Sequence Model

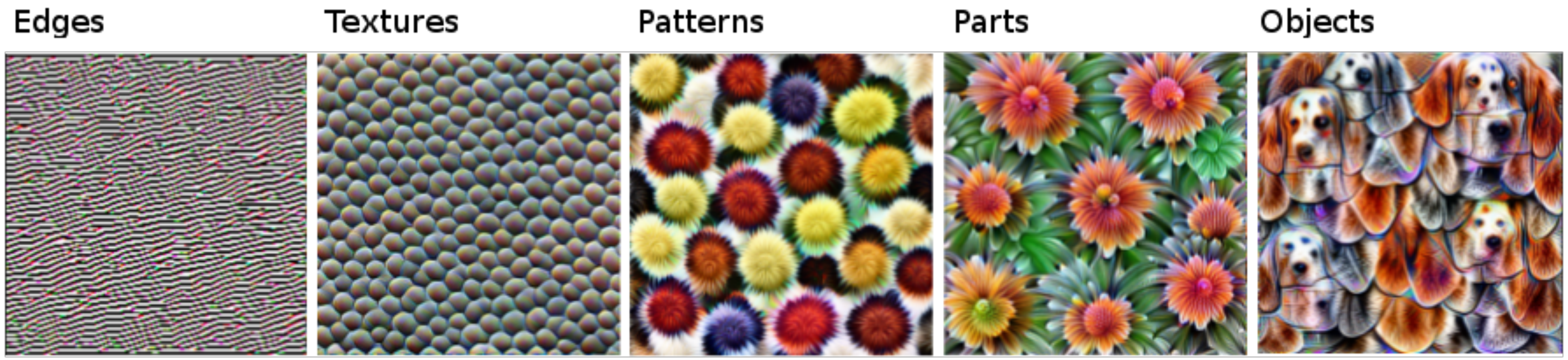
Announcements

1. Makeup exam Dec 11

2. We will release the last reading quiz today

Recap on Convolutional neural network

Learned feature representations in CNN



[https://distill.pub/2017/feature-visualization/]

Objective today

Understanding neural network structures that are suitable for natural language (i.e., sequences of words)

Outline today

1. Word-2-Vec embedding and positional embedding

2. Attention model

3. Putting things together: the Transformer model

Example: autocompletion

e.g., I went to the climbing gym and I ___

A Language model is a conditional probability model:

$$y_1 \sim P(Y = \cdot \quad x_1, \ldots, x_n) \in \mathbb{R}^{100k}$$

$$y_2 \sim P(Y = \cdot \quad x_1, \ldots, x_n, y_1)$$

$$y_m \sim P(Y = \cdot \quad x_1, \ldots, x_n, y_1, \ldots y_{m-1})$$

Word to Vector Embedding

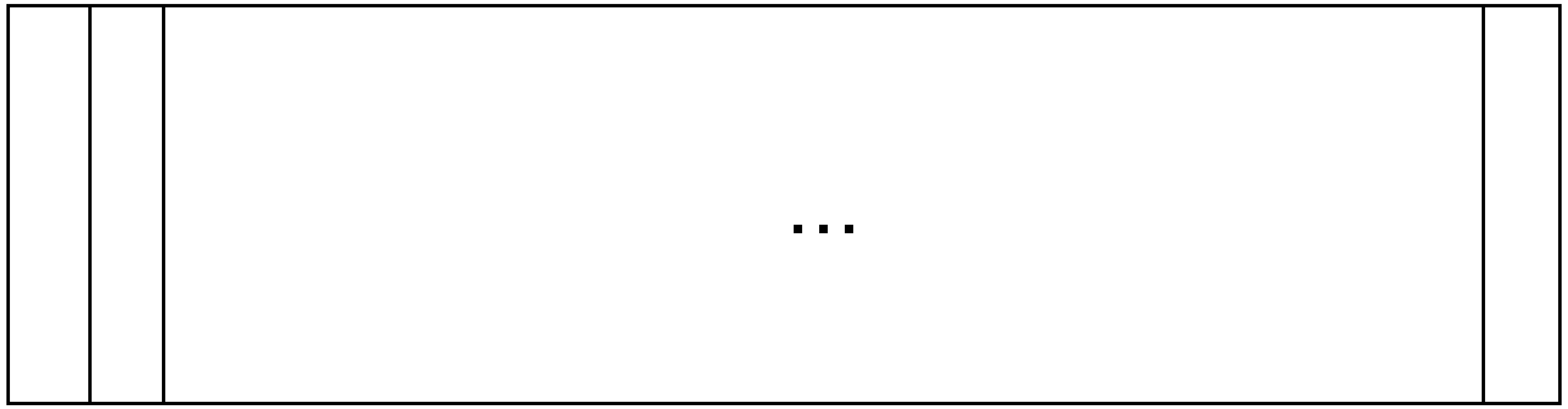
ML models only take vectors of real numbers as inputs…

