2022-06-12

주간보고

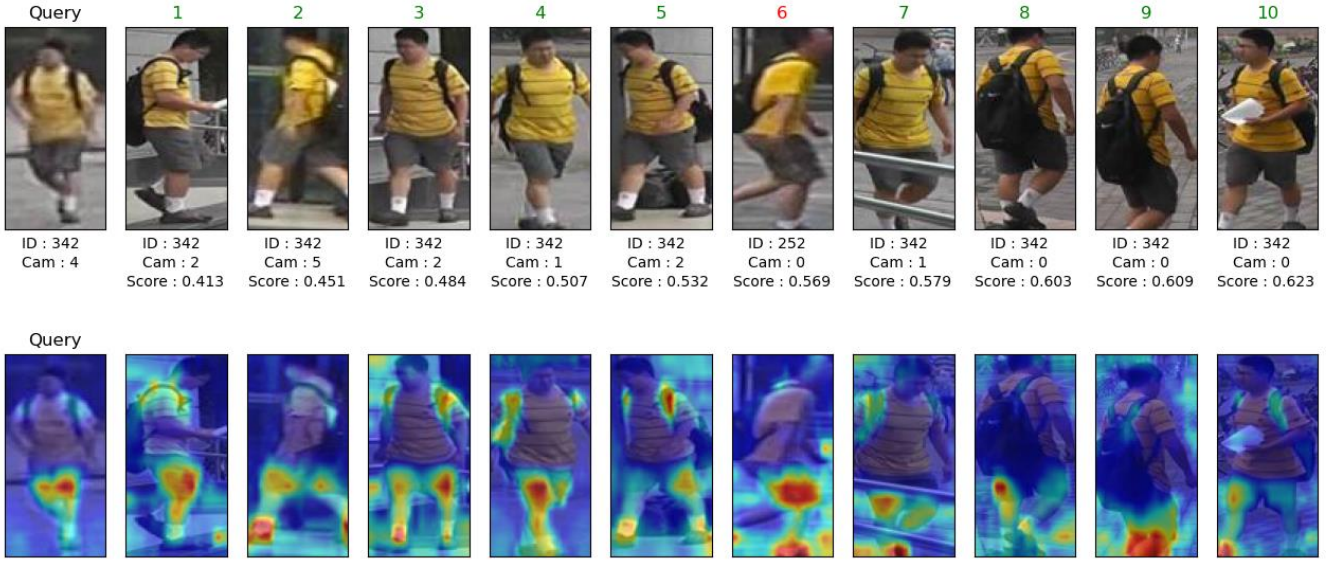
Man Machine Interface Lab.

This week

- *Person Re-identification using ViT*
  - *1. Patch Triplet loss for ViT (Market 1501 mAP 93.51%(0.31%↑), Rank1 97.1%(0.4% ↑) w/o RK)*

1	**CTL Model** (ResNet50, 256x128)	98.3	98.0	98.6	99.5	×	On the Unreasonable Effectiveness of Centroids in Image Retrieval		2021	ResNet

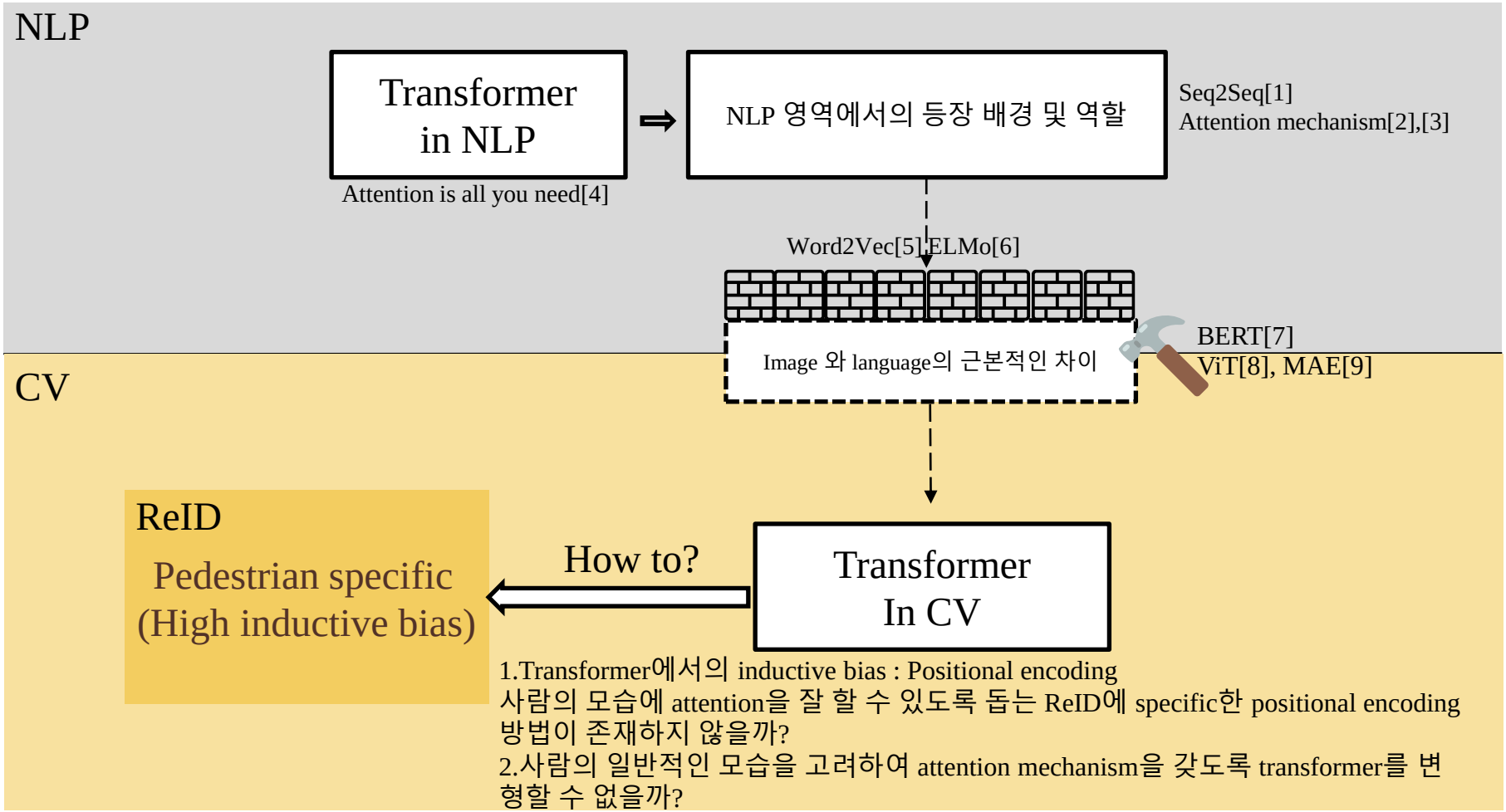
TEST_SIZE : [256, 256], METRIC: Euclidean, ATTENTION_VISUALIZE : mean



14	**Sets** (RK)	93.3	95.4	98.3	×	Sets for Person Re-identification		2019	ResNet
15	**TransReID-SSL** (ViT-B w/o RK)	93.2	96.7		✓	Self-Supervised Pre-Training for Transformer-Based Person Re-Identification		2021	Transformer

