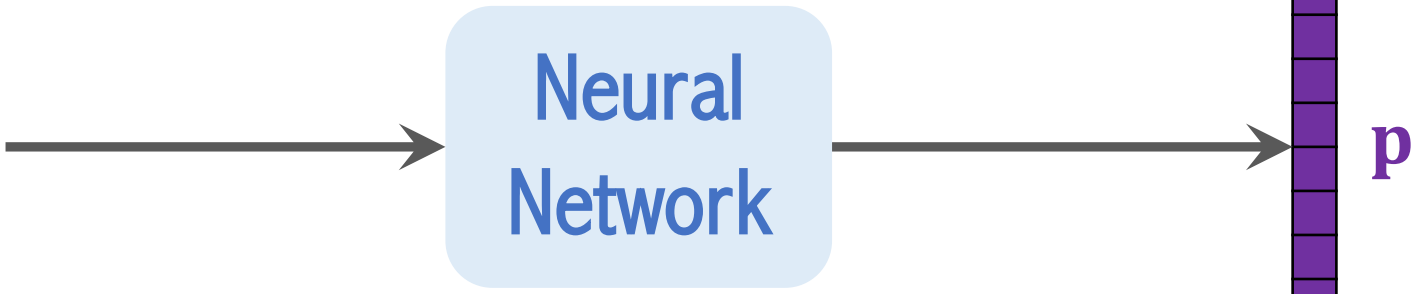
Vision Transformer (ViT)

Shusen Wang

Stevens Institute of Technology

http://wangshusen.github.io/

What is in the image?



p

Neural Network

p

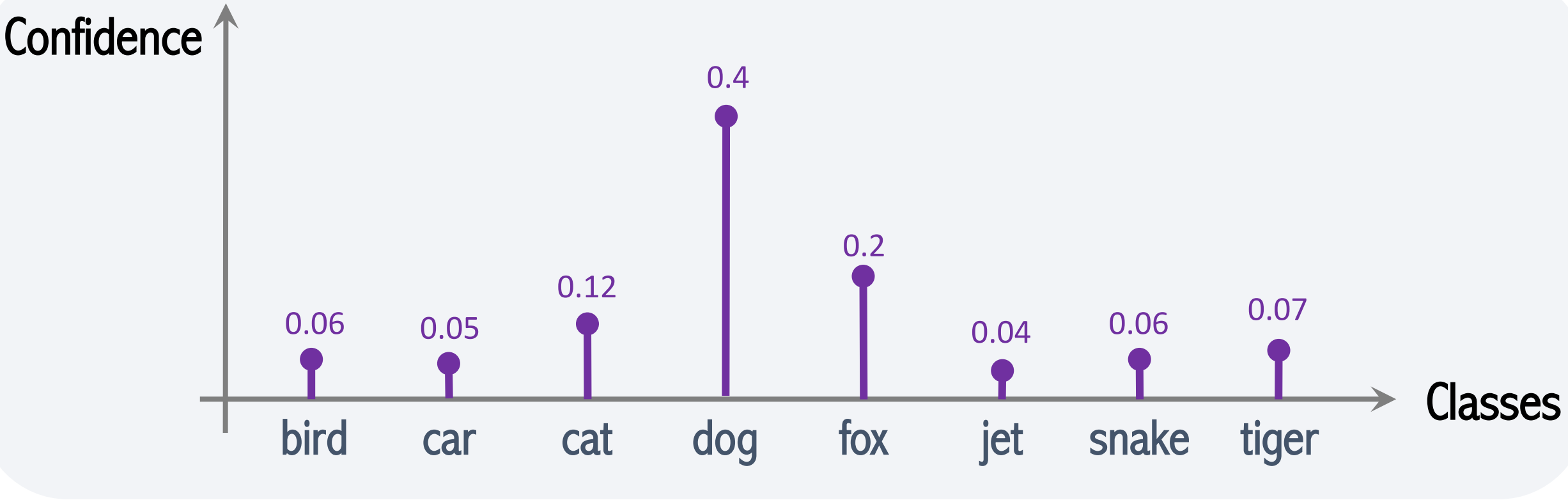


Image Classification

- CNNs, e.g., ResNet, were the best solutions to image classification.
- Vision Transformer (ViT) [1] beats CNNs (by a small margin), if the dataset for pretraining is sufficiently large (at least 100 million images).
- ViT is based on Transformer (for NLP) [2].

Reference

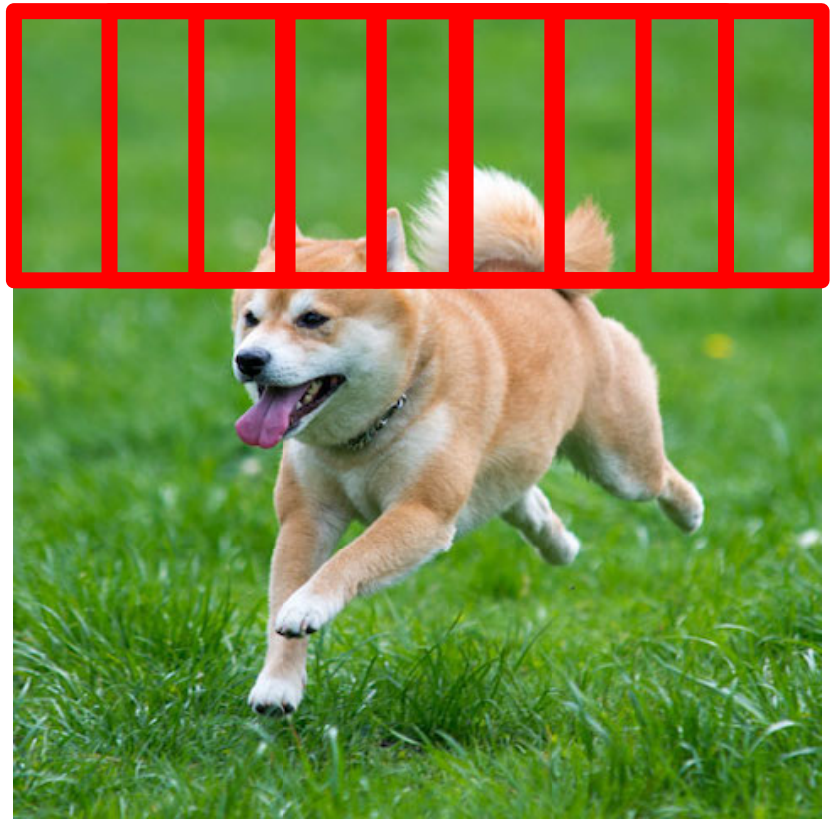
1. Dosovitskiy et al. An image is worth 16×16 words: transformers for image recognition at scale. In *ICLR*, 2021.
2. Vaswani et al. Attention Is All You Need. In *NIPS*, 2017.

Split Image into Patches

Split Image into Patches

- Here, the patches do not overlap.

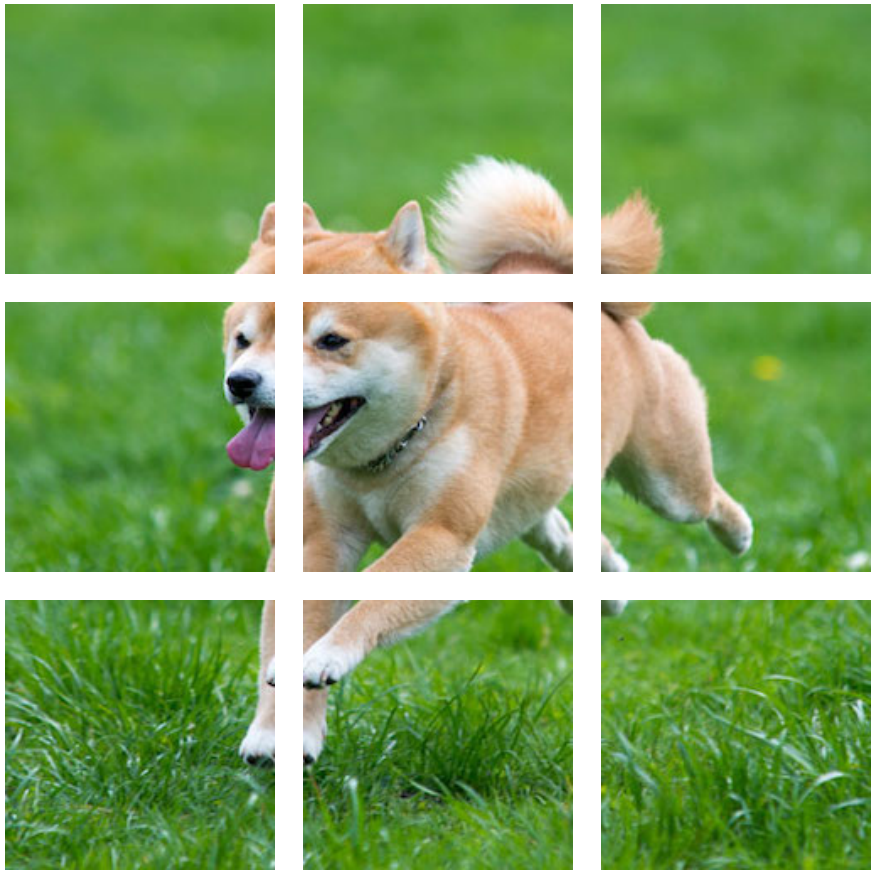
Split Image into Patches



- Here, the patches do not overlap.
- The patches can overlap.
- User specifies:
  - patch size, e.g., 16×16;
  - stride, e.g., 16×16.