e.g., I went to the climbing gym and I

$u_I \in \mathbb{R}^{128}$ $u_{went} \in \mathbb{R}^{128}$ $u_{and} \in \mathbb{R}^{128}$ $u_I \in \mathbb{R}^{128}$

Size of the English vocabulary (e.g., 100k)

128



Embedding matrix

Positional embedding

Order of the words and their positions matter…

e.g., When I say Transformer in ML, I do not mean the transformer in the movies

$u_{transformer} \in \mathbb{R}^{128}$

$+p_4 \in \mathbb{R}^{128}$

$u_{transformer} + p_{13} \in \mathbb{R}^{128}$

Create positional embedding using sin functions

High frequency

$$p_t = \begin{bmatrix} \sin(t/c_1) \\ \sin(t/c_2) \\ \cdots \\ \sin(t/c_{128}) \end{bmatrix}$$

Low frequency

Summary so far

We turn words into vectors of real numbers

e.g., When I say Transformer in ML, I do not mean the transformer in the movies

$u_{transformer} + p_4$

$u_{transformer} + p_{13} \in \mathbb{R}^{128}$

Feature of the word + feature of the position

Outline today

1. Word-2-Vec embedding and positional embedding

2. Attention model

3. Putting things together: the Transformer model

Motivation

e.g., When I say Transformer in ML, I do not mean the transformer in the movies

e.g., When I say Transformer, I literally mean the transformer in the movies

Contextual feature: feature of a word should depend on the context around it

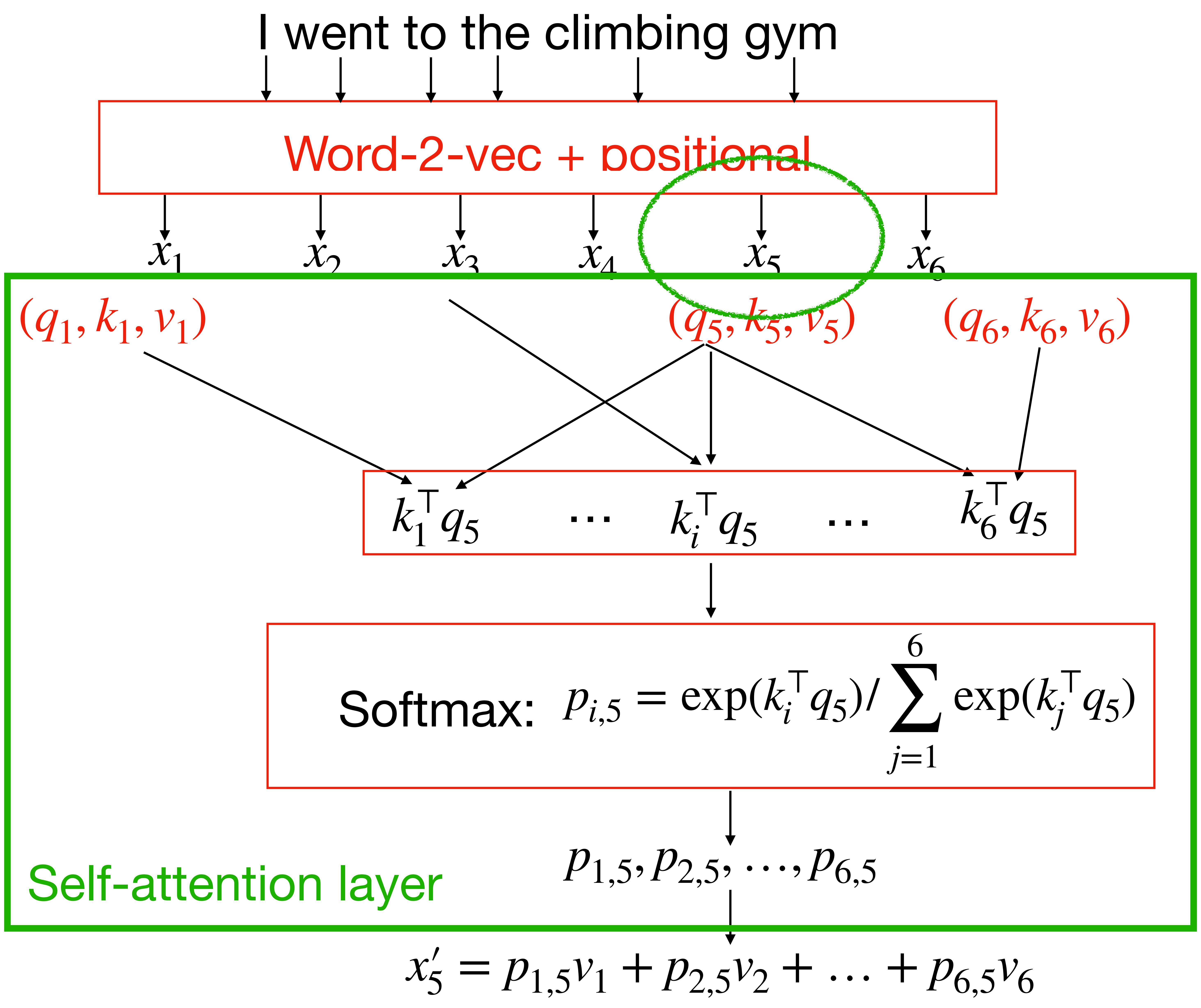
Self-attention

Attention head:
three matrices:

