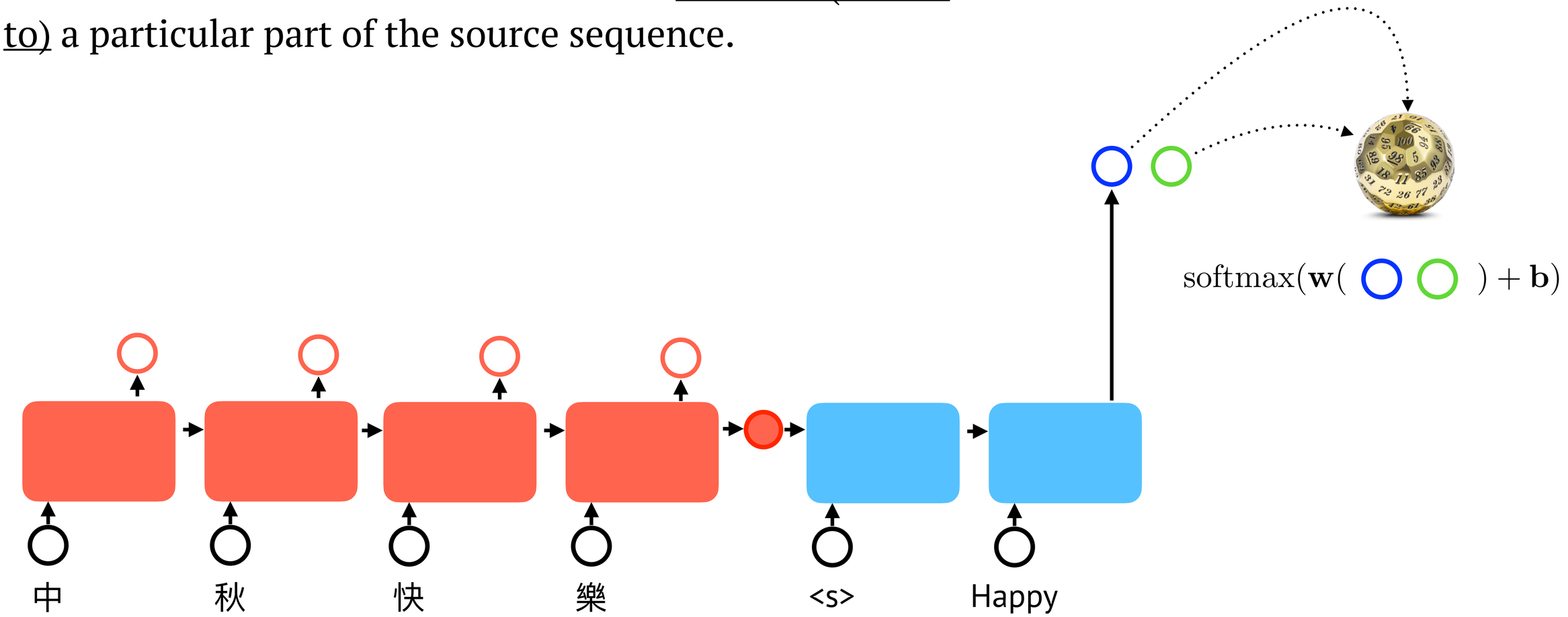
Attention Mechanism

Use direct connection to the encoder to focus on (attend to) a particular part of the source sequence.