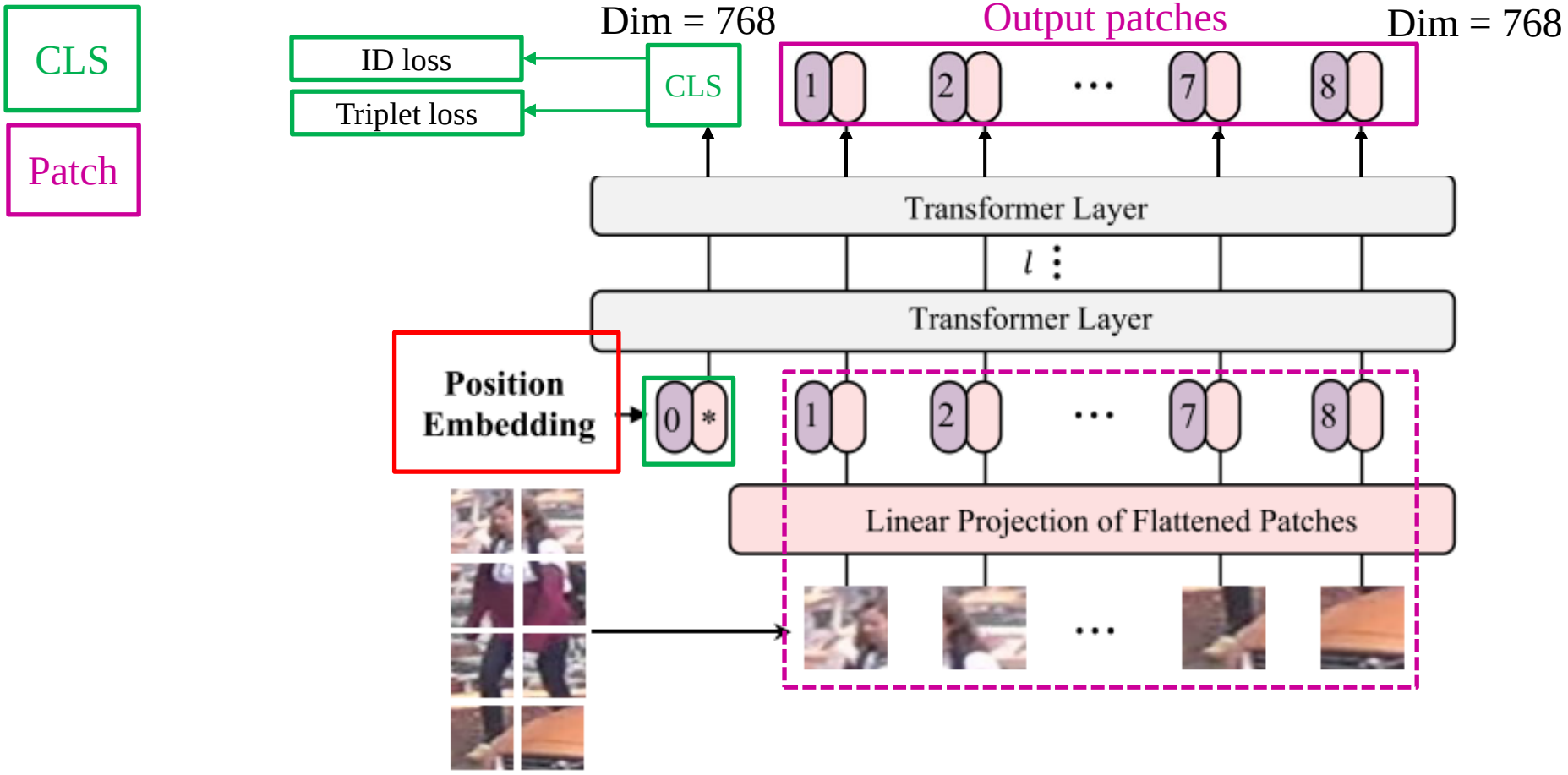
This week

- 2022/01/14 랩미팅 자료 中
  - *Transformer 기반의 model을 ReID에 사용할때 CNN에 비해 얻을 수 있는 장점은 무엇인가?*
  - *보행자의 모습만을 다루는 ReID에서 적용될 수 있는 Inductive bias가 있을까?*



This week

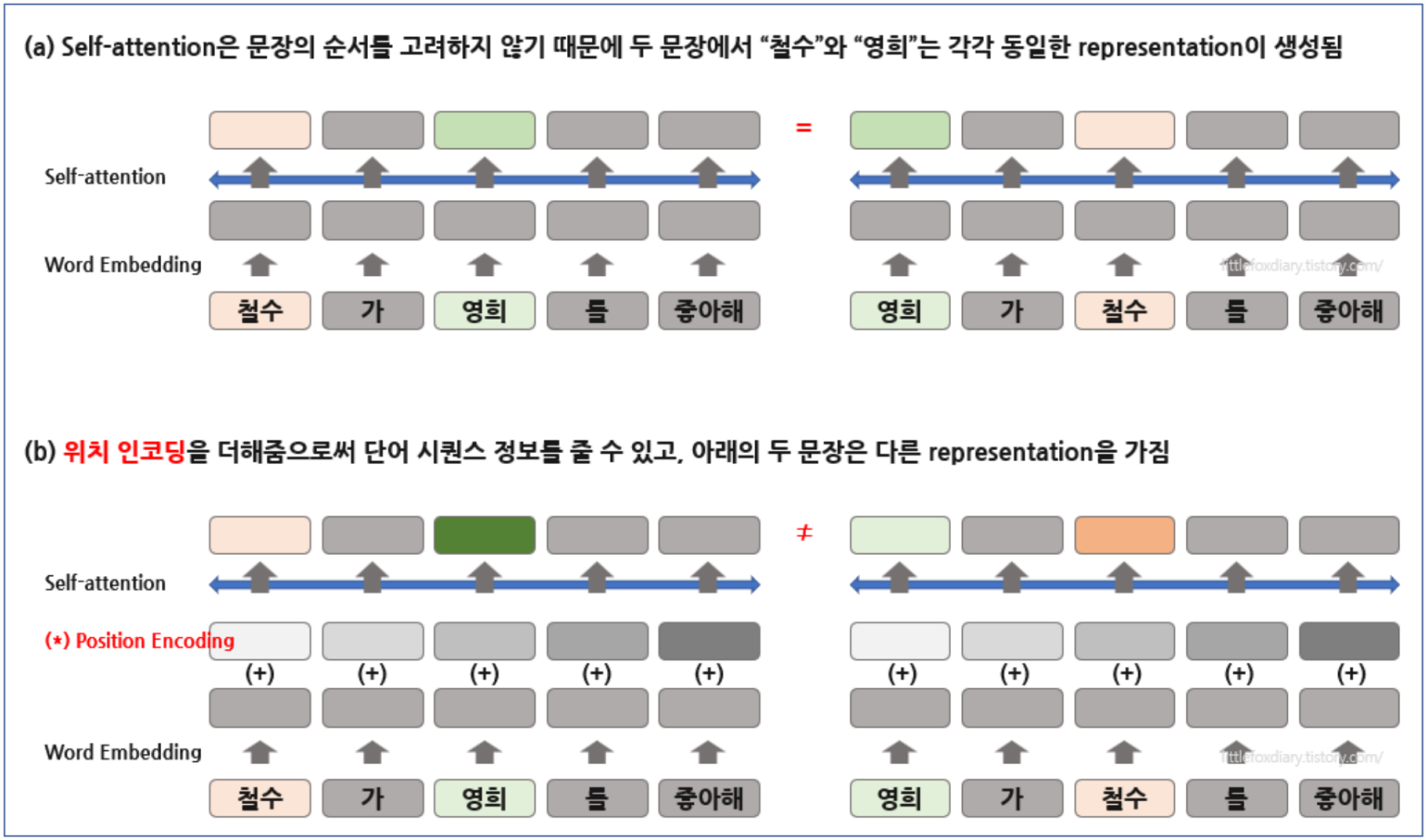
- *Person Re-identification using ViT*
  - *2. Relation between position encoding & person re-identification*



This week

[1]A. Vaswani *et al.*, *"Attention is all you need," in Advances in neural information processing systems, 2017, pp. 5998-6008.*

- *Positional encoding*
  - [1]의 attention mechanism에서 각 단어의 문장내에서의 위치를 학습 하기 위해 도입