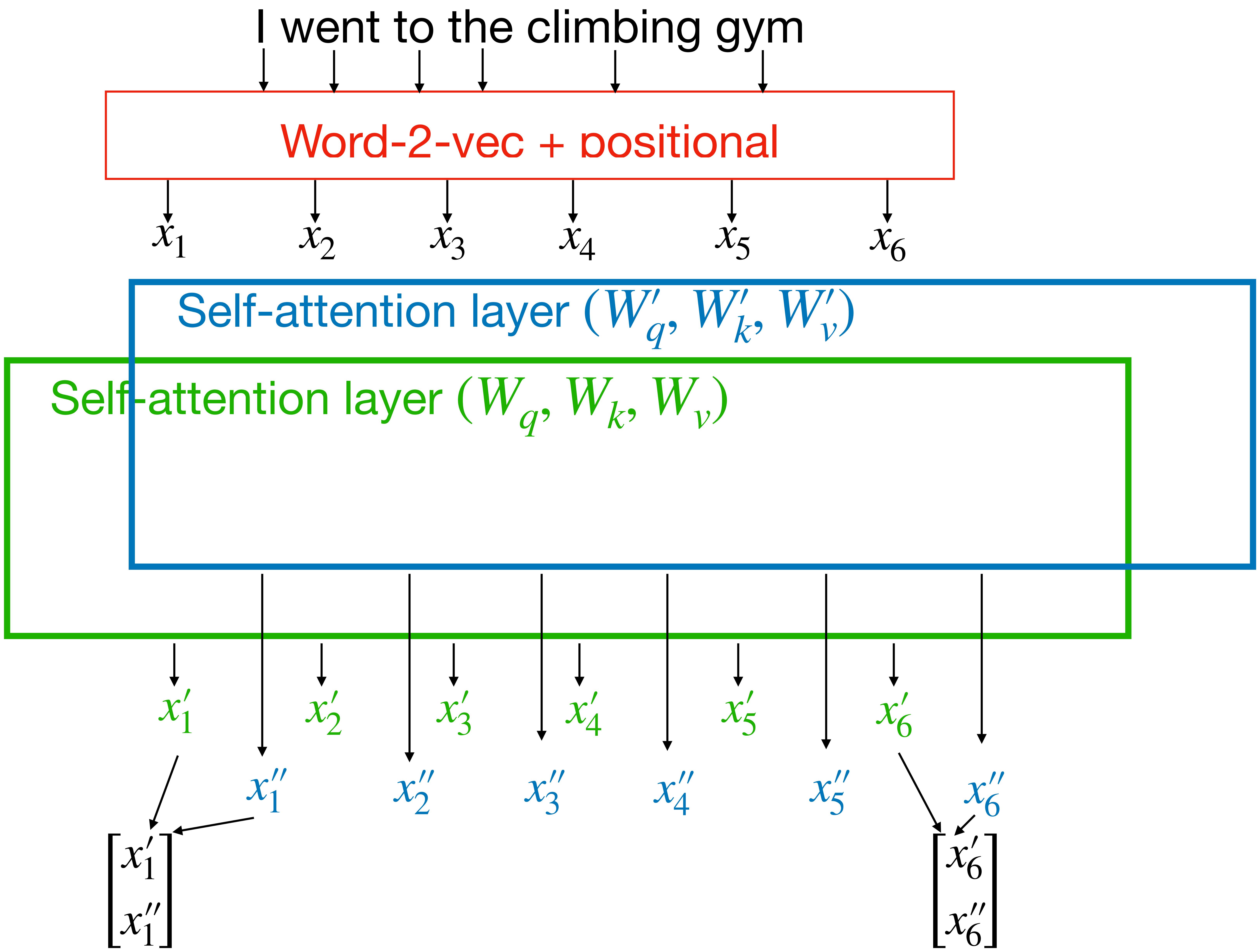
$$W_q, W_k, W_v$$

$q = W_q x$ $\quad$ $k = W_q x$ $\quad$ $v = W_q x$

