Transformers (self-attention)

CS 685, Spring 2021

Advanced Topics in Natural Language Processing

http://brenocon.com/cs685

https://people.cs.umass.edu/~brenocon/cs685_s21/

Brendan O’Connor

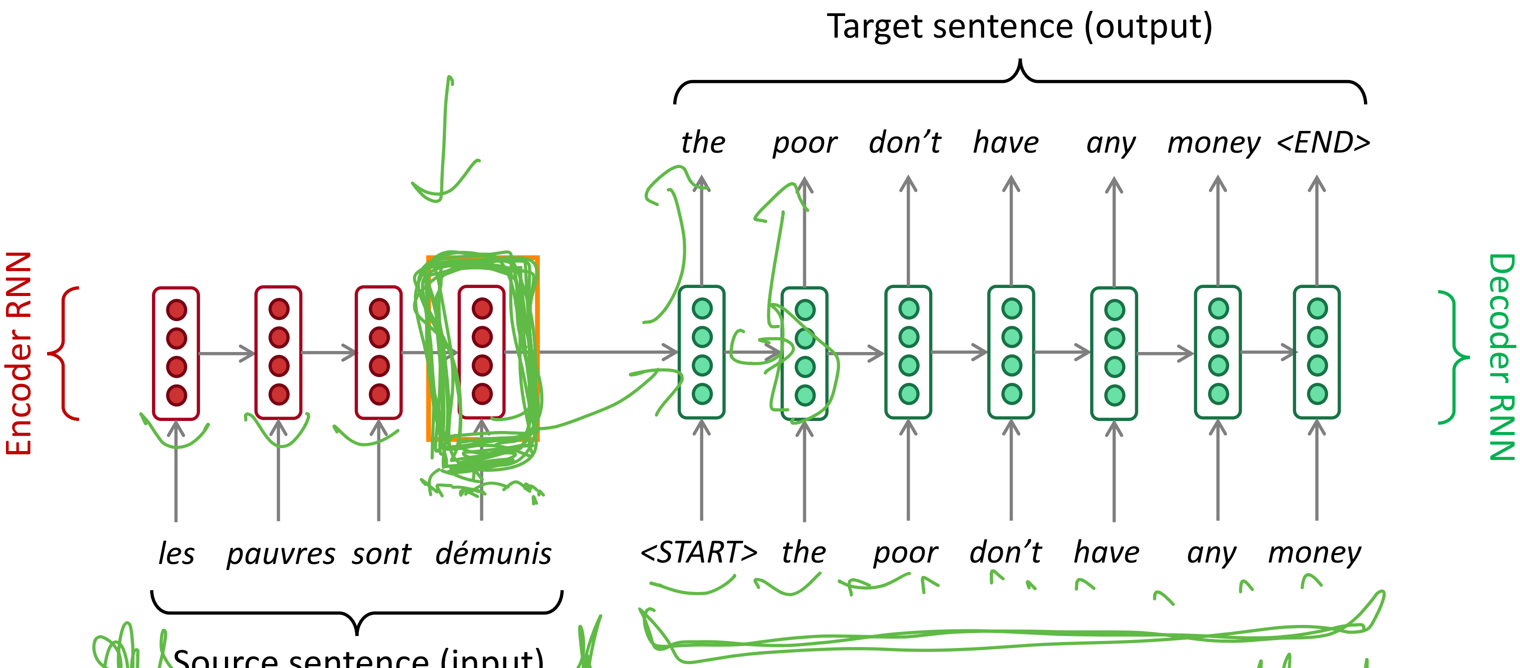
College of Information and Computer Sciences

University of Massachusetts Amherst

- Recurrent neural network: get state for a word position from neighboring position
- Attention: compose a new state by *choosing* previous states (words) to combine
- Major neural architecture for language
  - Originally: *cross-attention* for machine translation (seq2seq)
  - Self-attention ("Transformer"): for a single sentence
- Neural LMs based on a Transformer architecture
  - BERT (next week)
  - GPT-2/3

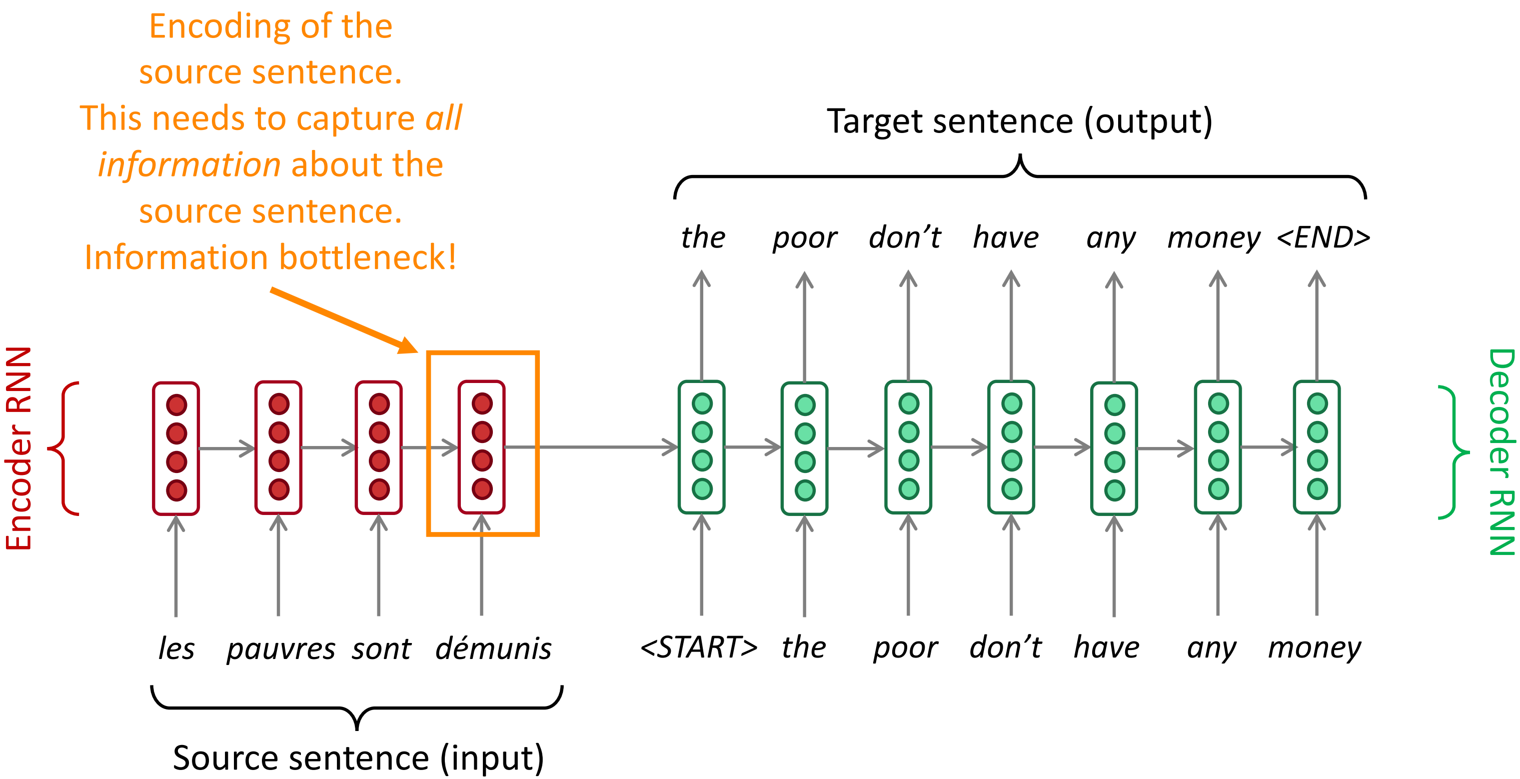
Sequence-to-sequence: the bottleneck problem

Holds all info ??



RNN LM, conditioned on src.

Sequence-to-sequence: the bottleneck problem

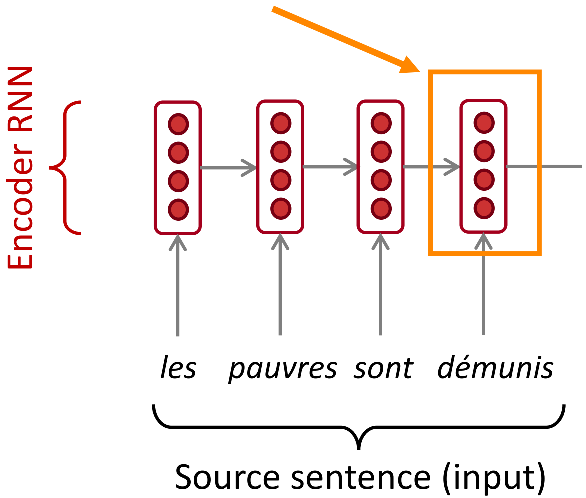


“you can’t cram the meaning of a whole %&@#&ing sentence into a single $*(&@ing vector!”

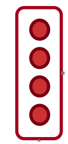
— Ray Mooney (famous NLP professor at UT Austin)

idea: what if we use multiple vectors?

Encoding of the
source sentence.
This needs to capture *all information* about the
source sentence.
Information bottleneck!

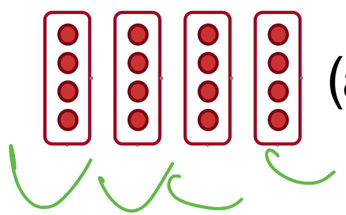


Instead of:

les pauvres sont démunis = 

Let's try:

les pauvres sont démunis =

 (all 4 hidden states!)

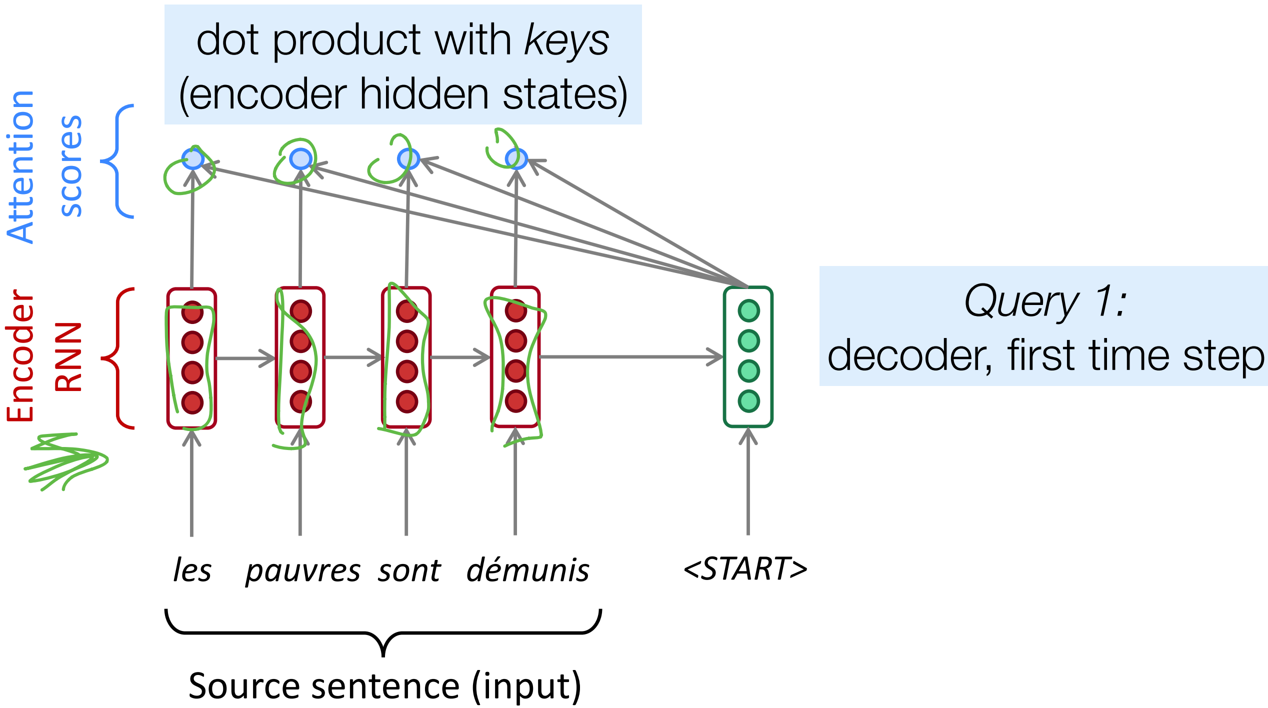
The solution: **attention**

- **Attention mechanisms** (Bahdanau et al., 2015) allow the decoder to focus on a particular part of the source sequence at each time step
  - Conceptually similar to *word alignments*

