Object Recognition – Vision Transformers

Danna Gurari

University of Colorado Boulder

Fall 2024

https://dannagurari.colorado.edu/course/recent-advances-in-computer-vision-fall-2024/

Review

- Last lecture: object recognition with CNNs
  - ImageNet Challenge Top Performers
  - Baseline Model: AlexNet
  - VGG
  - ResNet
  - Summary of CNN Era
- Assignments (Canvas)
  - Reading assignment was due earlier today
  - Next reading assignments due next Monday and Wednesday
  - Project proposal due in 2 weeks
- Questions?

Today’s Topics

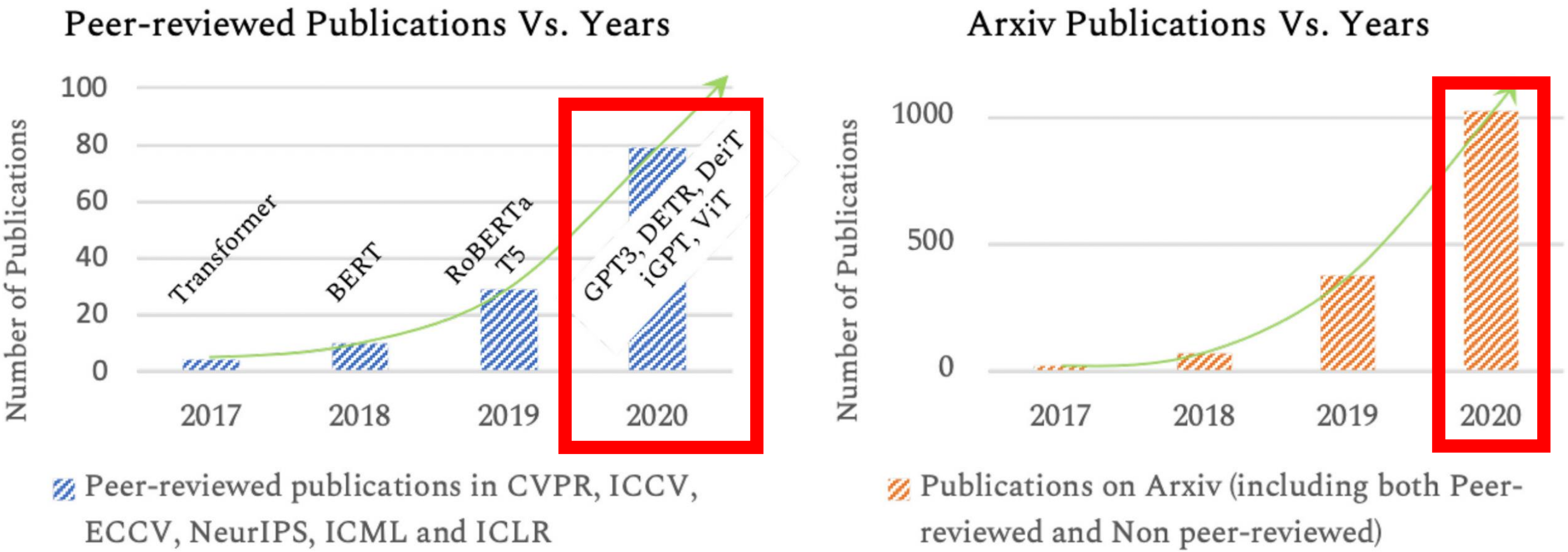
- Motivation
- ViT architecture
- ViT training
- Guidance for student-led lectures

Today’s Topics

- Motivation
- ViT architecture
- ViT training
- Guidance for student-led lectures

Introduced in 2017, Transformers Achieved Astonishing Performance for NLP Problems

(*Frequency of certain words appearing in paper title*)



Inspired, researchers in the computer vision community explored transformers for many vision problems and discovered they perform well!

Khan et al. Transformers in Vision: A Survey. CSUR 2022

Transformer: A Suggested Definition

"Any architecture designed to process a connected set of units—such as the tokens in a sequence or the pixels in an image—where the only interaction between units is through self-attention."

http://peterbloem.nl/blog/transformers

Why ViT?

Named after the proposed technique: Vision Transformer

Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

Novelty

- First paper to demonstrate that a pure transformer architecture can achieve strong performance on vision tasks, achieving comparable or better image classification results to the best methods at the time

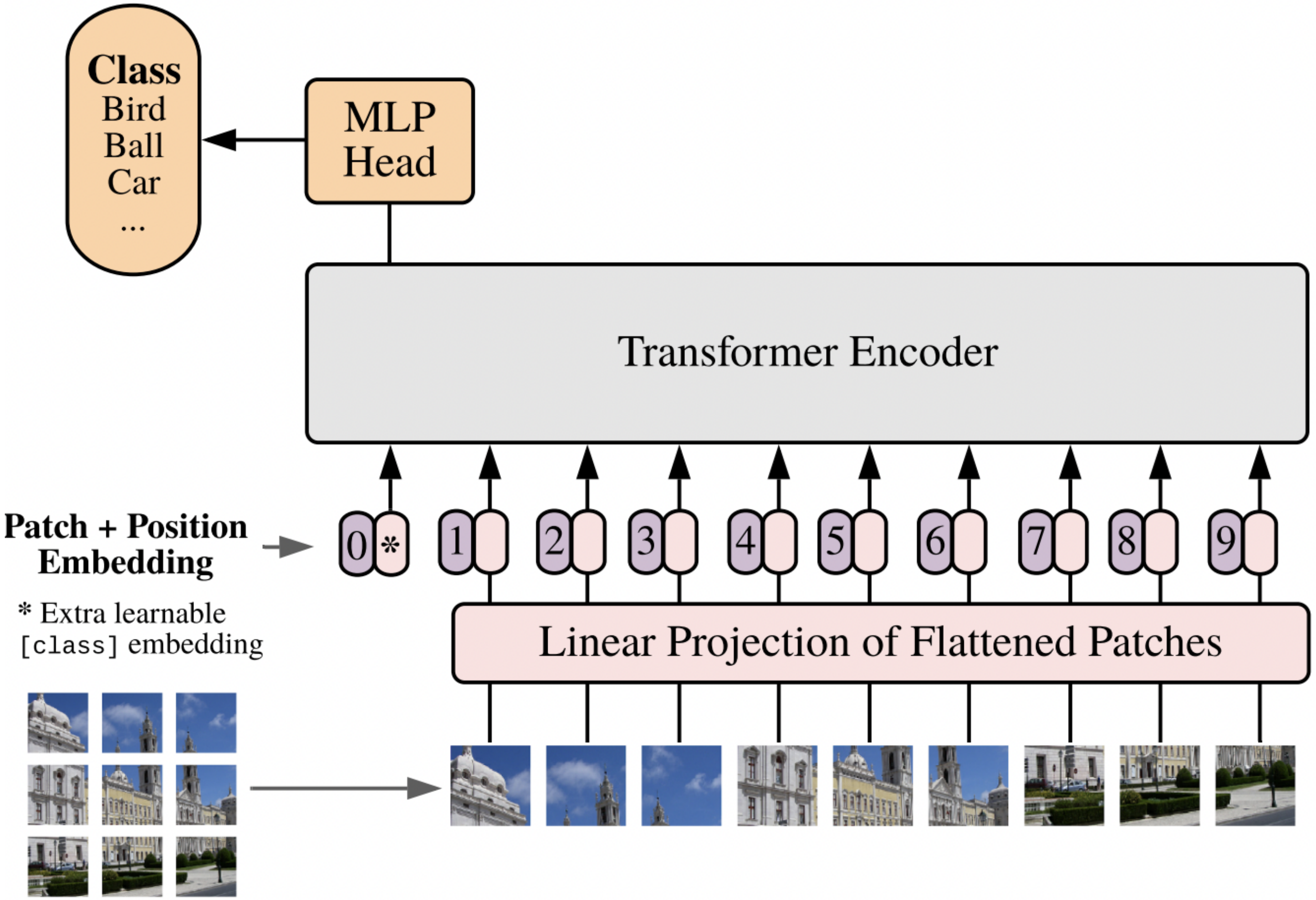
ViT: Key Ingredients for Success

- Transformer architecture (embeds self-attention)
- Pre-training with massive amounts of data

Today’s Topics

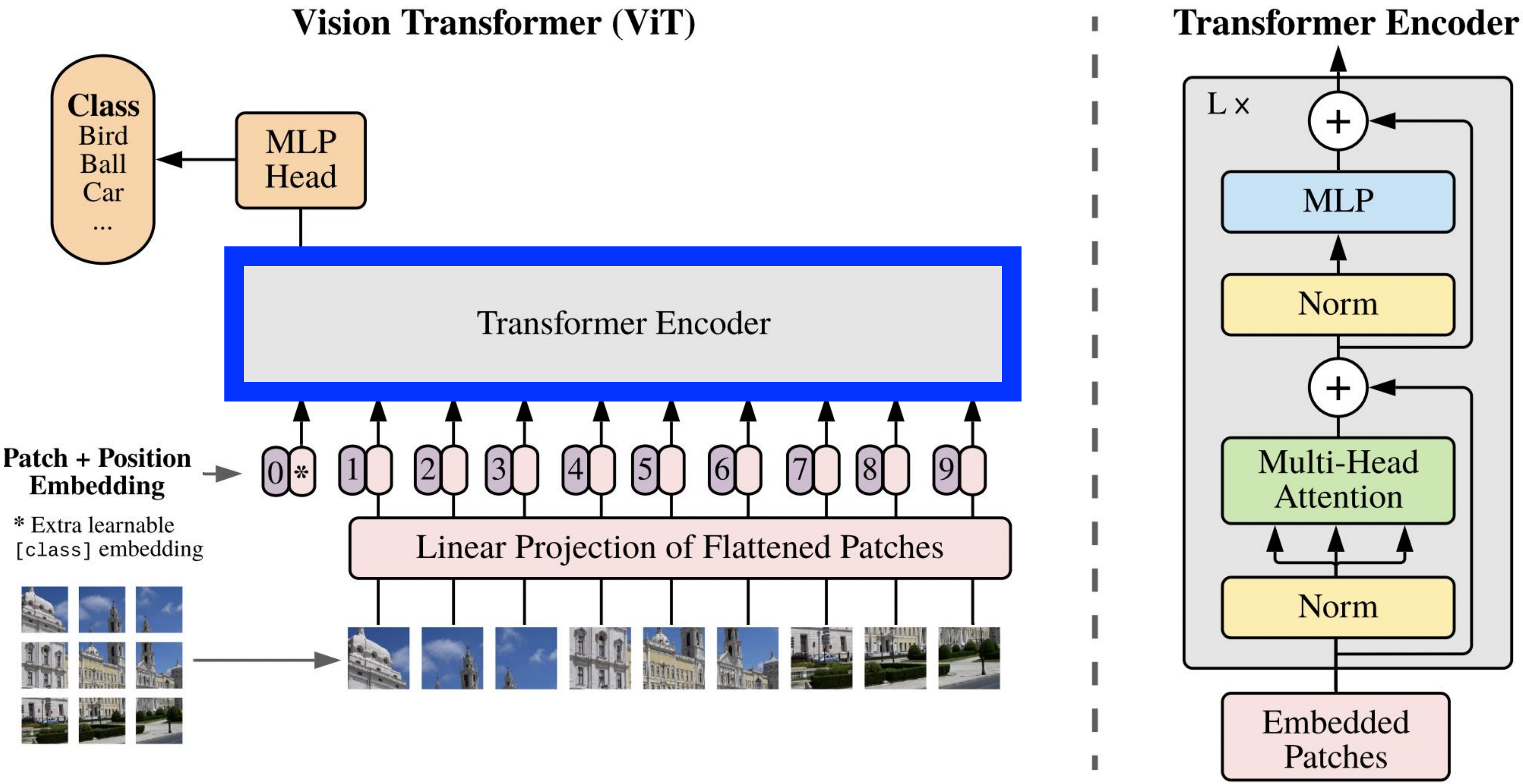
- Motivation
- ViT architecture
- ViT training
- Guidance for student-led lectures

Architecture



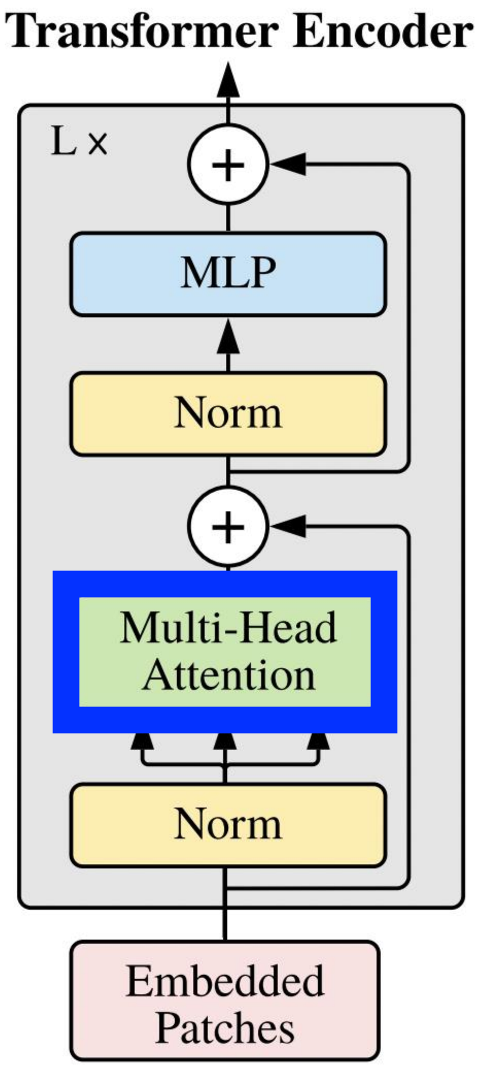
Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021

Architecture: Uses Popular BERT Architecture



Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021

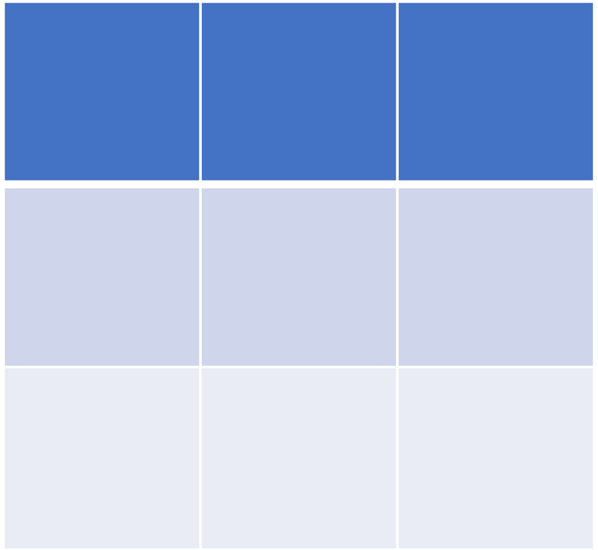
Architecture: Key Ingredient is Self-Attention



Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021

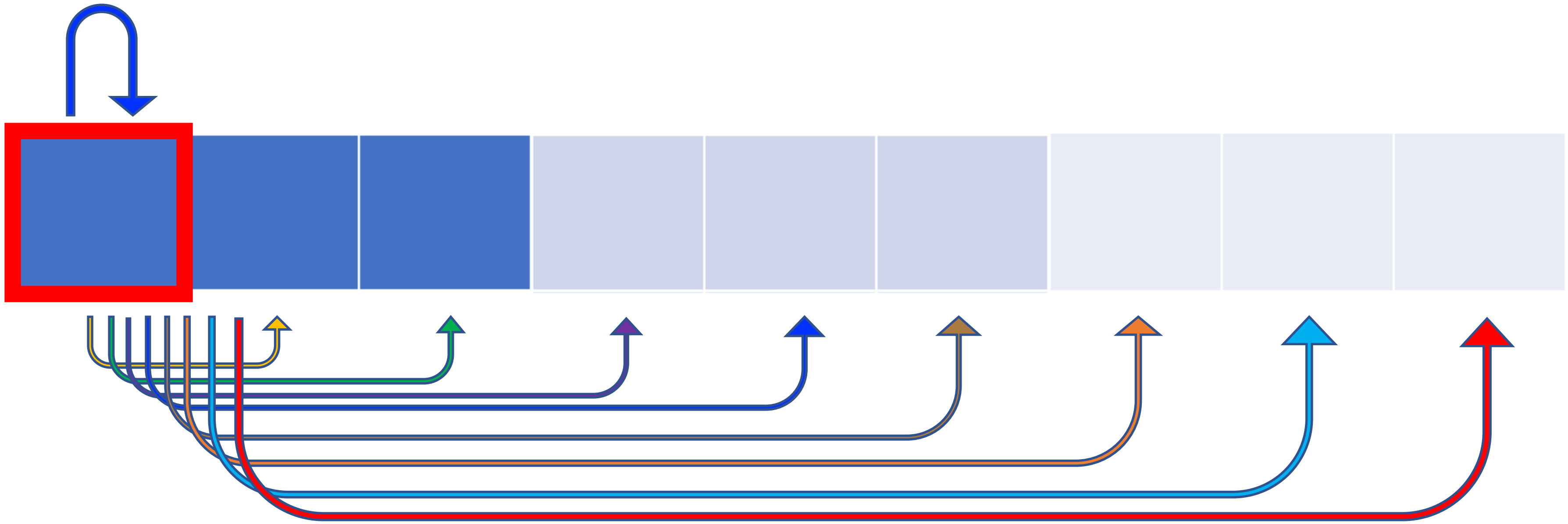
Self-Attention: Idea

New representation of each pixel showing its relationship to all pixels; e.g., assume this 3x3 image



Self-Attention: Idea

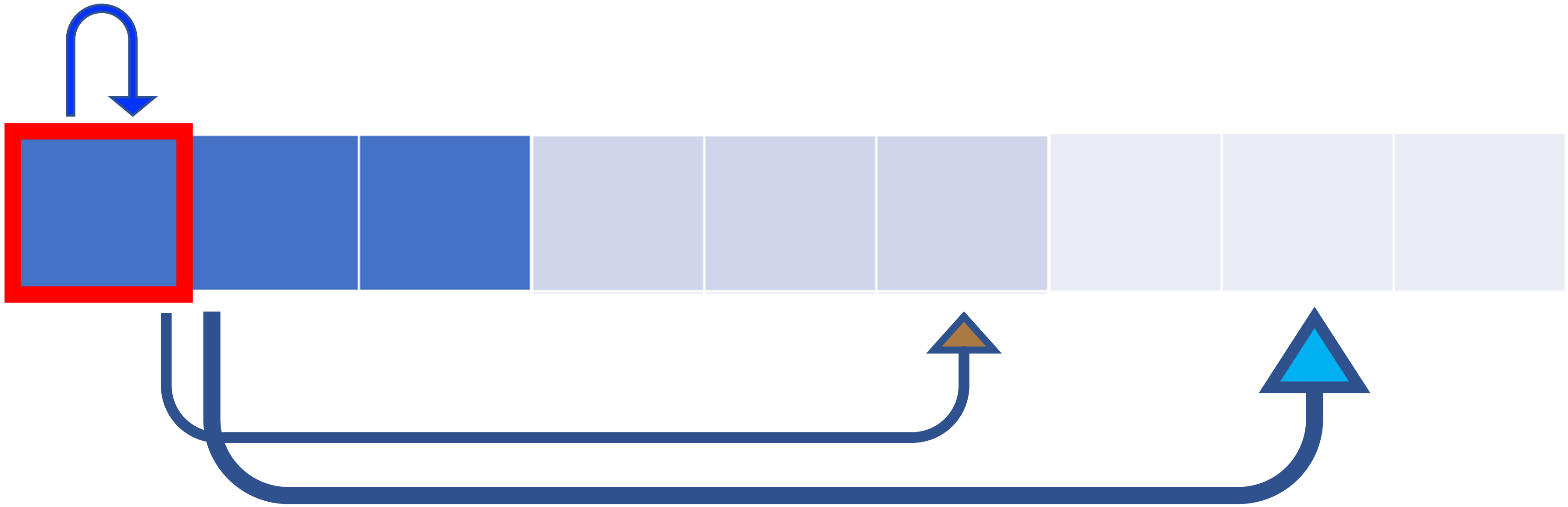
New representation of each pixel showing its relationship to all pixels; e.g., assume this 3x3 image