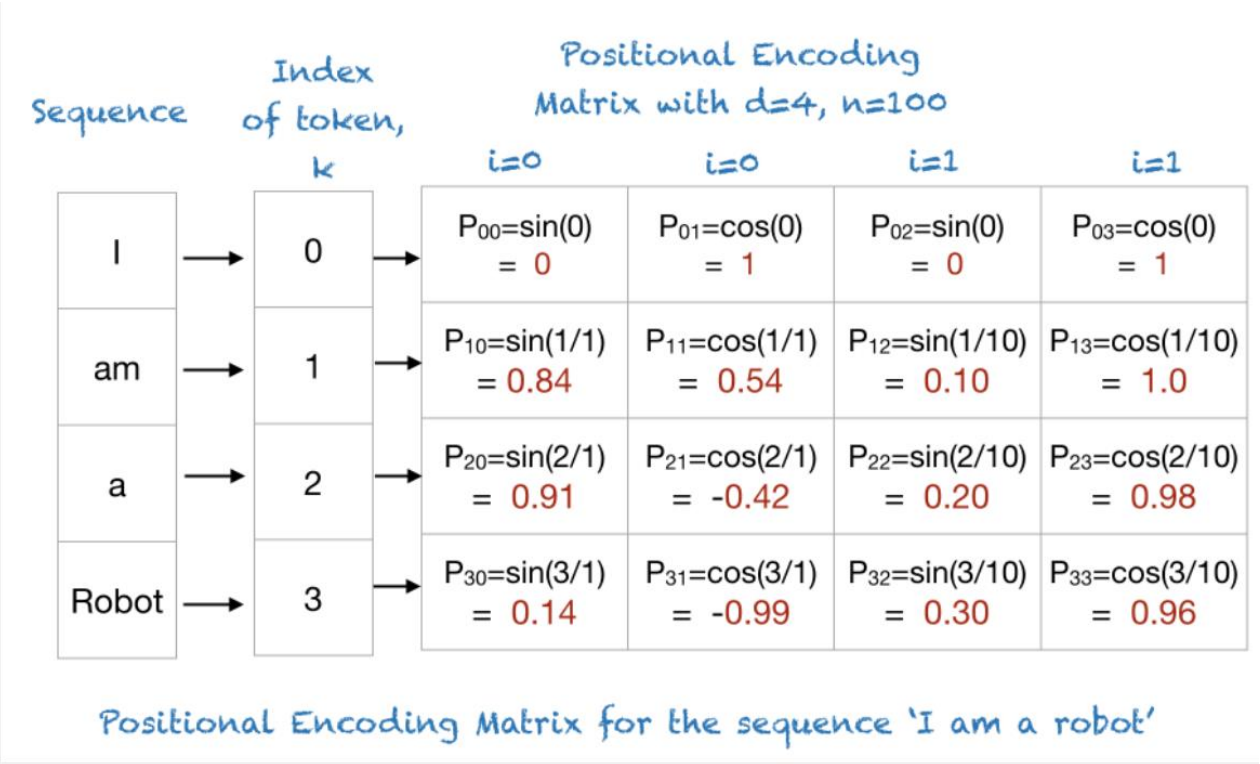


This week

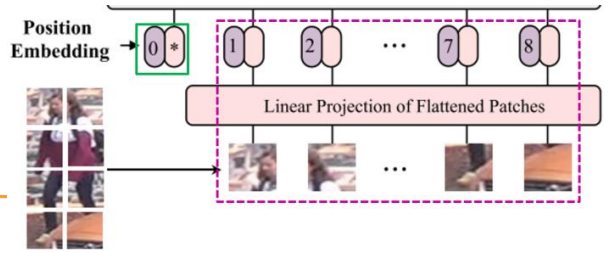
[1]A. Vaswani *et al., "Attention is all you need," in Advances in neural information processing systems, 2017, pp. 5998-6008.*

- *Positional encoding*
  - [1]의 attention mechanism에서 각 단어의 문장내에서의 위치를 학습 하기 위해 도입

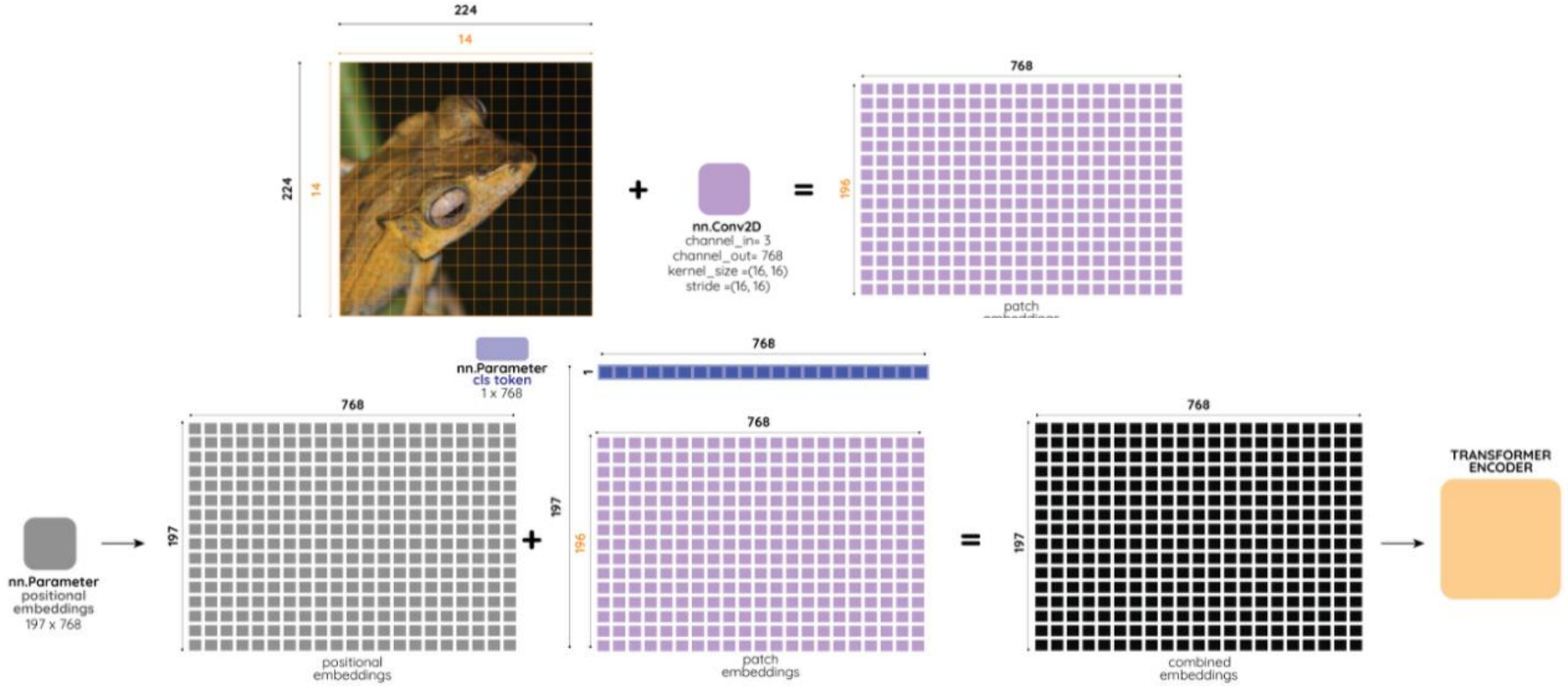
$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$



This week



- *Positional encoding in ViT*
  - *Input projection layer를 거친 patch token들에 summation 되는 learnable parameters*
  - *Model 전체 parameter의 극히 일부를 차지하나, 없을 경우 정상적인 학습이 이루어지지 않음*
    - *ViT # params : 89782312*
    - *Positional encoding # params : 768 * (192 + 1) = 148224 (0.165% of model parameters)*