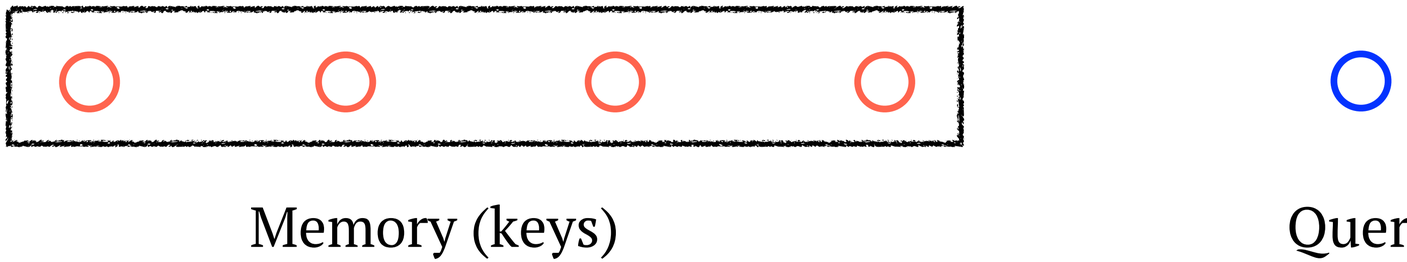


Attention Mechanism

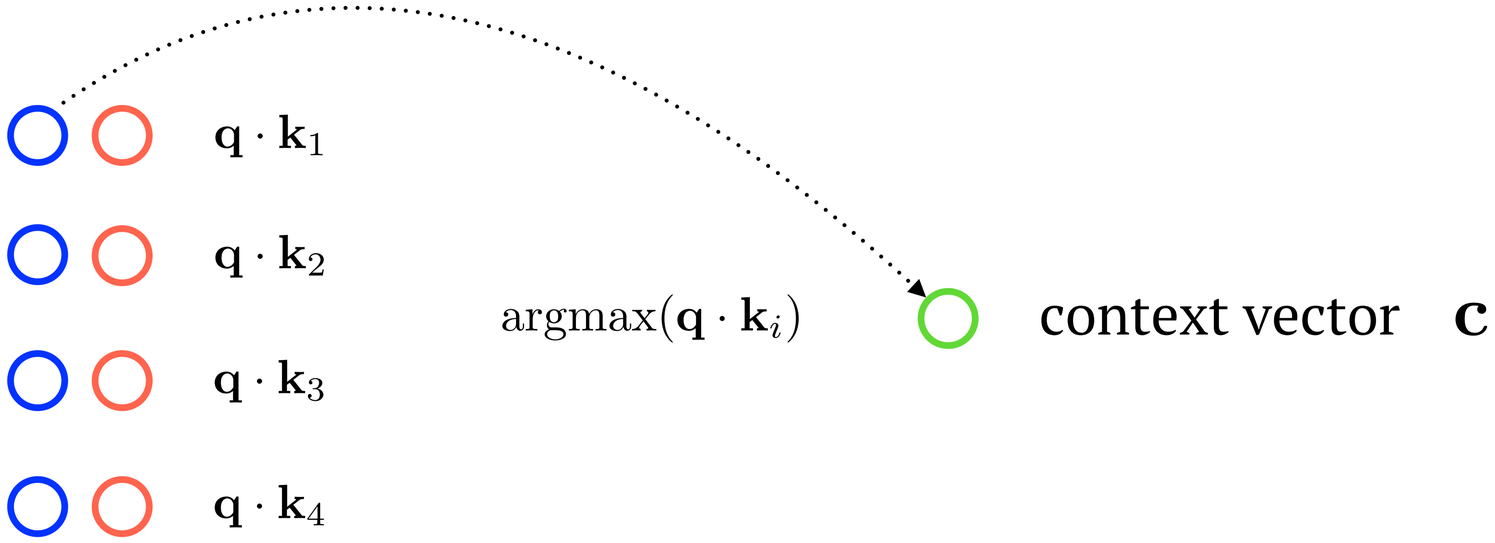
Use direct connection to the encoder to focus on (attend to) a particular part of the source sequence.



Memory Abstraction



Task: Finding the most "relevant" item in the memory.

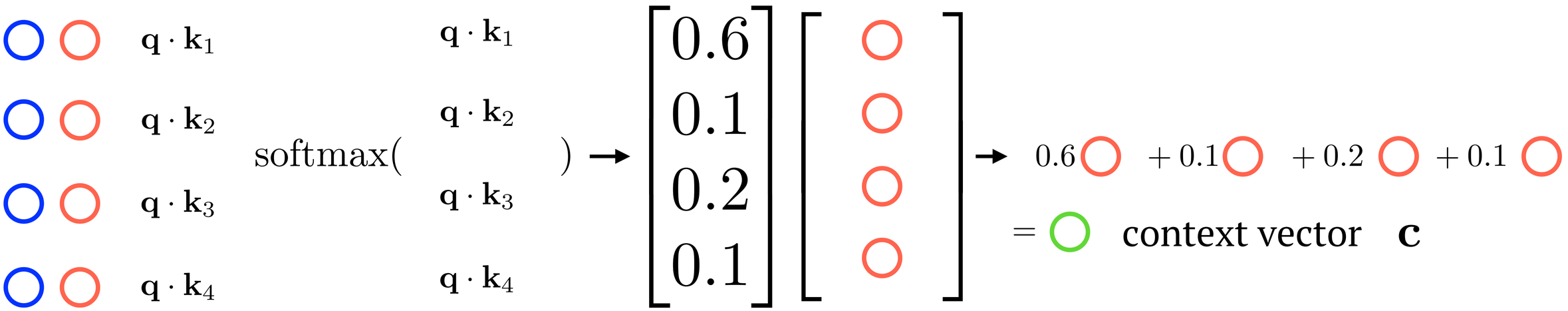


Dot-Product-Softmax Attention

Memory (keys)