Learned new representation indicates which global information clarifies a pixel's meaning (e.g., include in the representation of a pixel of an eye context of what animal it belongs to)

Self-Attention: Idea

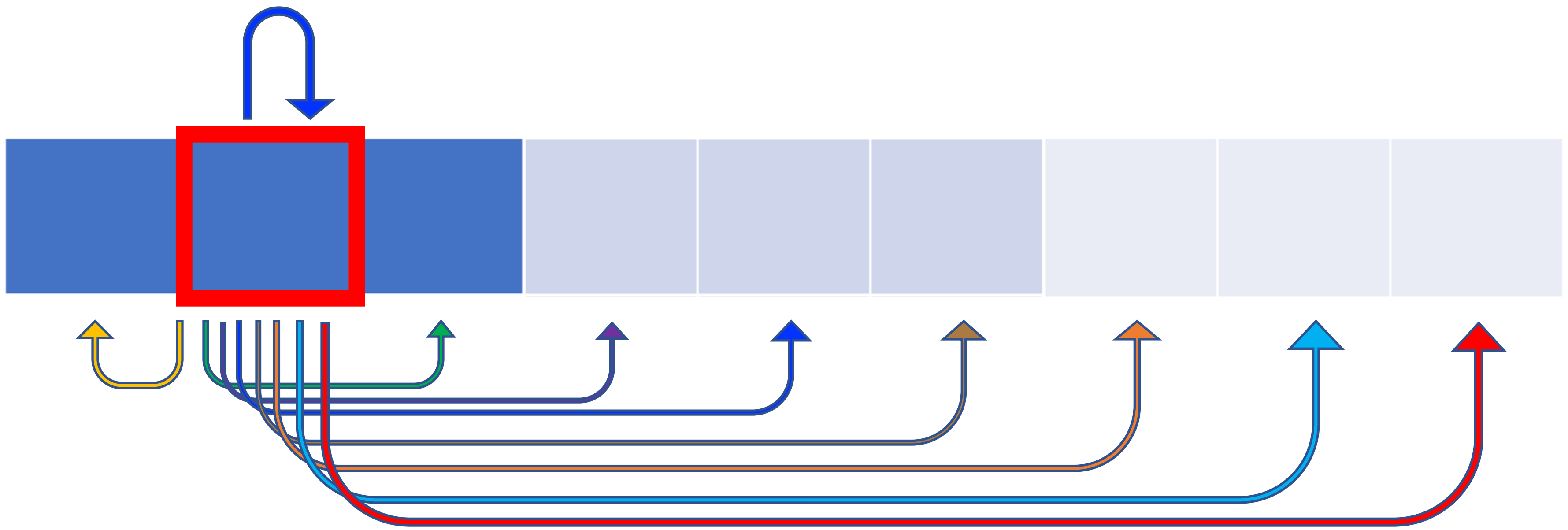
New representation of each pixel showing its relationship to all pixels; e.g., assume this 3x3 image



Learned new representation indicates which global information clarifies a pixel's meaning (e.g., include in the representation of a pixel of an eye context of what animal it belongs to)

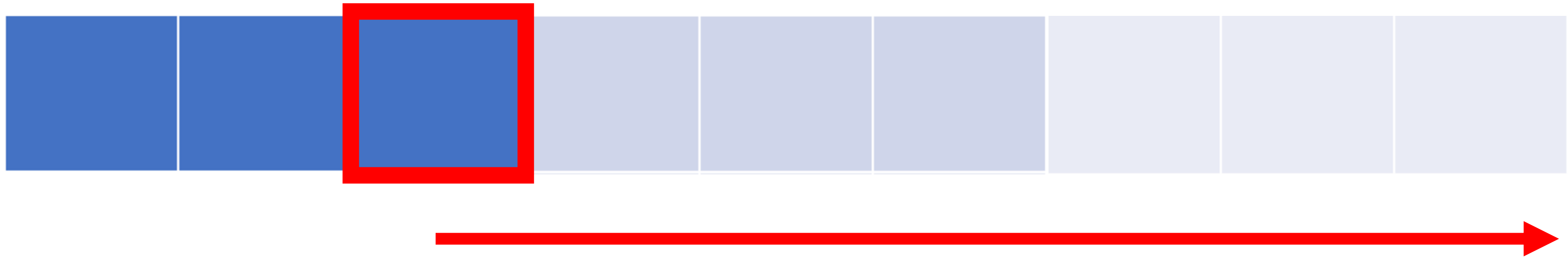
Self-Attention: Idea

New representation of each pixel showing its relationship to all pixels; e.g., assume this 3x3 image



Self-Attention: Idea

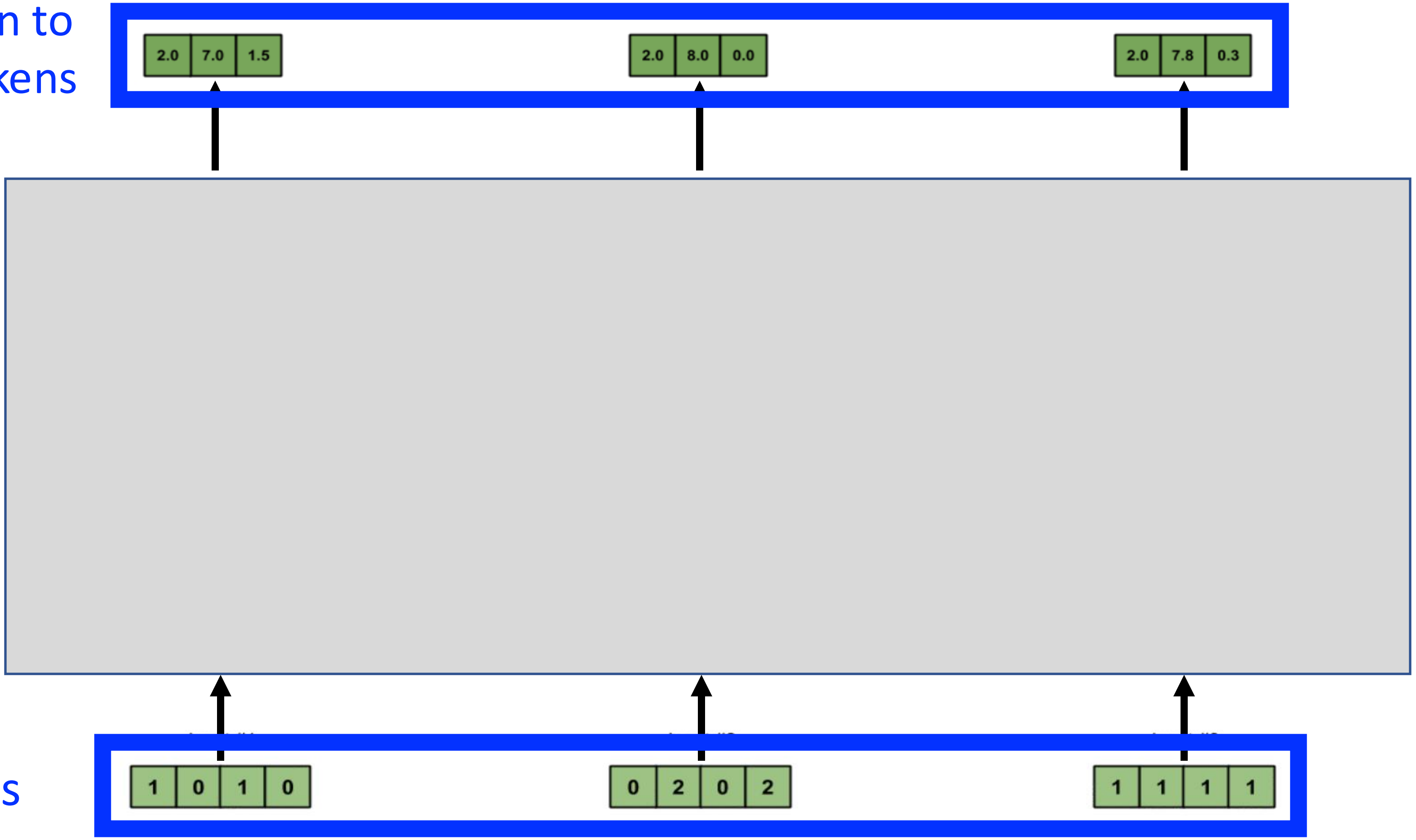
New representation of each pixel showing its relationship to all pixels; e.g., assume this 3x3 image



And so on for remaining image pixels…

Computing Self-Attention: Example

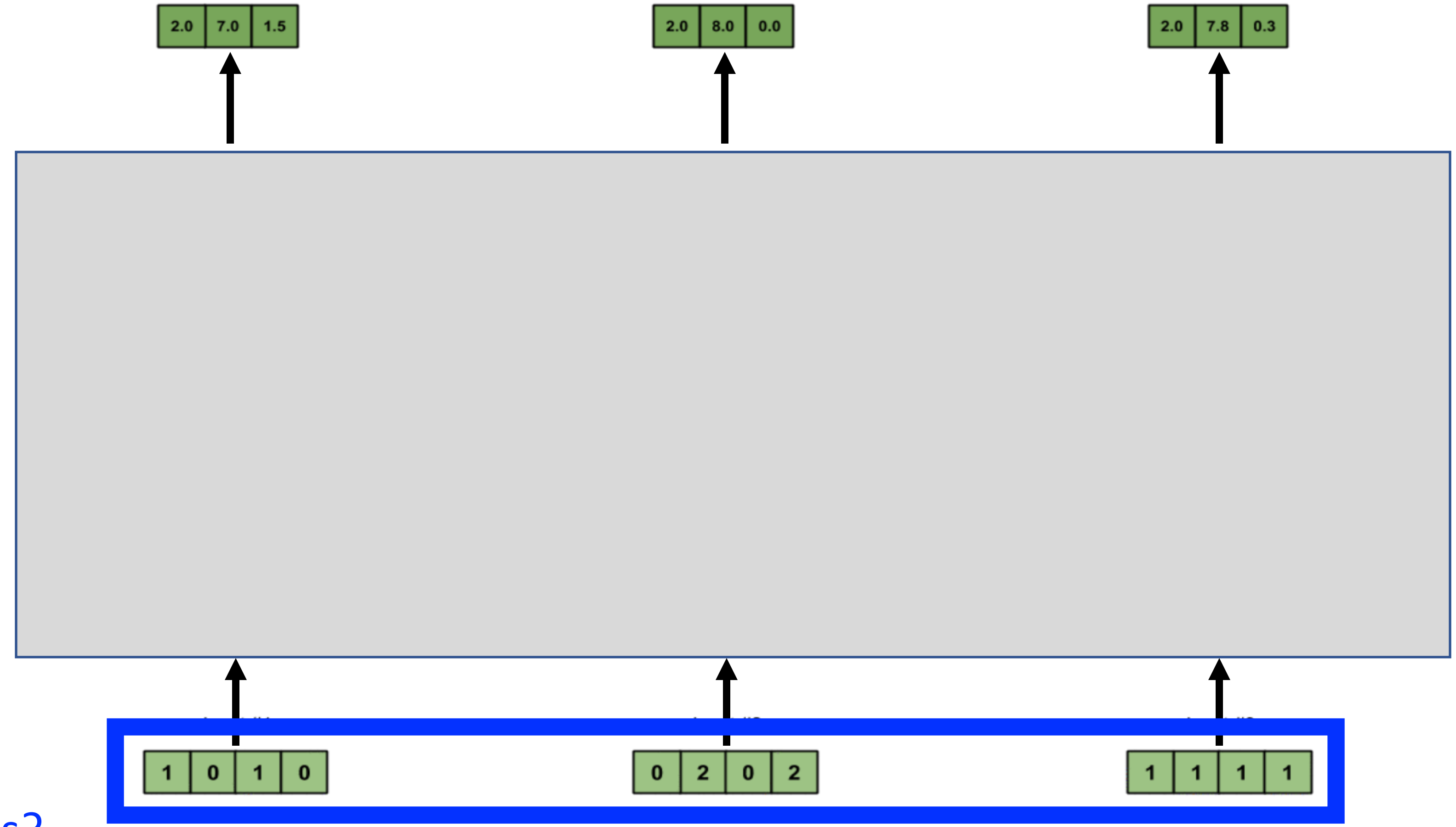
New representation of each input token to reflect each one's relationship to all tokens



Input tokens

https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

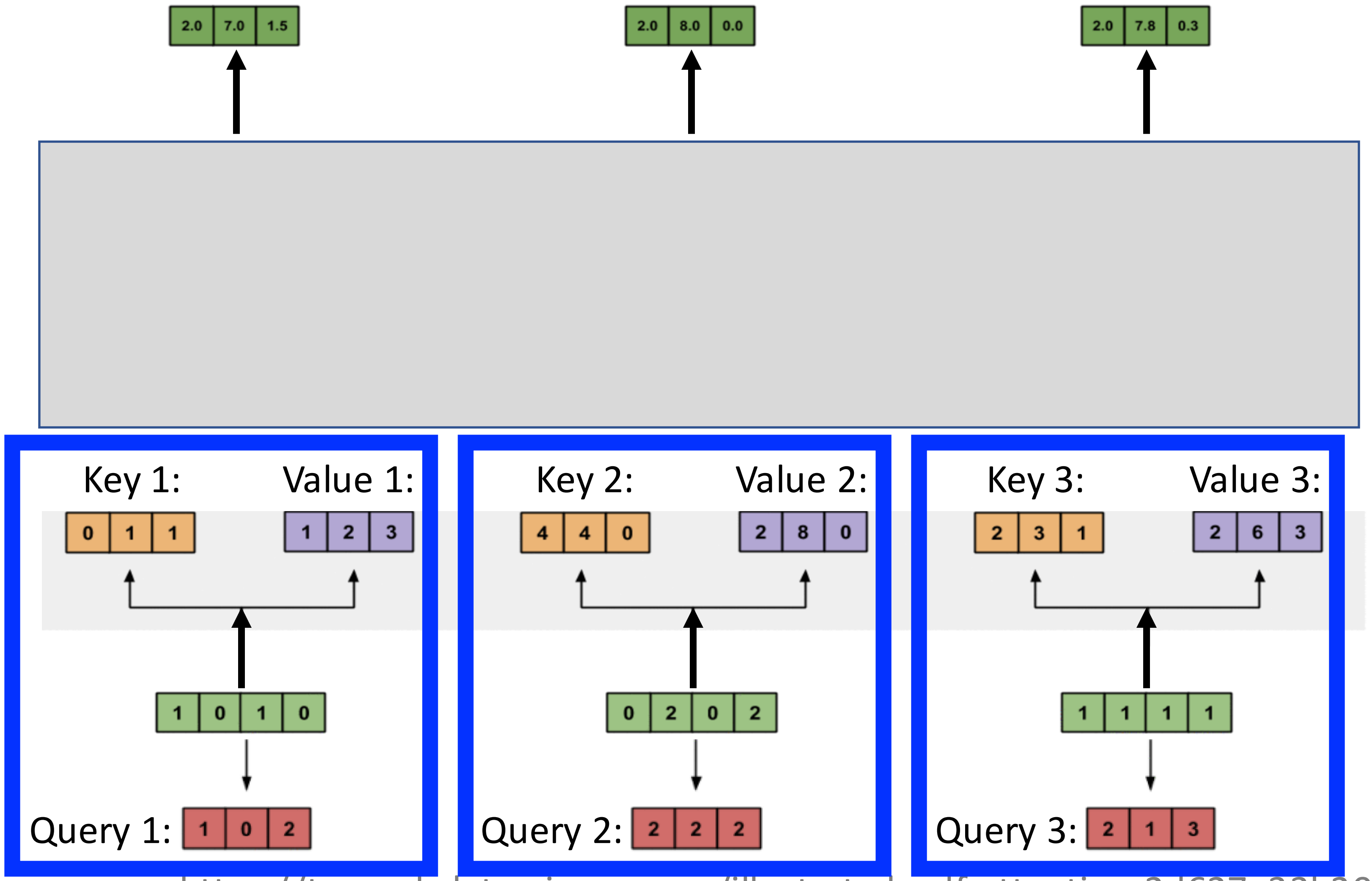
Computing Self-Attention: Example



- How many input tokens are there?
- What is each token’s dimensionality?
- How to support arbitrary length inputs?
    * Input length is a hyperparameter: pad shorter sequences with zeros and truncate longer sequences