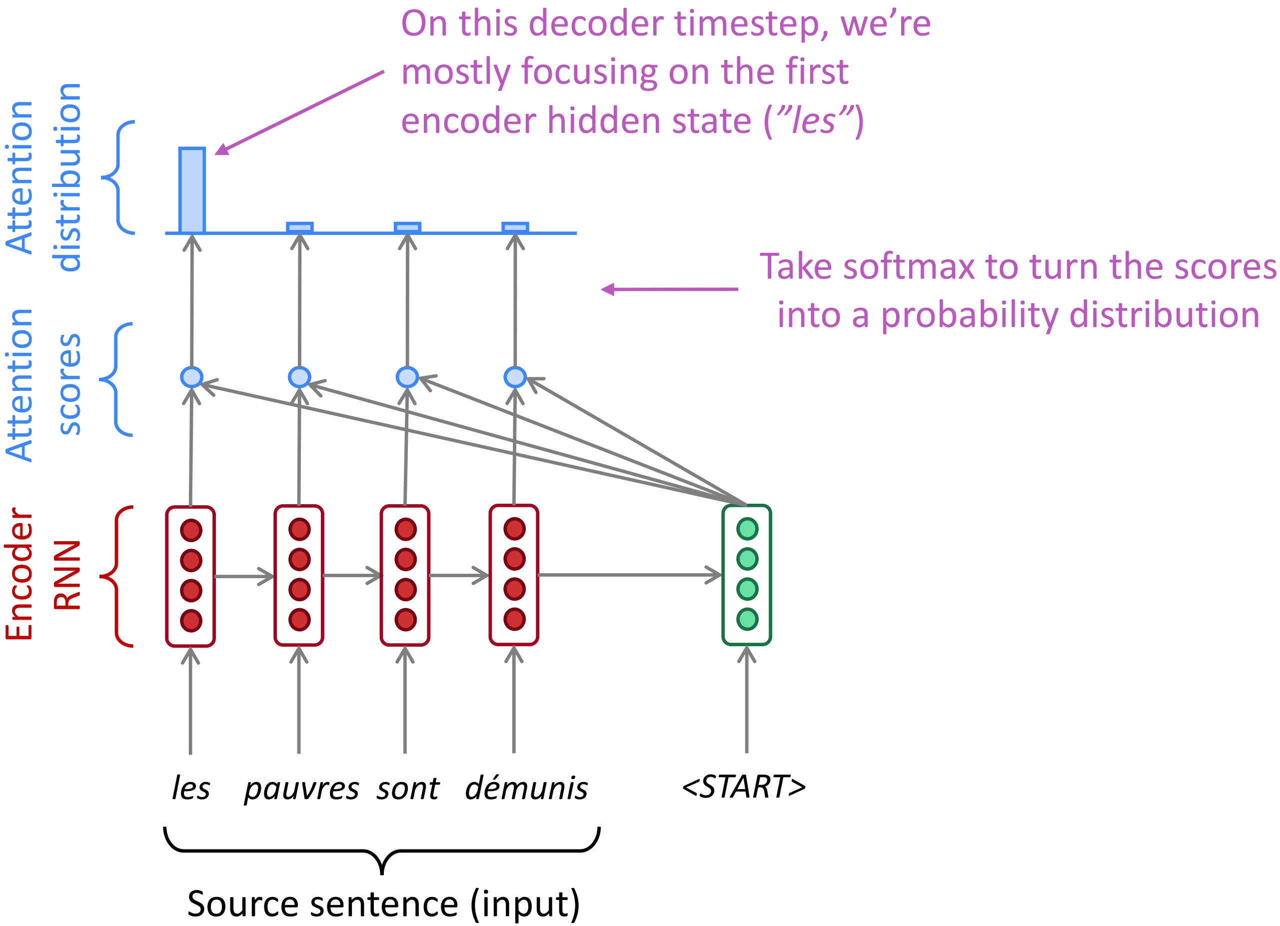
How does it work?

- in general, we have a single *query* vector and multiple *key* vectors. We want to score each query-key pair

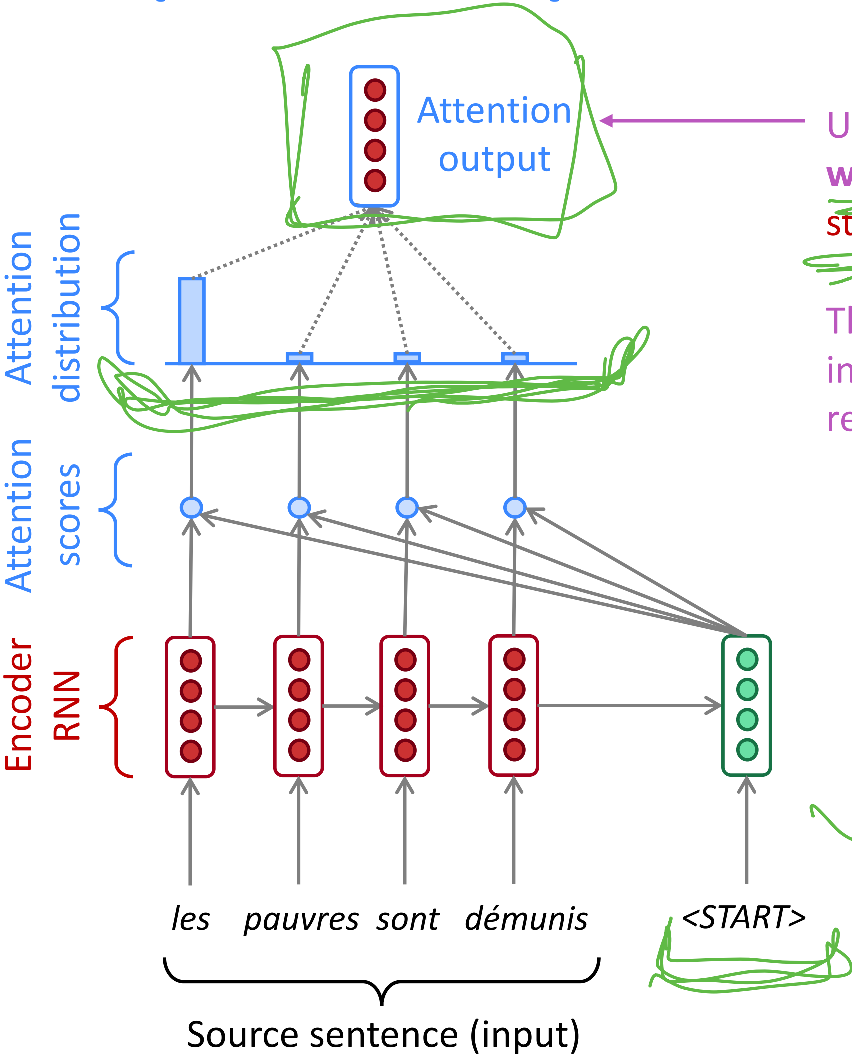
Sequence-to-sequence with attention



Sequence-to-sequence with attention



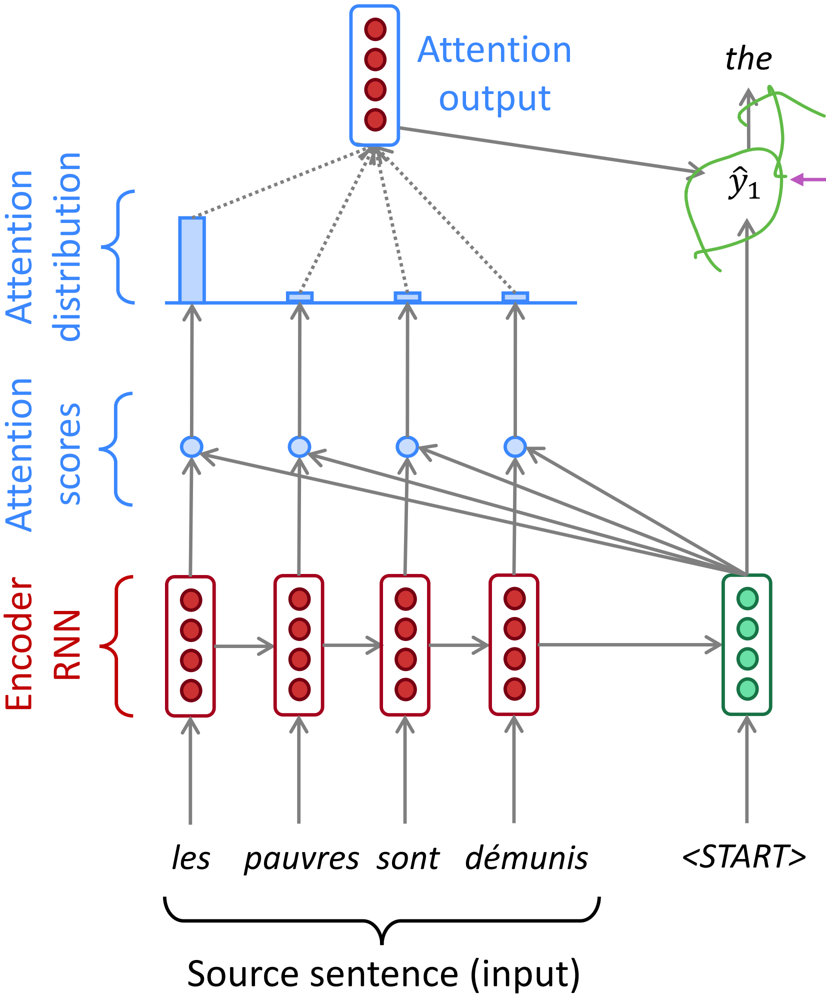
Sequence-to-sequence with attention



Use the attention distribution to take a **weighted sum** of the encoder hidden states.