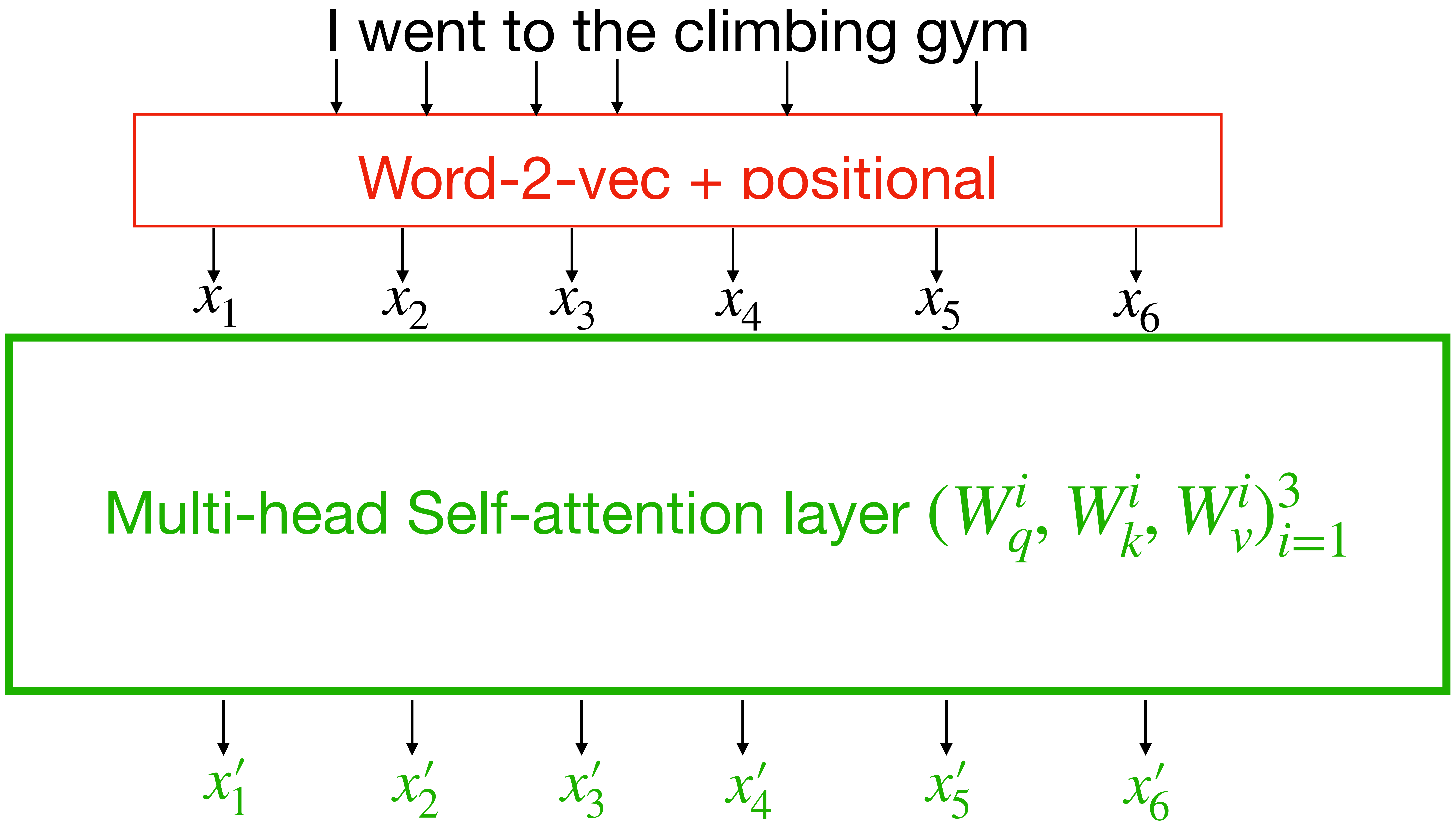
Query $\quad$ key $\quad$ value



Multi-head self-attention



Summary so far



Contextual features: e.g., x'_4 