https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example



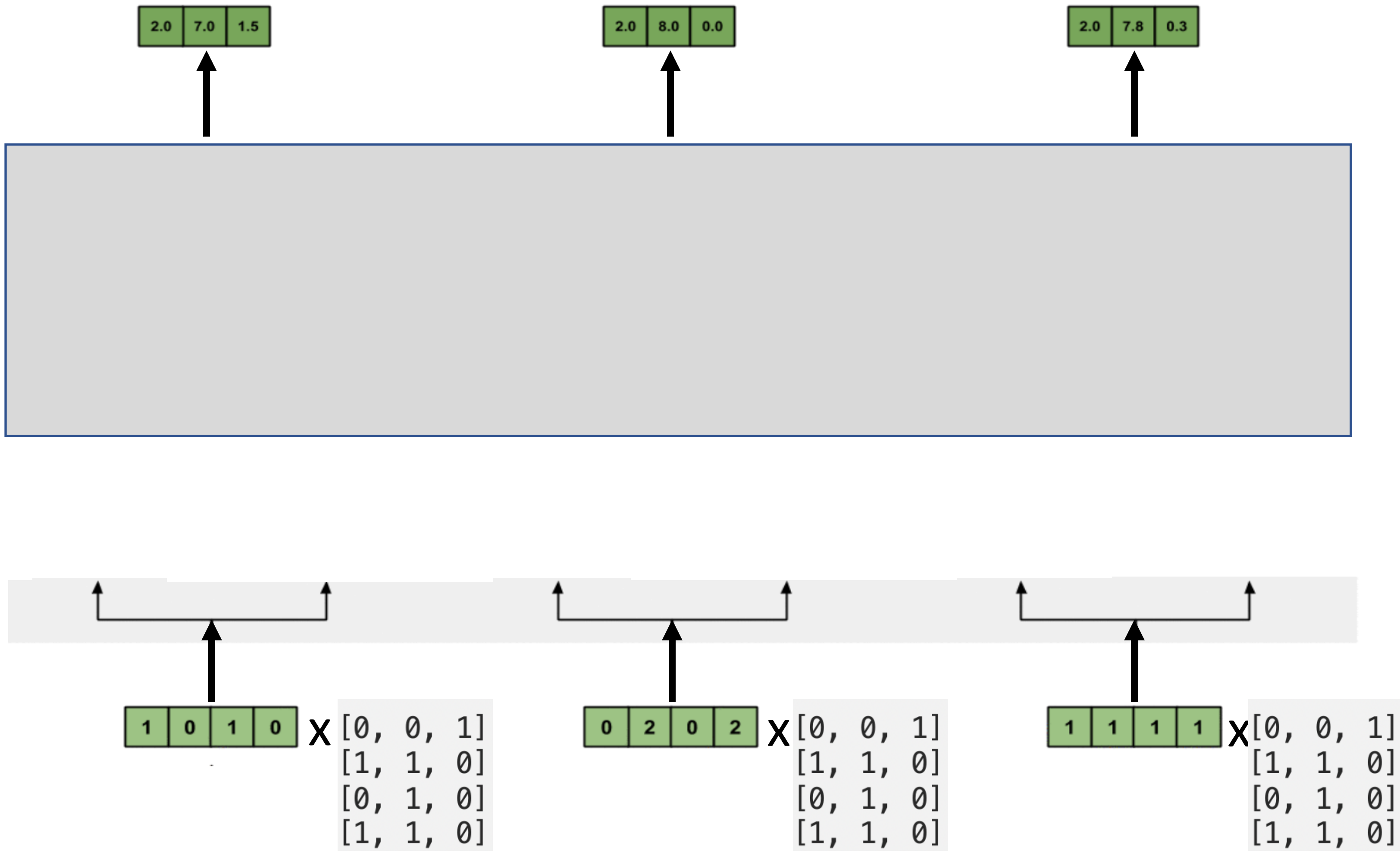
Three vectors are derived for each input by multiplying with three weight matrices (learned during training): query, key, and value

https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

e.g., key weights

```
[0, 0, 1]
[1, 1, 0]
[0, 1, 0]
[1, 1, 0]
```

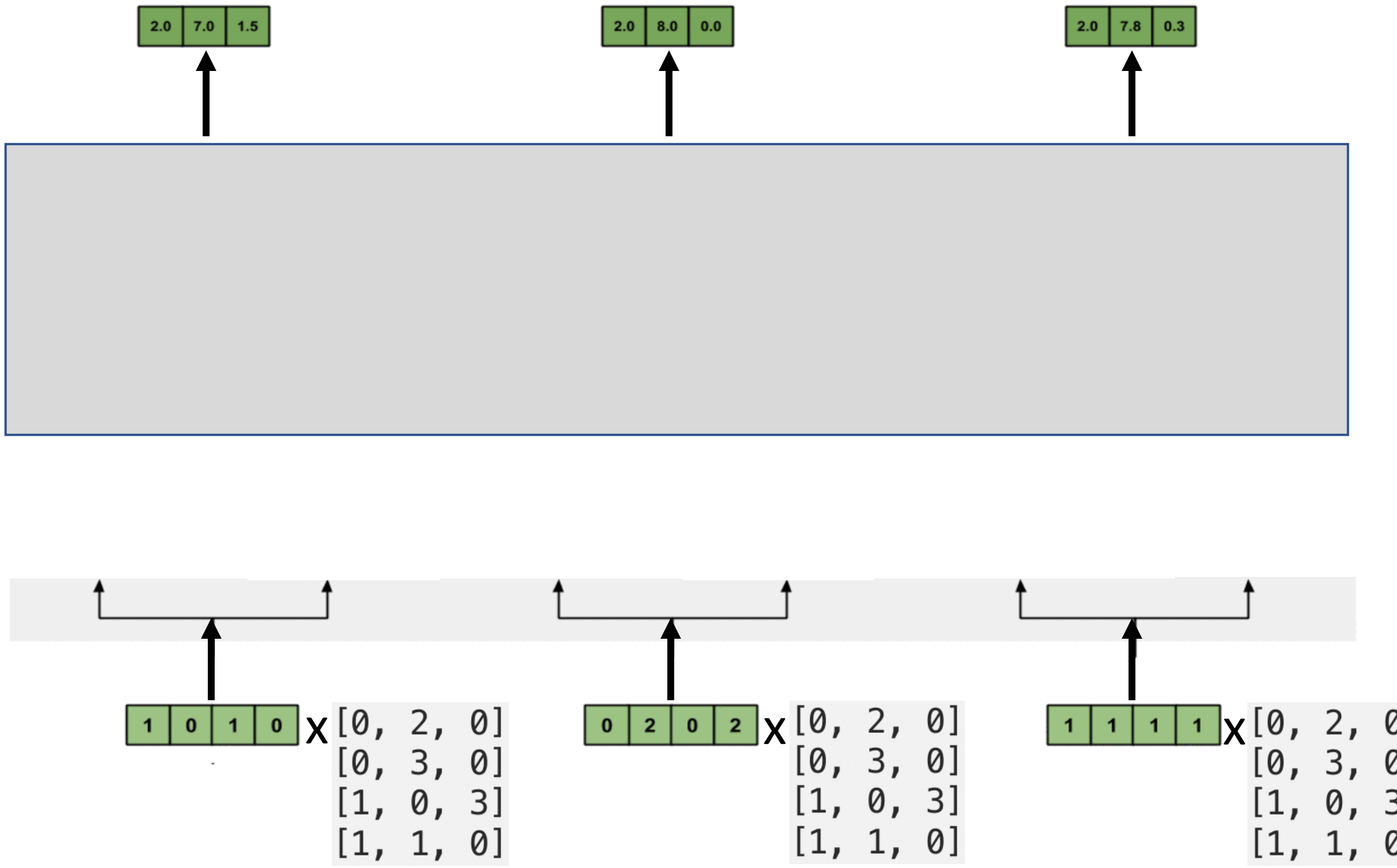


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

e.g., value weights

```
[0, 2, 0]
[0, 3, 0]
[1, 0, 3]
[1, 1, 0]
```

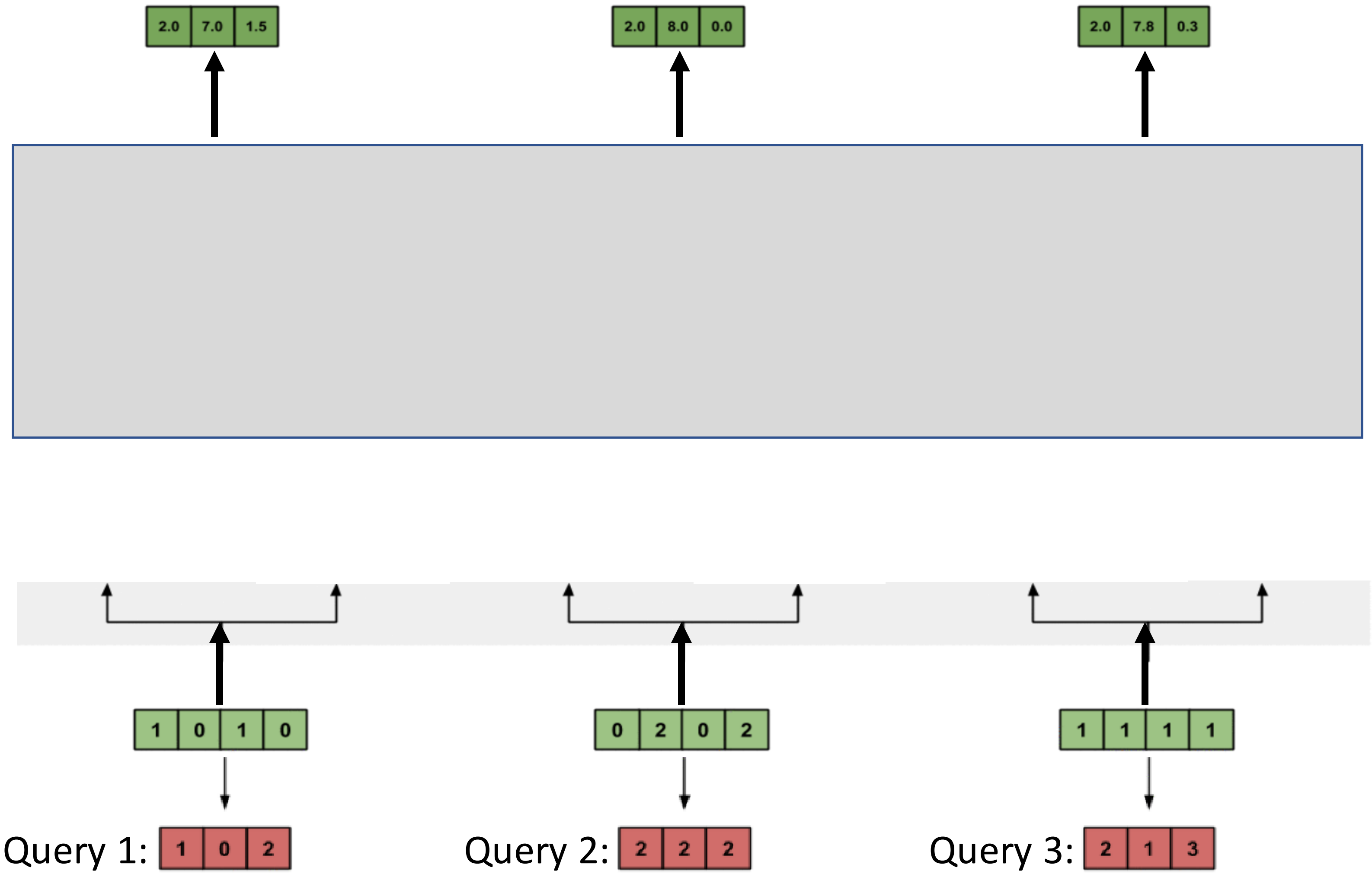


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

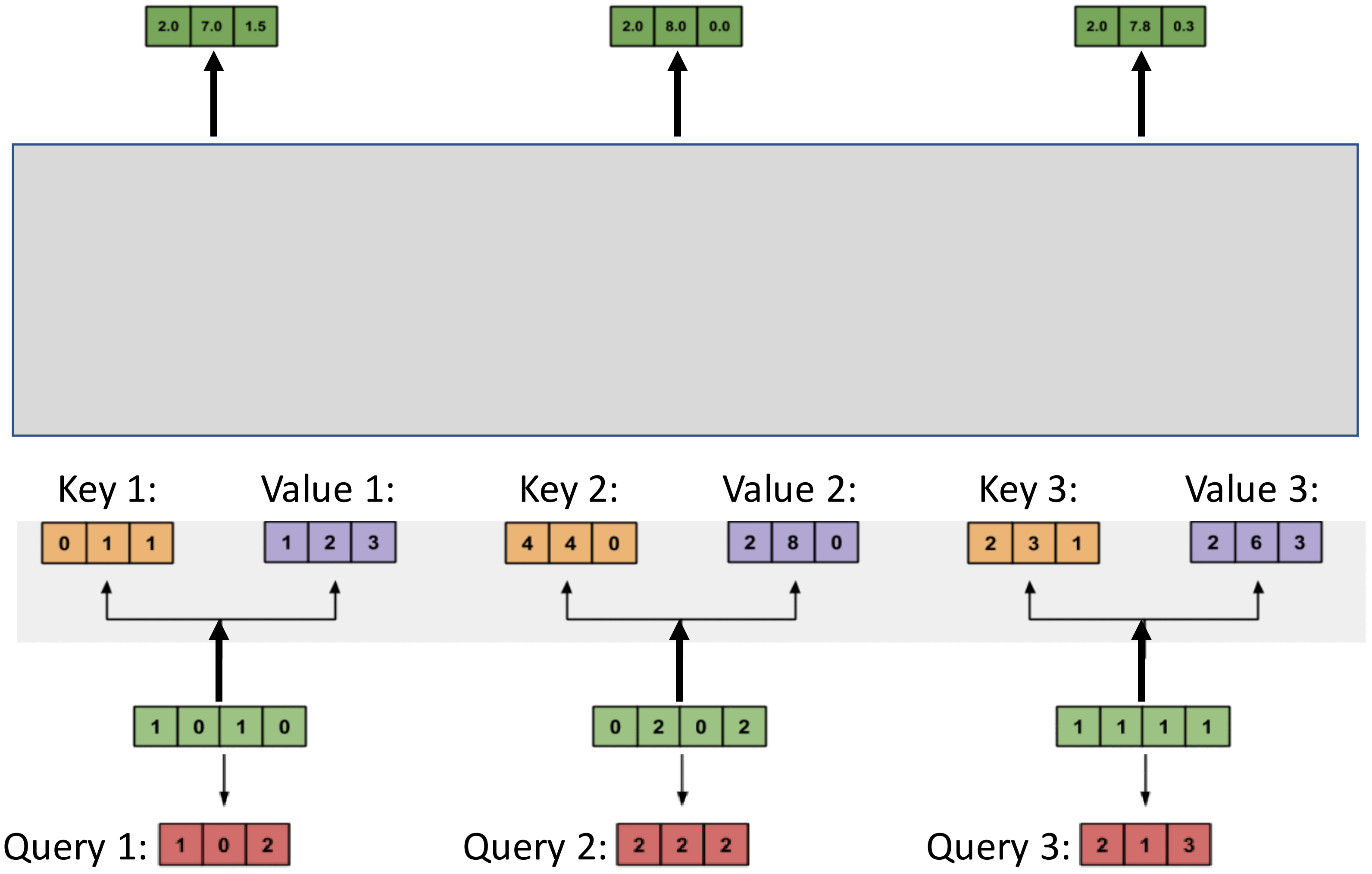
e.g., query weights

```
[1, 0, 1]
[1, 0, 0]
[0, 0, 1]
[0, 1, 1]
```



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example



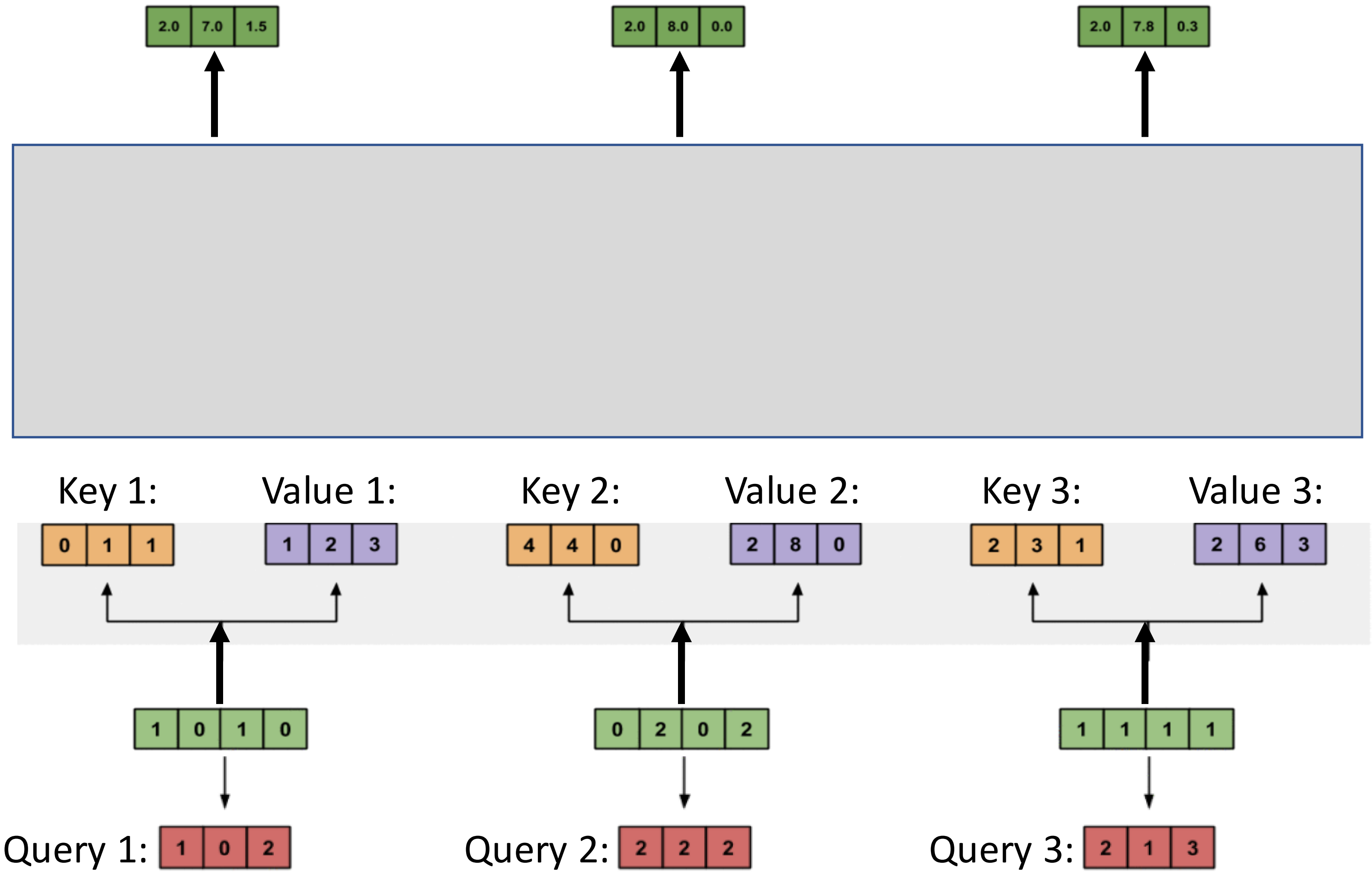
How many weight matrices are learned in this example?

https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

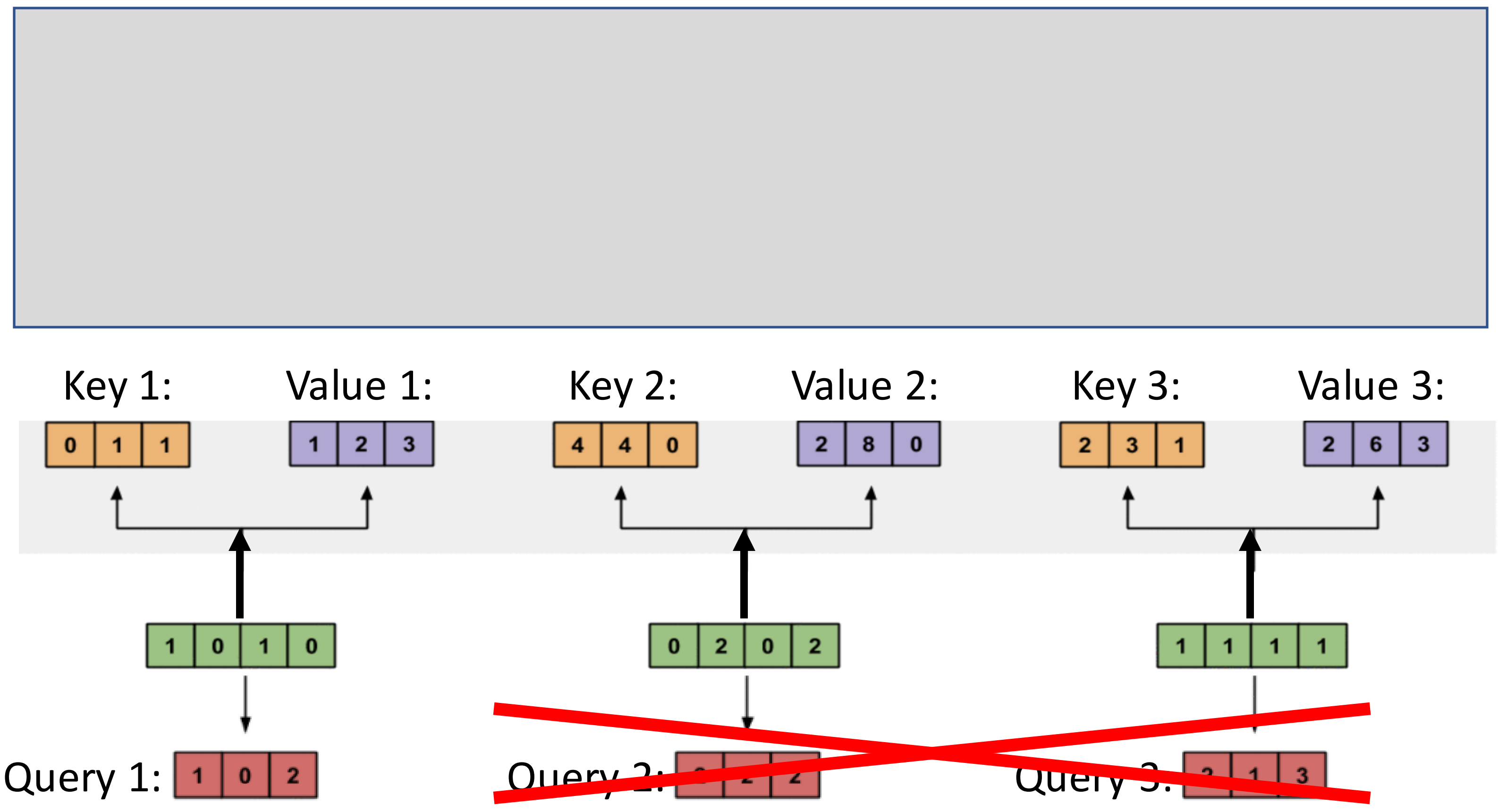
What is the purpose of the three weight matrices?