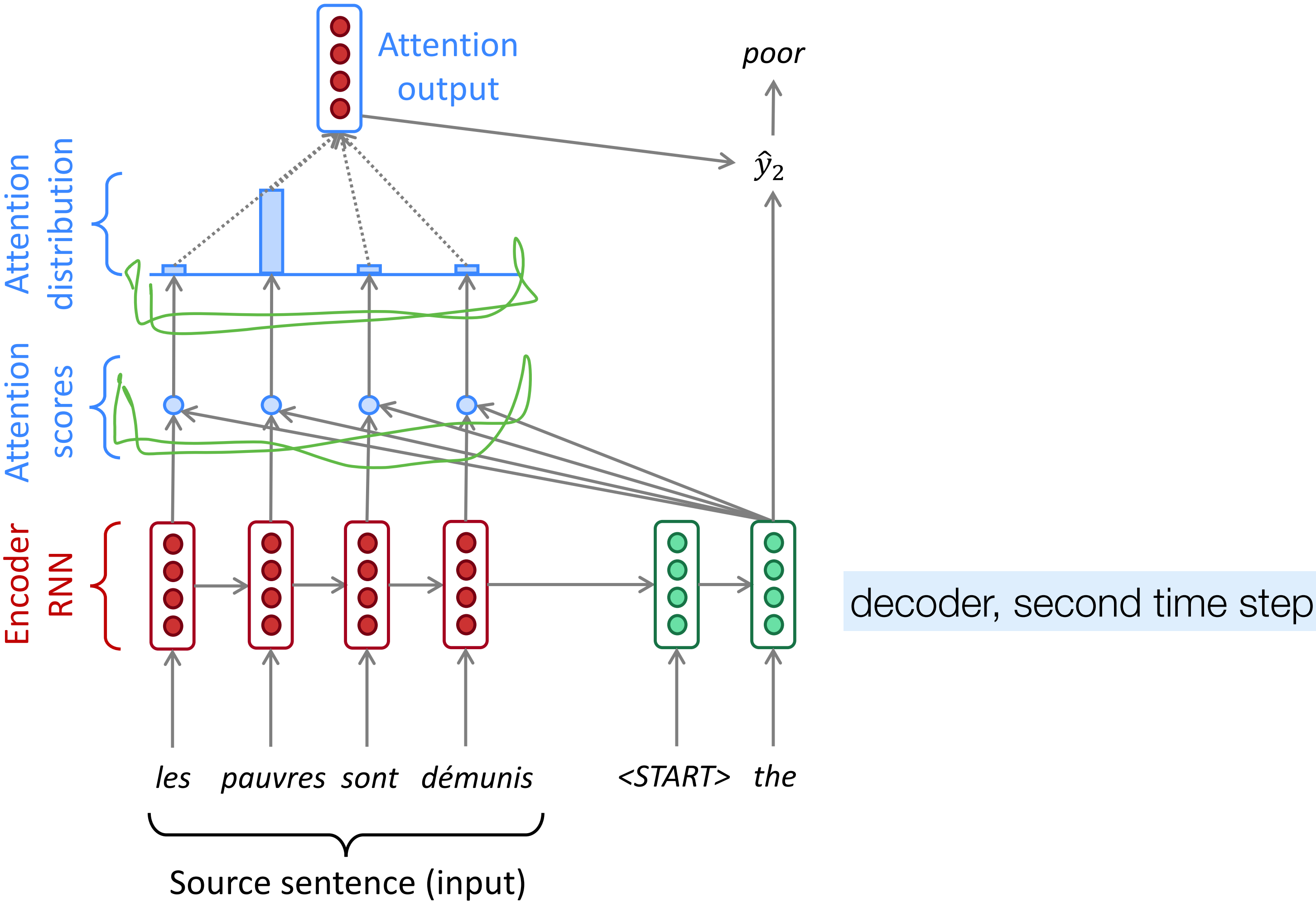
The attention output mostly contains information the hidden states that received high attention.

Sequence-to-sequence with attention



Concatenate attention output with decoder hidden state, then use to compute $\hat{y}_1$ as before

Sequence-to-sequence with attention

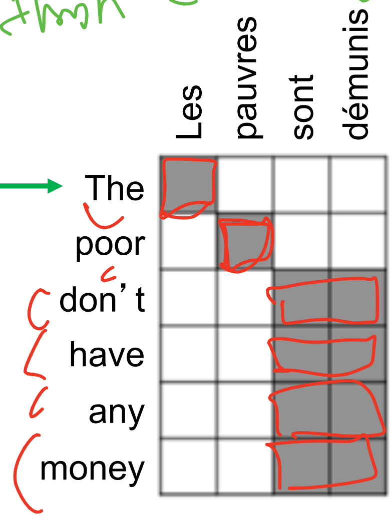


Attention is great

- Attention significantly improves NMT performance
  - It’s very useful to allow decoder to focus on certain parts of the source
- Attention solves the bottleneck problem
  - Attention allows decoder to look directly at source; bypass bottleneck
- Attention helps with vanishing gradient problem
  - Provides shortcut to faraway states

(though controversy)

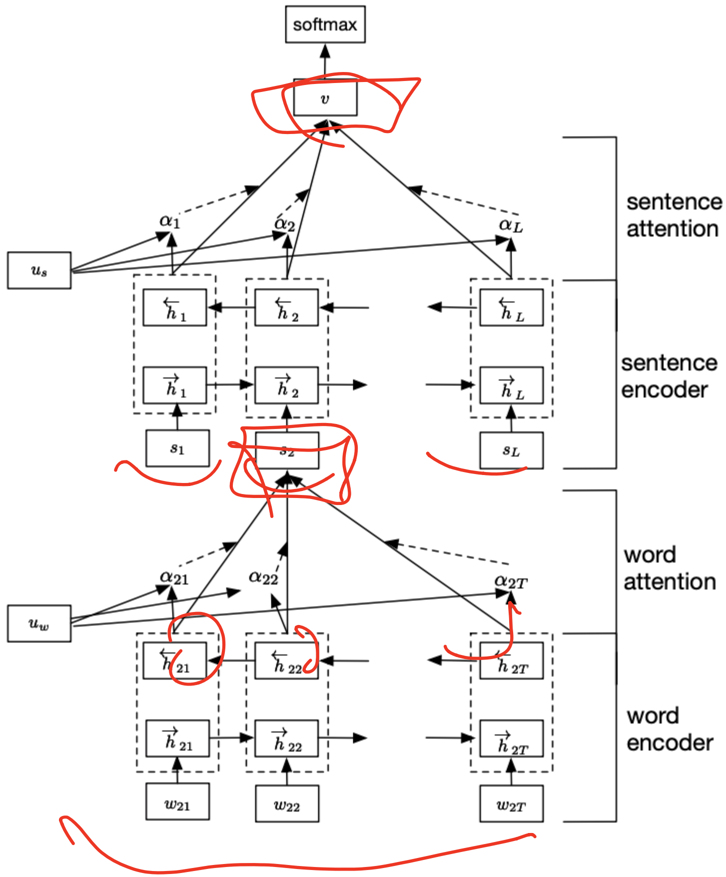
- Attention provides some interpretability
  - By inspecting attention distribution, we can see what the decoder was focusing on
  - We get alignment for free!
  - This is cool because we never explicitly trained an alignment system
  - The network just learned alignment by itself



Many variants of attention

- Original formulation: $a(\mathbf{q}, \mathbf{k}) = w_2^T \tanh(W_1[\mathbf{q}; \mathbf{k}])$
- Bilinear product: $a(\mathbf{q}, \mathbf{k}) = \mathbf{q}^T W \mathbf{k}$ Luong et al., 2015
- Dot product: $a(\mathbf{q}, \mathbf{k}) = \mathbf{q}^T \mathbf{k}$ Luong et al., 2015
- Scaled dot product: $a(\mathbf{q}, \mathbf{k}) = \dfrac{\mathbf{q}^T \mathbf{k}}{\sqrt{|\mathbf{k}|}}$ Vaswani et al., 2017

Hierarchical attention



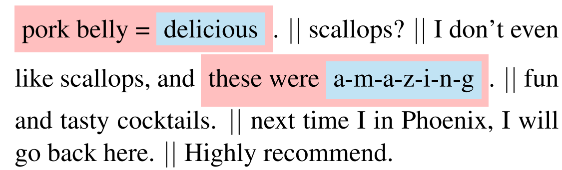


Figure 1: A simple example review from Yelp 2013 that consists of five sentences, delimited by period, question mark. The first and third sentence delivers stronger meaning and inside, the word *delicious, a-m-a-z-i-n-g* contributes the most in defining sentiment of the two sentences.

- Yang et al., 2016: hierarchical attention for document classification

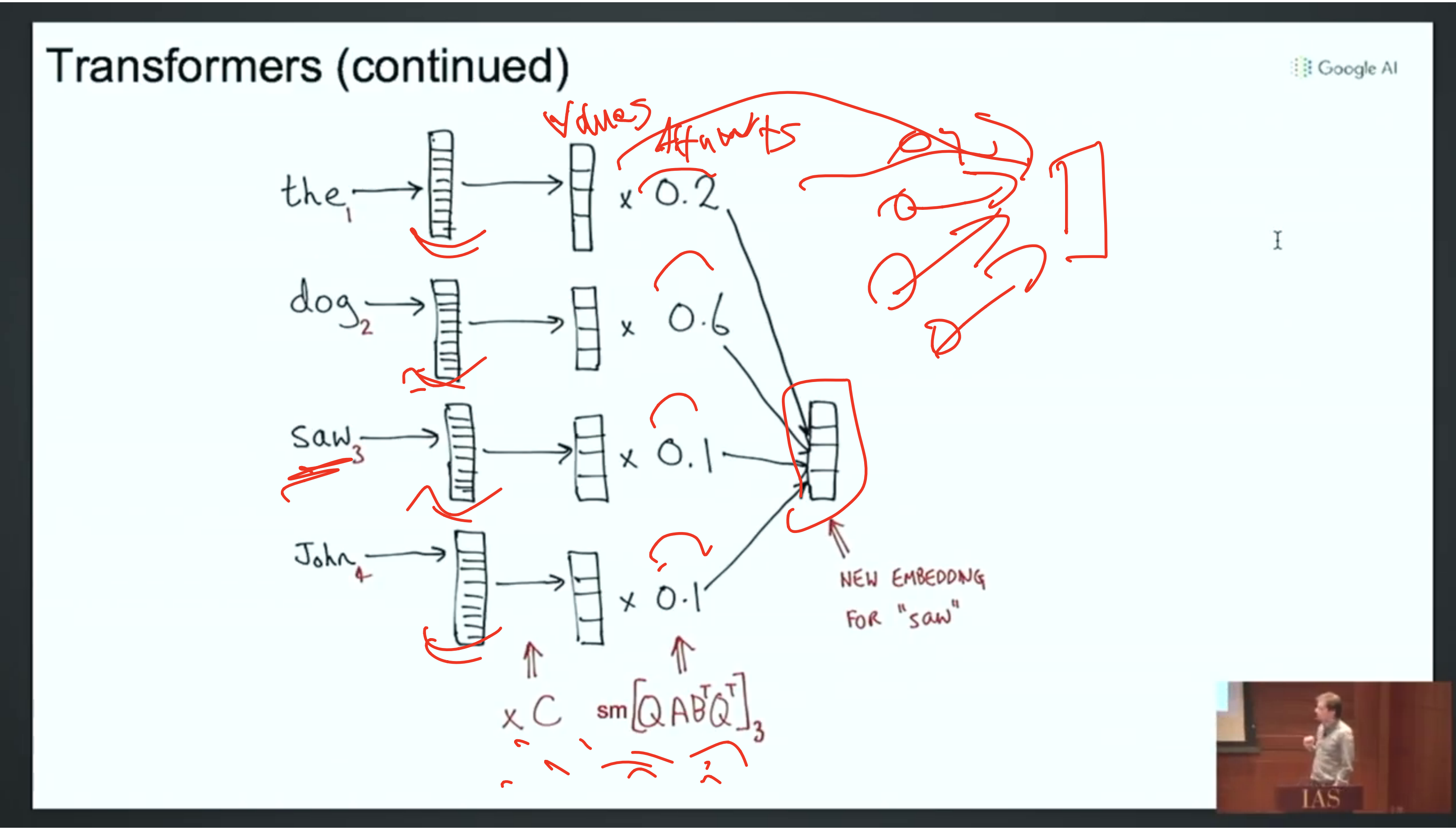
Transformers: Self-attention

Transformers (Attention is All You Need, Vaswani et al. 2017)

Google AI

- Assume we have a sequence of words $w_1 \ldots w_n$
- We can map this to a sequence of vectors $x_1 \ldots x_n$ where each $x_i \in \mathbb{R}^d$ (e.g., $d = 512$), and each x_i is the word embedding for w_i
- How do we map this to a new sequence $z_1 \ldots z_n$ where each $z_i \in \mathbb{R}^d$, where z_i's now take context into account?