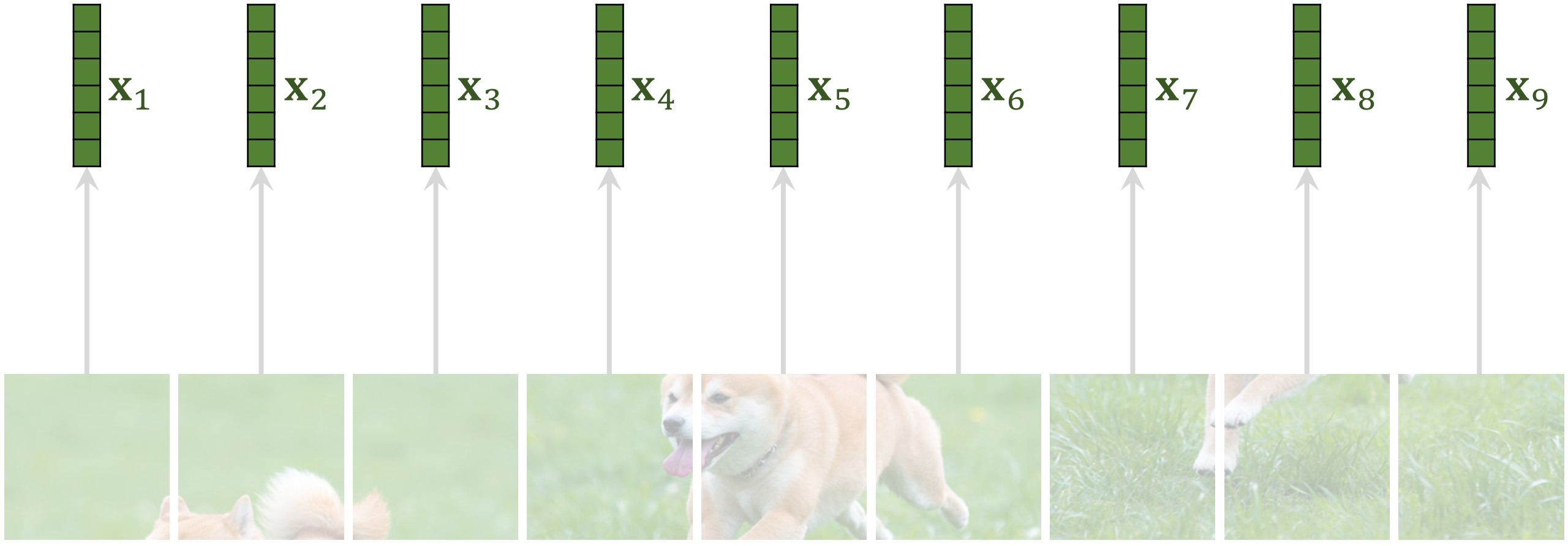
Vectorization

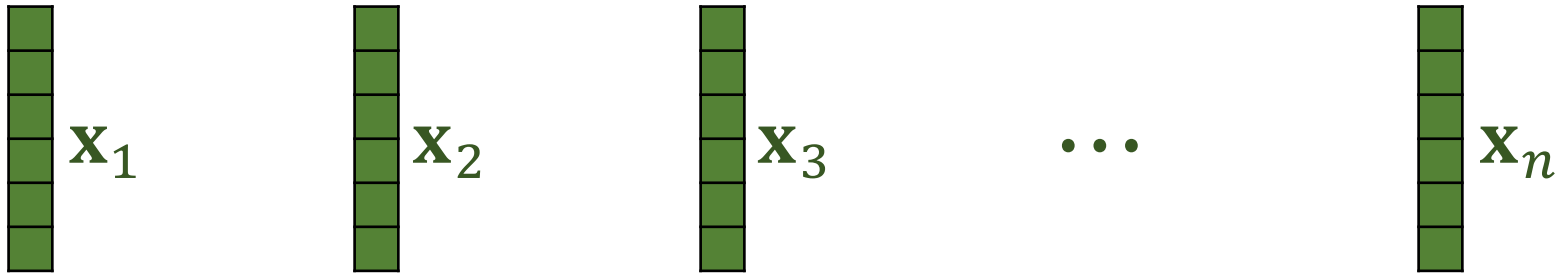
Vectorization

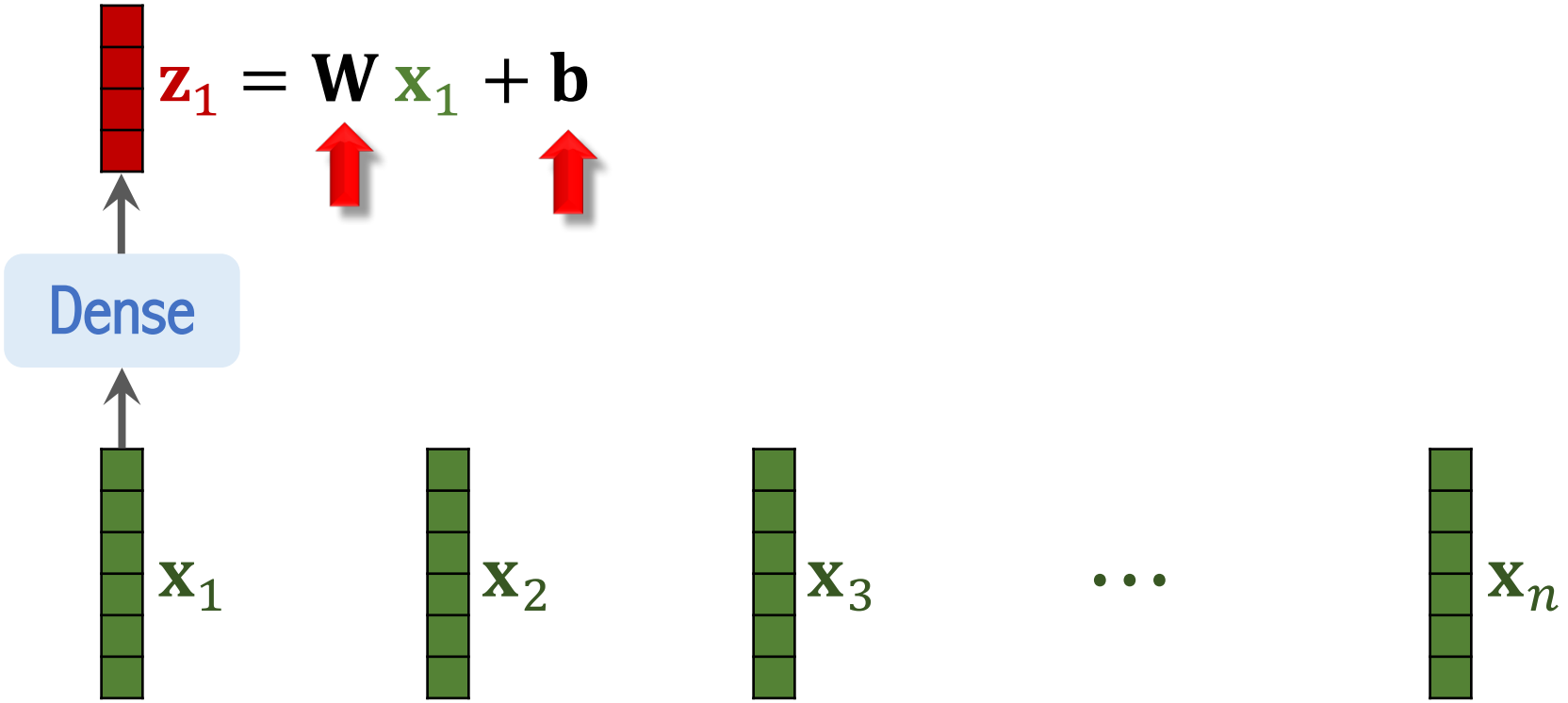


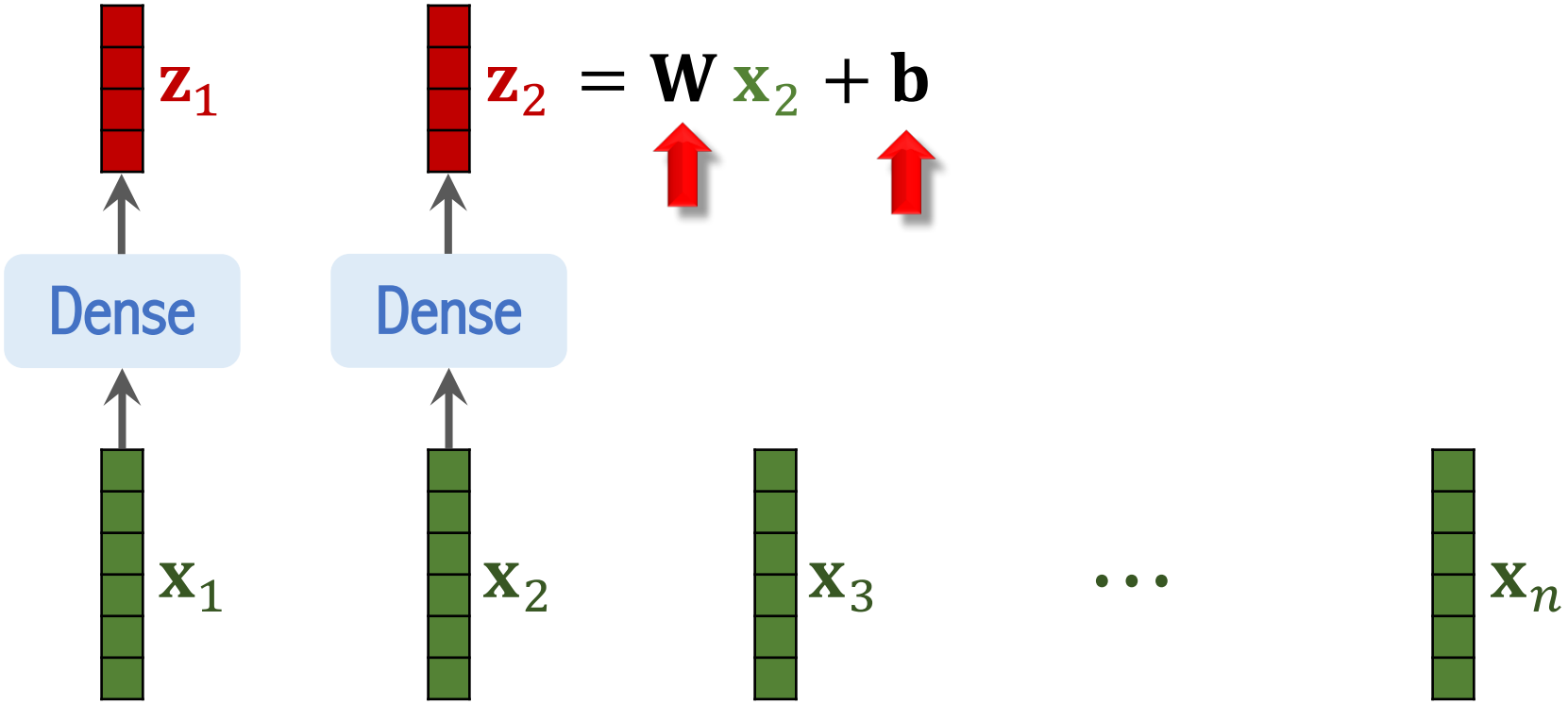
Vectorization

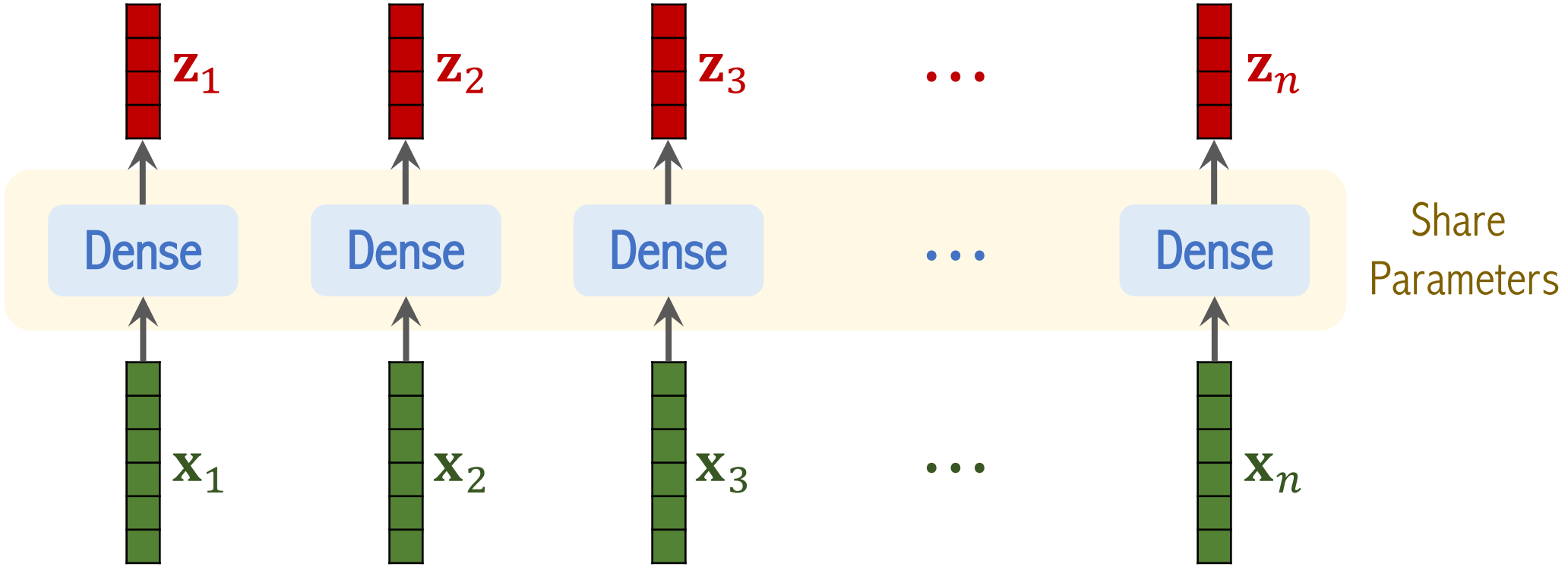
If the patches are $d_1 \times d_2 \times d_3$ tensors, then the vectors are $d_1 d_2 d_3 \times 1$.