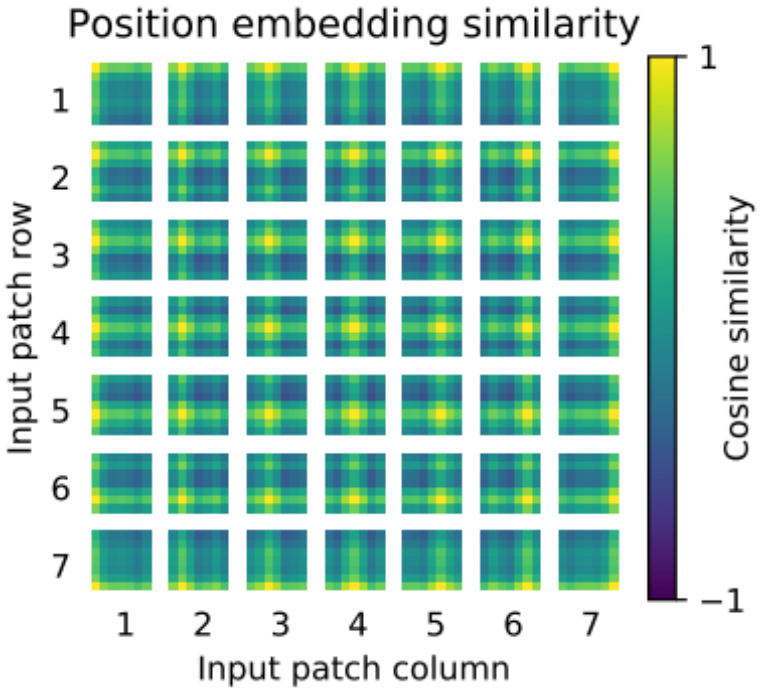


This week

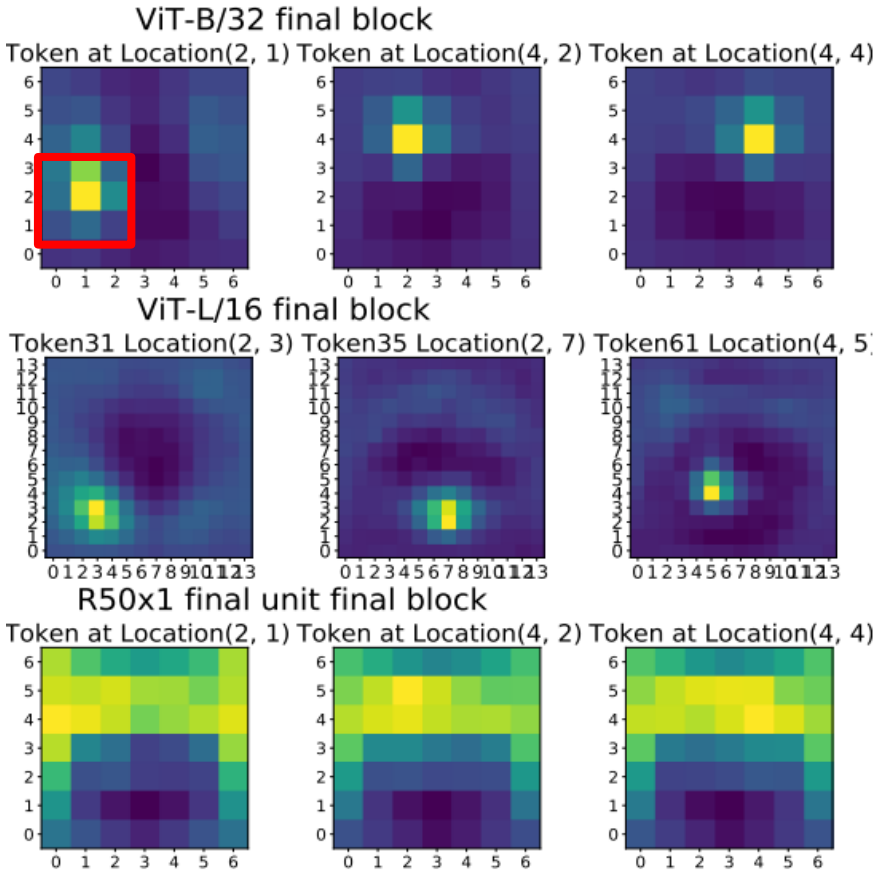
[2]A. Dosovitskiy *et al., "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.*
[3] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy, "Do vision transformers see like convolutional neural networks?," *Advances in Neural Information Processing Systems, vol. 34, 2021.*

김성수

- *Positional encoding in ViT*
  - *Input projection layer를 거친 patch token들에 summation 되는 learnable parameters*



Similarity between position embeddings[2]



Similarity between Input tokens and output tokens[3]

This week

- *Two kinds of positional encoding*
  - *Absolute positional encoding(APE) : 각 token의 절대적인 위치에만 의존하는 positional encoding*
  - *"문장의 의미는 단어들의 상대적인 위치에 의해 결정되지 않는가?"*
  - Ex) With APE :

철수가 영희를 좋아해

0 1 2

아 그러고보니, 철수가 영희를 좋아해

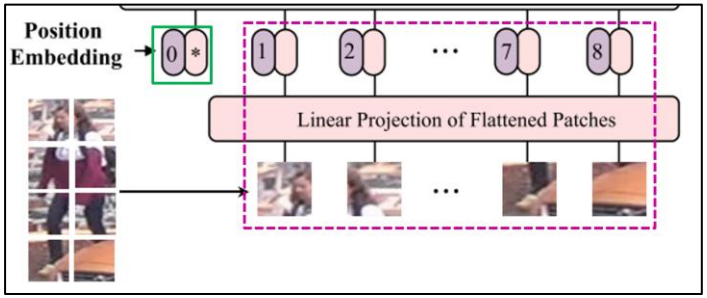
0 1 2 3 4

- 두 문장은 길이와 각 단어의 APE가 다르지만, 의미는 동일
  - 철수(주어)가 영희(목적어)보다 앞선 위치에 있음
  - 즉, 철수,영희,좋아해 세 단어의 상대적인 위치가 동일
    - → 단어간 상대적인 위치를 학습시키는 Relative positional encoding(RPE)이 등장

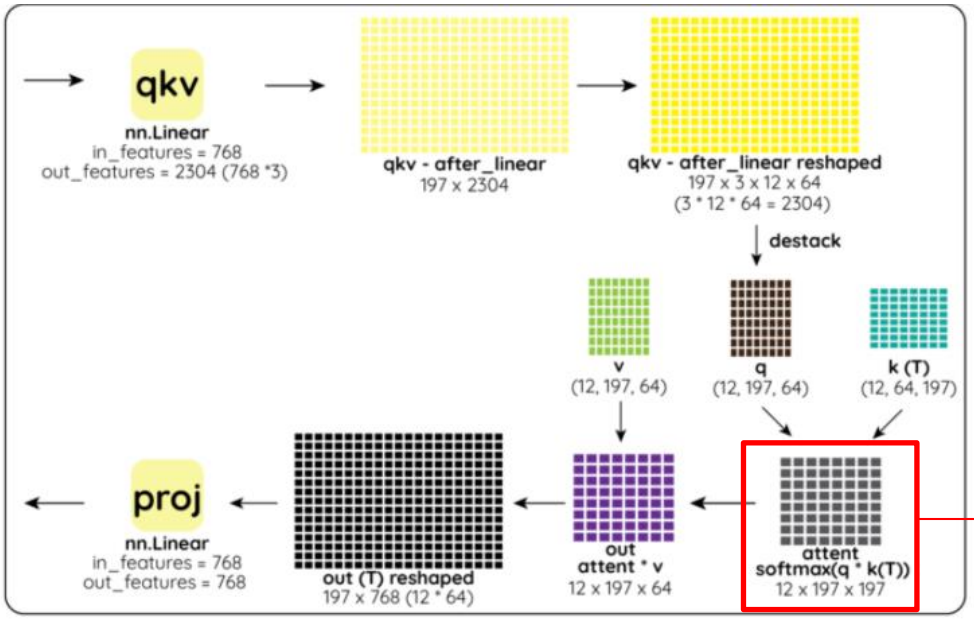
This week

[4]P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," *arXiv preprint arXiv:1803.02155, 2018.*

- *Relative positional encoding(RPE)*
  - Token들간의 상대적인 관계를 parameterize 할 수 있는 곳이 어디 인가?
    - APE : Patch가 encoder에 들어가기 직전



    - RPE[4] : Self-attention mechanism 내부



$$\text{Attention}(Q,K,V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Learnable Bias term