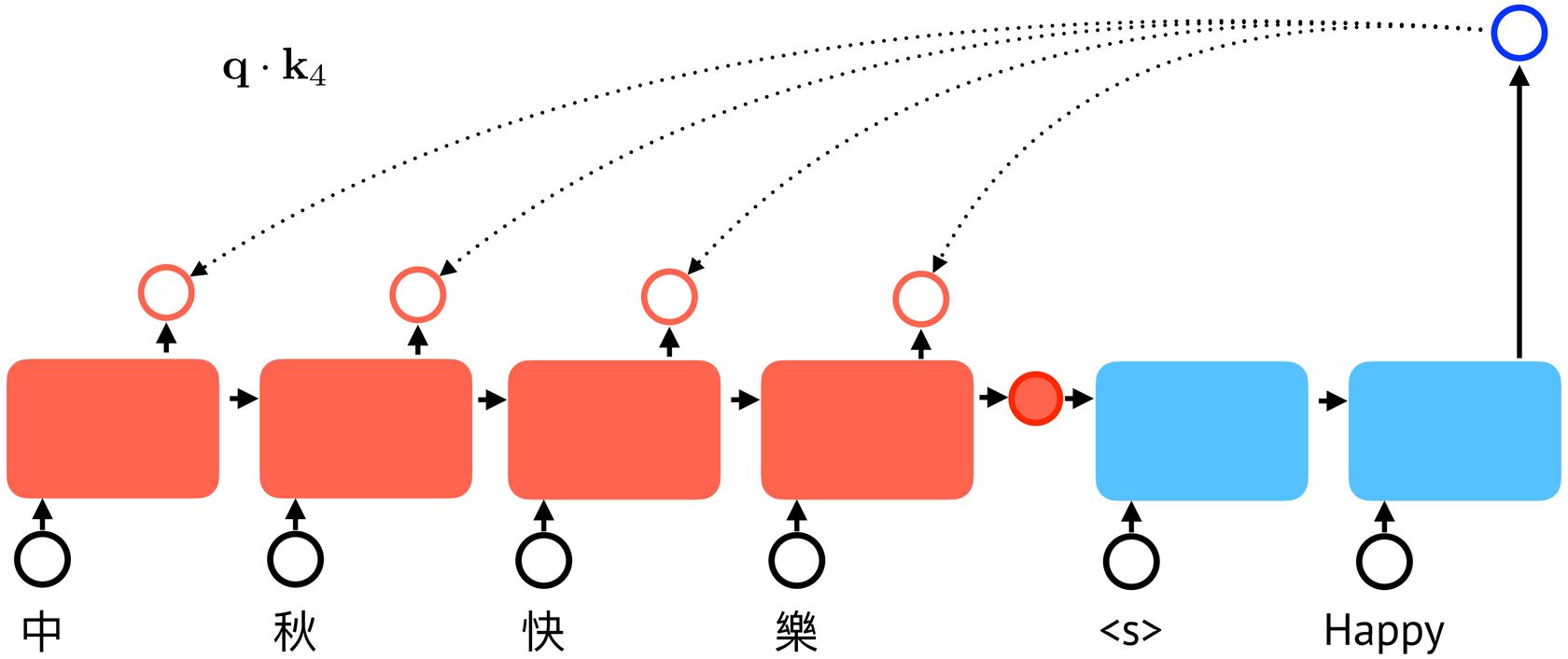
Query

Task: Finding the most "relevant" item in the memory.