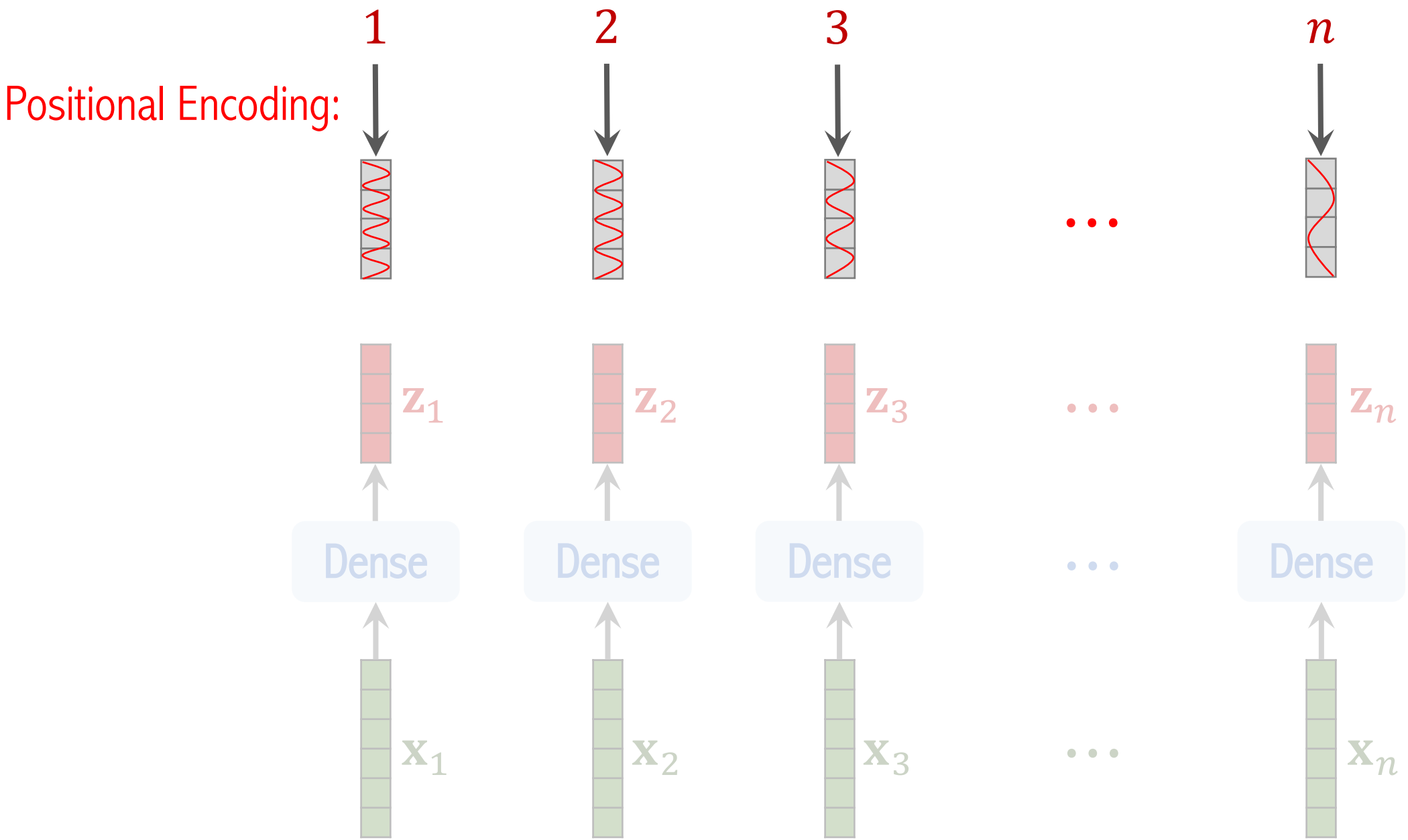



$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\cdots$
$\mathbf{x}_n$

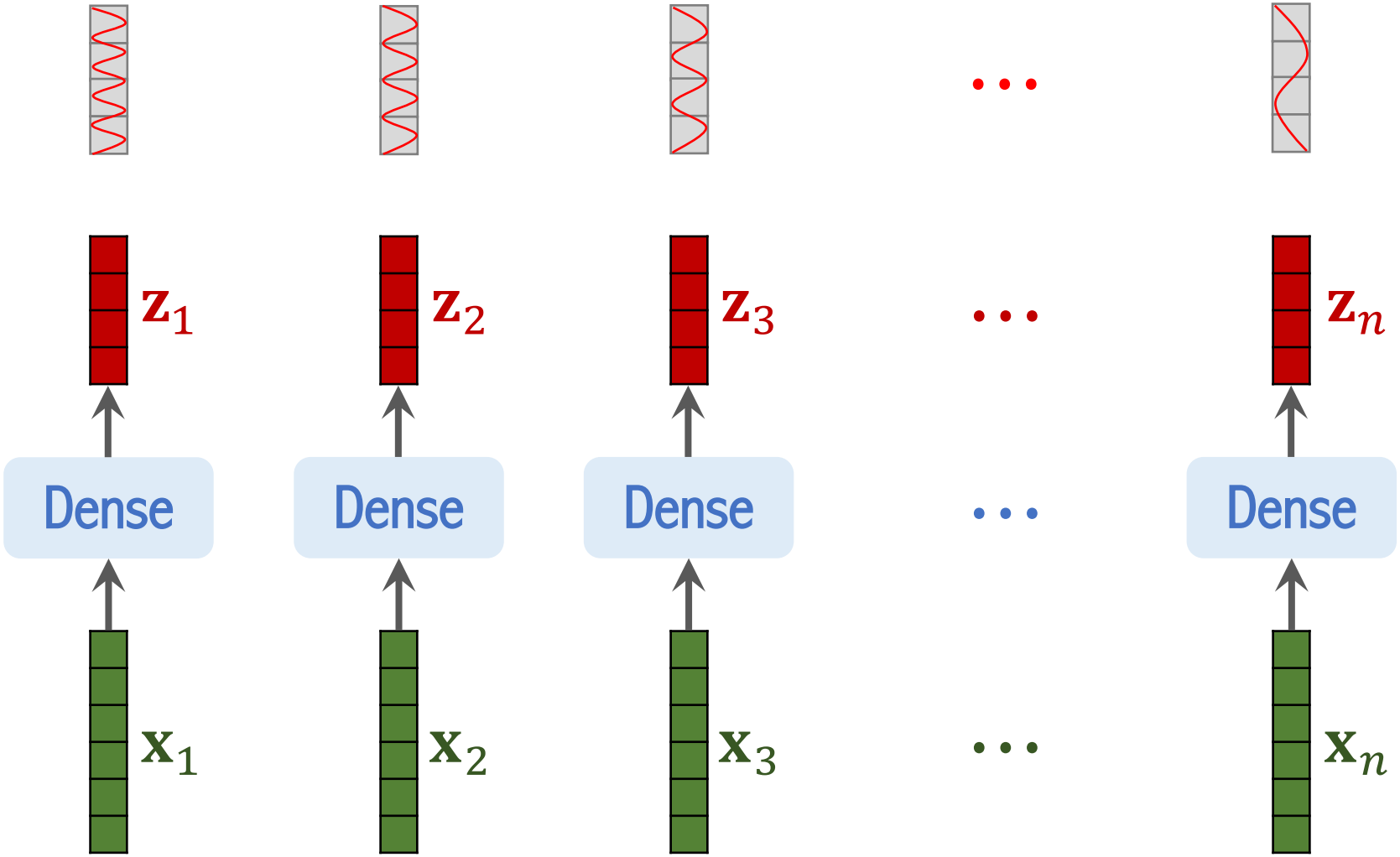

$\mathbf{z}_1 = \mathbf{W}\,\mathbf{x}_1 + \mathbf{b}$
Dense
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\cdots$
$\mathbf{x}_n$