For each input, 2 of the derived vectors are used to compute **attention weights** (query and key) and the 3rd is **information** passed on for the new representation (value)



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example



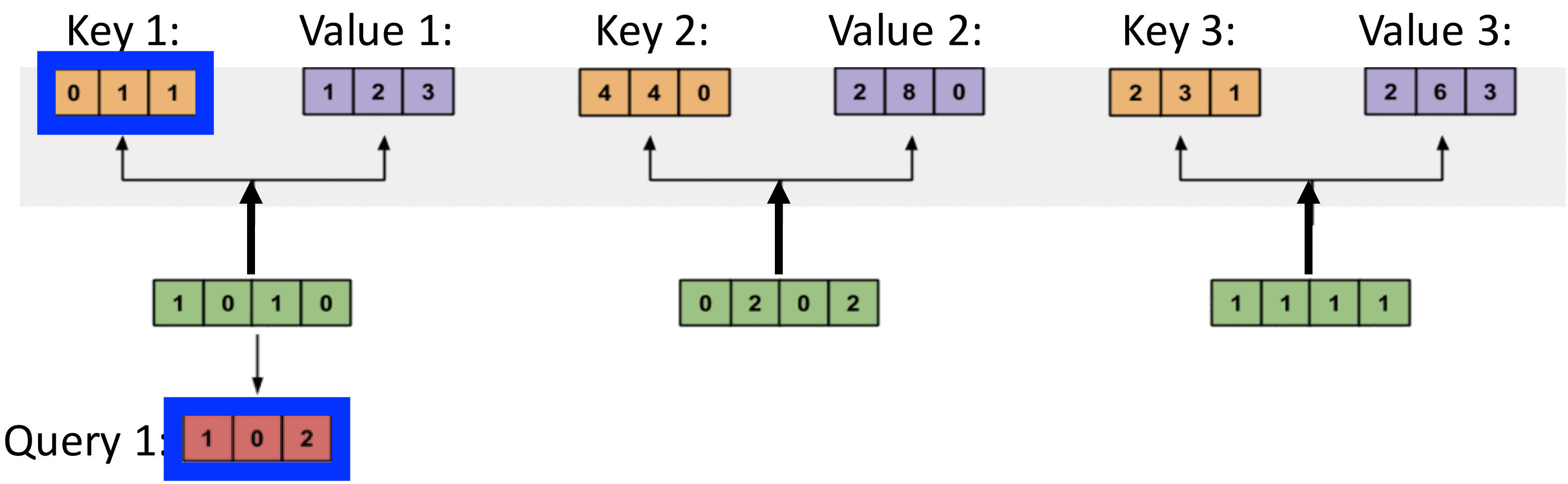
Let’s compute the new representation for the inputs…

https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Attention score: dot product of query ("what am I looking for") with all keys ("what I have") to identify relevant tokens (higher scores are better matches); e.g.,

$$\begin{bmatrix} 1 & 0 & 2 \end{bmatrix} \times \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = ?$$

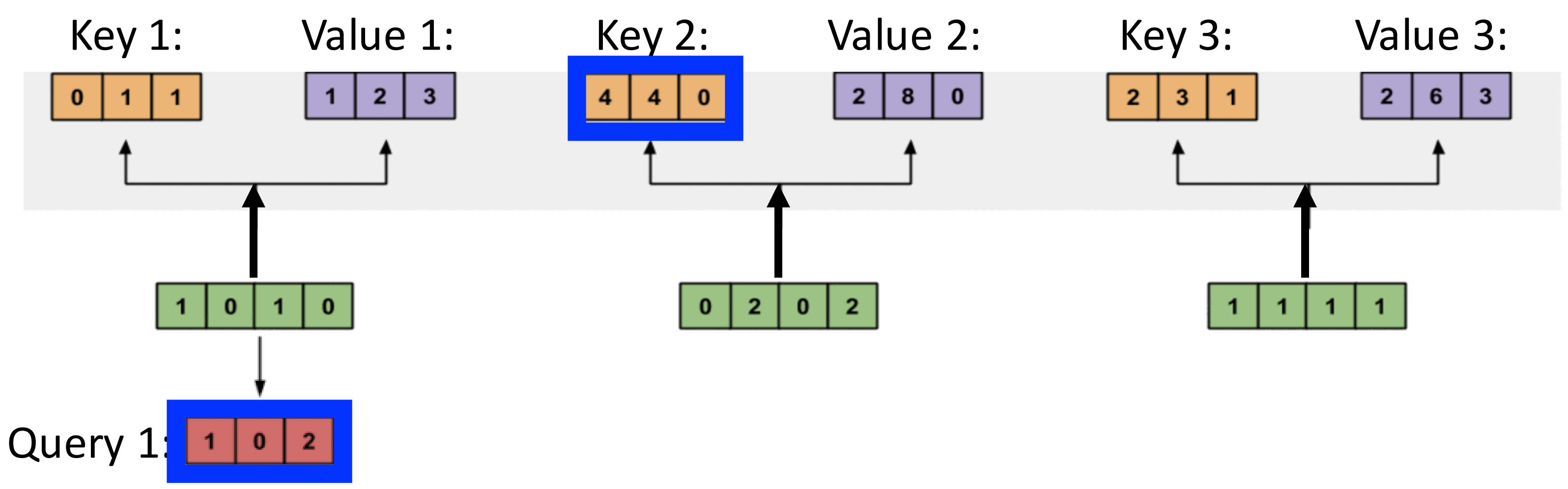


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Attention score: dot product of query ("what am I looking for") with all keys ("what I have") to identify relevant tokens (higher scores are better matches); e.g.,

$$\begin{bmatrix} 1 & 0 & 2 \end{bmatrix} \times \begin{bmatrix} 4 \\ 4 \\ 0 \end{bmatrix} = ?$$

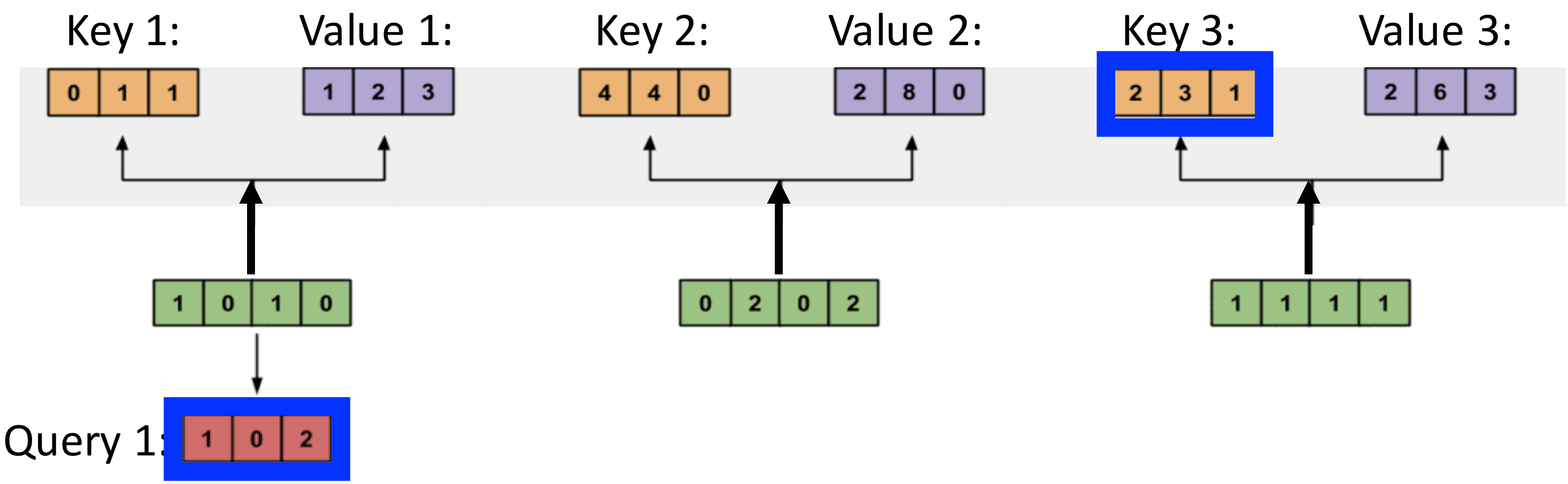


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Attention score: dot product of query ("what am I looking for") with all keys ("what I have") to identify relevant tokens (higher scores are better matches); e.g.,

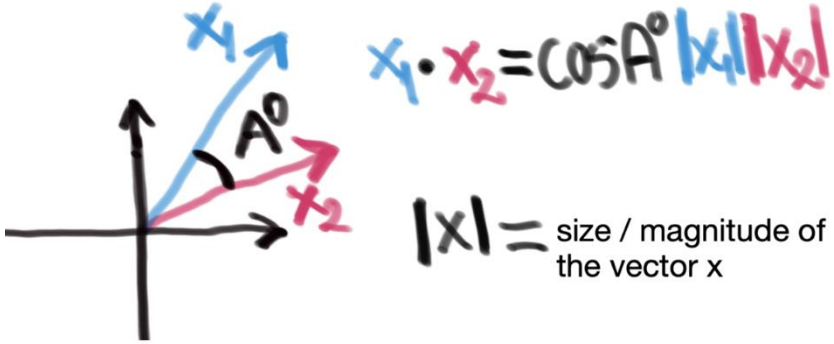
$$\begin{bmatrix} 1 & 0 & 2 \end{bmatrix} \times \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} = ?$$



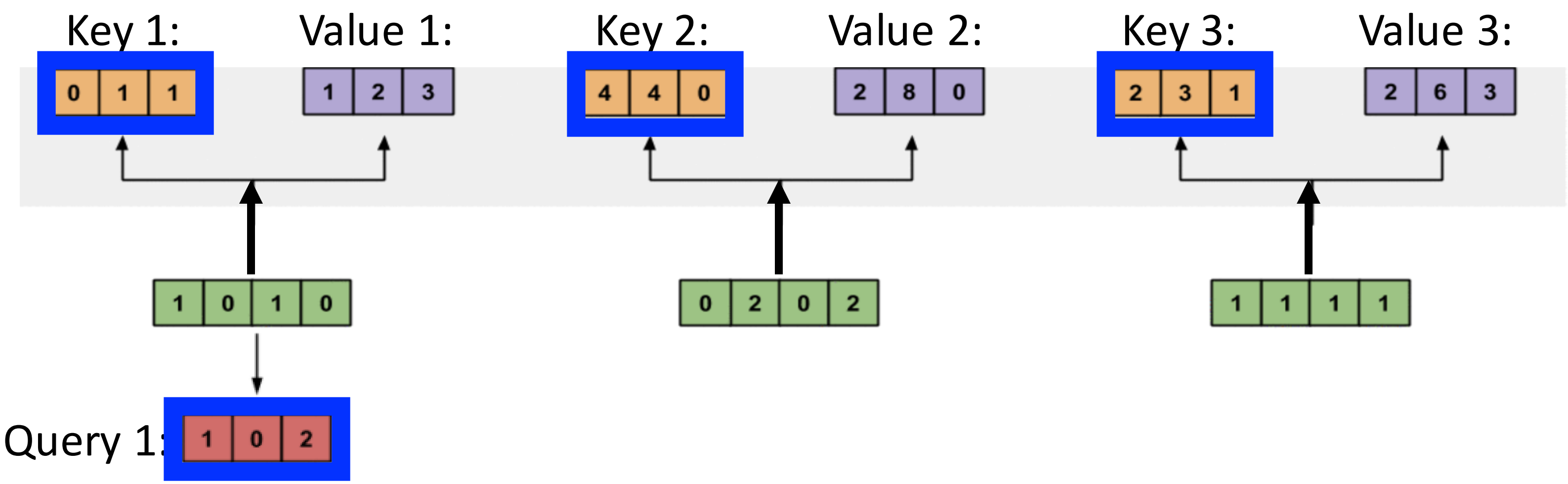
https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Why dot product? Indicates similarity of two vectors
- Match = 1 (i.e., cos(0))
- Opposites = -1 (i.e., cos(180))



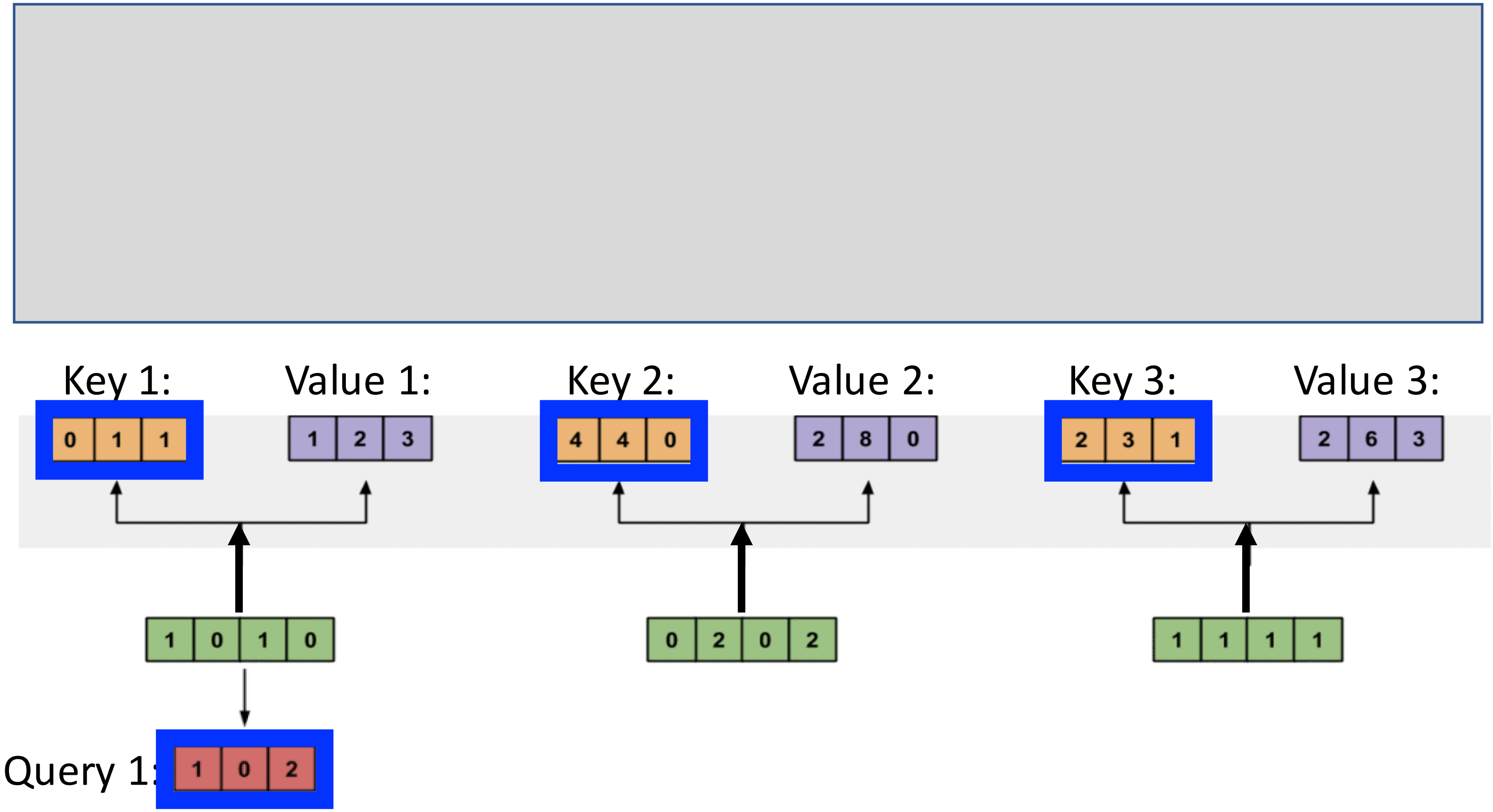
https://towardsdatascience.com/self-attention-5b95ea164f61



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Note: there are many similarity measures



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Attention weights: softmax scores for all inputs to quantify each token's relevance; e.g.,

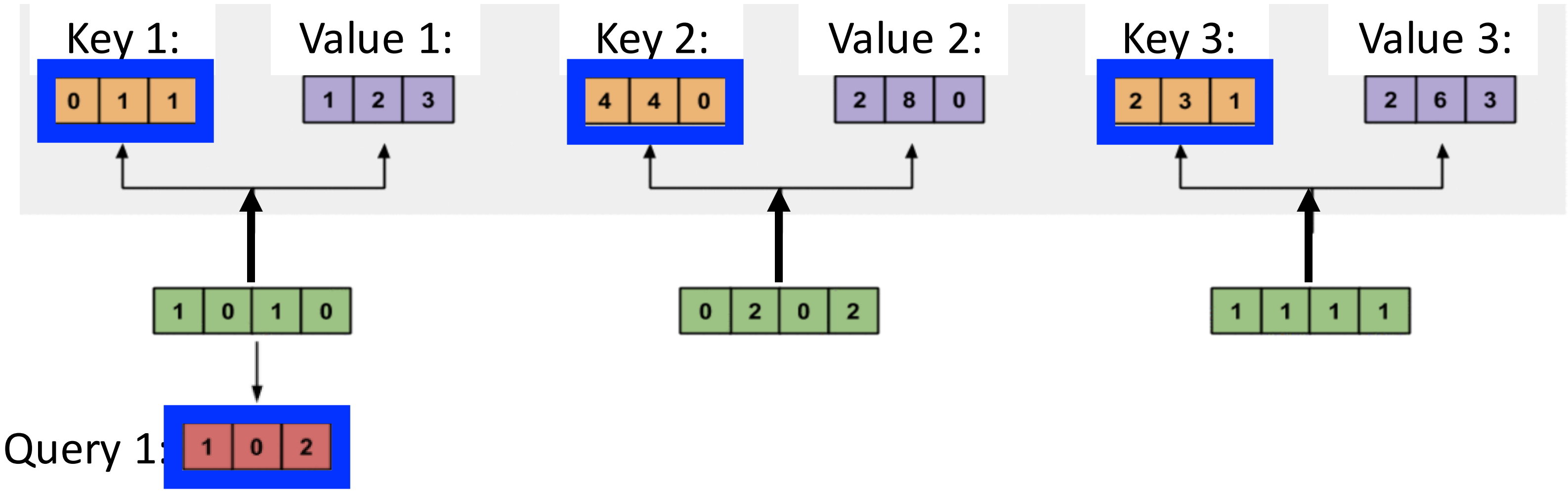
= softmax([2, 4, 4])

= [0.0, 0.5, 0.5])

Note: 0 from softmax can arise from rounding

To which input(s) is input 1 **least** related?