encodes information from all words

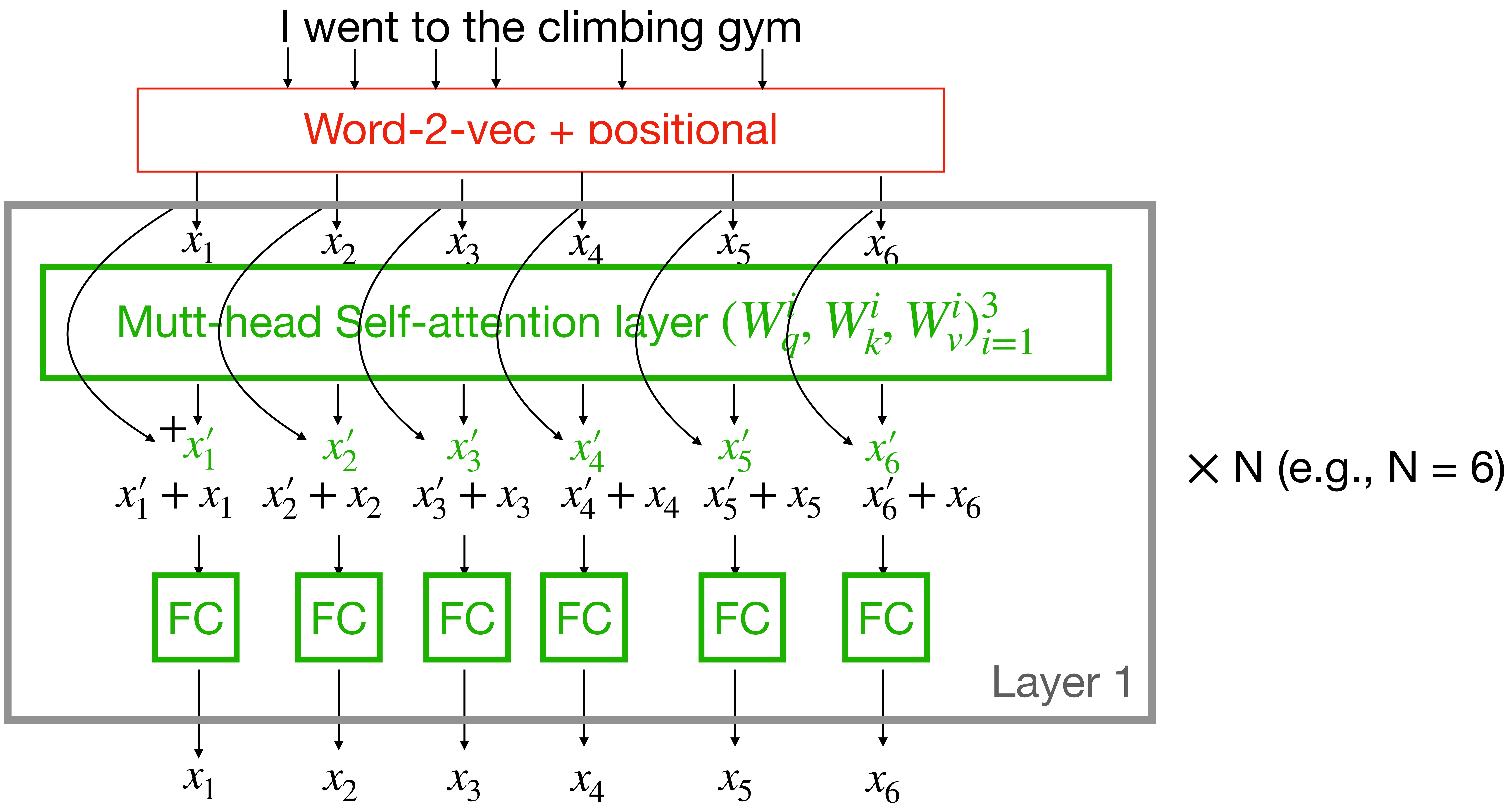
Outline today

1. Word-2-Vec embedding and positional embedding