$\mathbf{z}_1$
$\mathbf{z}_2 = \mathbf{W}\,\mathbf{x}_2 + \mathbf{b}$
Dense
Dense
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\cdots$
$\mathbf{x}_n$


$\mathbf{z}_1$
$\mathbf{z}_2$
$\mathbf{z}_3$
$\cdots$
$\mathbf{z}_n$
Dense
Dense
Dense
$\cdots$
Dense
Share Parameters
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\cdots$
$\mathbf{x}_n$

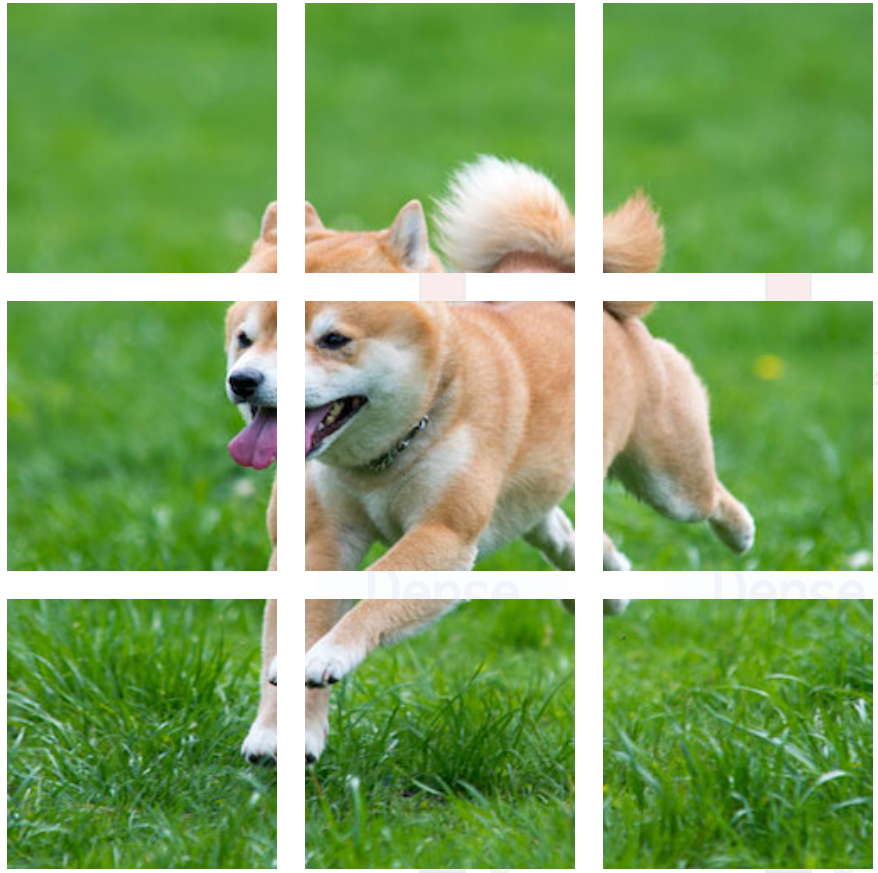

Positional Encoding:
1
2
3
n
z1
z2
z3
zn
Dense
Dense
Dense
Dense
x1
x2
x3
xn

Add positional encoding vectors to $\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_n$.

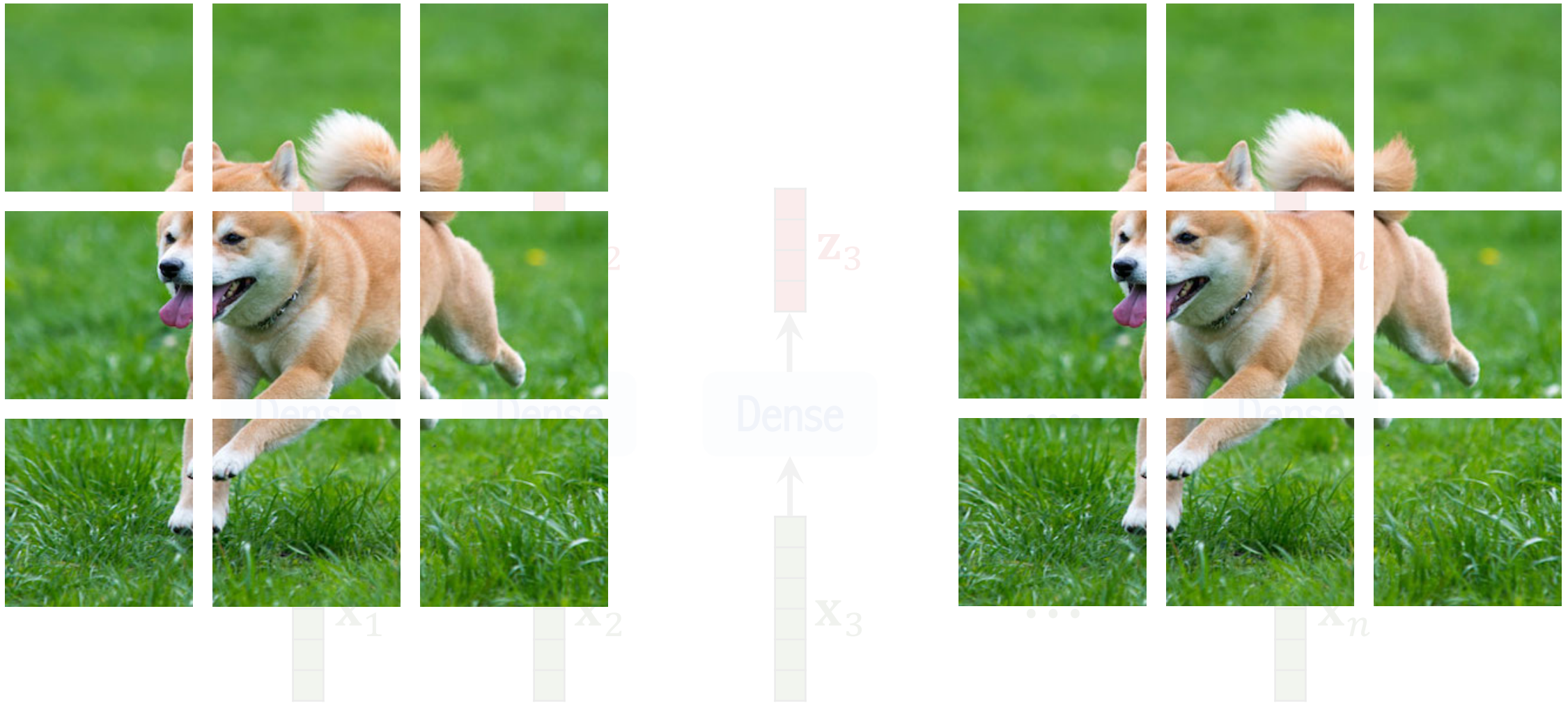


Add positional encoding vectors to $\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_n$. (Why?)

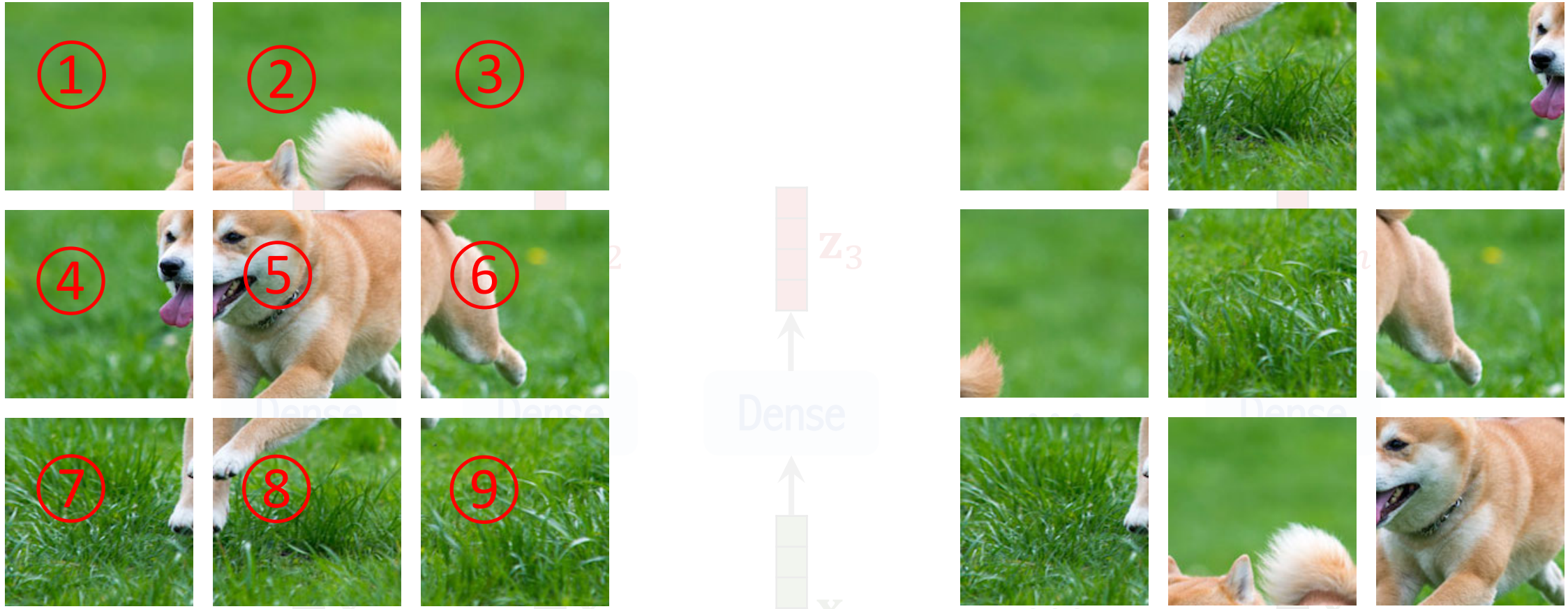
Add positional encoding vectors to $\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_n$. (Why?)