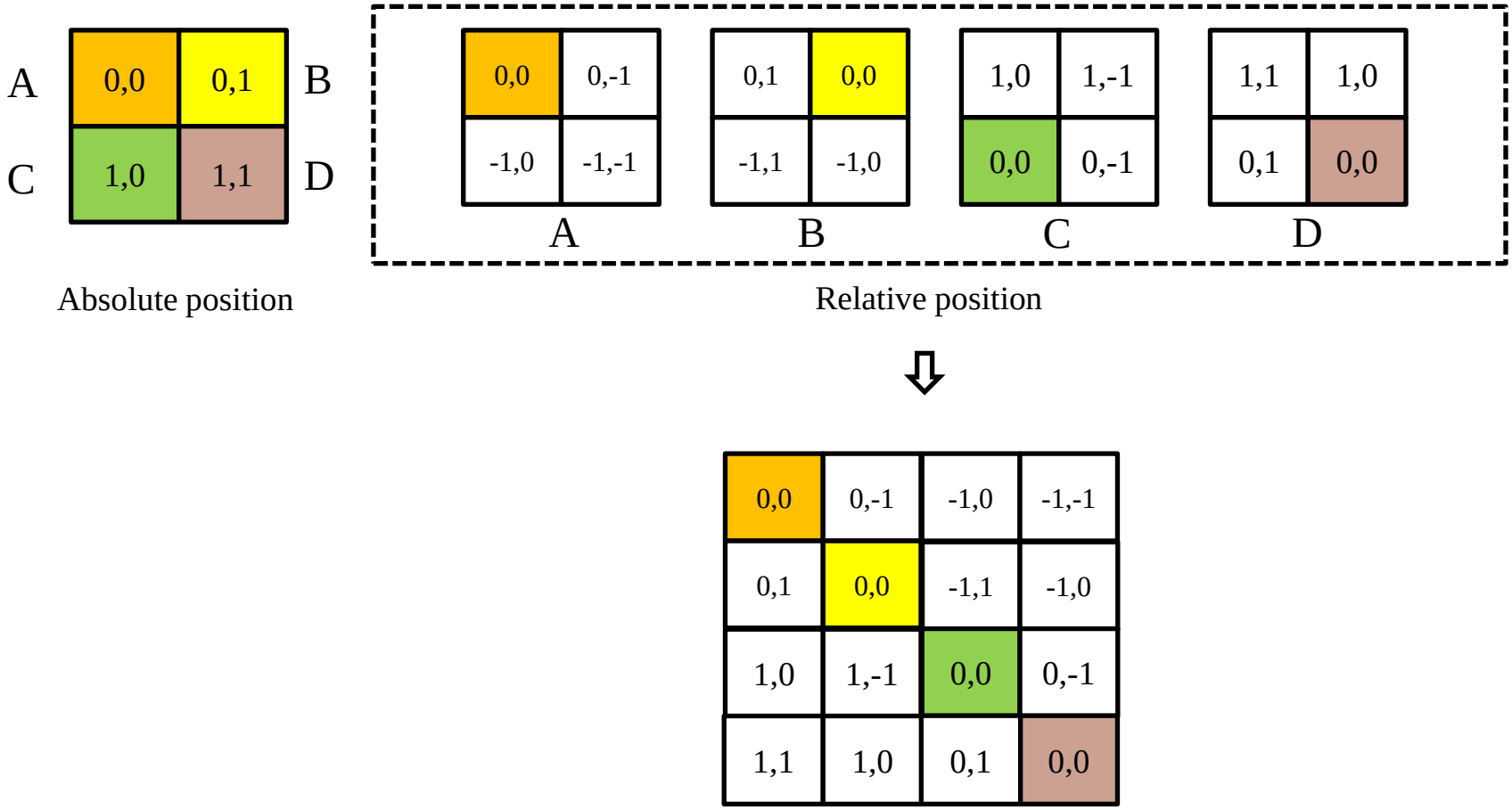
$$e_{ij} = \frac{x_i W^Q (x_j W^K + a_{ij}^K)^T}{\sqrt{d_z}}$$

Size of relative position parameters per layer
→ 12 x 197 x 197

RPE # params?