To which input(s) is input 1 **most** related?

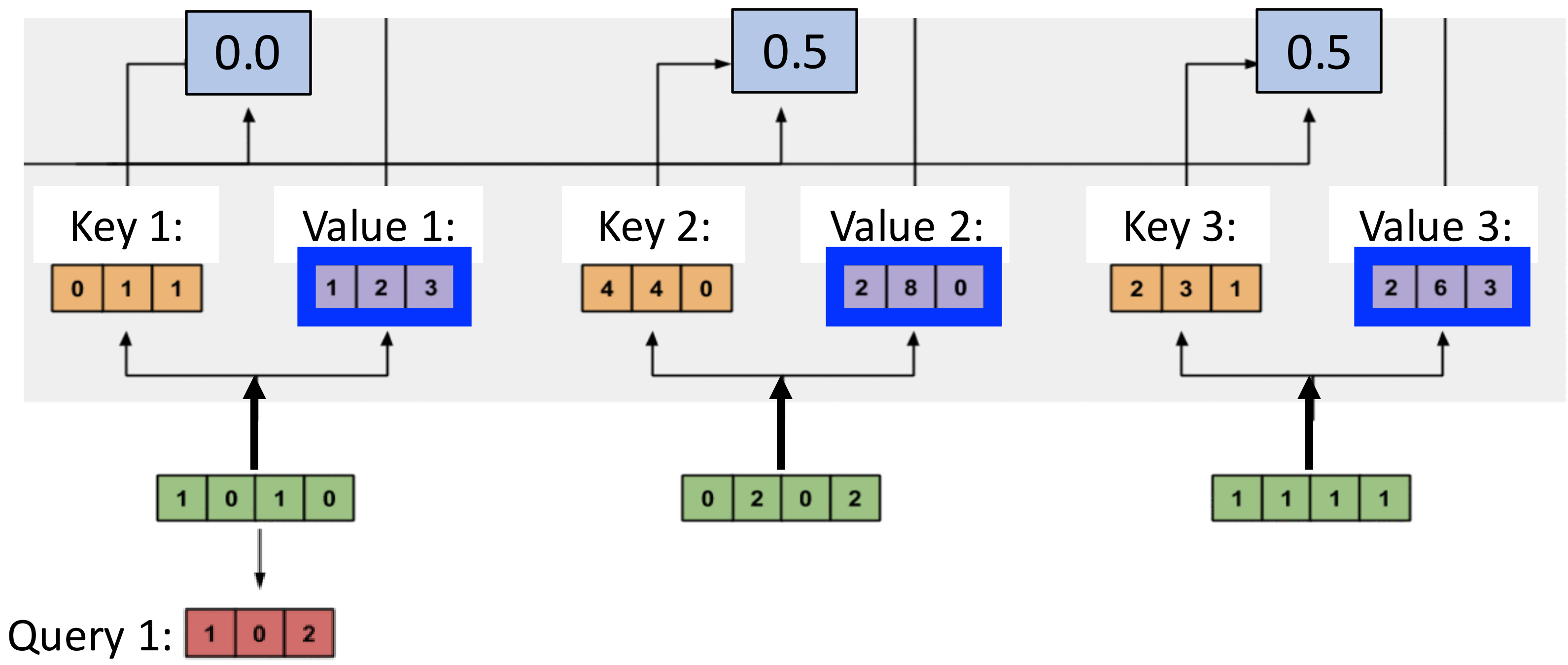


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Compute new representation of input token that reflects entire input:

1. Attention weights x Values



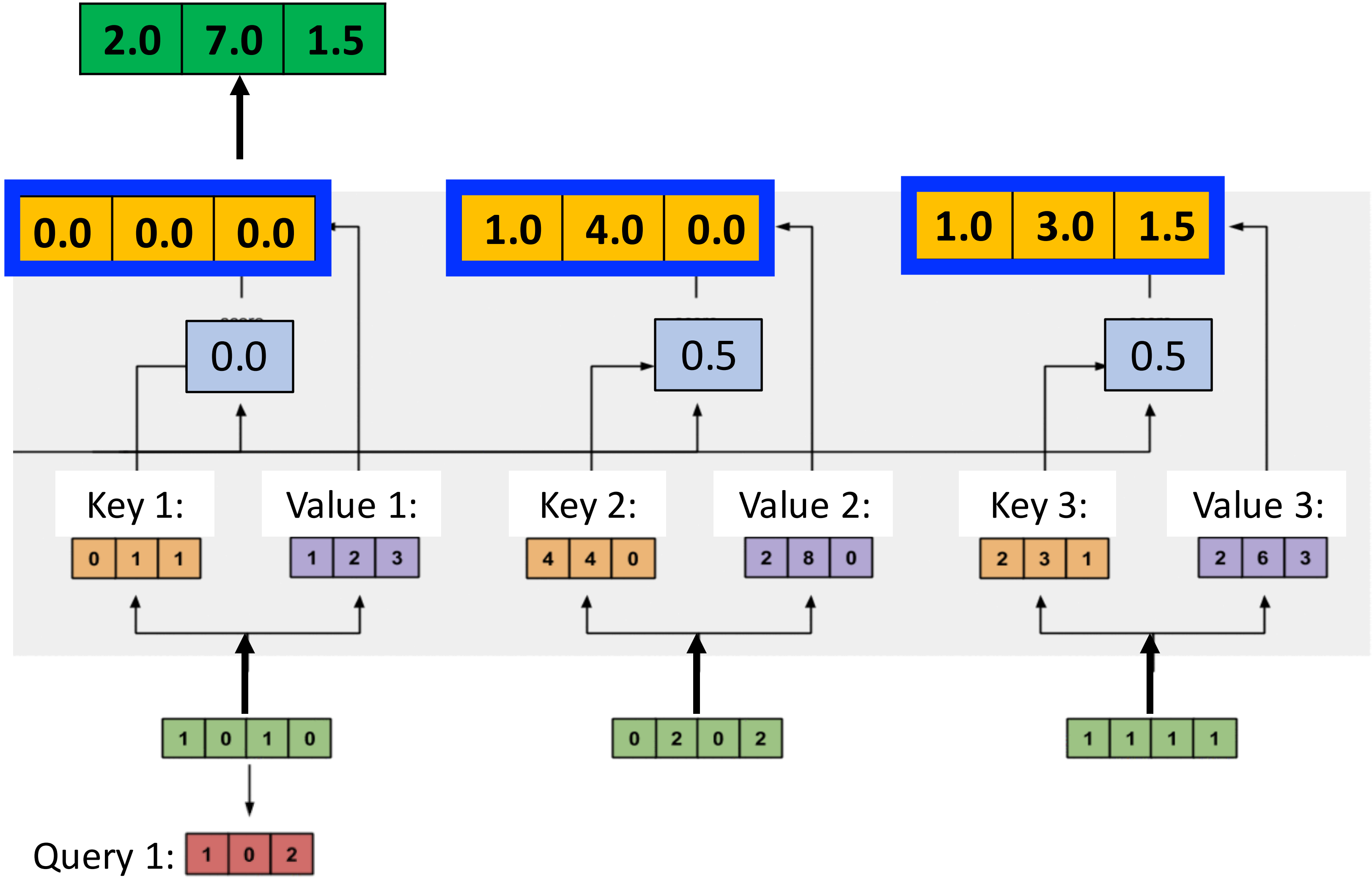
https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Compute new representation of input token that reflects entire input:

1. Attention weights x Values

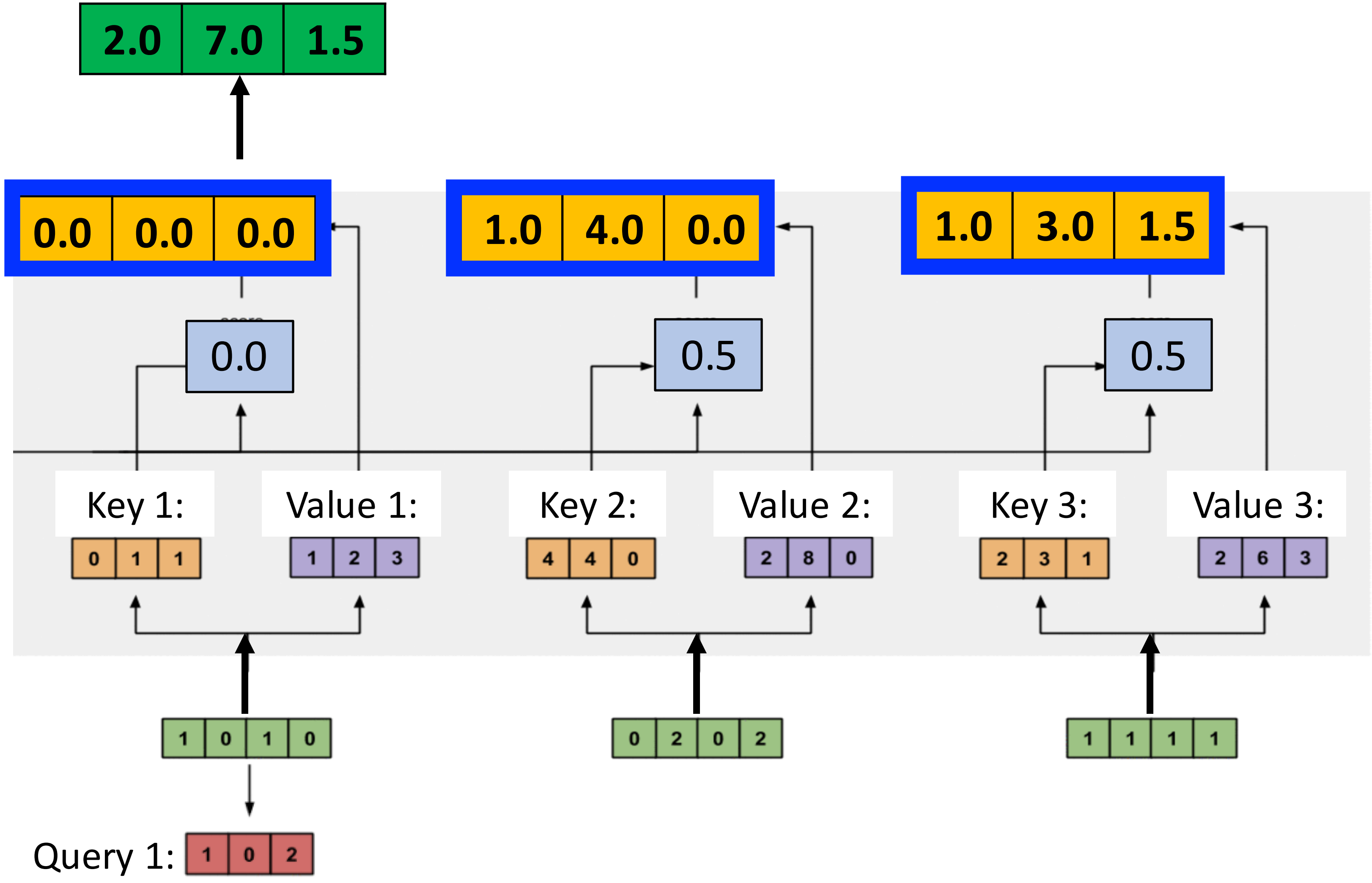
2. Sum all weighted vectors



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

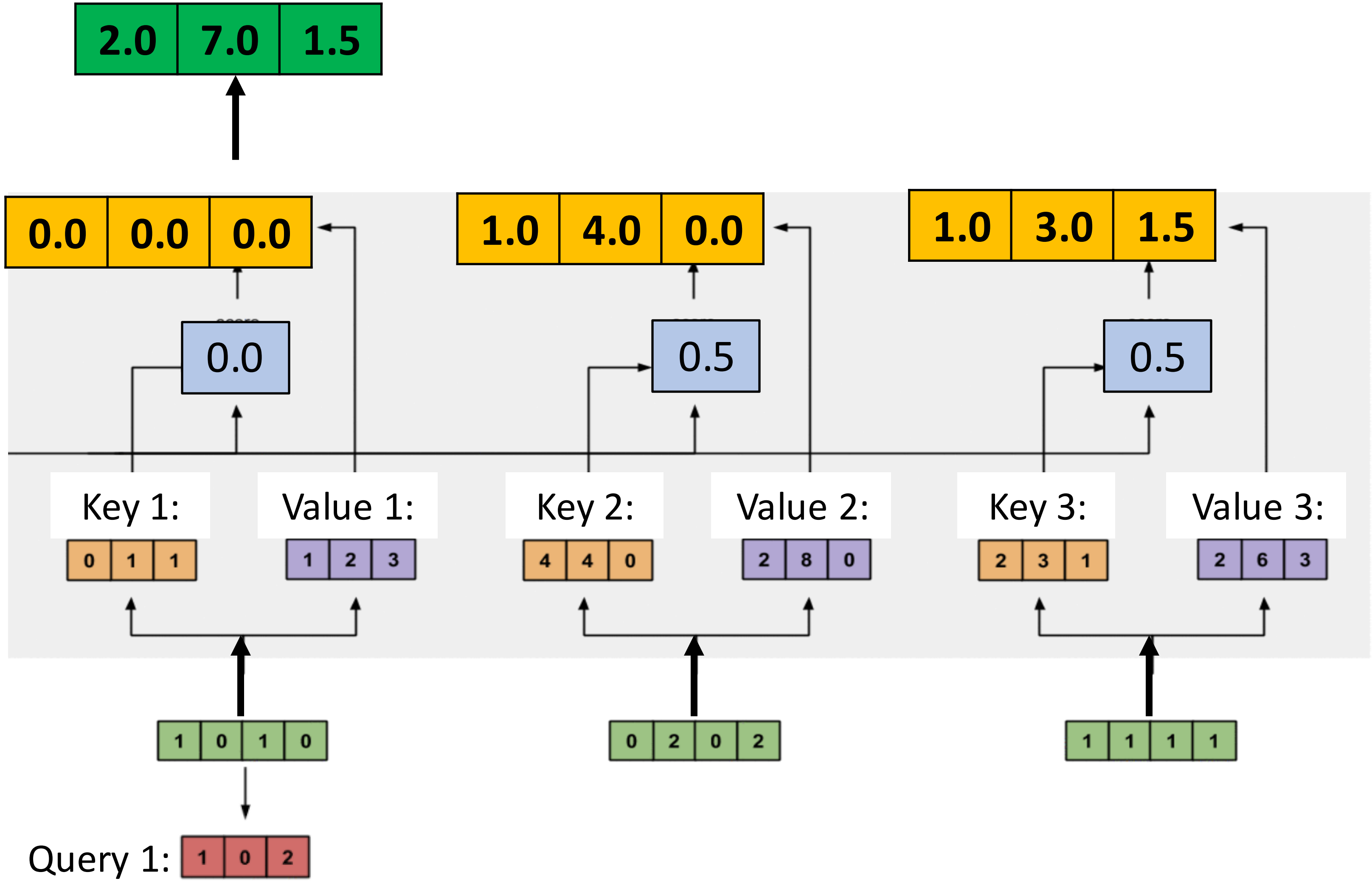
Attention weights amplify input representations (values) that we want to pay attention to and repress the rest



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

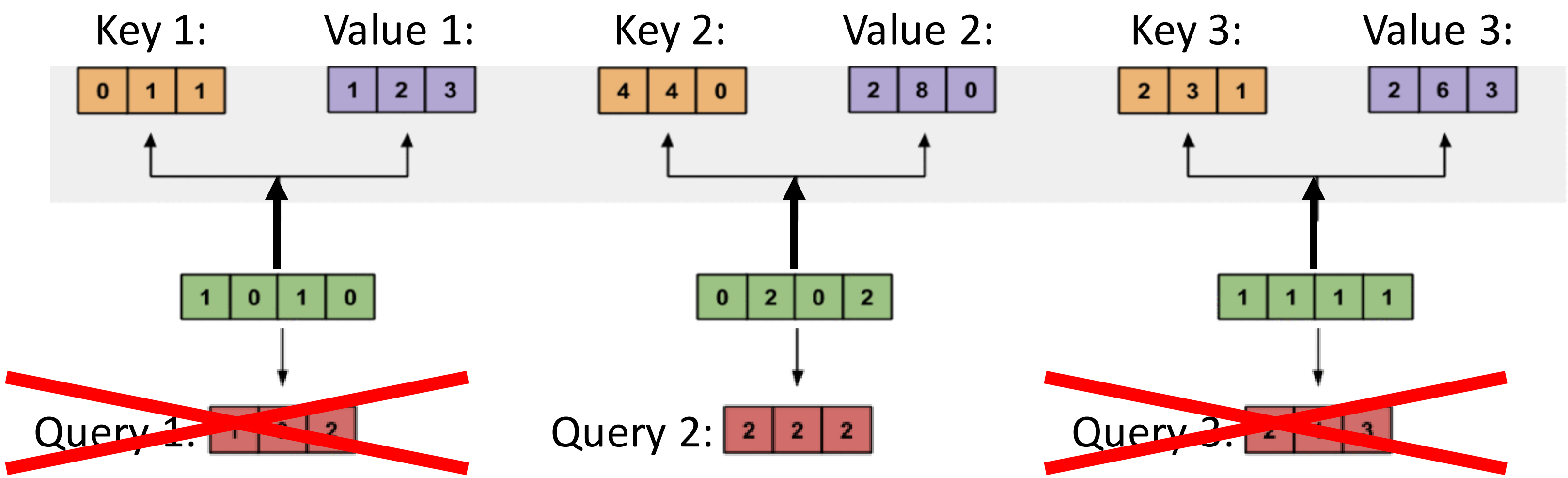
Attention weights amplify input representations (values) that we want to pay attention to and repress the rest