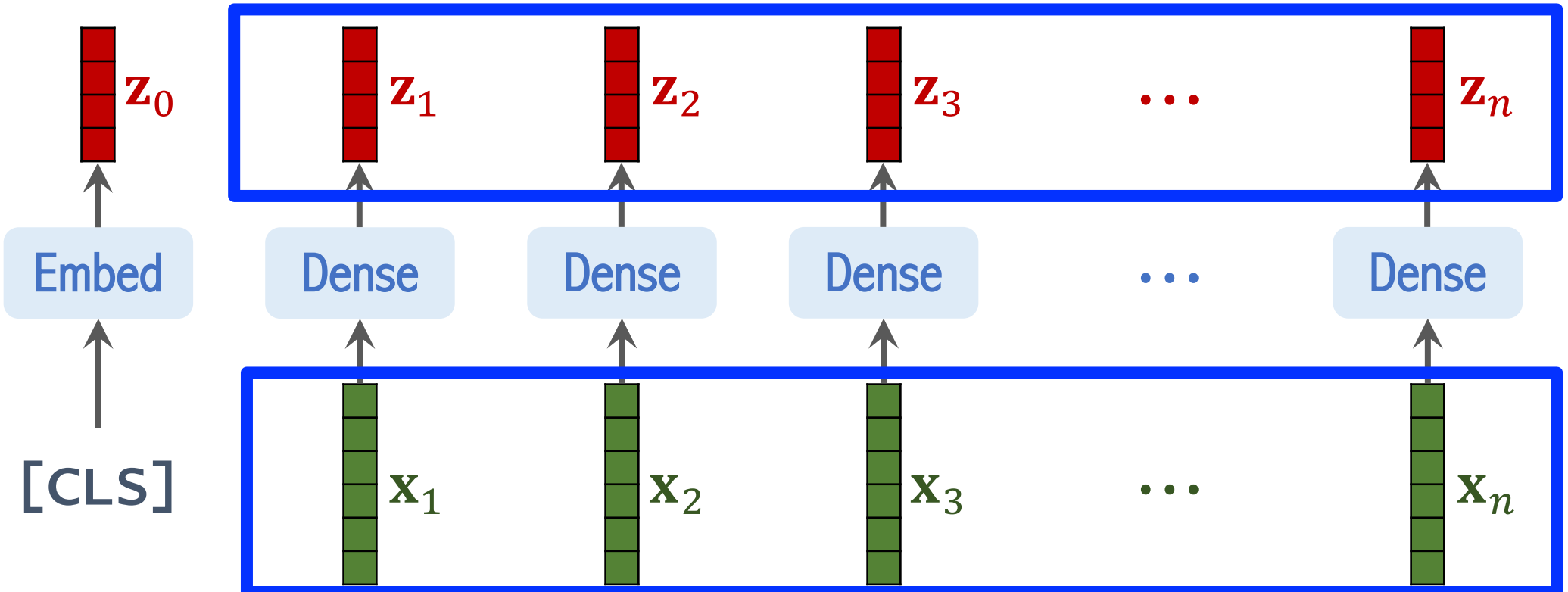


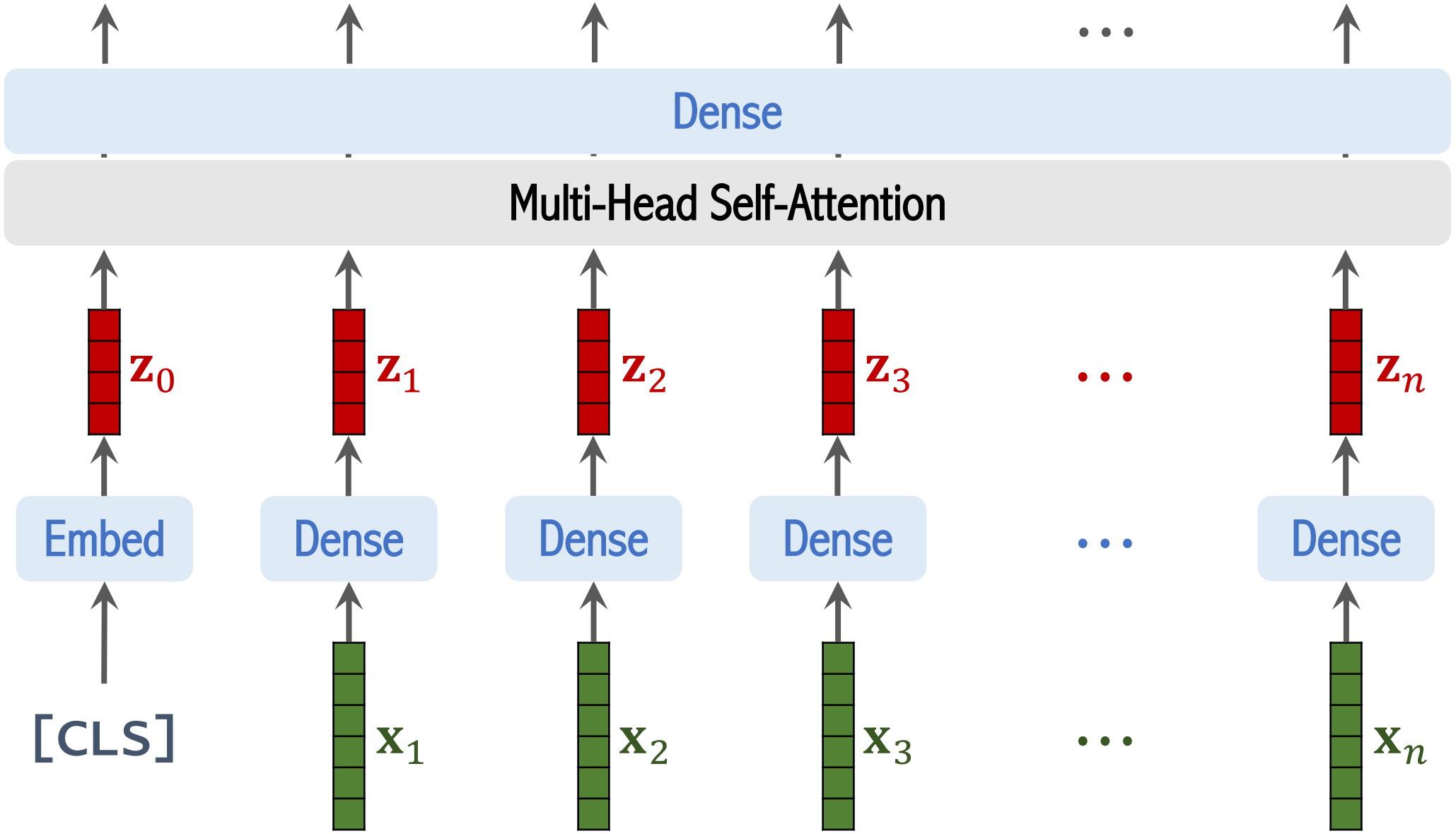
Add positional encoding vectors to $\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_n$. (Why?)