[Michael Collins 2019 lecture]



[Michael Collins 2019 lecture]

https://namedtensor.github.io/

In our notation, the above equation becomes

$$\text{Attention}\colon \mathbb{R}^{\mathsf{key}} \times \mathbb{R}^{\mathsf{seq}\times\mathsf{key}} \times \mathbb{R}^{\mathsf{seq}\times\mathsf{val}} \to \mathbb{R}^{\mathsf{val}}$$

$$\text{Attention}(Q,K,V) = \underset{\mathsf{seq}}{\text{softmax}}\left(\frac{Q \underset{\mathsf{key}}{\odot} K}{\sqrt{|\mathsf{key}|}}\right) \underset{\mathsf{seq}}{\odot} V.$$

computes the value for one position!

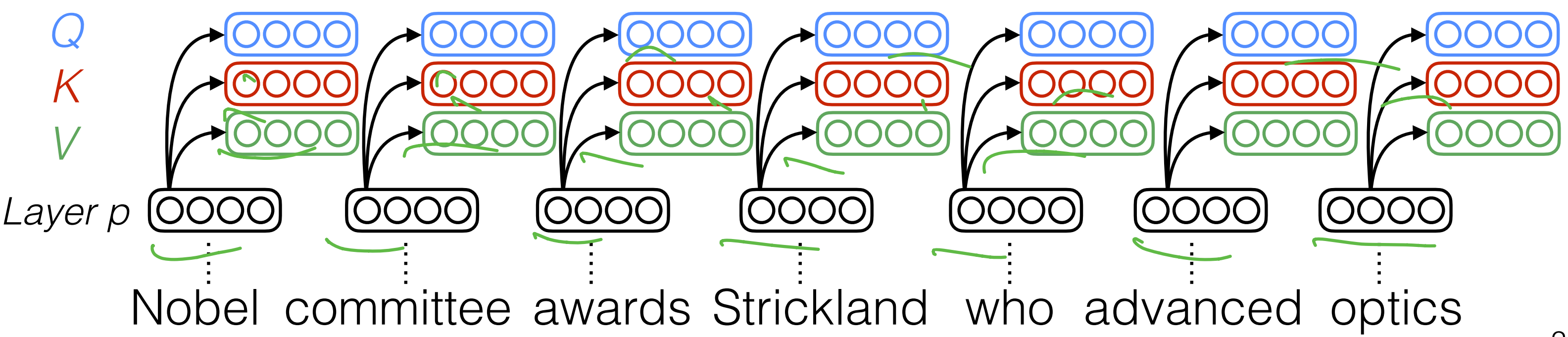
$$\sum_{t=1}^{seq} w_t \vec{V}_t = \quad \vec{w} \in \mathbb{R}^{seq}$$

$$w_t = \frac{\exp\left(\frac{1}{\sqrt{key}}\, \vec{Q} \cdot \vec{K}_t\right)}{\sum_{s=1}^{seq} \exp\left(\frac{1}{\sqrt{key}}\, \vec{Q} \cdot \vec{K}_s\right)}$$

Self-attention

[Vaswani et al. 2017 original notation, slide: Emma Strubell]

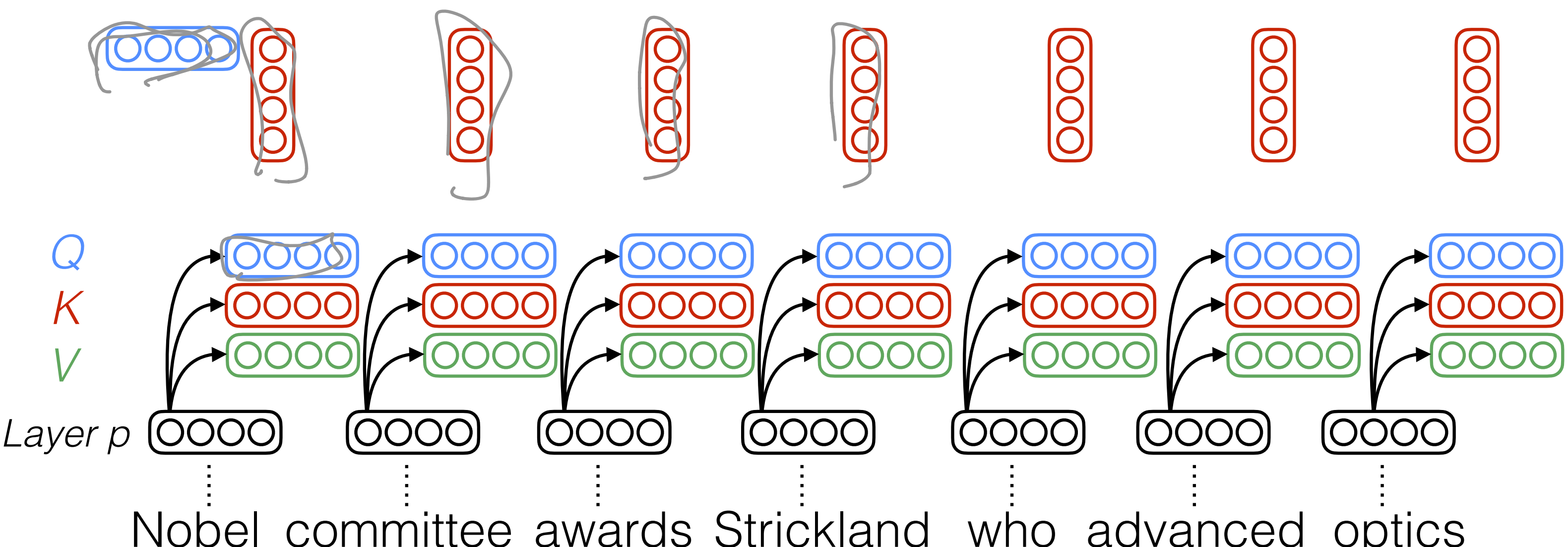
$$\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$



Self-attention

[Vaswani et al. 2017 original notation, slide: Emma Strubell]

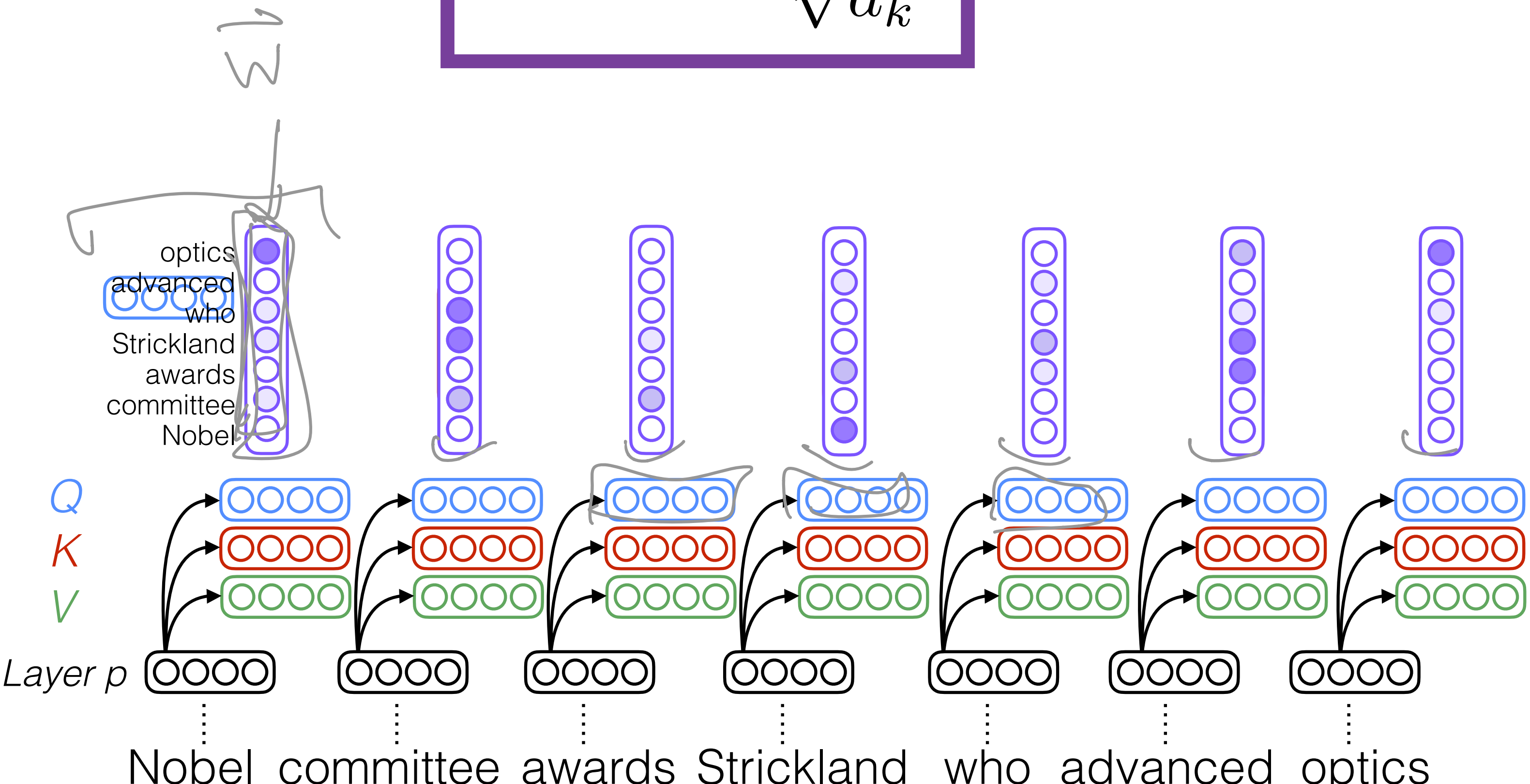
$$\mathrm{softmax}\left(\frac{\boxed{QK^T}}{\sqrt{d_k}}\right)V$$



Self-attention

[Vaswani et al. 2017
original notation,
slide: Emma Strubell]