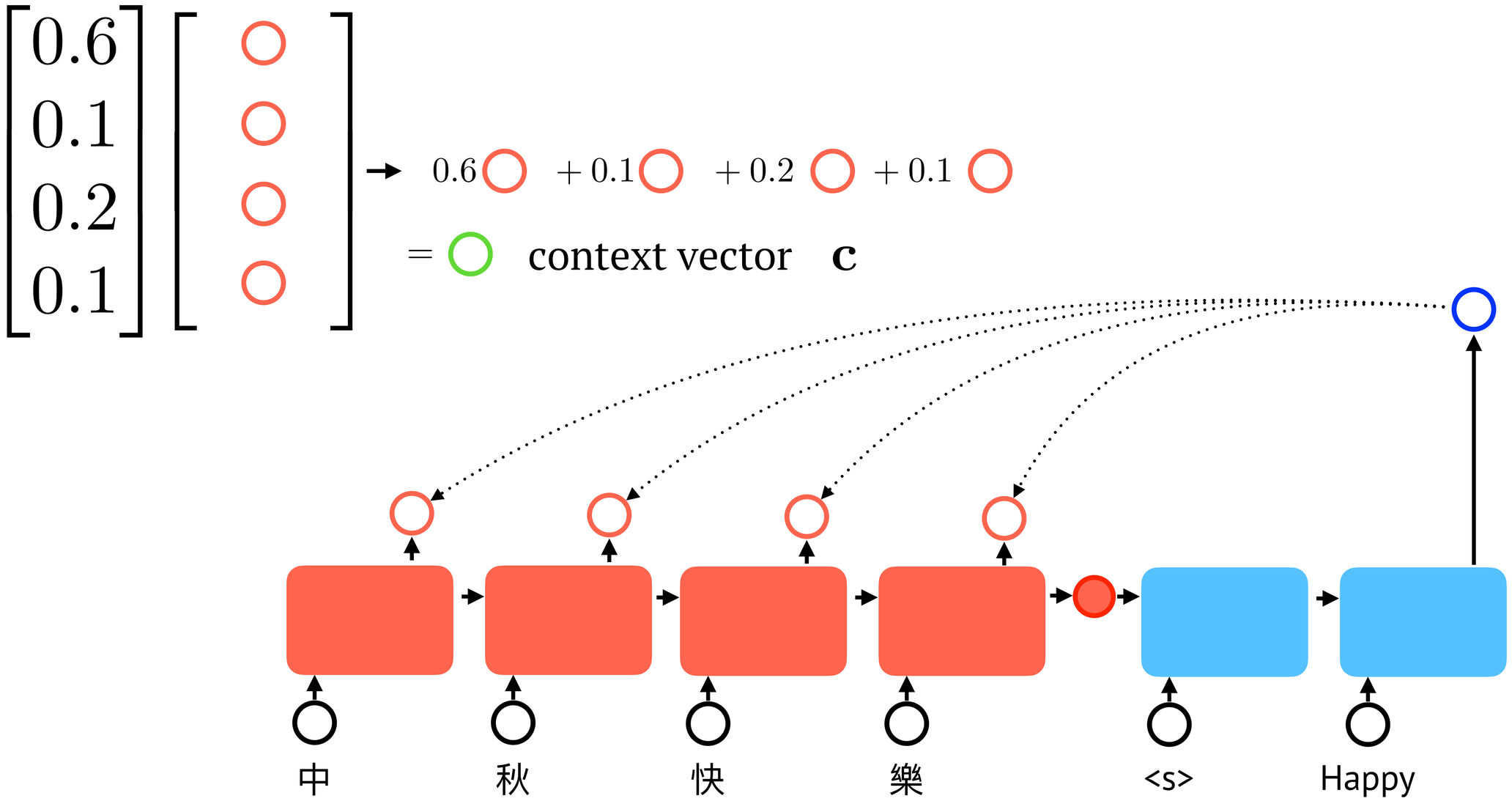


Attention Mechanism

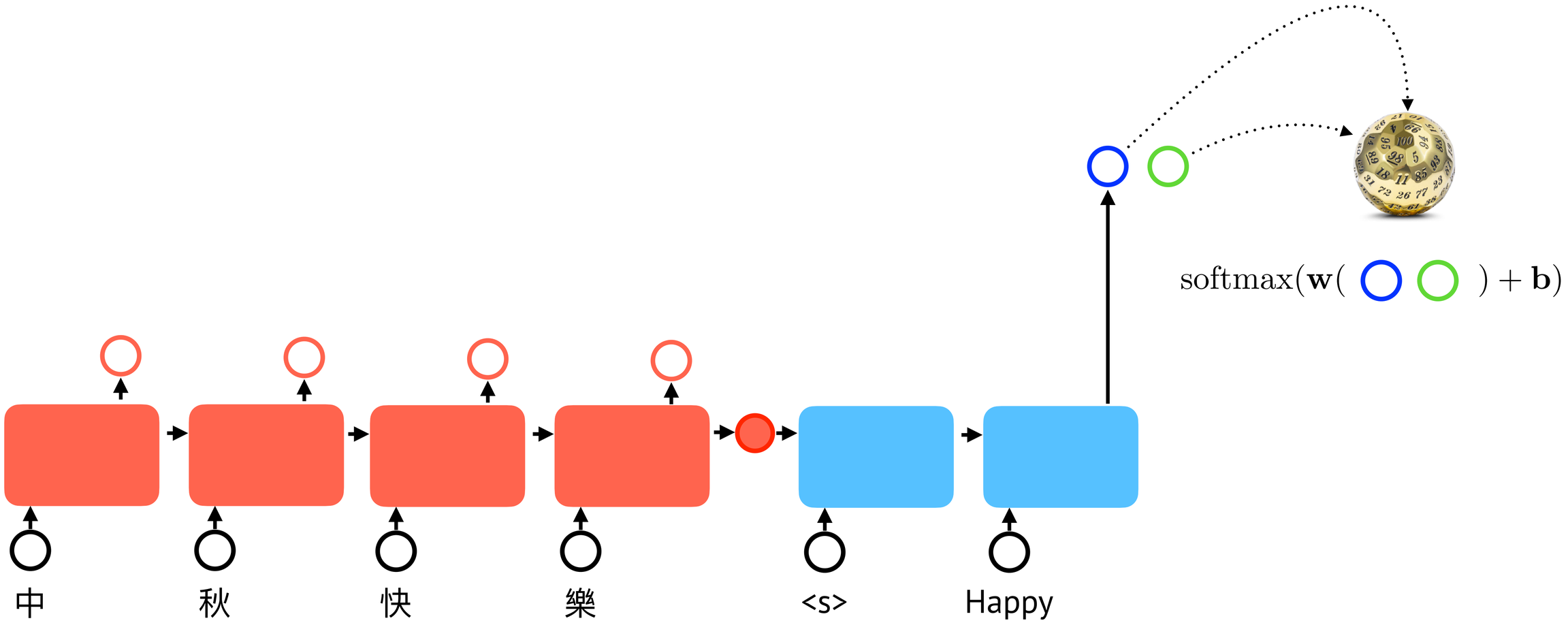
$$\text{softmax}\begin{pmatrix} \mathbf{q}\cdot\mathbf{k}_1 \\ \mathbf{q}\cdot\mathbf{k}_2 \\ \mathbf{q}\cdot\mathbf{k}_3 \\ \mathbf{q}\cdot\mathbf{k}_4 \end{pmatrix}$$