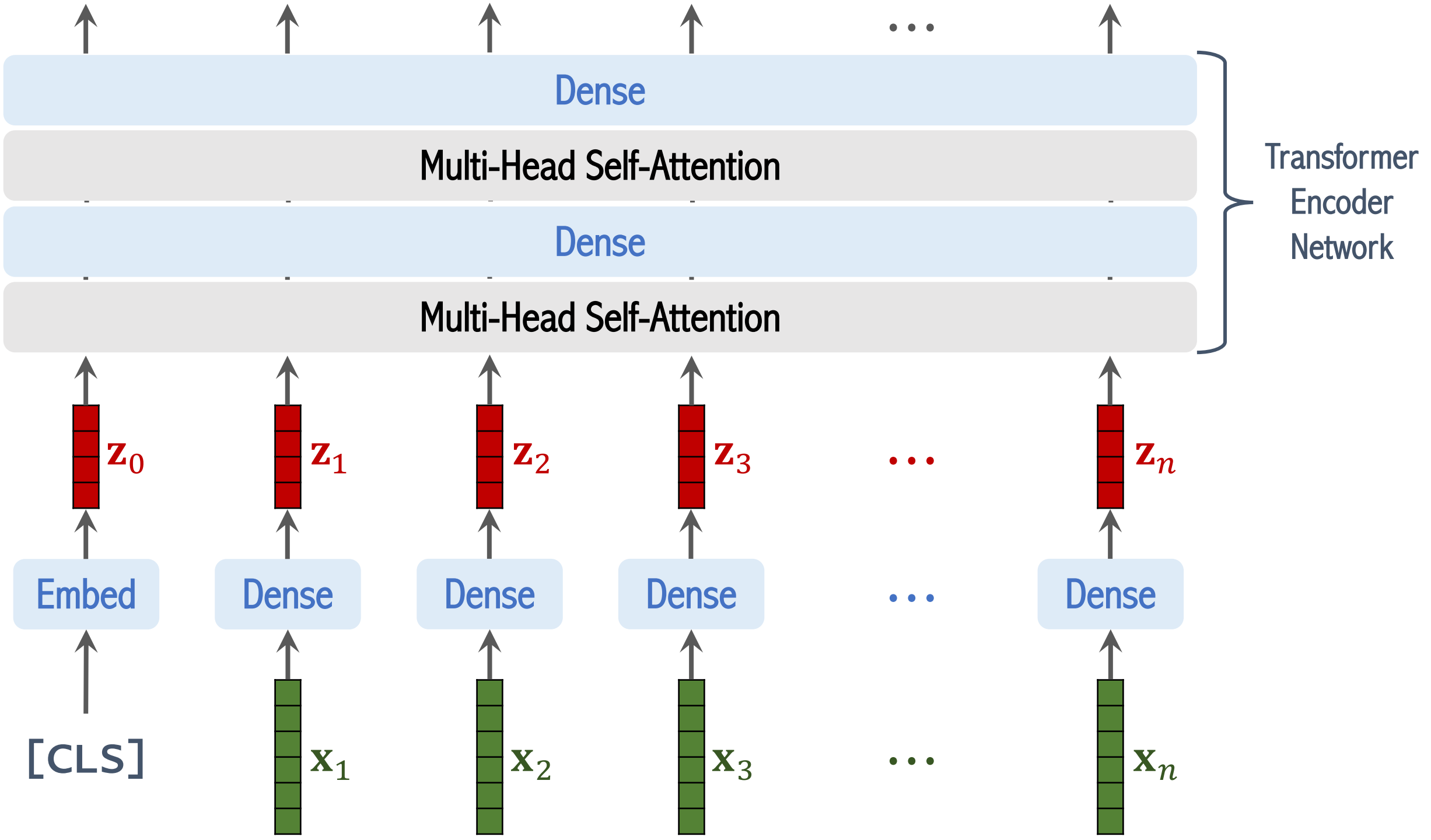


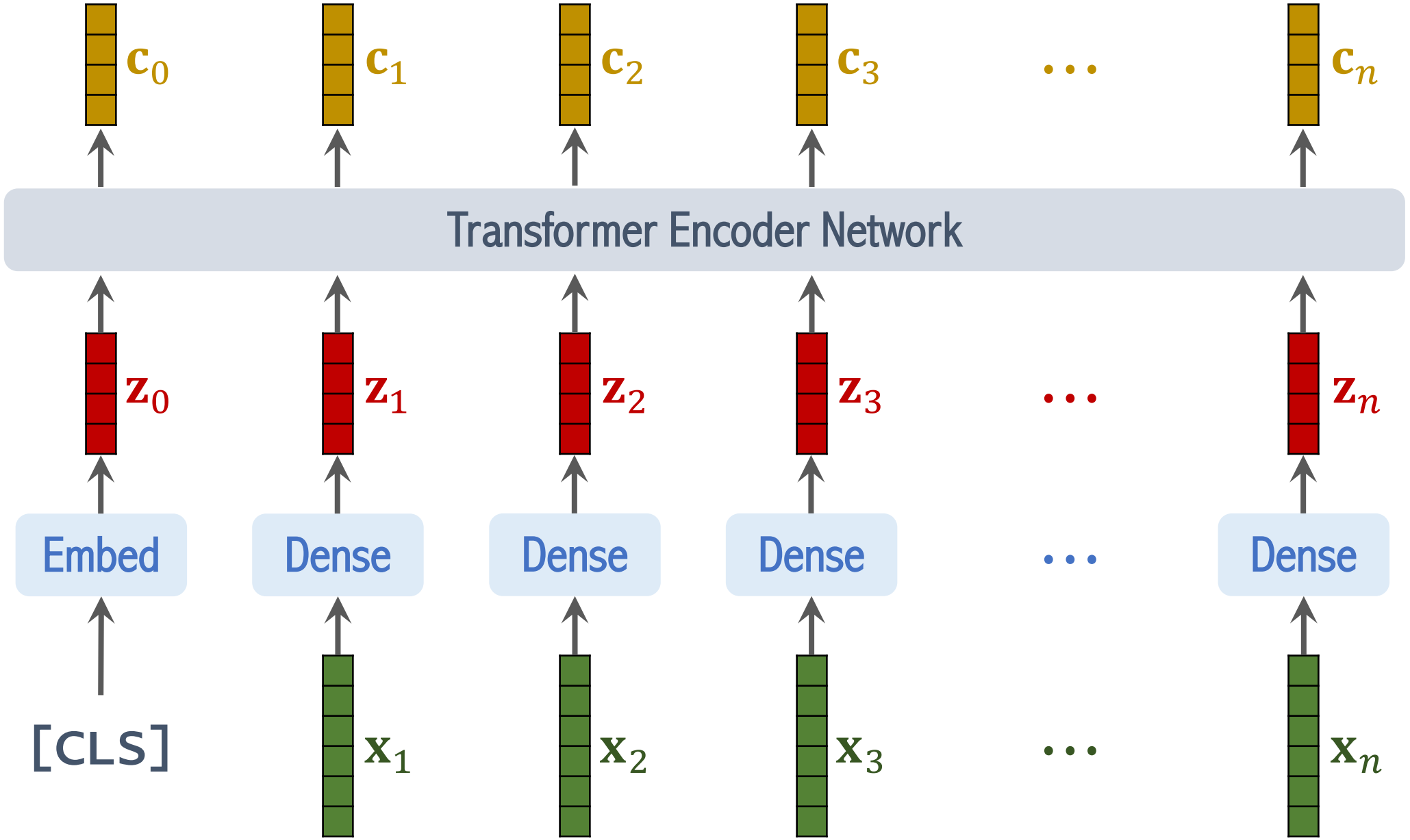
Add positional encoding vectors to $\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_n$. (Why?)

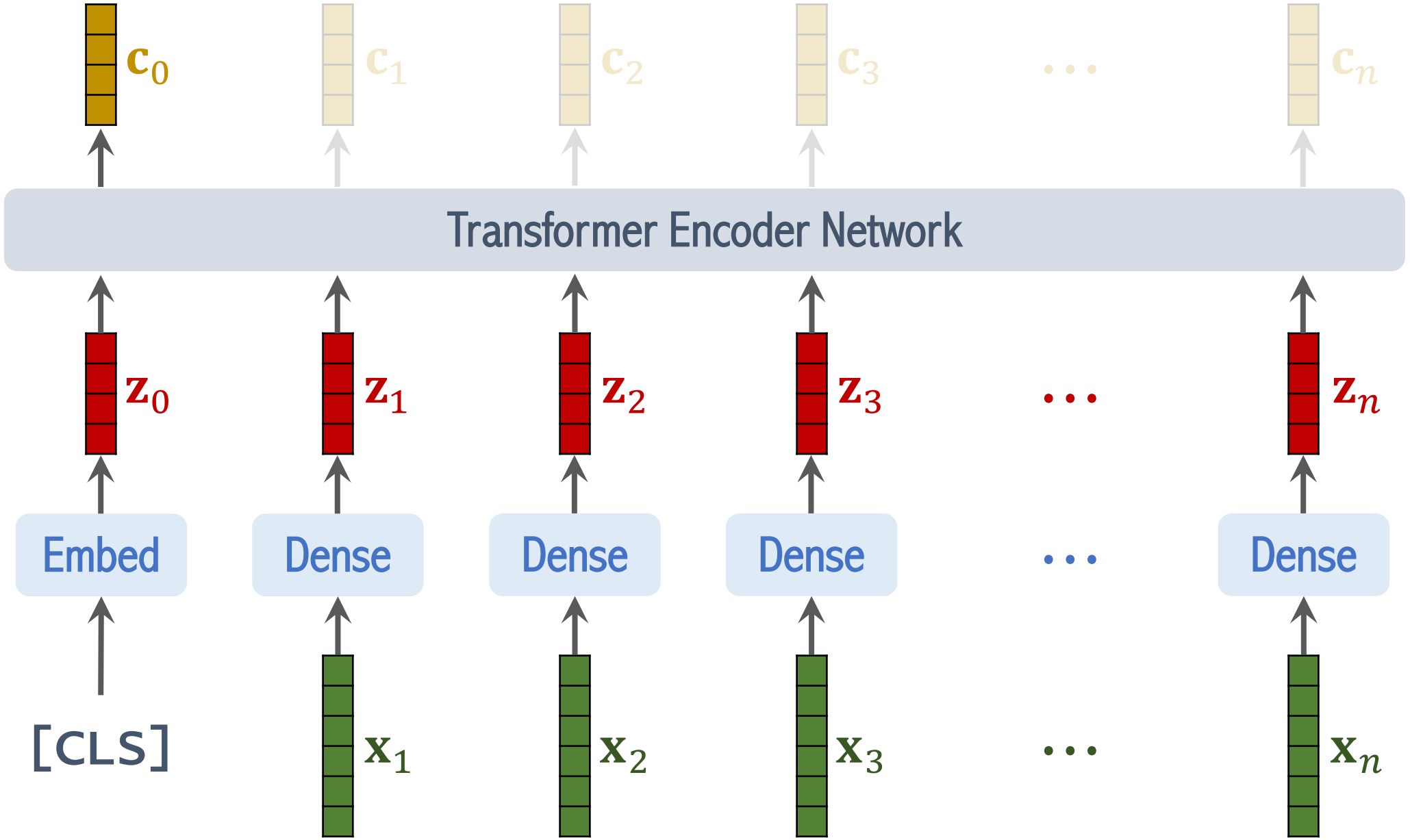


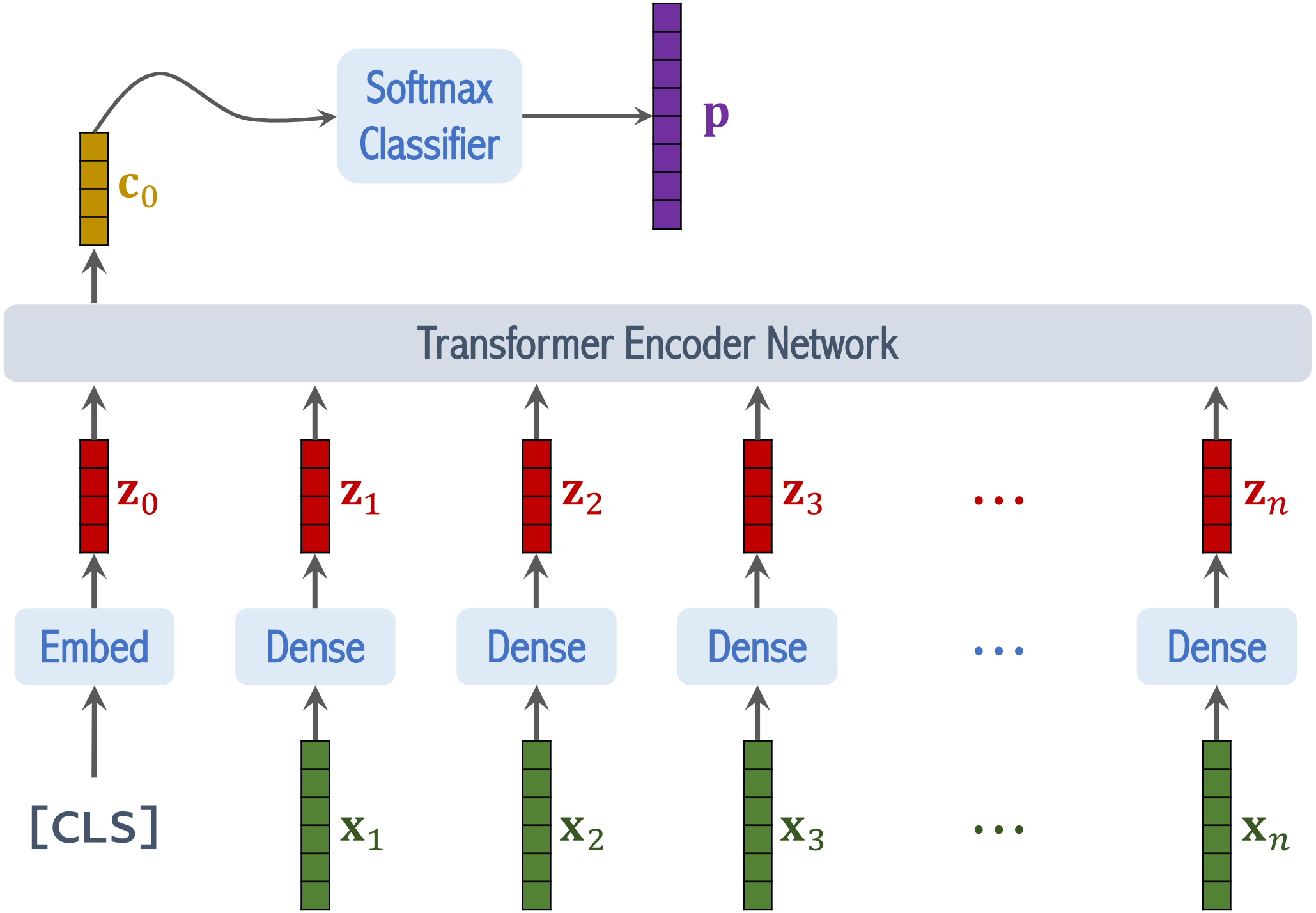

$\mathbf{z}_0$
$\mathbf{z}_1$
$\mathbf{z}_2$
$\mathbf{z}_3$
• • •
$\mathbf{z}_n$
Embed
Dense
Dense
Dense
• • •
Dense
[CLS]
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
• • •
$\mathbf{x}_n$