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Repeat the same process for each remaining input token

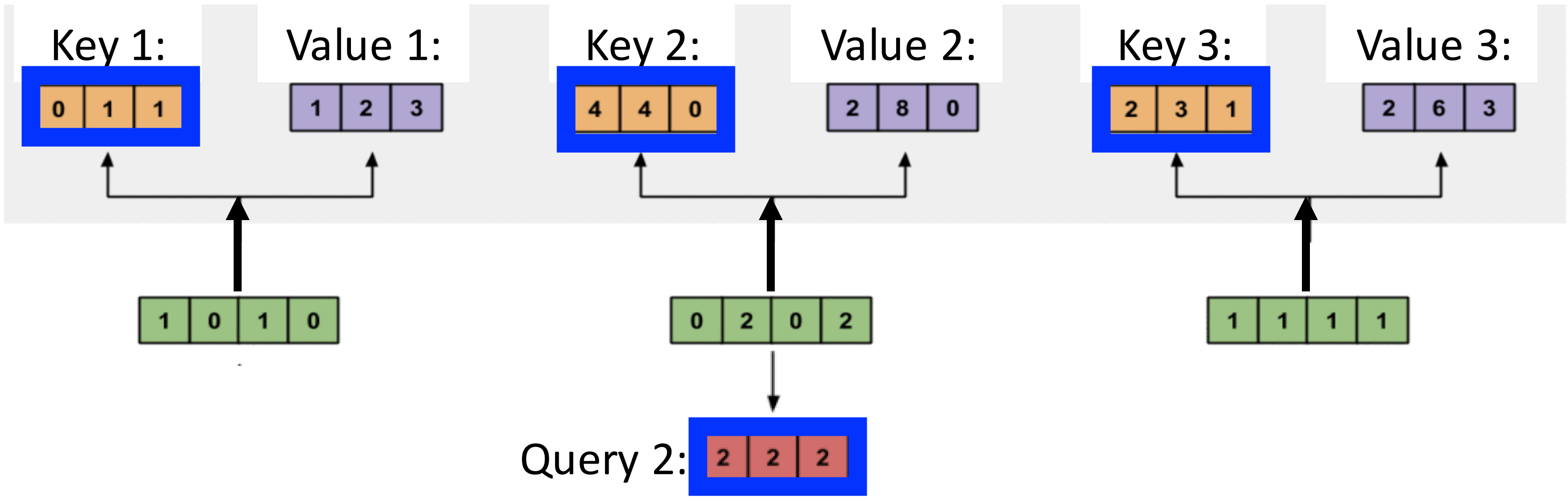


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

1. Compute attention weights
- Softmax resulting 3 scores from query x keys

To which input(s) is input 2 most related?

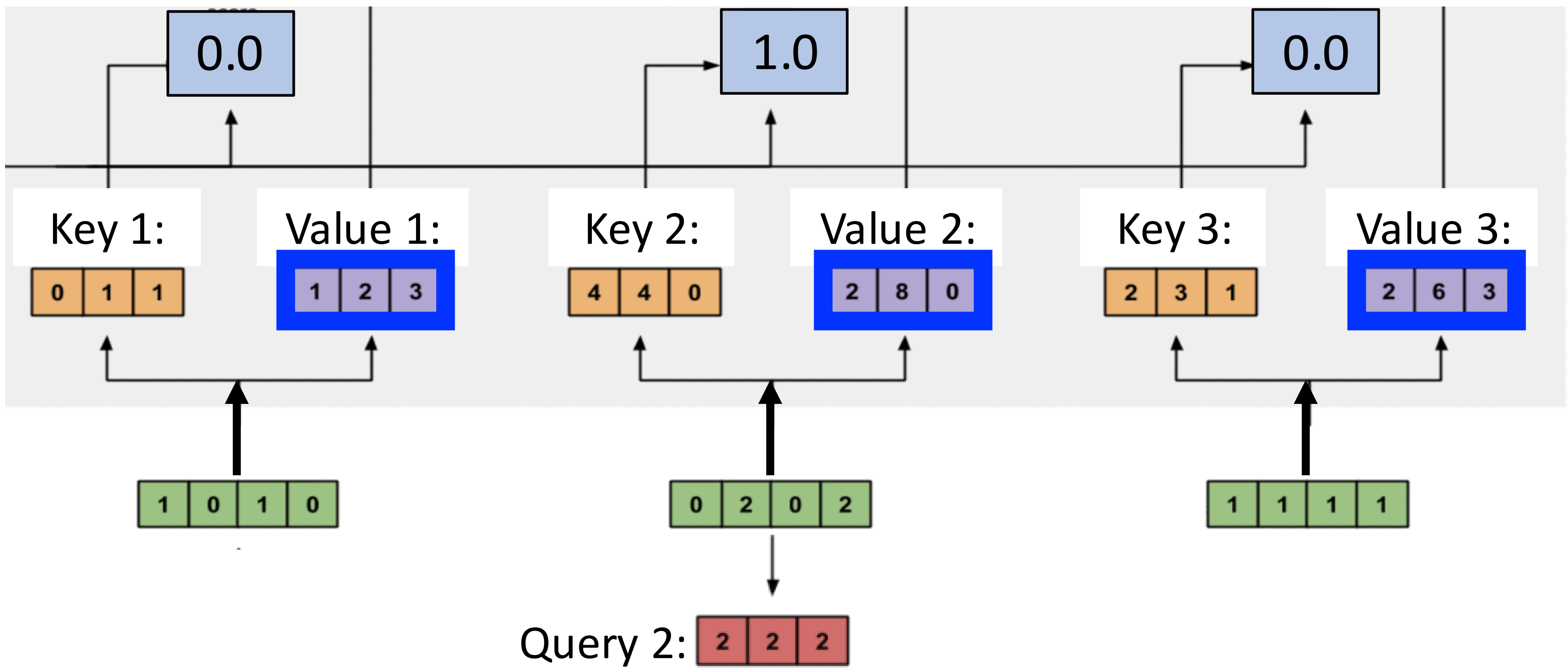


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

1. Compute attention weights
- Softmax resulting 3 scores from query x keys

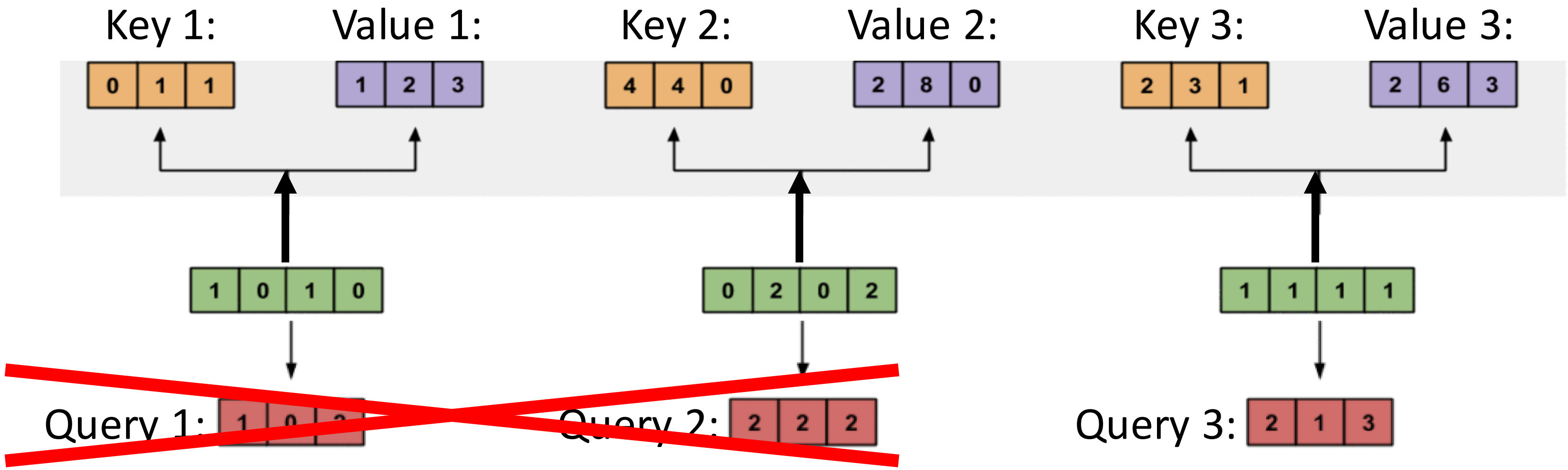
2. Compute weighted sum of values using attention scores



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

Repeat the same process for each remaining input token

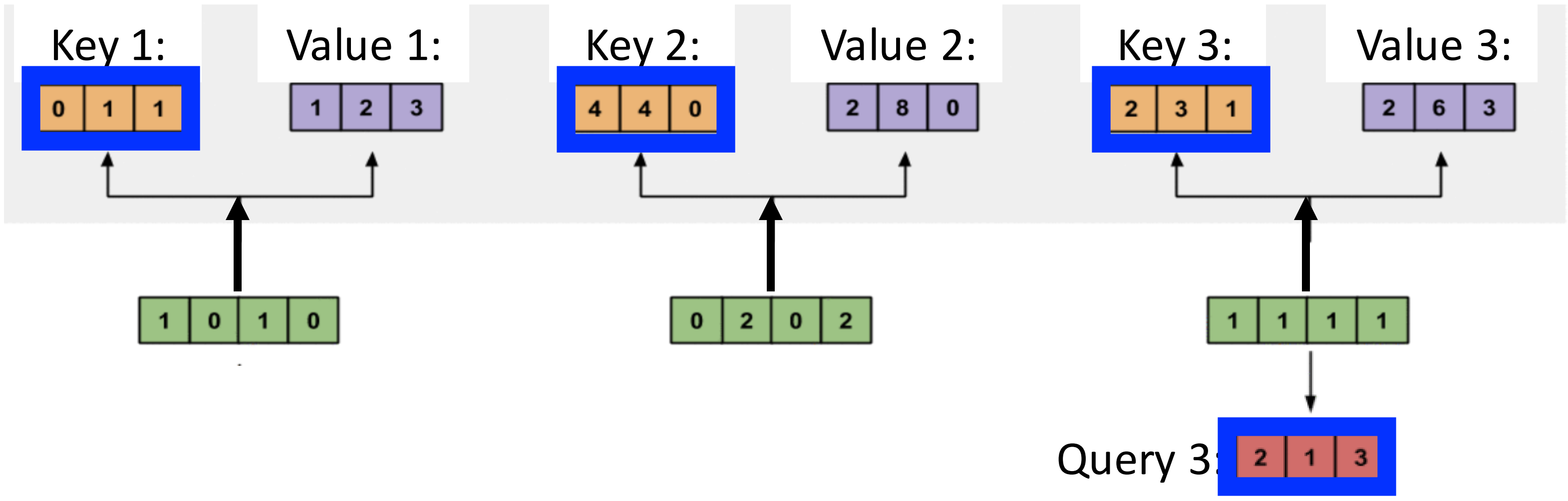


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

1. Compute attention weights
- Softmax resulting 3 scores from query x keys

To which input(s) is input 3 most related?

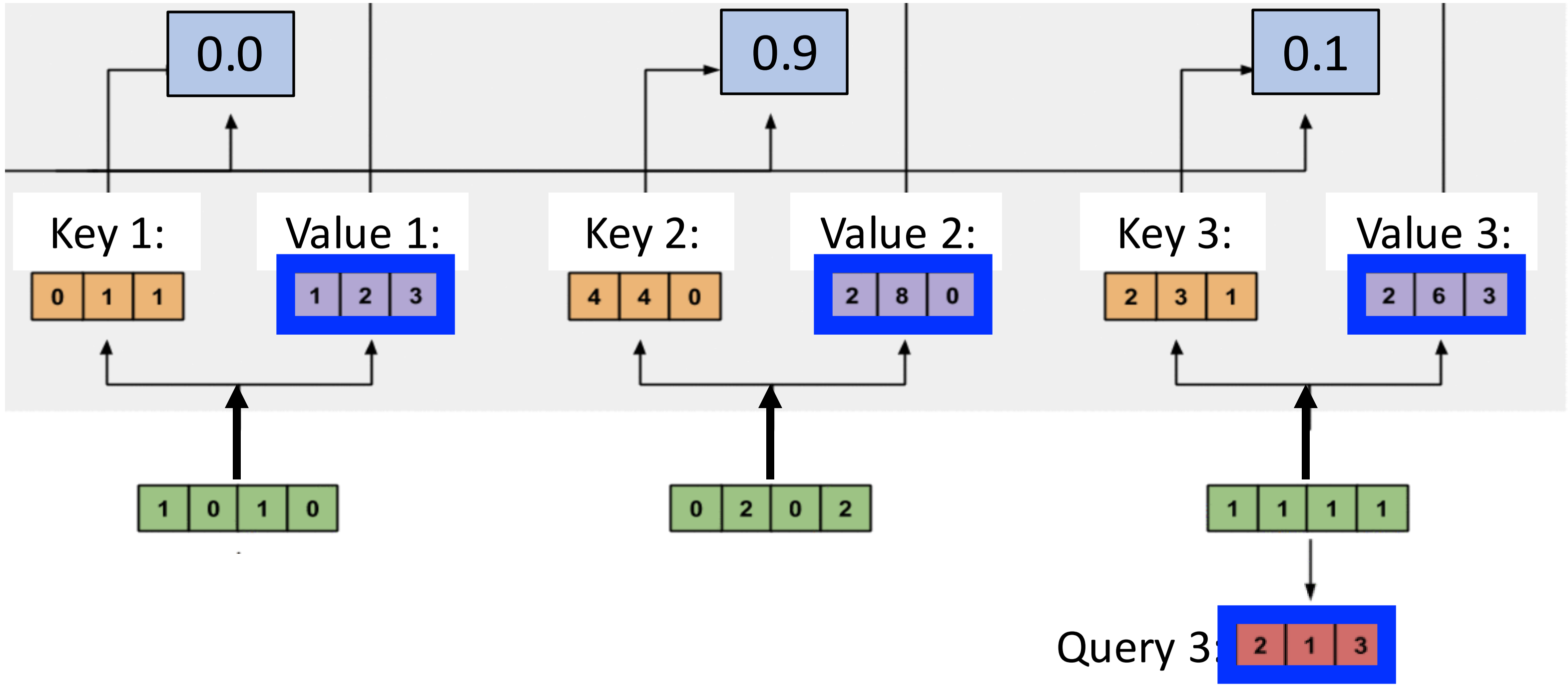


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example

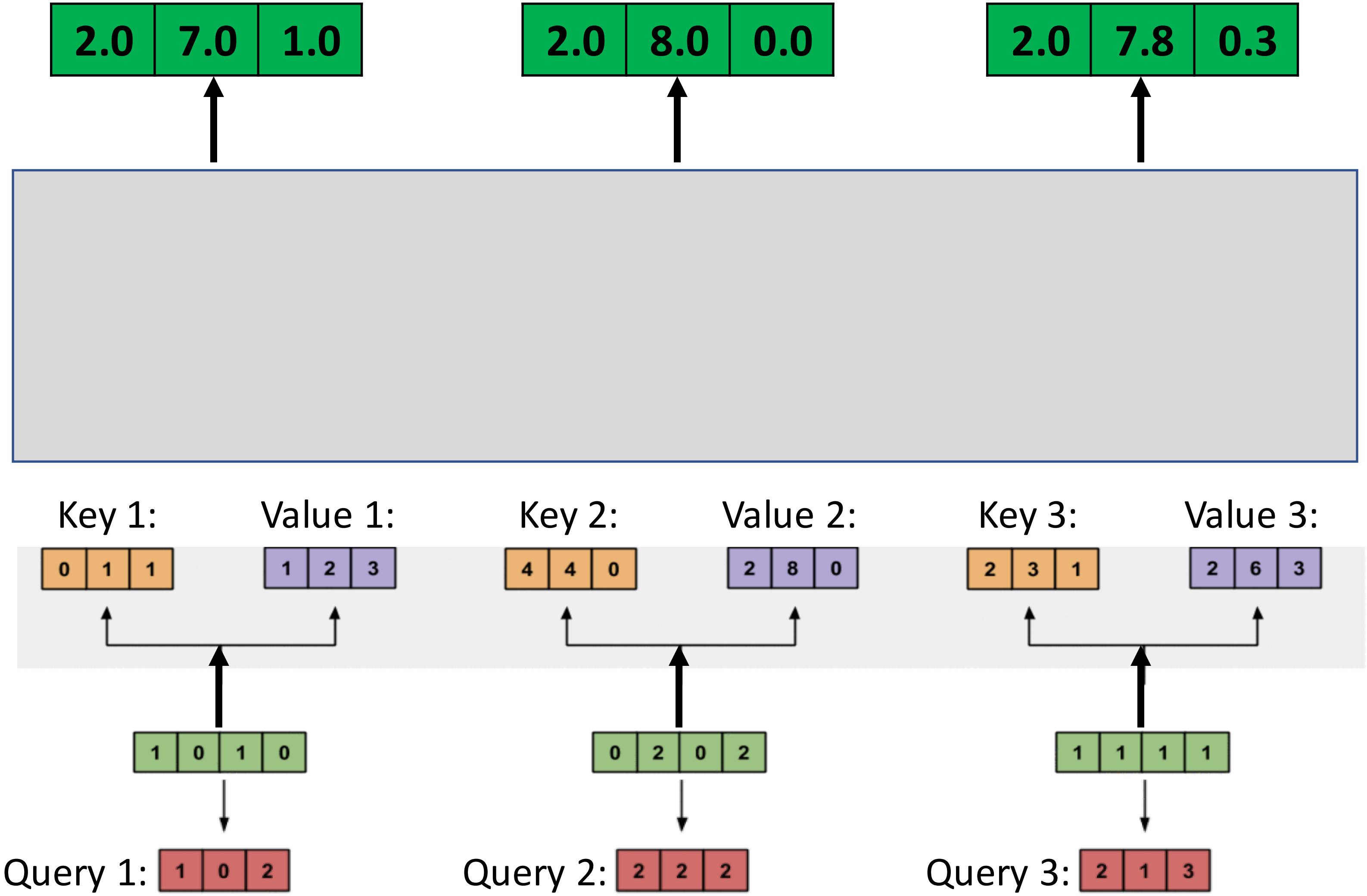
1. Compute attention weights
- Softmax resulting 3 scores from query x keys

2. Compute weighted sum of values using attention scores



https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Computing Self-Attention: Example