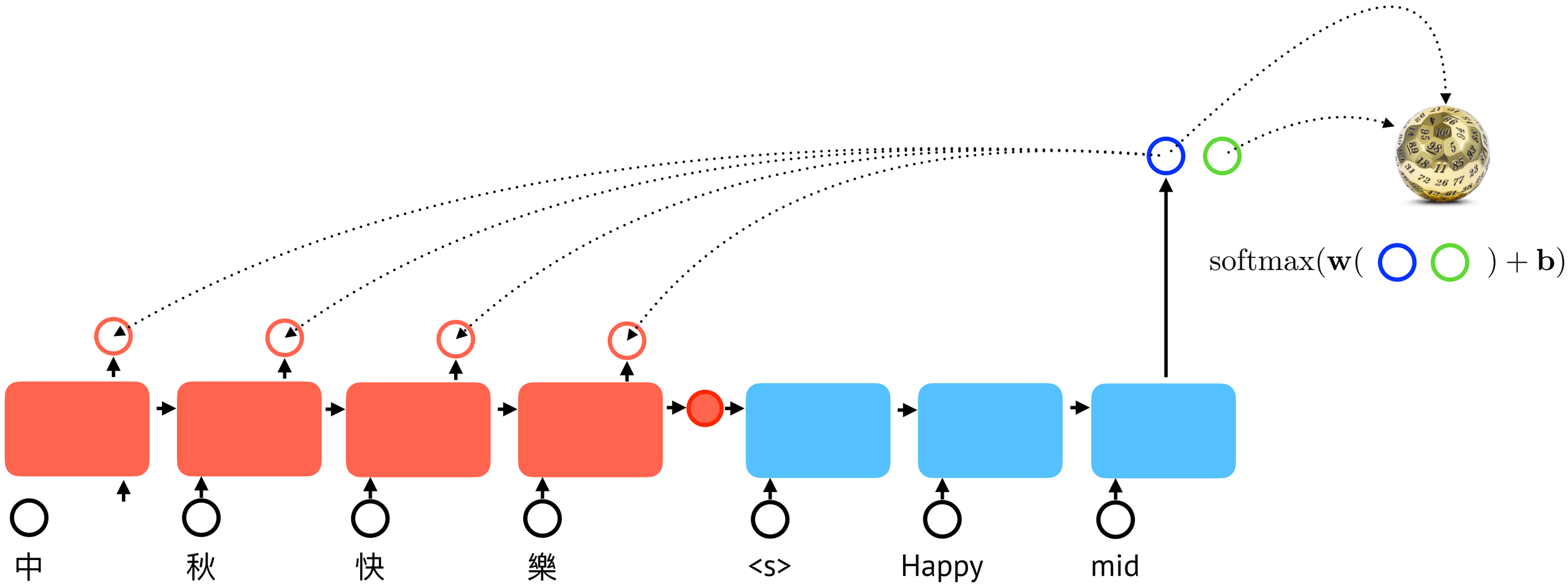
Attention Mechanism



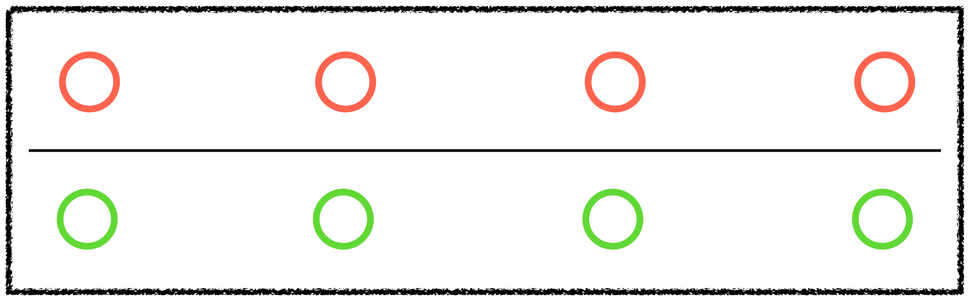
Attention Mechanism



Attention Mechanism



Dot-Product-Softmax Attention



Memory (key-value pairs)

Query

$\mathbf{q} \cdot \mathbf{k}_1$

$\mathbf{q} \cdot \mathbf{k}_2$

$\mathbf{q} \cdot \mathbf{k}_3$

$\mathbf{q} \cdot \mathbf{k}_4$

$$\text{softmax}\left(\begin{matrix}\mathbf{q}\cdot\mathbf{k}_1\\ \mathbf{q}\cdot\mathbf{k}_2\\ \mathbf{q}\cdot\mathbf{k}_3\\ \mathbf{q}\cdot\mathbf{k}_4\end{matrix}\right) \rightarrow \begin{bmatrix}0.6\\0.1\\0.2\\0.1\end{bmatrix}$$