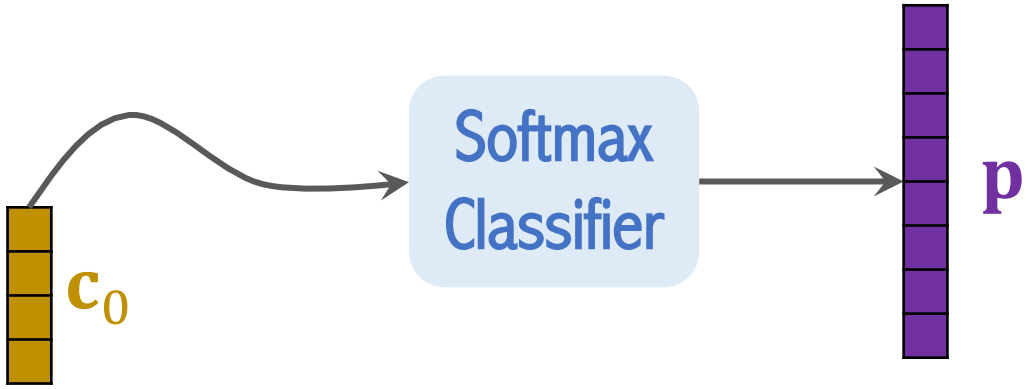

Dense
Multi-Head Self-Attention
$\mathbf{z}_0$
$\mathbf{z}_1$
$\mathbf{z}_2$
$\mathbf{z}_3$
$\mathbf{z}_n$
Embed
Dense
Dense
Dense
Dense
[CLS]
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\mathbf{x}_n$

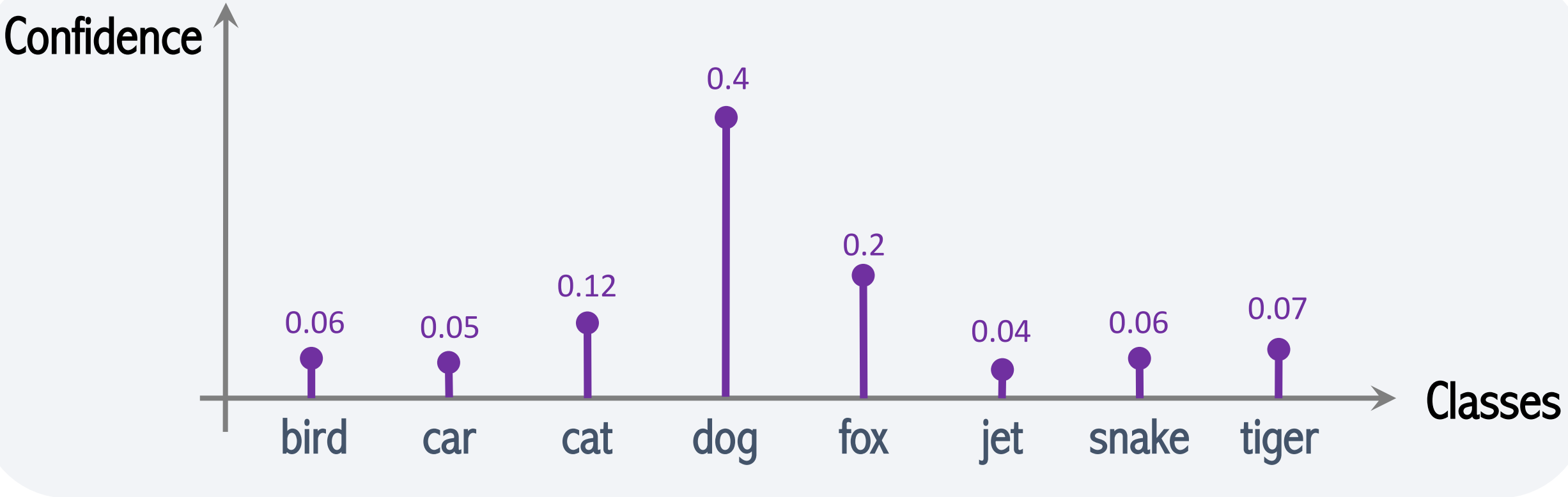

Dense
Multi-Head Self-Attention
Dense
Multi-Head Self-Attention
Transformer Encoder Network
$\mathbf{z}_0$
$\mathbf{z}_1$
$\mathbf{z}_2$
$\mathbf{z}_3$
$\mathbf{z}_n$
Embed
Dense
Dense
Dense
Dense
[CLS]
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\mathbf{x}_n$

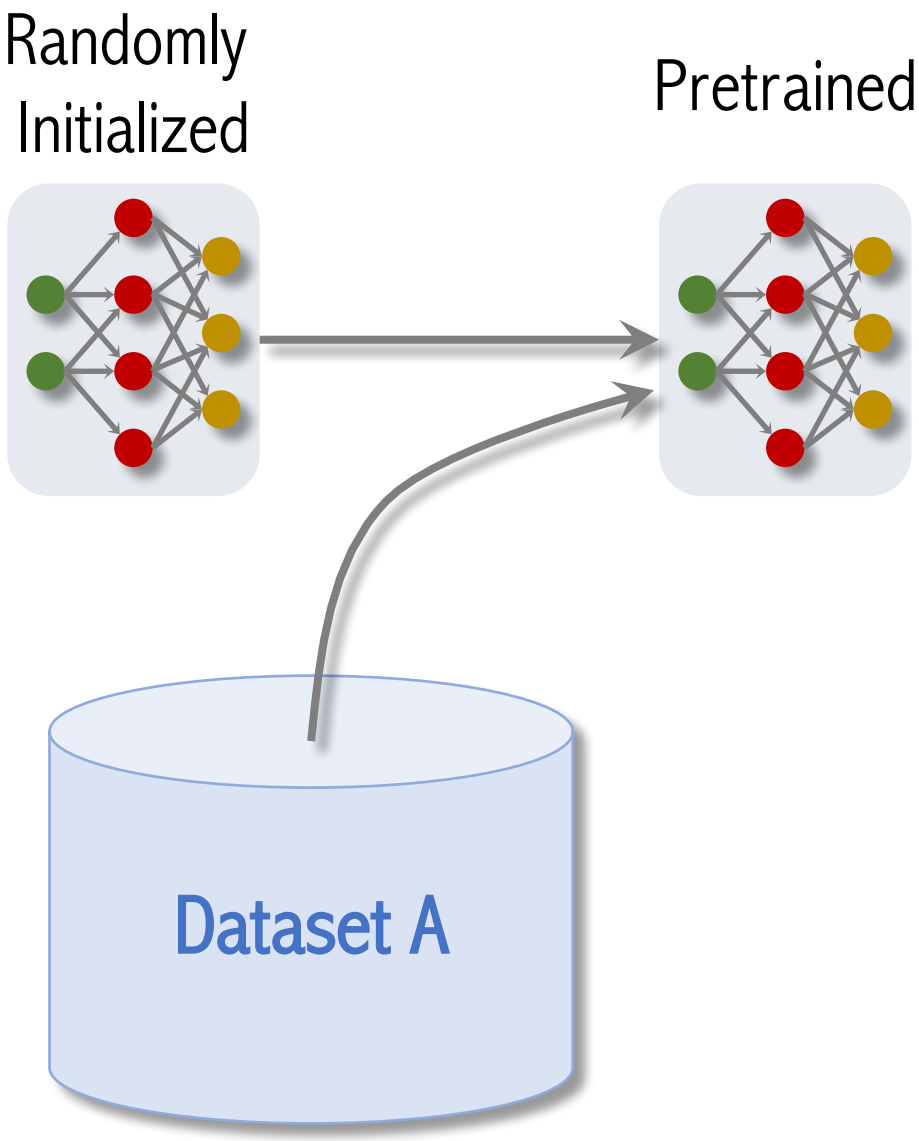

$\mathbf{c}_0$
$\mathbf{c}_1$
$\mathbf{c}_2$
$\mathbf{c}_3$
$\cdots$
$\mathbf{c}_n$
Transformer Encoder Network
$\mathbf{z}_0$
$\mathbf{z}_1$
$\mathbf{z}_2$
$\mathbf{z}_3$
$\cdots$
$\mathbf{z}_n$
Embed
Dense
Dense
Dense
$\cdots$
Dense
[CLS]
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\cdots$
$\mathbf{x}_n$