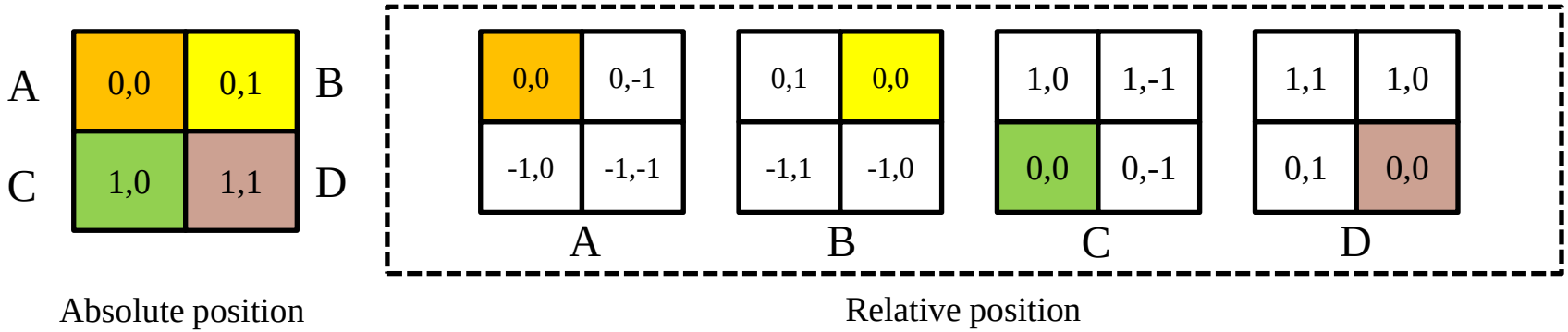
This week

- *Example*
  - *2x2 patch*



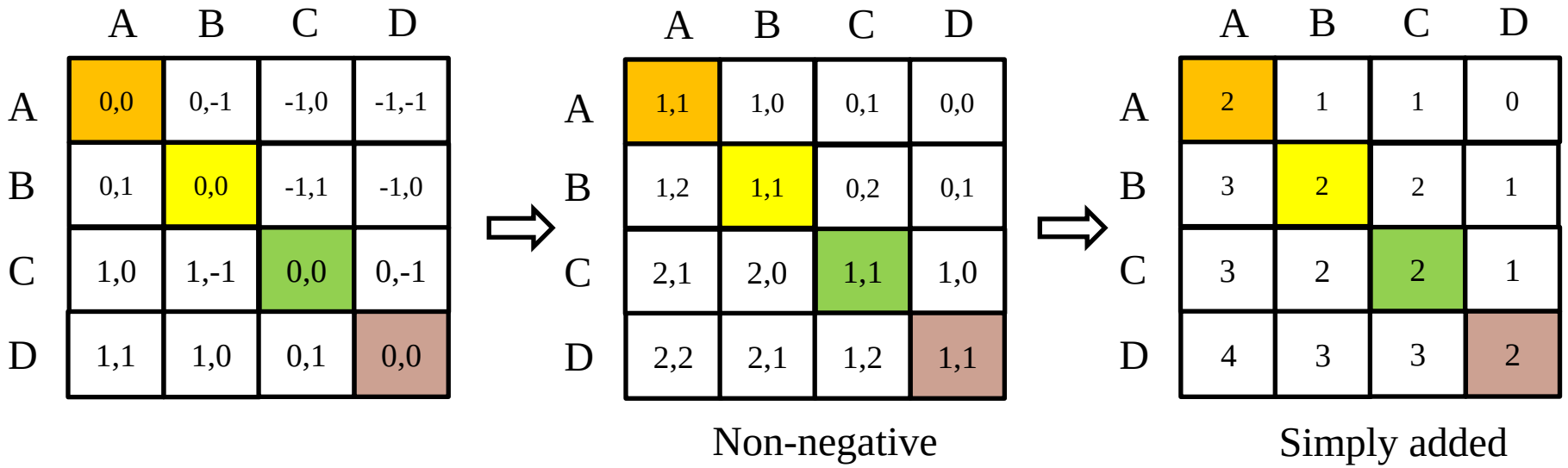
This week

- *Example*
  - *2x2 patch*



Absolute position

Relative position

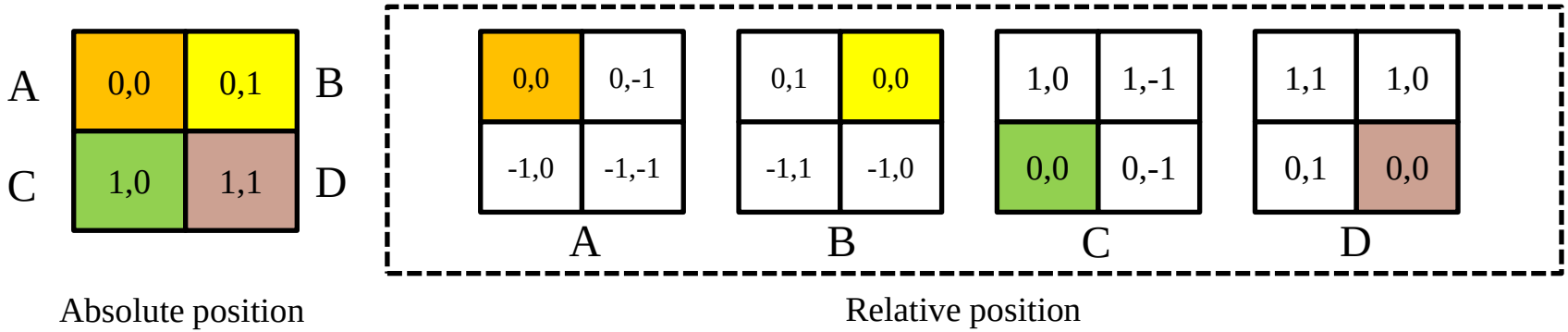


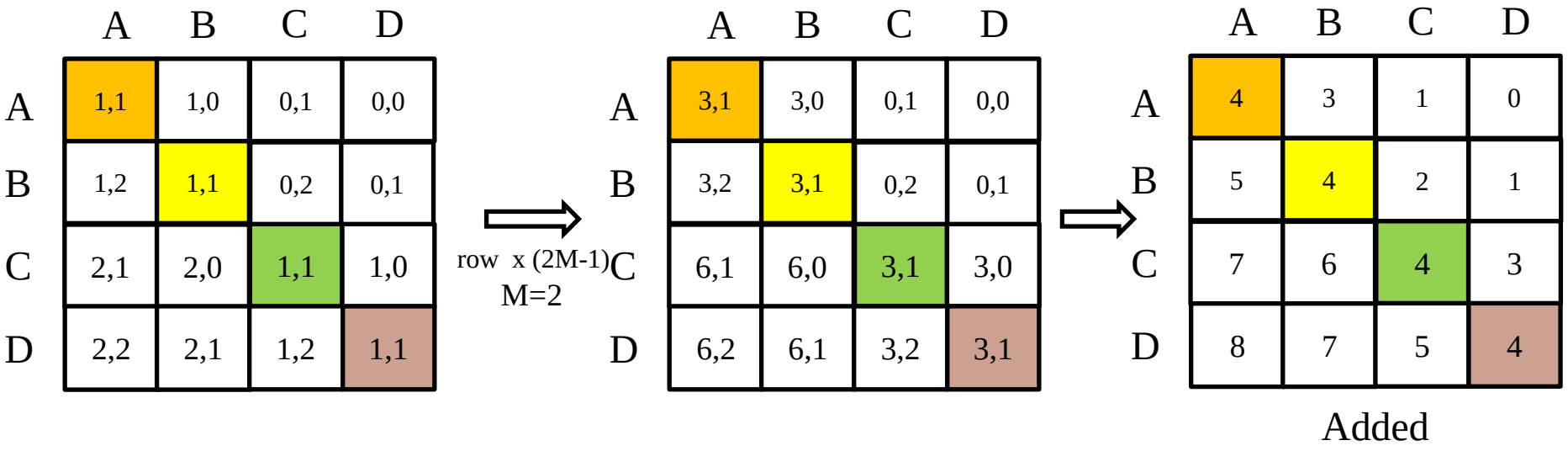
Non-negative

Simply added

This week

- *Example*
  - *2x2 patch*





This week

- *Example*
  - *2x2 patch*

0	1	2	3	4	5	6	7	8
0.1	0.03	0.8	0.5	0.07	0.2	0.4	0.01	0.06

Relative position bias table

A	0,0	0,1 B
C	1,0	1,1 D

Absolute position

	A	B	C	D
A	4	3	1	0
B	5	4	2	1
C	7	6	4	3
D	8	7	5	4

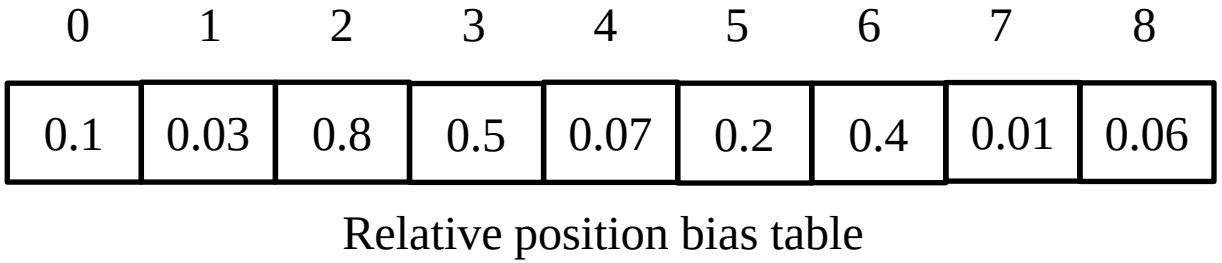
Relative position index

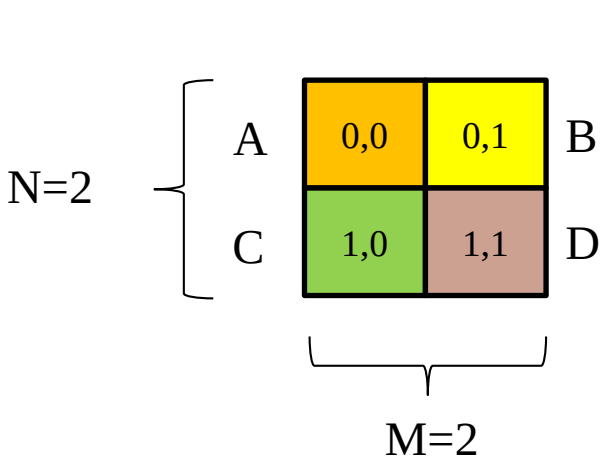
	A	B	C	D
A	0.07	0.5	0.03	0.1
B	0.2	0.07	0.8	0.03
C	0.01	0.4	0.07	0.5
D	0.06	0.01	0.2	0.07

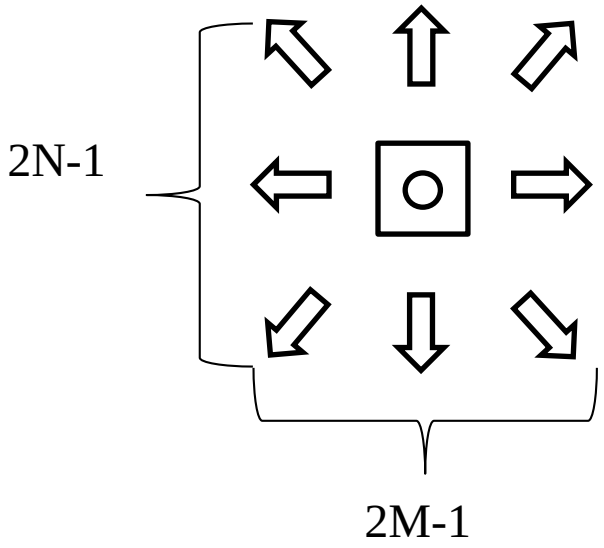
Relative position bias

This week

- *Example*
  - *2x2 patch*





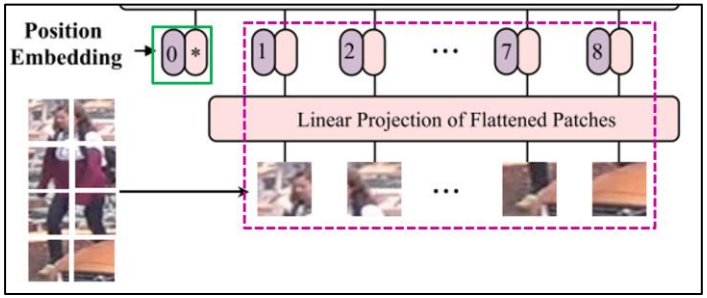


  - *Patch의 개수가 N x M 이라 할때, relative position parameter는 (2N-1) x (2M-1)개*

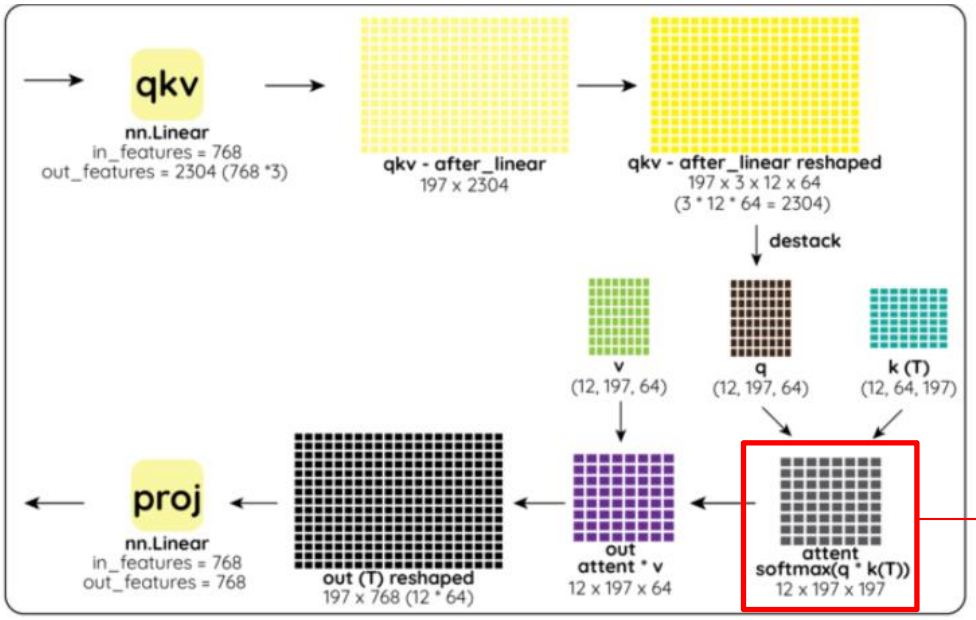
This week

[4]P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," *arXiv preprint arXiv:1803.02155, 2018.*