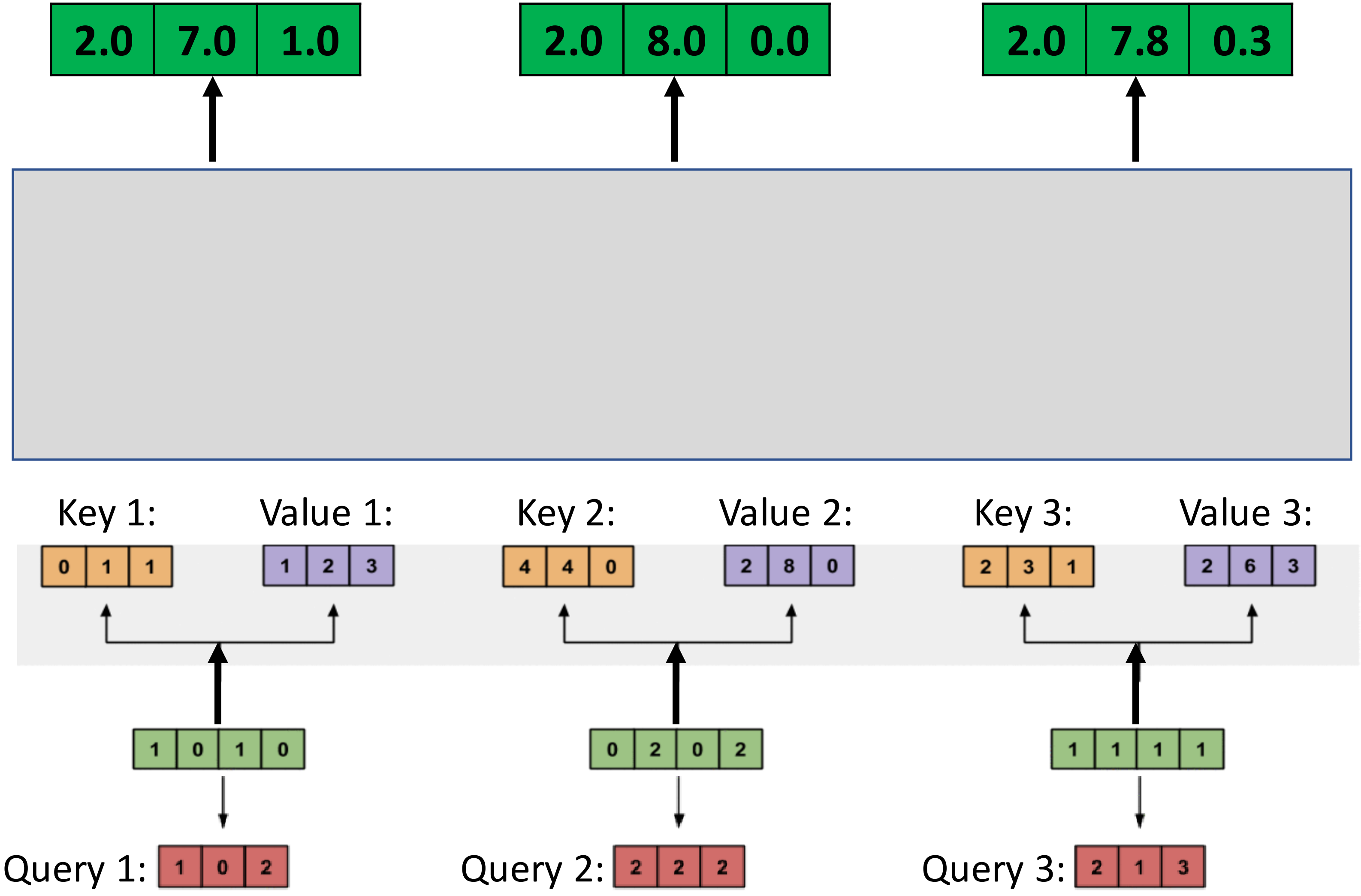
https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Hyperparameters

A developer chooses **input token length** and **number of matrices' columns** (and thus vector sizes)

Dimension of **query** and **key** must match to assess similarity (e.g., dot product).

Dimension of **value** can differ from that of **query** and **key** and is output dimension.

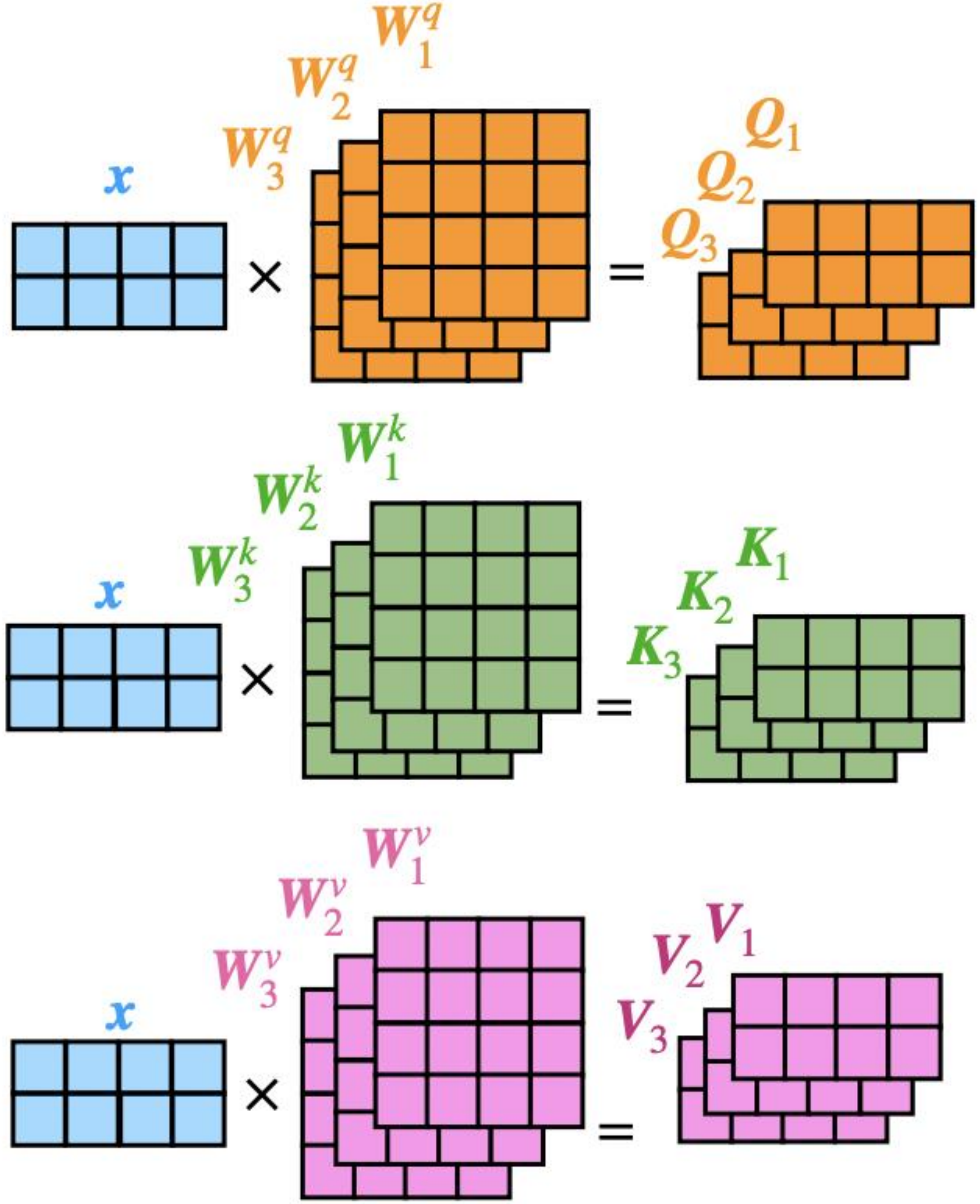


https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Multi-head Attention

- **Idea**: enable each token to relate to other tokens in multiple ways

- **Approach**: multiple self-attention heads, each with their own key, value and query matrices

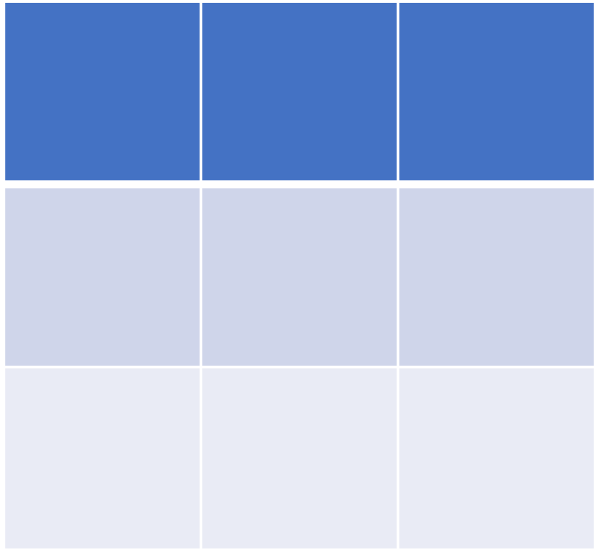


https://sebastianraschka.com/pdf/lecture-notes/stat453ss21/L19_seq2seq_rnn-transformers__slides.pdf

Problem: Self-Attention's Computational Expense

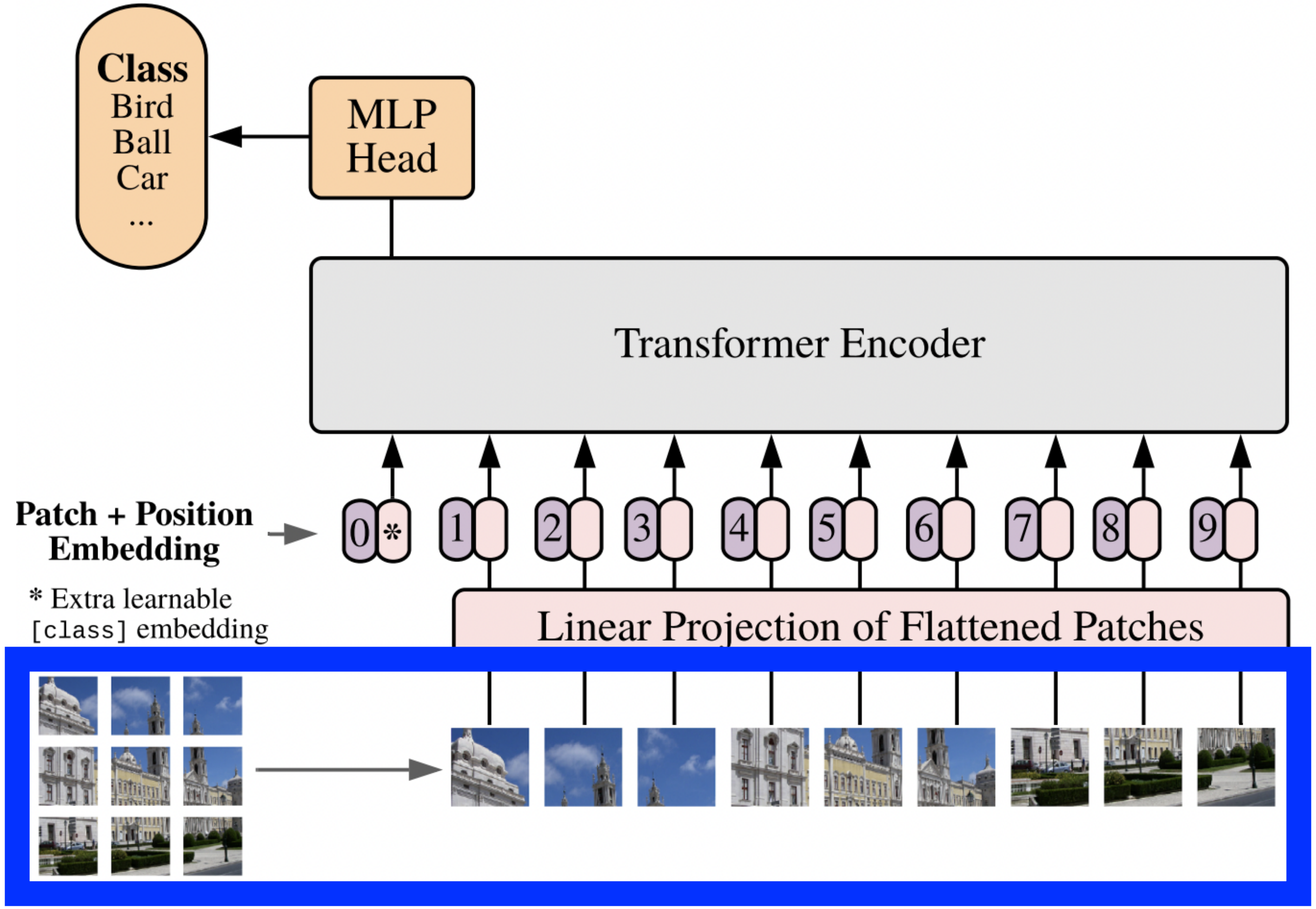
e.g., instead of using 3x3 image, what if a 1920 x 1080 image was used? How many self-attention computations would be needed?

- $(1920 \times 1080)^2 = 4{,}299{,}816{,}960{,}000$ (i.e., ~4.3 trillion)



Quadratic cost of self-attention in transformers is often impractical for pixels!

Architecture: Input (Patches Instead of Pixels)

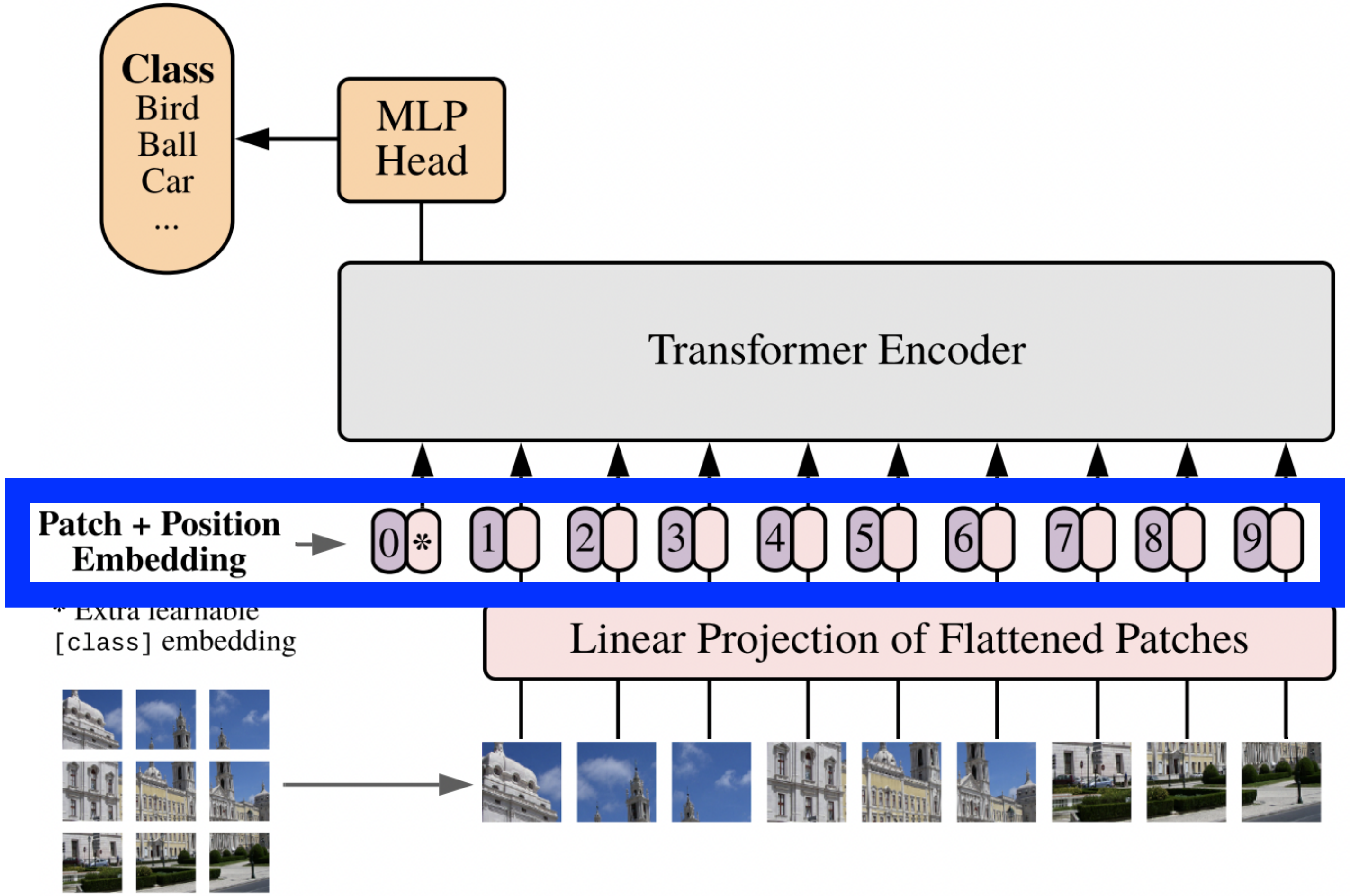


Assuming a 160 x 160 pixel image, how many patches (and so inputs) would be extracted?
- 100

image decomposed into 16x16 patches (example simplified); representations include "flattened" and ResNet features

Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021

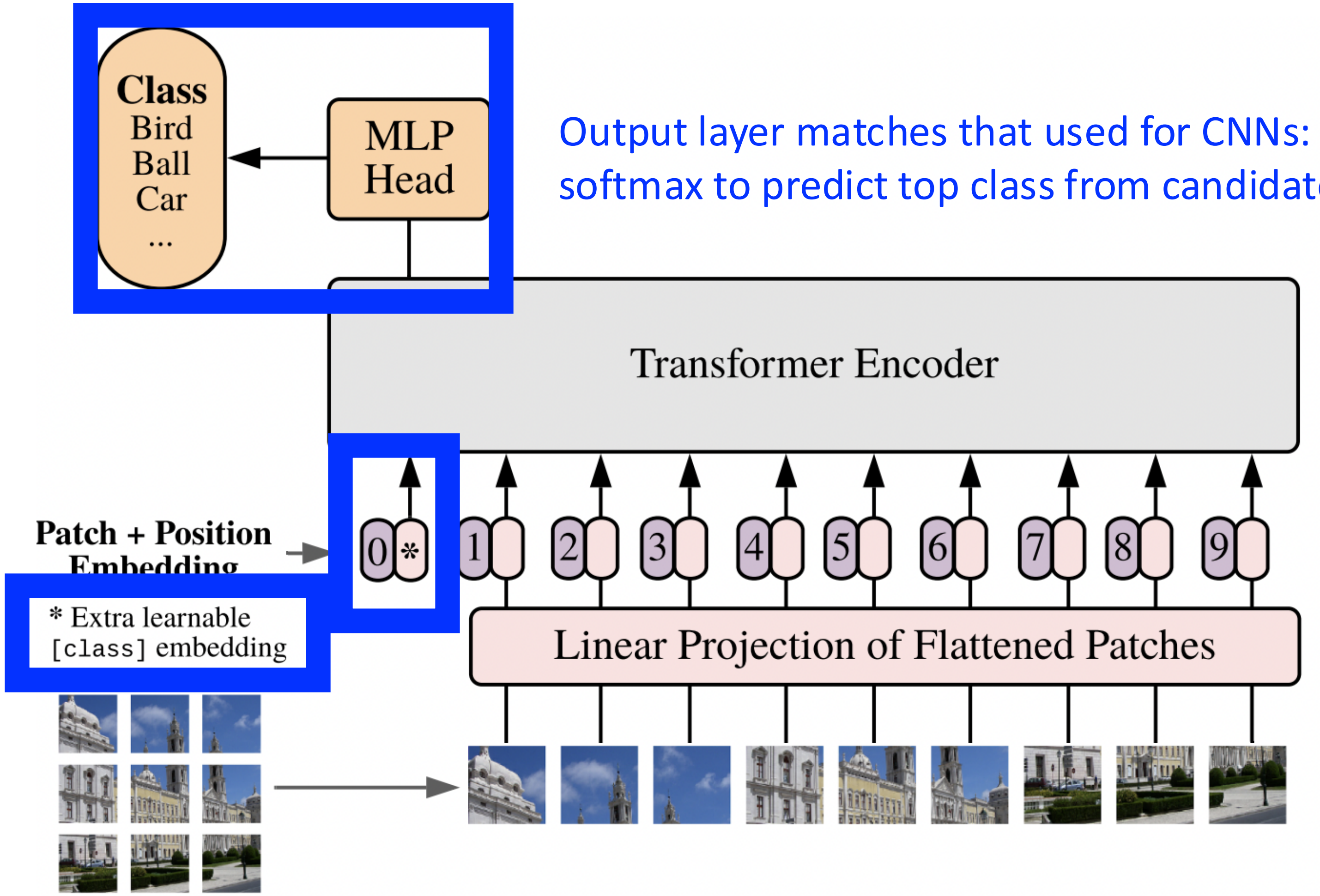
Architecture: Input Position Embedding



Position embedding added for learning order of input tokens

Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021

Architecture: Classification with CLS Token



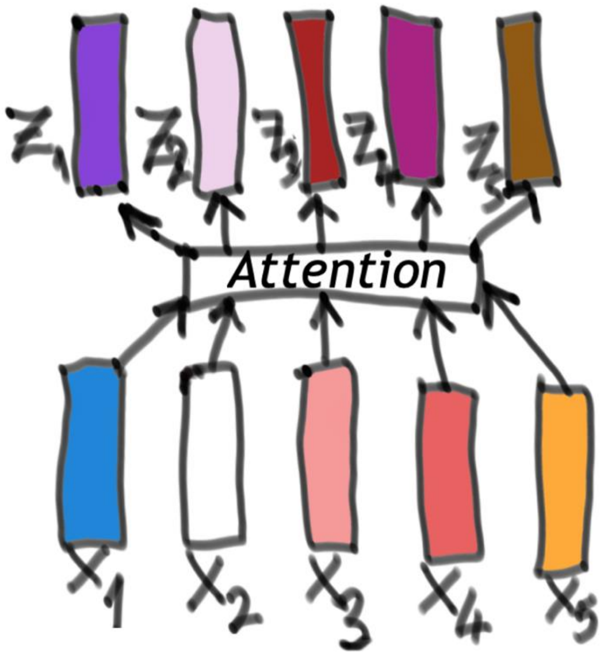
Output layer matches that used for CNNs: softmax to predict top class from candidates

[CLS] token represents entire image

Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021

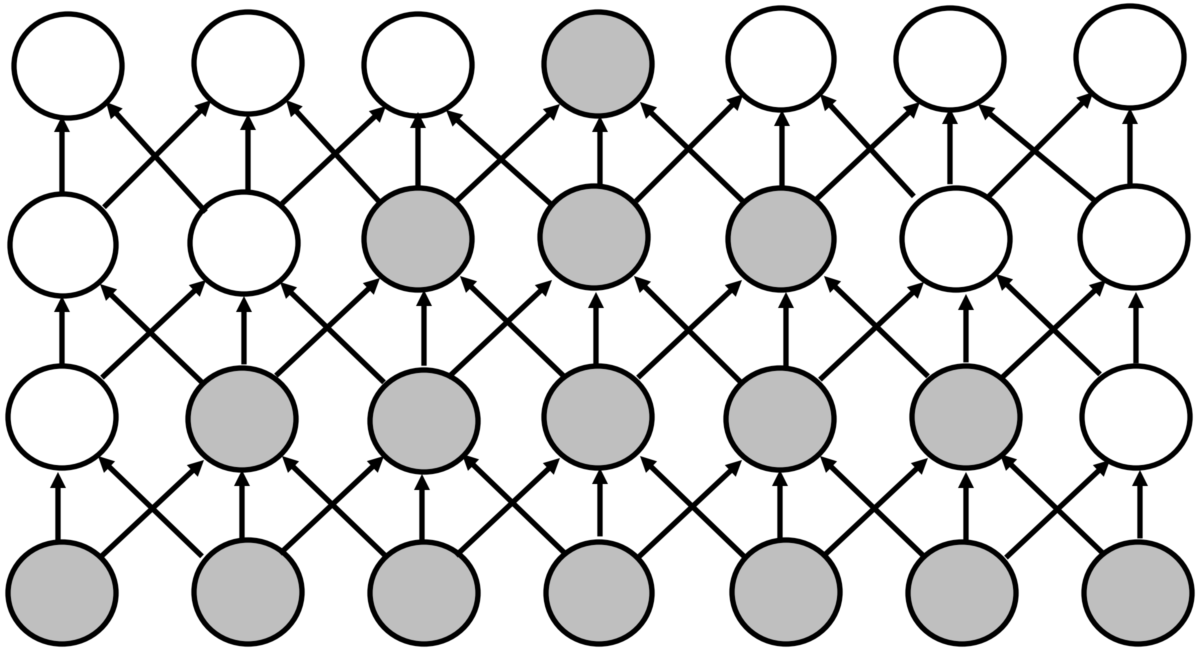
Transformers vs CNNs

Self-attention: each layer has a global receptive field



https://towardsdatascience.com/self-attention-5b95ea164f61

Convolutional layers: deeper layers have increasingly more global receptive fields

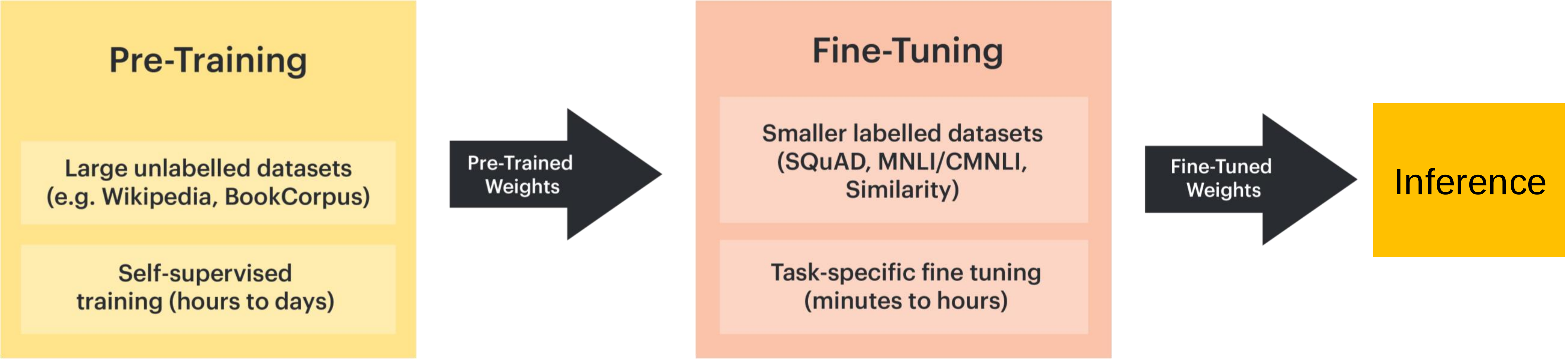


https://www.deeplearningbook.org/contents/convnets.html

Today’s Topics

- Motivation
- ViT architecture
- ViT training
- Guidance for student-led lectures

Common Paradigm for NLP Transformers



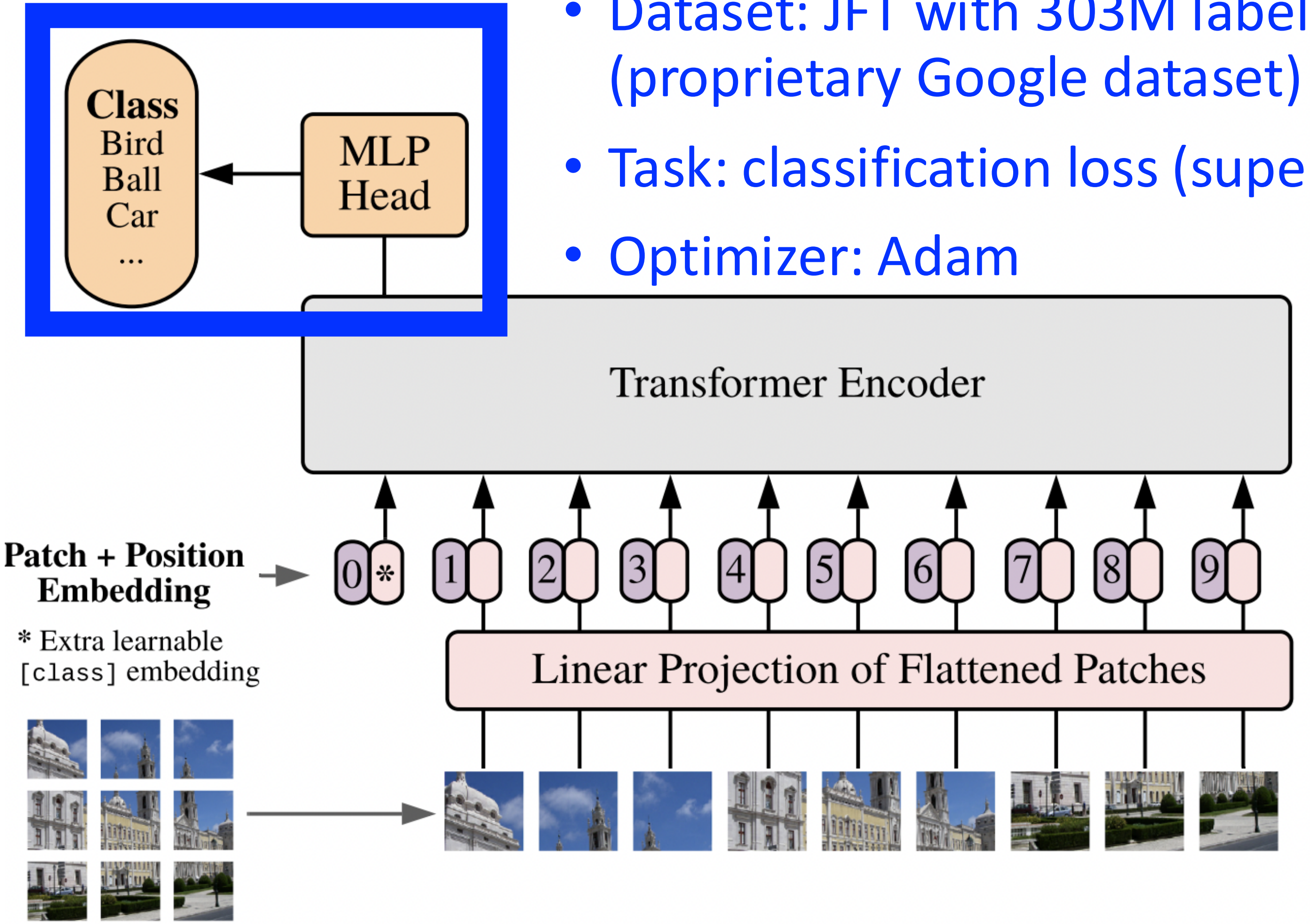
Transformers can provide effective features for downstream tasks!

https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

ViT Training Approach

https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

ViT Pre-Training

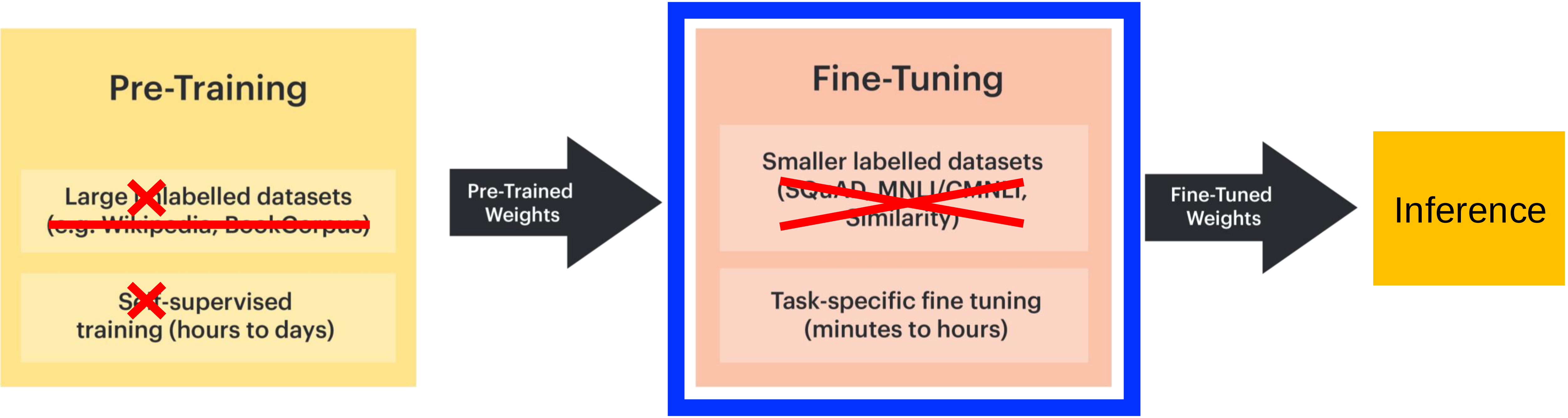


- Dataset: JFT with 303M labeled images (proprietary Google dataset)
- Task: classification loss (supervised)
- Optimizer: Adam

* Note: research also shows how smaller datasets can be effective; e.g., data efficient image transformers (DeiT)

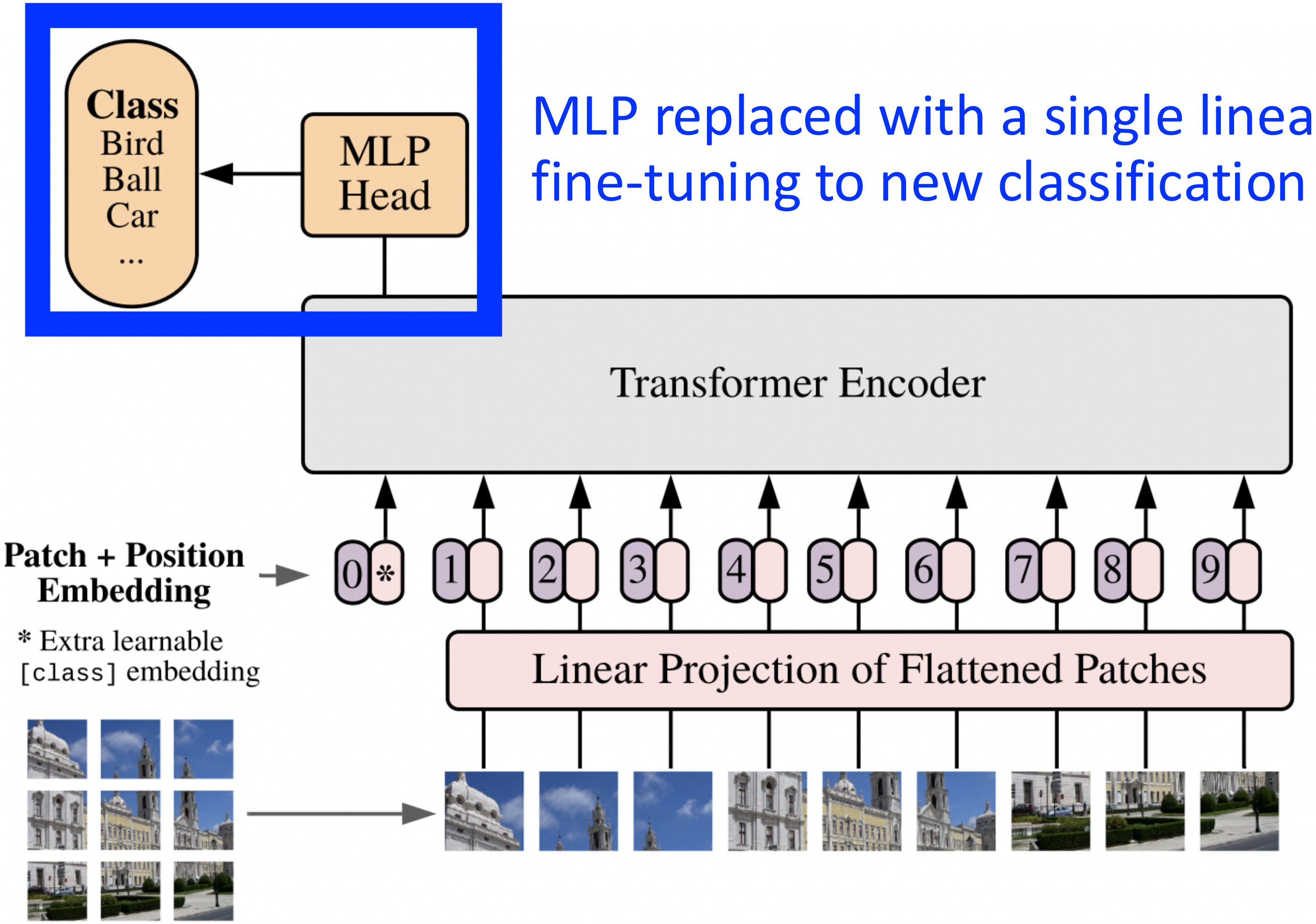
Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021

ViT Training



https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

ViT Fine-Tuning: Other Image Classification Tasks



MLP replaced with a single linear layer when fine-tuning to new classification categories

Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021

Experimental Findings

Achieved strong results on five image classification datasets

Transformers vs CNNs

- An open debate remains about which architecture to prefer
- Ideas from both architectures are infused into each other; e.g.,
  - https://arxiv.org/pdf/2201.03545.pdf
  - https://proceedings.neurips.cc/paper_files/paper/2022/file/5e0b46975d1bfe6030b1687b0a da1b85-Paper-Conference.pdf
  - https://arxiv.org/pdf/2207.13317.pdf
  - https://arxiv.org/pdf/2201.09792.pdf
- Benchmarks compare their robustness; e.g.,
  - https://arxiv.org/pdf/2207.11347.pdf
  - https://arxiv.org/pdf/2206.03452.pdf
  - https://proceedings.neurips.cc/paper_files/paper/2022/file/5ce3a49415f78db65a714b4f05c 62f4e-Paper-Conference.pdf

- **To Do**: test both model types following programming tutorial in Canvas

Today’s Topics

- Motivation
- ViT architecture
- ViT training
- Guidance for student-led lectures

Summary

- Expectations and recommendations
  - https://dannagurari.colorado.edu/course/recent-advances-in-computer-vision-fall-2024/student-lecture/
- Meeting sign-up
  - Link provided in Canvas
- Google form
  - Fill out your topic preferences

Today’s Topics

- Motivation
- ViT architecture
- ViT training
- Guidance for student-led lectures


The End