$0.6 + 0.1 + 0.2 + 0.1 =$ context vector $\mathbf{c}$

Dot-Product-Softmax Attention

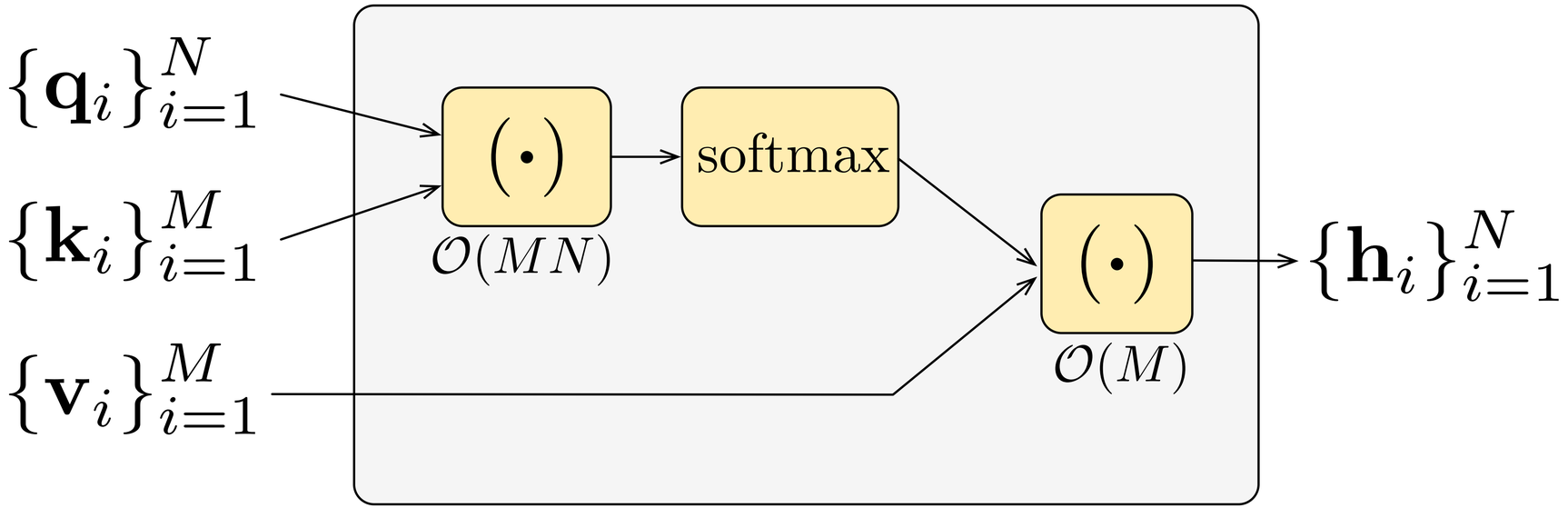
$$\sum_{m=1}^{M} \frac{\exp\left(\boldsymbol{q}_n \boldsymbol{k}_m\right)}{\sum_{m'=1}^{M} \exp\left(\boldsymbol{q}_n \boldsymbol{k}_{m'}\right)} \boldsymbol{v}_m^\top = \mathbf{V}^\top \mathrm{softmax}(\mathbf{K}\boldsymbol{q}_n)$$