- *Relative positional encoding(RPE)*
  - Token들간의 상대적인 관계를 parameterize 할 수 있는 곳이 어디 인가?
    - APE : Patch가 encoder에 들어가기 직전



    - RPE[4] : Self-attention mechanism 내부



$$\text{Attention}(Q,K,V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Learnable Bias term

$$e_{ij} = \frac{x_i W^Q (x_j W^K + a_{ij}^K)^T}{\sqrt{d_z}}$$

Size of relative position parameters per layer
→ 12 x 197 x 197

RPE # params?
197 = (14 x 14 +1)
→ 12 (heads) x 27 x 27 x 12 (layers)
→ 104796

This week

- *Relative positional encoding(RPE)*
  - "Image에서 relative position embedding은 어떤 의미를 갖는가?"
    - Ex) ViT로 ImageNet을 학습

    - 모델이 학습하는 것
      - APE : n번째 patch와 m번째 patch 사이의 관계 (ex: n 번째와 n+row 번째 patch간의 관계)
      - RPE : 거리 (방향이 고려된) 가 k인 두 patch 간의 관계
    - Image간 correlation이 낮은 dataset은 patch 간 relative distance가 약함

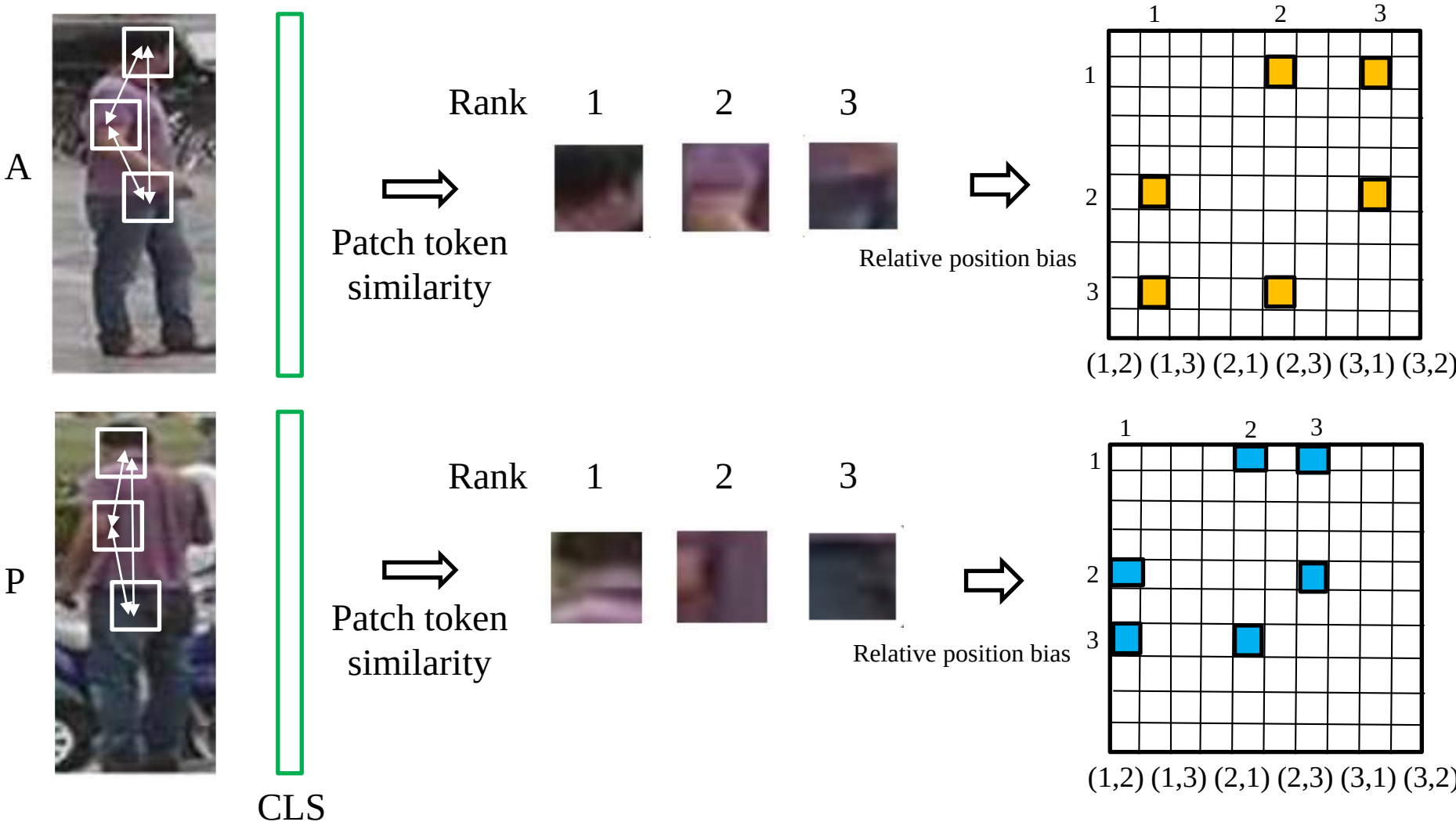
This week

- *Relative positional encoding(RPE)*
  - "Image에서 relative position embedding은 어떤 의미를 갖는가?"
    - Ex) ViT로 ReID dataset을 학습

    - ReID dataset : 모든 data의 대상이 사람이기 때문에 Sample간 correlation이 높음
    - 모델이 학습하는 것
      - APE : n번째 patch와 m번째 patch 사이의 관계 (ex: n 번째와 n+row ,n + 2*row, n + 3*row..patch간의 관계)
      - RPE : k만큼 떨어진 두 신체부위를 나타내는 patch사이의 관계

Proposed method

- *Relative position triplet loss*
  - *Idea : "같은 ID의 image 끼리는 주요 patch 들간 relative position bias 가 비슷할 것이다."*



Proposed method

- *Relative position triplet loss*
  - *Idea : "같은 ID의 image끼리는 주요 patch들간 relative position bias가 비슷할 것이다."*

0,0	0,1	B
1,0	1,1	D

A (top-left), C (bottom-left)