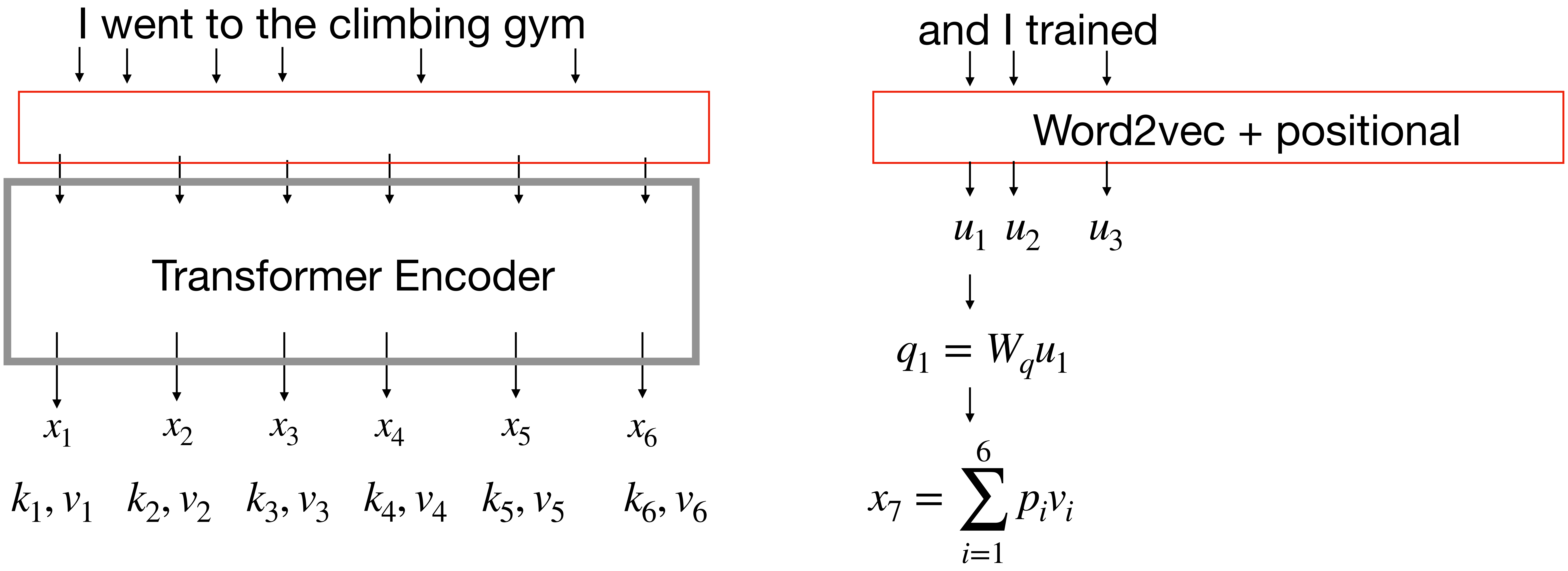
2. Attention model

3. Putting things together: the Transformer model