similarity

normalized similarity

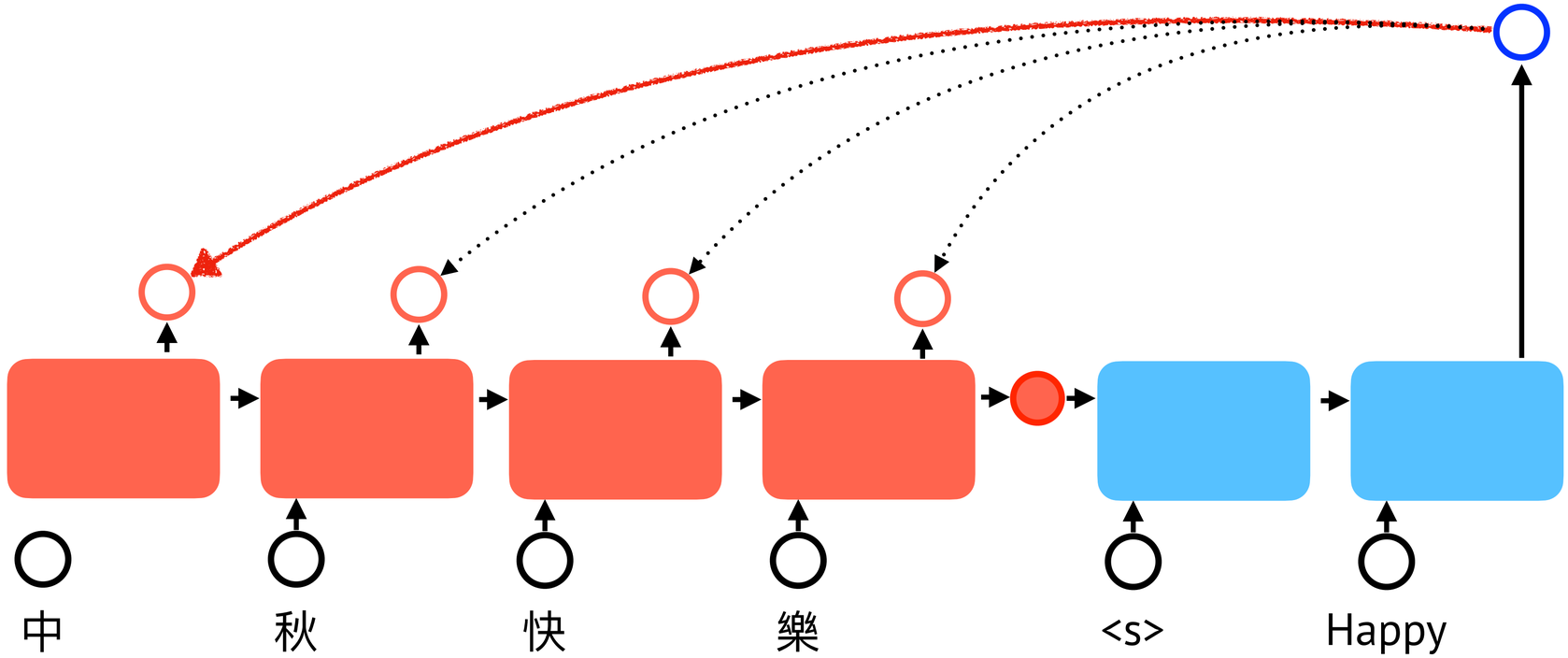
weighted sum

Computational Complexity



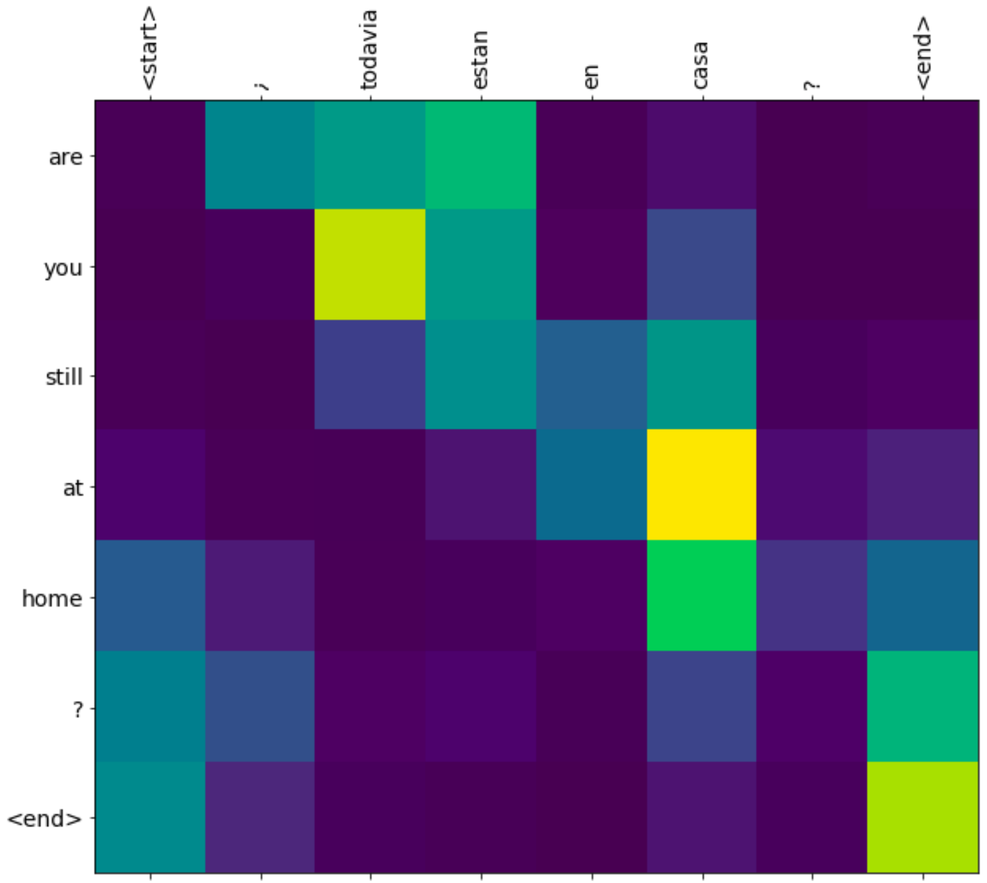
Attention Mechanism

It helps with vanishing gradient problem.



Attention Mechanism

It offers some interpretability.



https://www.tensorflow.org/text/tutorials/nmt_with_attention