Absolute position

0	1	2	3	4	5	6	7	8
0.1	0.03	0.8	0.5	0.07	0.2	0.4	0.01	0.06

Relative position bias table

	A	B	C	D
A	4	3	1	0
B	5	4	2	1
C	7	6	4	3
D	8	7	5	4

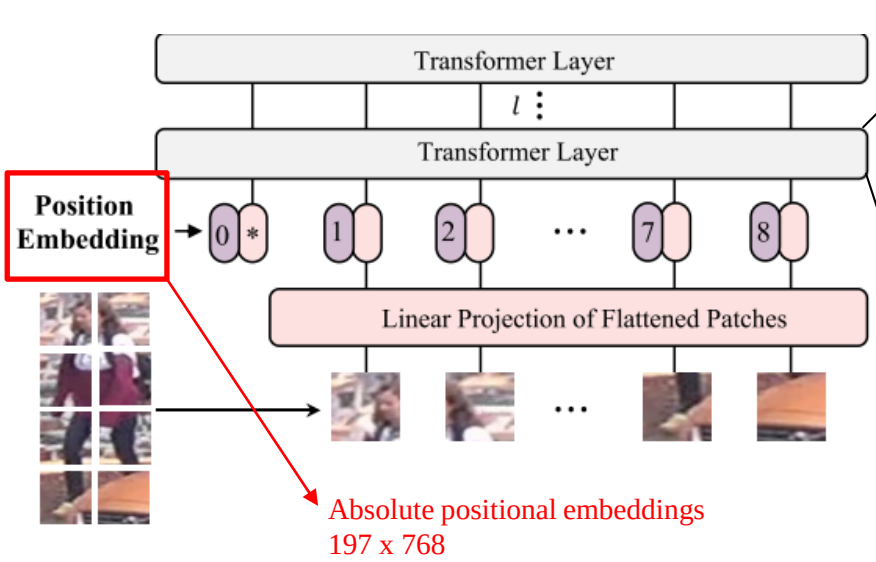
Relative position index

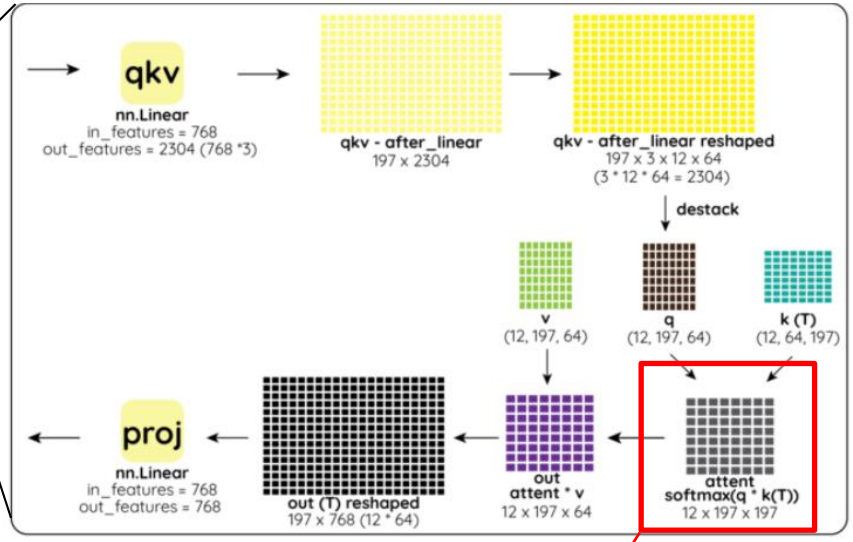
	A	B	C	D
A	0.07	0.5	0.03	0.1
B	0.2	0.07	0.8	0.03
C	0.01	0.4	0.07	0.5
D	0.06	0.01	0.2	0.07

Relative position bias

Proposed method

- *Relative position triplet loss*
  - *같은 relative position 을 갖는 patch 들이 영향을 최소화 하도록 absolute position 을 함께 사용*





Relative positional embeddings

$$e_{ij} = \frac{x_i W^Q (x_j W^K + a_{ij}^K)^T}{\sqrt{d_z}}$$

Learnable Bias term

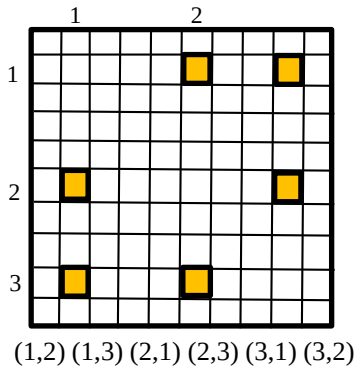
12 x 197 x 197 x 12

CLS

Rank 1 2 3

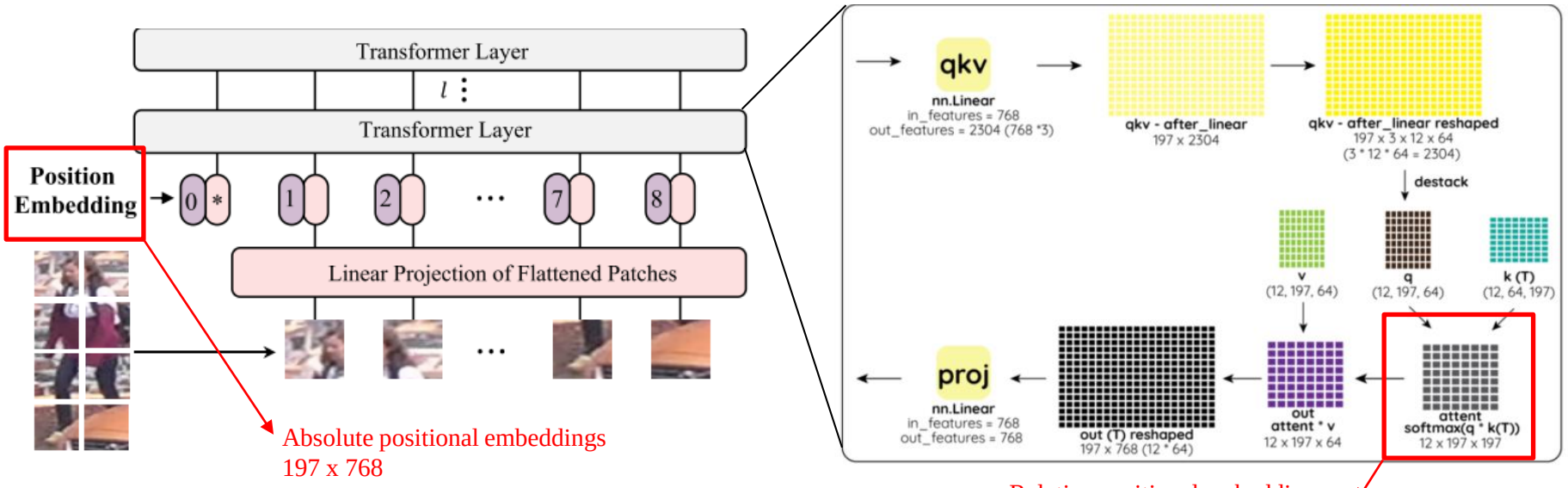
Patch token similarity

Relative position bias



Proposed method

- *Relative position triplet loss*
  - *같은 relative position 을 갖는 patch 들이 영향을 최소화 하도록 absolute position 을 함께 사용*



$$e_{ij} = \frac{x_i W^Q (x_j W^K + a_{ij}^K)^T}{\sqrt{d_z}}$$

Attention score $= (Q_1 + P_{abs1})(K_2 + P_{abs2})^T + P_{rel12}$

Positional relationship between patch 1 and 2

→ $P_{abs1} \cdot {P_{abs2}}^T + P_{rel12}$

Proposed method

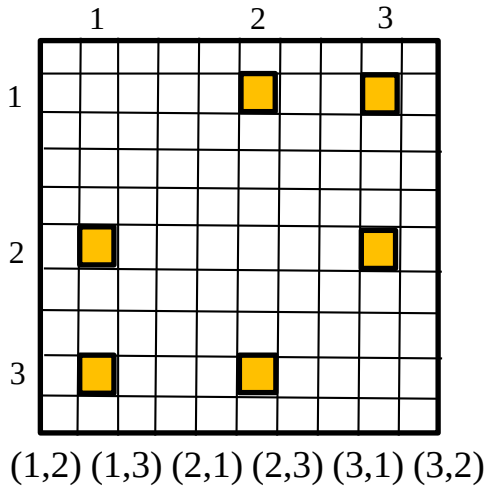
- *Relative position triplet loss*
  - *같은 relative position 을 갖는 patch 들이 영향을 최소화 하도록 absolute position 을 함께 사용*
    - Positional relationship between patch 1 and 2 : $P_{abs1} \cdot {P_{abs2}}^{T} + P_{rel12}$

CLS

Patch token similarity

Rank 1 2 3

Relative position bias



- *Algorithm*

1. *Anchor, Positive, Negative sample 각각에 대해 CLS token 과 similarity 가 높은 N 개의 patch 선별 (sorted)*
2. *N 개의 patch 에 대해 가능한 combination ${}_{N}C_{2}$ 개의 쌍 (a,b) 에 대해 positional relation: $P_{abs\_a} \cdot {P_{abs\_b}}^{T} + P_{rel\_ab}$ 를 계산하여 원소의 개수가 ${}_{N}C_{2}$ 개인 patch distance vector (ex: [relation(1,2), relation(1,3), relation(2,3)]) 생성*
3. *Anchor, Positive, Negative 의 positional relation vector 에 대해 triplet loss 를 적용*

Ex)

Anchor : 3,4,5 번 patch

Positive : 7,10,11 번 patch

→ Anchor 의 (3,4) 번 patch 간 position 관계는 positive 의 (7,10) 번 patch 간 position 관계와 유사해야 한다!

Proposed method

- *Relative position triplet loss*
  - *Algorithm*
  1. *Anchor, Positive, Negative sample 각각에 대해 CLS token과 similarity가 높은 N개의 patch 선별 (sorted)*
  2. *N개의 patch에 대해 가능한 combination ${}_NC_2$ 개의 쌍 (a,b)에 대해 positional relation: $P_{abs_a} \cdot P_{abs_b}{}^T + P_{rel_ab}$를 계산하여 원소의 개수가 ${}_NC_2$ 개인 patch distance vector (ex: [relation(1,2), relation(1,3), relation(2,3)]) 생성*
  3. *Anchor, Positive, Negative의 positional relation vector에 대해 triplet loss를 적용*
- *Experimental result*
  - *초반 학습이 기존 방식보다 꽤 앞서지만, 후반부 loss가 수렴하지 못하는 현상이 매번 발생 (Learning rate 문제 X)*
  - *Negative와 Anchor의 positional relation vector간 거리를 크게 만드는 것에서 loss가 발산*