The Transformer model: encoder

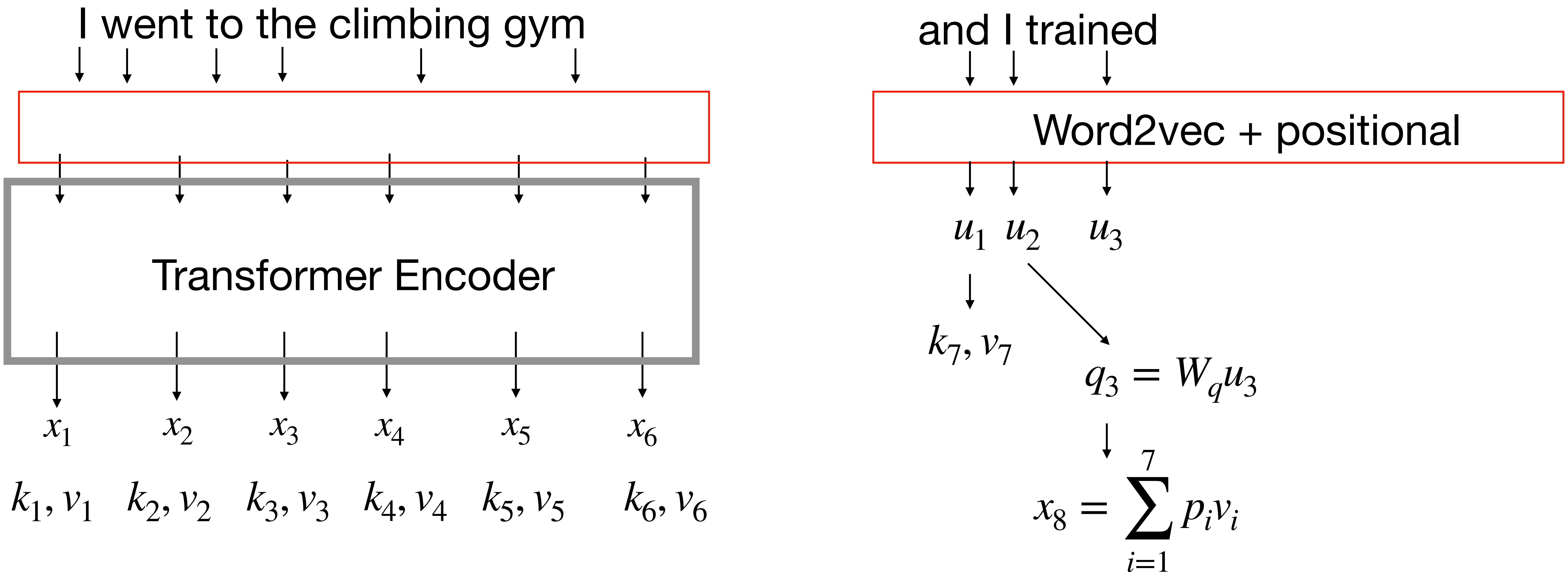


The Transformer model: decoder



cross-attention (W_q, W_k, W_v)