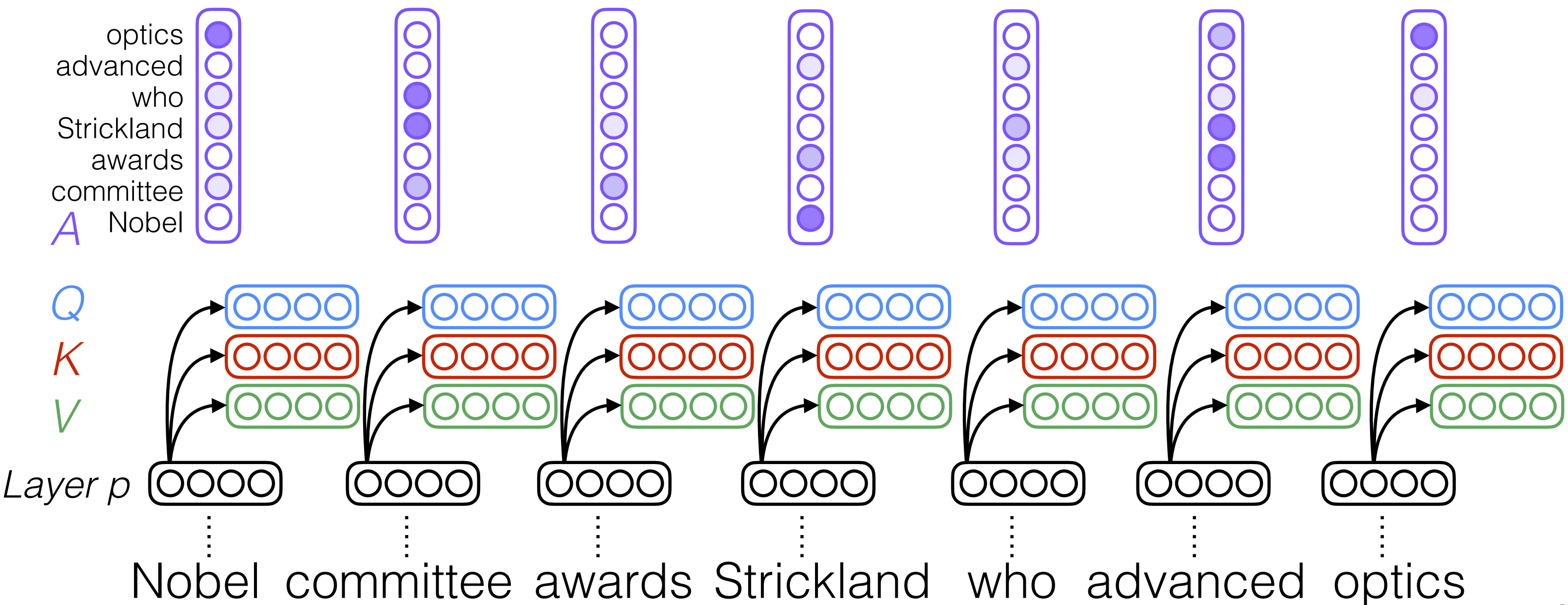
$$\boxed{\mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)}V$$



Self-attention

[Vaswani et al. 2017 original notation, slide: Emma Strubell]

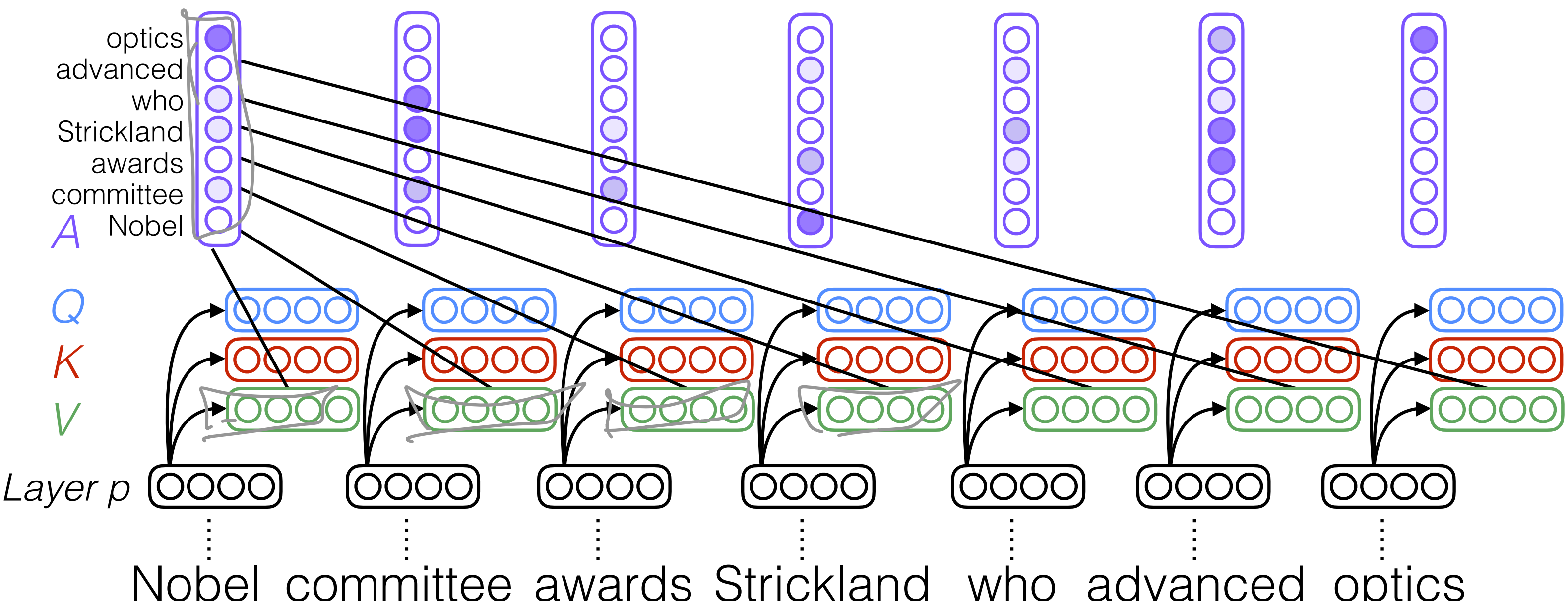
$$\boxed{\mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})}V$$



Self-attention

[Vaswani et al. 2017 original notation, slide: Emma Strubell]

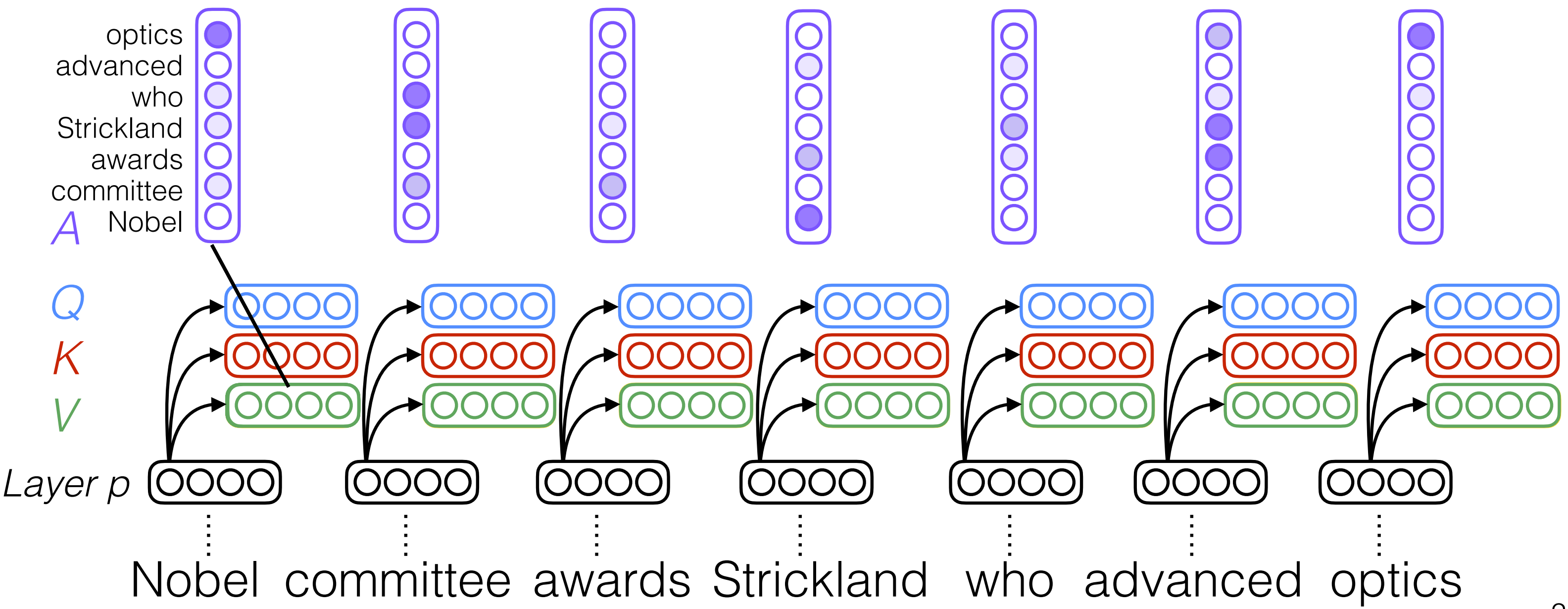
$$\boxed{\mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V}$$



Self-attention

[Vaswani et al. 2017 original notation, slide: Emma Strubell]

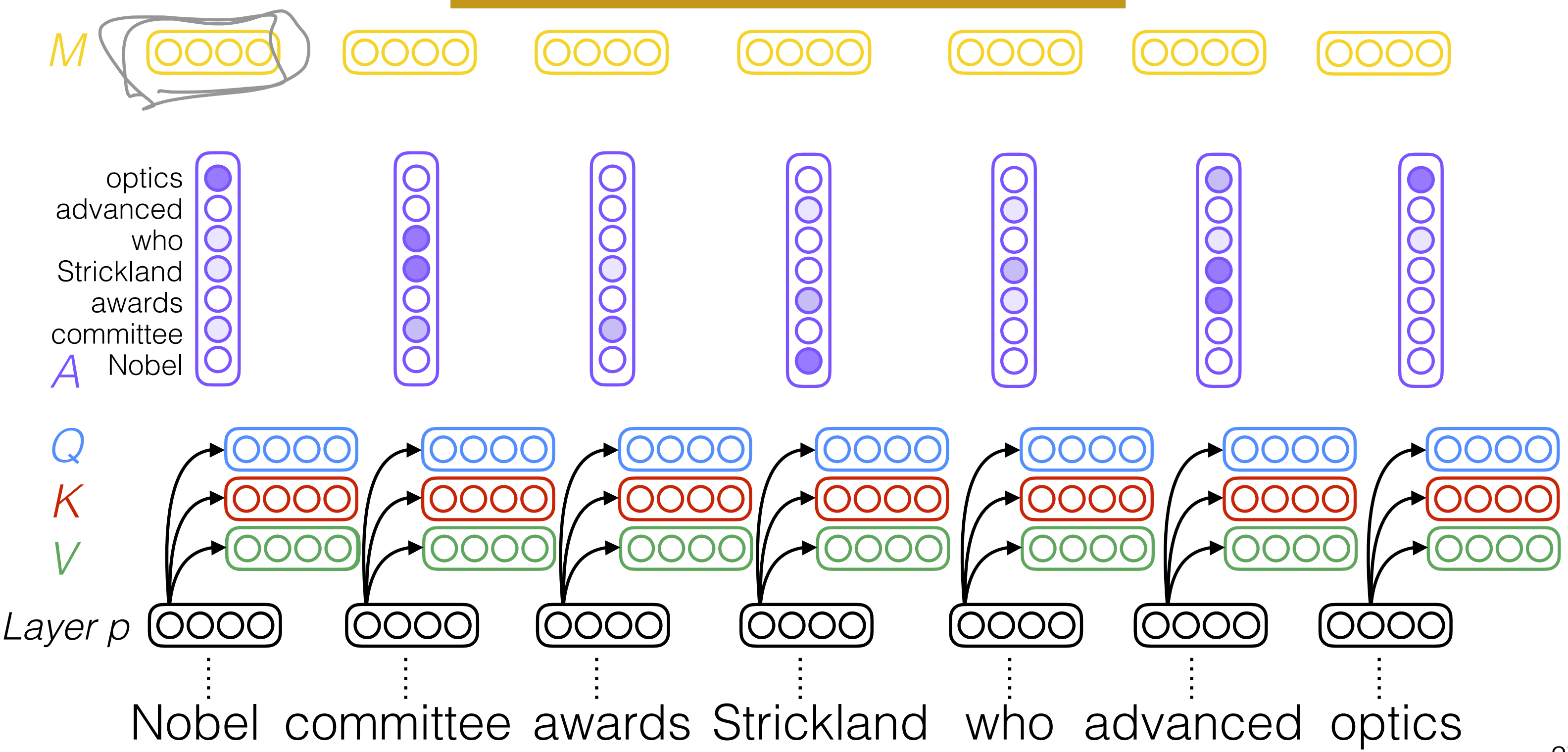
$$\mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$