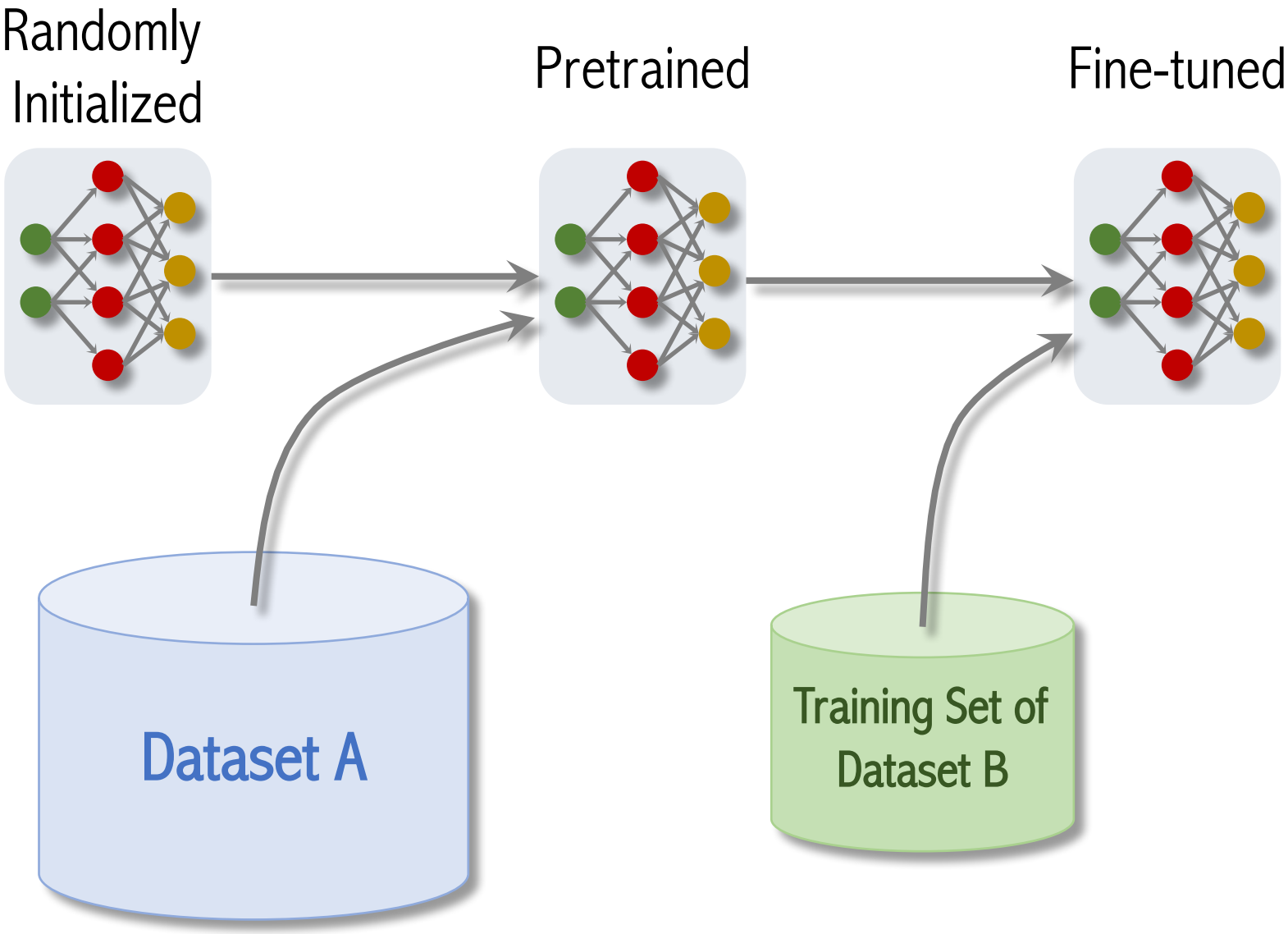

$\mathbf{c}_0$
$\mathbf{c}_1$
$\mathbf{c}_2$
$\mathbf{c}_3$
$\cdots$
$\mathbf{c}_n$
Transformer Encoder Network
$\mathbf{z}_0$
$\mathbf{z}_1$
$\mathbf{z}_2$
$\mathbf{z}_3$
$\cdots$
$\mathbf{z}_n$
Embed
Dense
Dense
Dense
$\cdots$
Dense
[CLS]
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\cdots$
$\mathbf{x}_n$

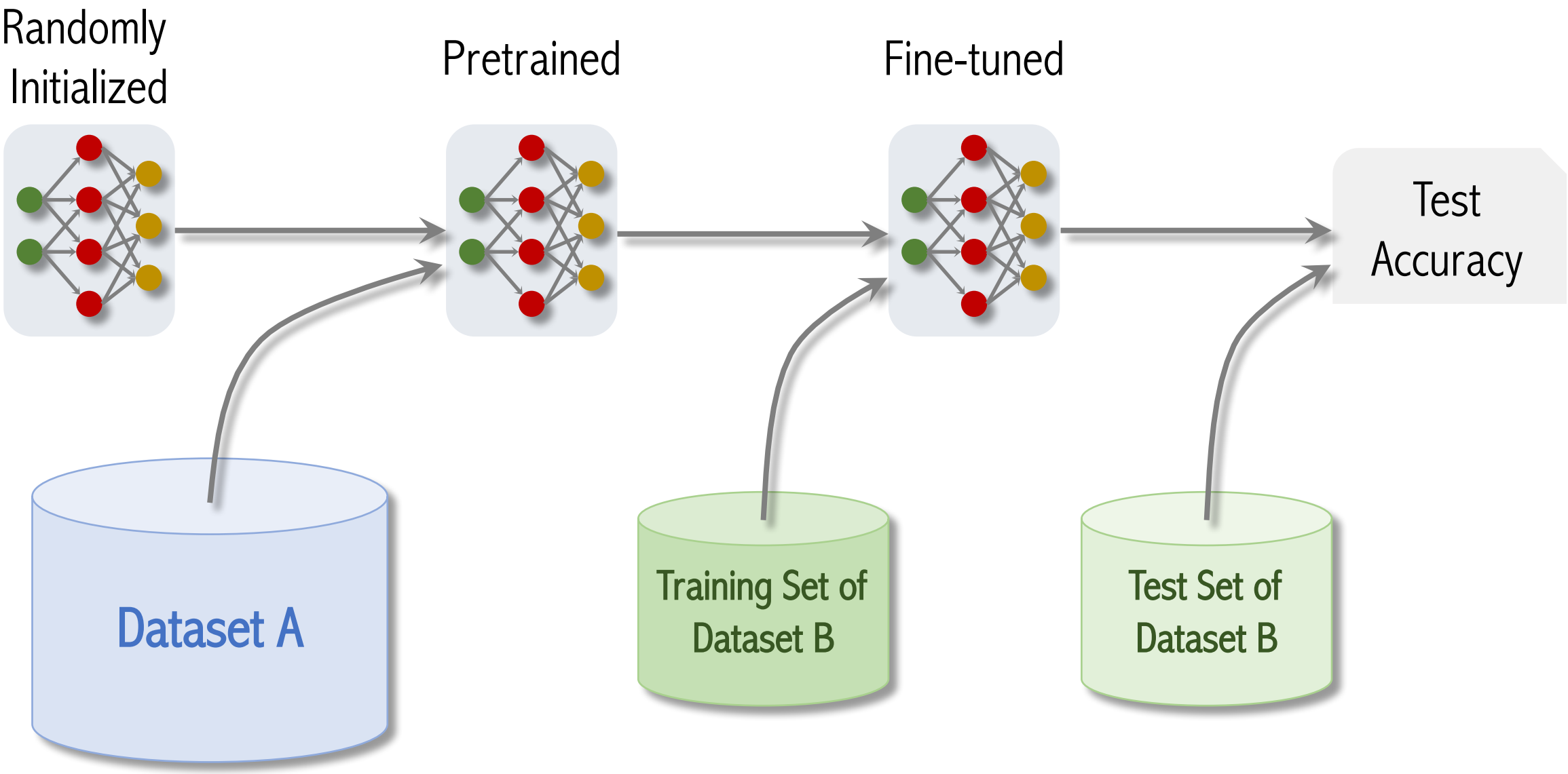

Softmax Classifier
p
$\mathbf{c}_0$
Transformer Encoder Network
$\mathbf{z}_0$
$\mathbf{z}_1$
$\mathbf{z}_2$
$\mathbf{z}_3$
$\mathbf{z}_n$
Embed
Dense
Dense
Dense
Dense
[CLS]
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\mathbf{x}_n$


Softmax
Classifier
$\mathbf{c}_0$
$\mathbf{p}$


Confidence
0.06
0.05
0.12
0.4
0.2
0.04
0.06
0.07
bird
car
cat
dog
fox
jet
snake
tiger
Classes


Randomly Initialized
Pretrained
Dataset A


Randomly Initialized
Pretrained
Fine-tuned
Dataset A
Training Set of Dataset B


Randomly Initialized
Pretrained
Fine-tuned
Test Accuracy
Dataset A
Training Set of Dataset B
Test Set of Dataset B

Datasets

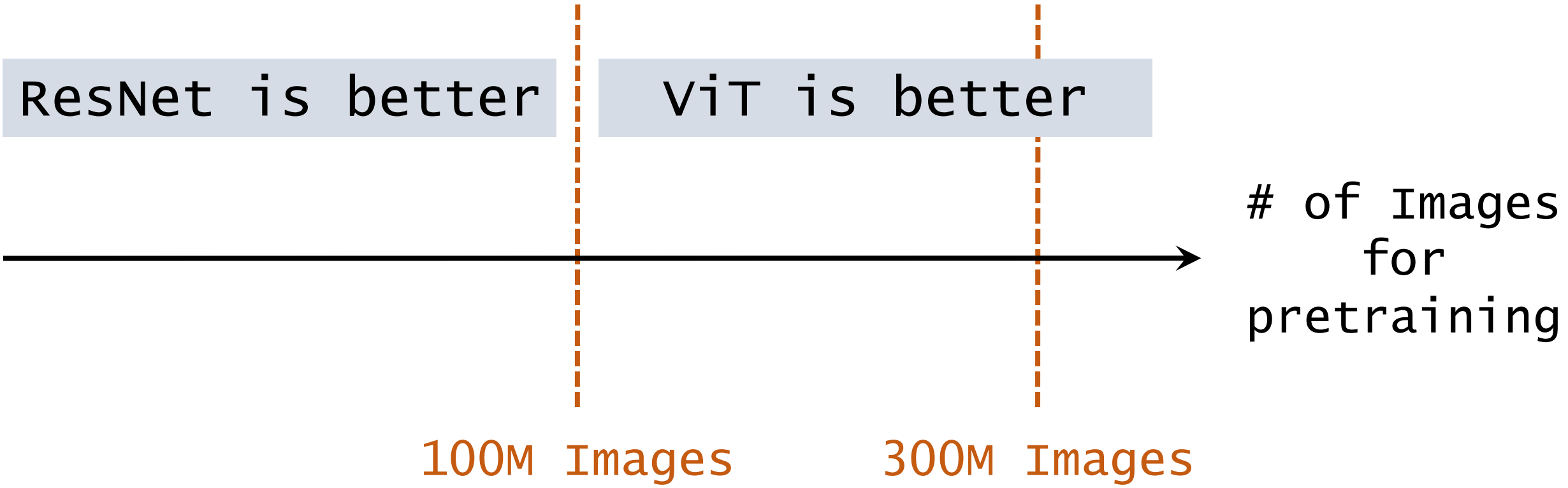
	# of Images	# of Classes
ImageNet (Small)	1.3 Million	1 Thousand
ImageNet-21K (Medium)	14 Million	21 Thousand
JFT (Big)	300 Million	18 Thousand

Image Classification Accuracies

- Pretrain the model on Dataset A, fine-tune the model on Dataset B, and evaluate the model on Dataset B.

- Pretrained on ImageNet (small), ViT is slightly worse than ResNet.
- Pretrained on ImageNet-21K (medium), ViT is comparable to ResNet.
- Pretrained on JFT (large), ViT is slightly better than ResNet.

Image Classification Accuracies



Thank You!

http://wangshusen.github.io/