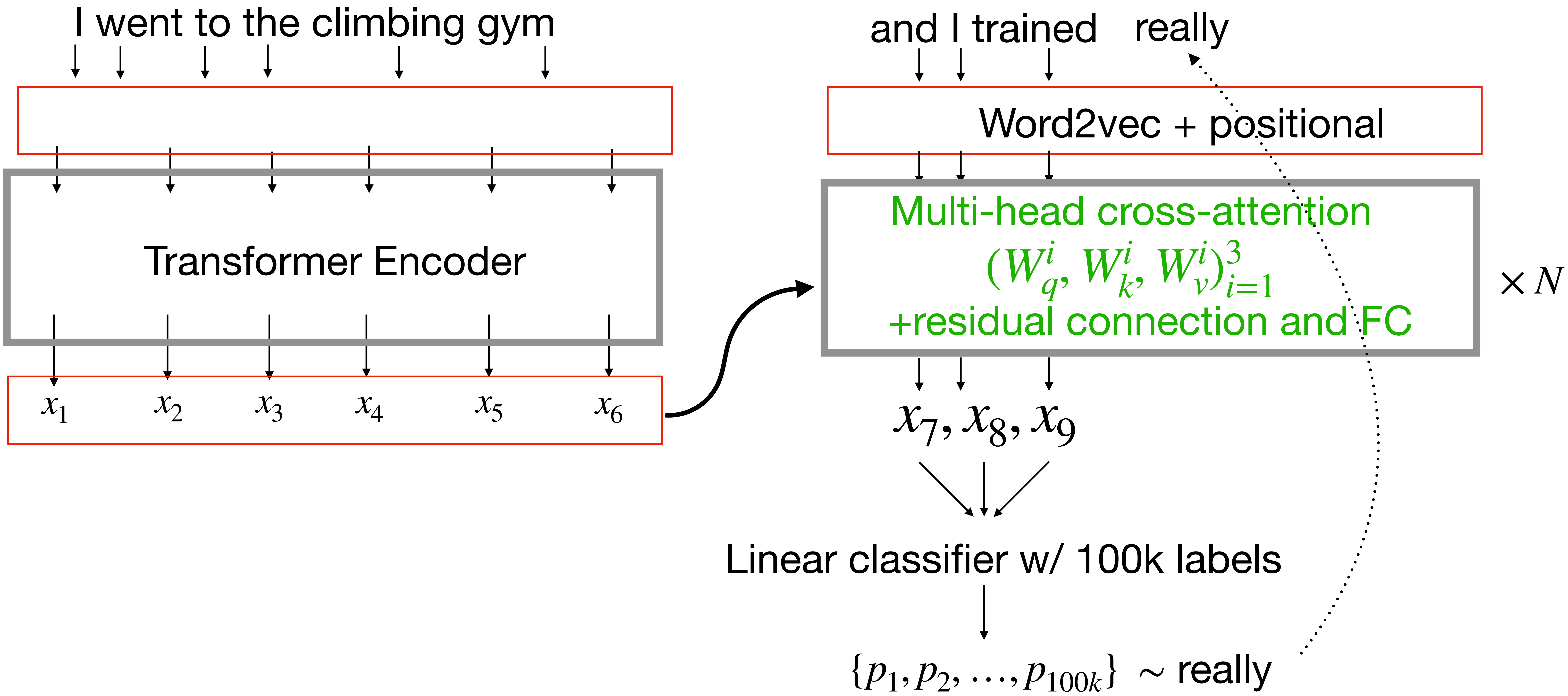
The Transformer model: decoder



cross-attention (W_q, W_k, W_v)

Note: we do not pay attention to future words

The Transformer model: decoder



Take home task:

Check out the the original paper (not too hard to read!)

Attention Is All You Need

Ashish Vaswani[*]
Google Brain
avaswani@google.com

Noam Shazeer[*]
Google Brain
noam@google.com

Niki Parmar[*]
Google Research
nikip@google.com

Jakob Uszkoreit[*]
Google Research
usz@google.com

Llion Jones[*]
Google Research
llion@google.com

Aidan N. Gomez[* †]
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser[*]
Google Brain
lukaszkaiser@google.com

Illia Polosukhin[* ‡]
illia.polosukhin@gmail.com