Self-attention

[Vaswani et al. 2017 original notation, slide: Emma Strubell]

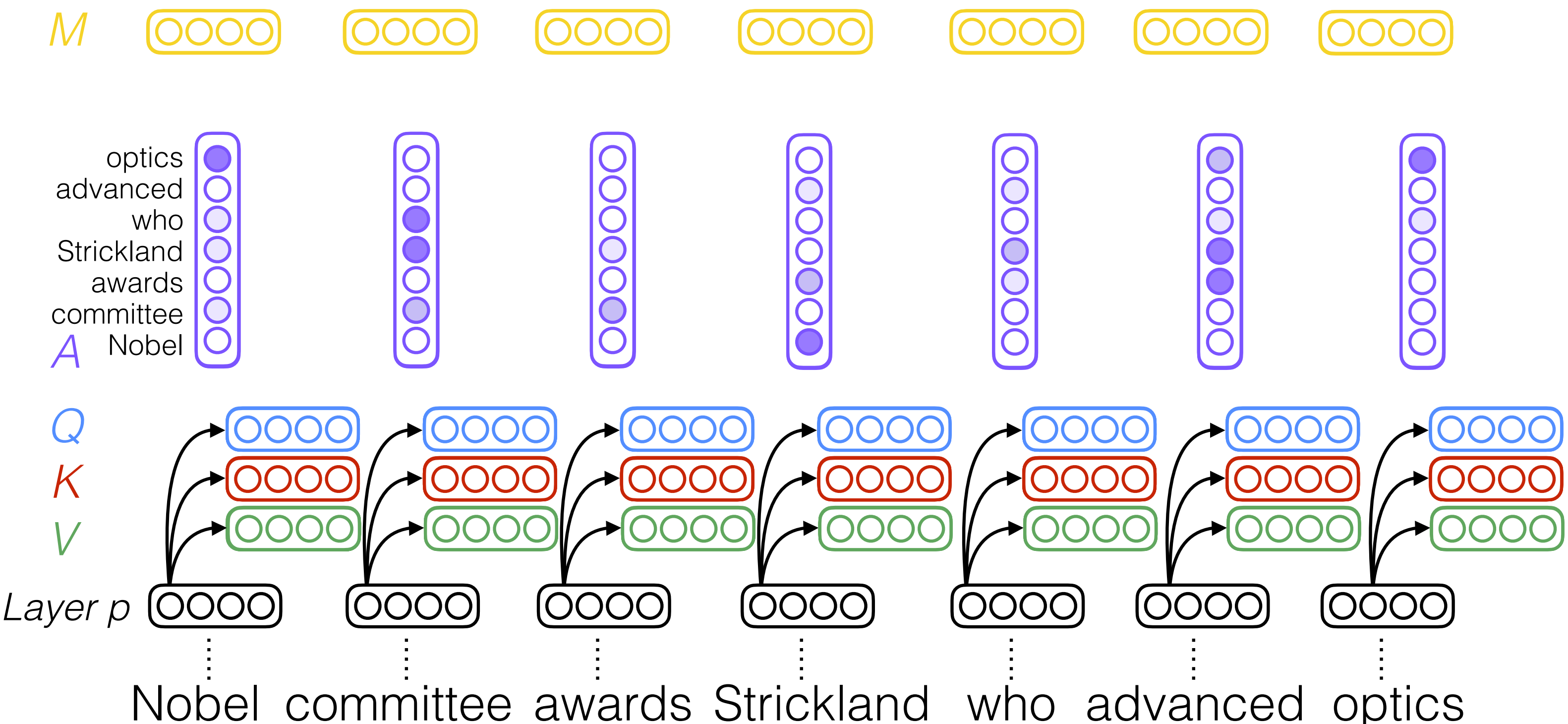
$$\mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$



Self-attention

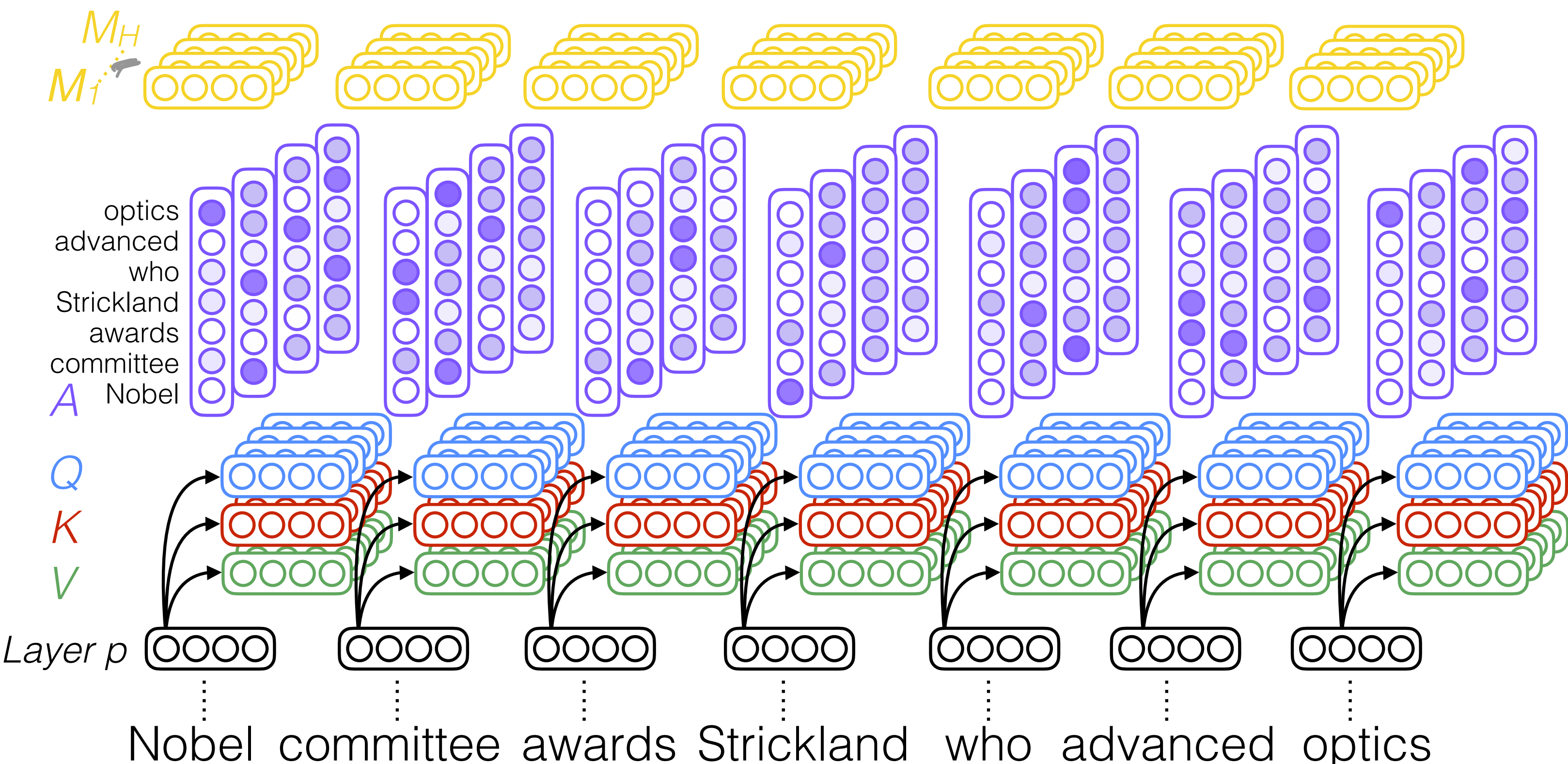
[Vaswani et al. 2017 original notation, slide: Emma Strubell]

$$\boxed{\mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V}$$



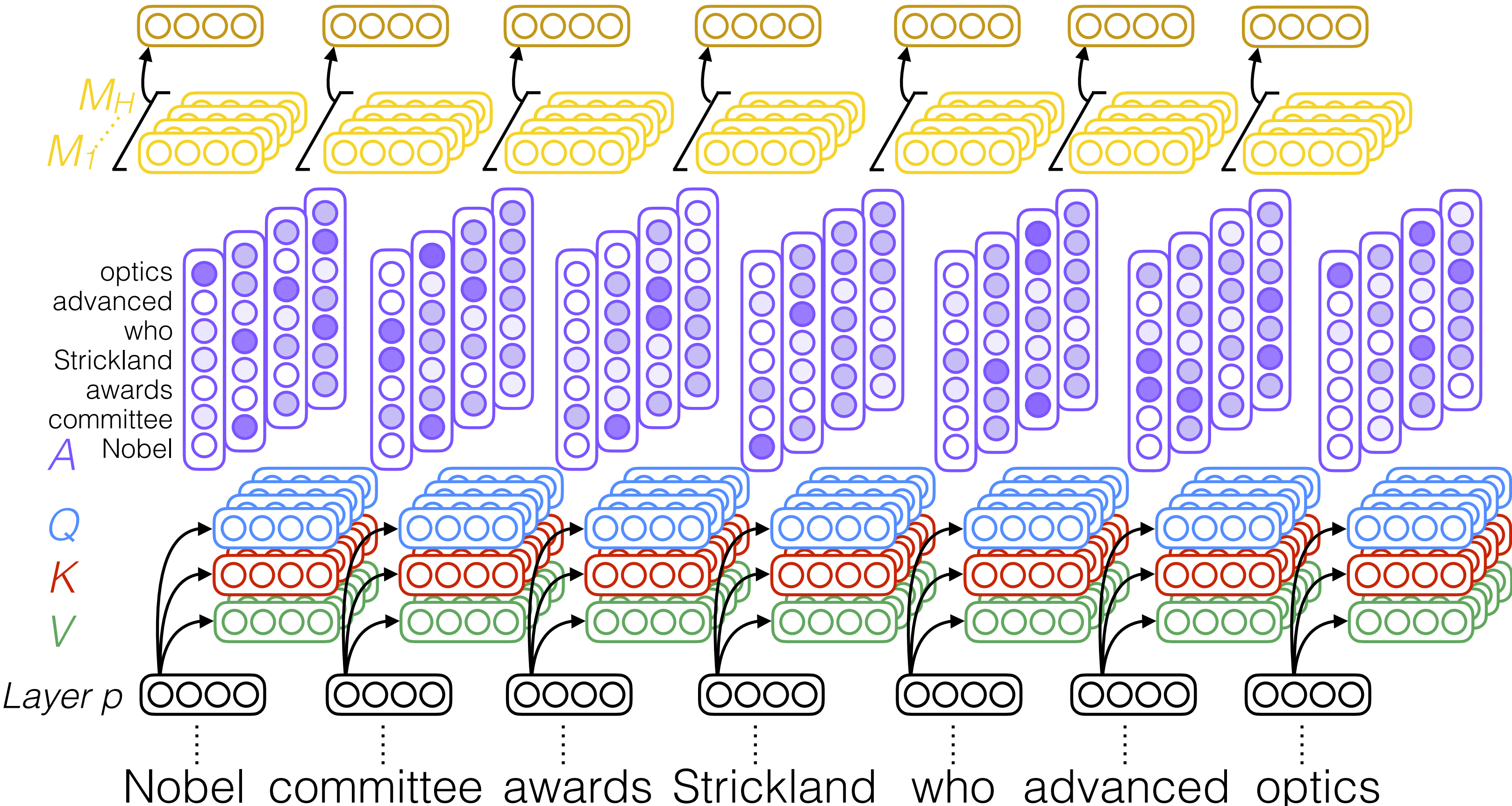
Multi-head self-attention

[Vaswani et al. 2017 original notation]



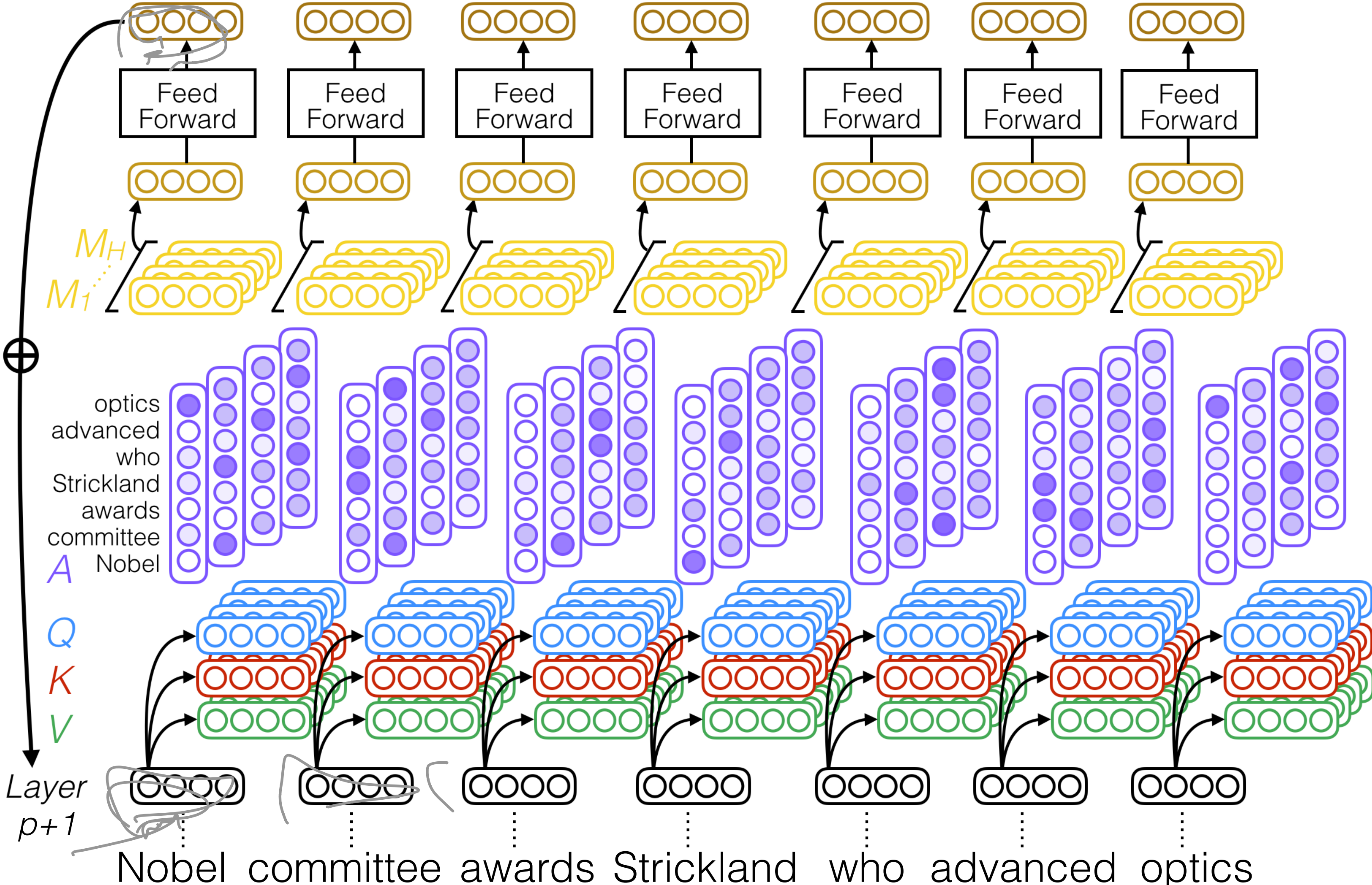
Multi-head self-attention

[Vaswani et al. 2017 original notation]



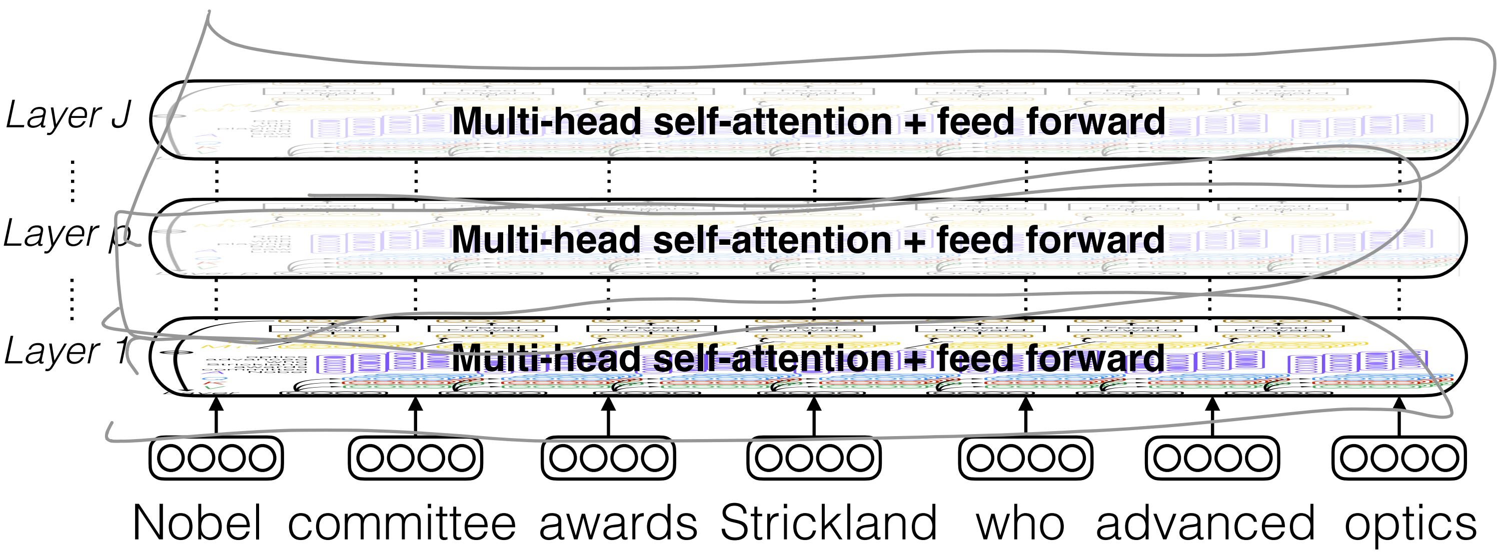
Multi-head self-attention

[Vaswani et al. 2017 original notation]

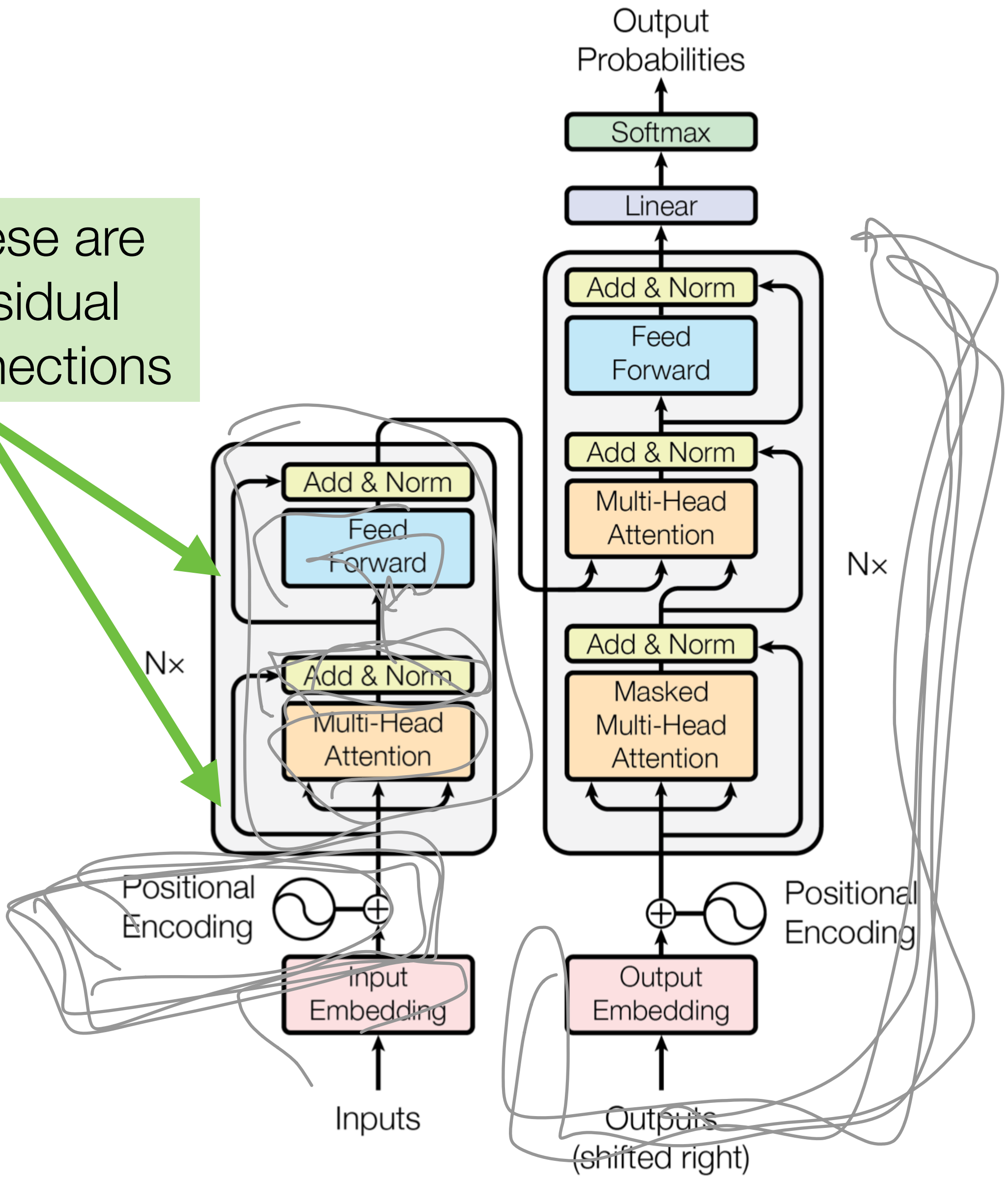


Multi-head self-attention

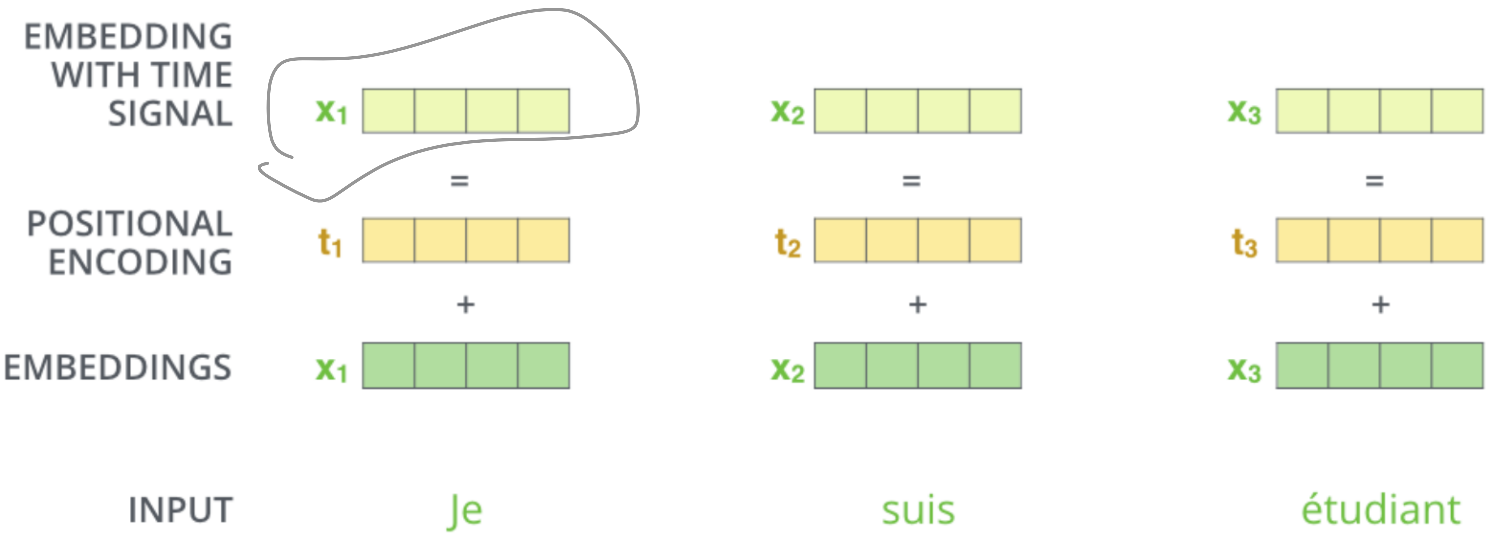
[Vaswani et al. 2017 original notation]



These are residual connections



Positional encoding



Why this function???

“We chose this function because we hypothesized it would allow the model to easily learn to attend by relative positions, since for any fixed offset k, PE_{pos+k} can be represented as a linear function of PE_{pos}.”

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d_{model}}}\right)$$

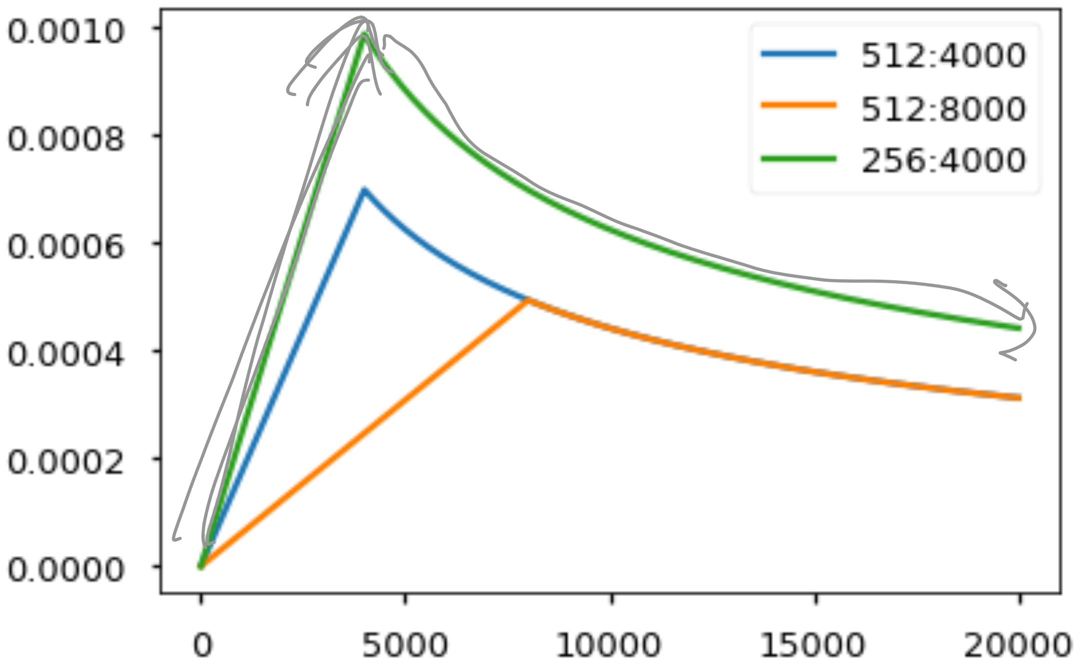
$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d_{model}}}\right)$$

Hacks to get it to work:

Optimizer

We used the Adam optimizer (cite) with $\beta_1 = 0.9$, $\beta_2 = 0.98$ and $\epsilon = 10^{-9}$. We varied the learning rate over the course of training, according to the formula: $lrate = d_{\text{model}}^{-0.5} \cdot \min(step\_num^{-0.5}, step\_num \cdot warmup\_steps^{-1.5})$ This corresponds to increasing the learning rate linearly for the first $warmup_steps$ training steps, and decreasing it thereafter proportionally to the inverse square root of the step number. We used $warmup_steps = 4000$.

> *Note: This part is very important. Need to train with this setup of the model.*



Label Smoothing

During training, we employed label smoothing of value $\epsilon_{ls} = 0.1$ (cite). This hurts perplexity, as the model learns to be more unsure, but improves accuracy and BLEU score.

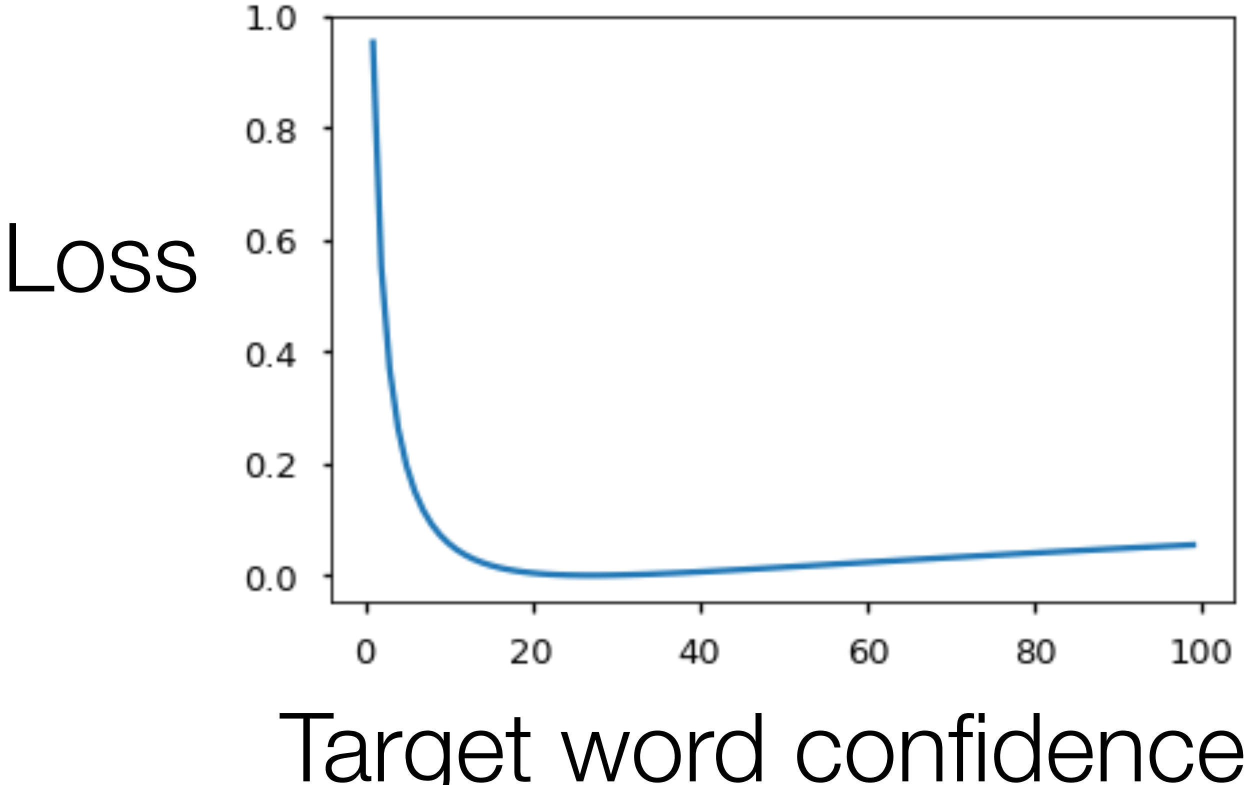
> *We implement label smoothing using the KL div loss. Instead of using a one-hot target distribution, we create a distribution that has `confidence` of the correct word and the rest of the `smoothing` mass distributed throughout the vocabulary.*

I went to class and took ___

cats	TV	notes	took	sofa
0	0	1	0	0
0.025	0.025	0.9	0.025	0.025

with label smoothing

Get penalized for overconfidence!



- Training instability is a notorious issue
  - Esp. with many layers, >10 or >20
- Yet something is going right. Not clear why!
- Next week: BERT
  - Like ELMO, but with self-attention: pretrained LM intended for downstream tasks
  - Extensive analysis has been done it; why it's good is still not totally understood?

Byte pair encoding (BPE)

- Deal with rare words / large vocabulary by instead using *subword* tokenization

system	sentence
source	health research institutes
reference	Gesundheitsforschungsinstitute
WDict	Forschungsinstitute
C2-50k	Fo|rs|ch|un|gs|in|st|it|ut|io|ne|n
BPE-60k	Gesundheits|forsch|ungsinstitu|ten
BPE-J90k	Gesundheits|forsch|ungsin|stitute
source	asinine situation
reference	dumme Situation
WDict	asinine situation → UNK → asinine
C2-50k	as|in|in|e situation → As|in|en|si|tu|at|io|n
BPE-60k	as|in|ine situation → A|in|line-|Situation
BPE-J90K	as|in|ine situation → As|in|in-|Situation

Sennrich et al., ACL 2016