Point-BERT

Pre-training 3D Point Cloud Transformers with Masked Point Modeling

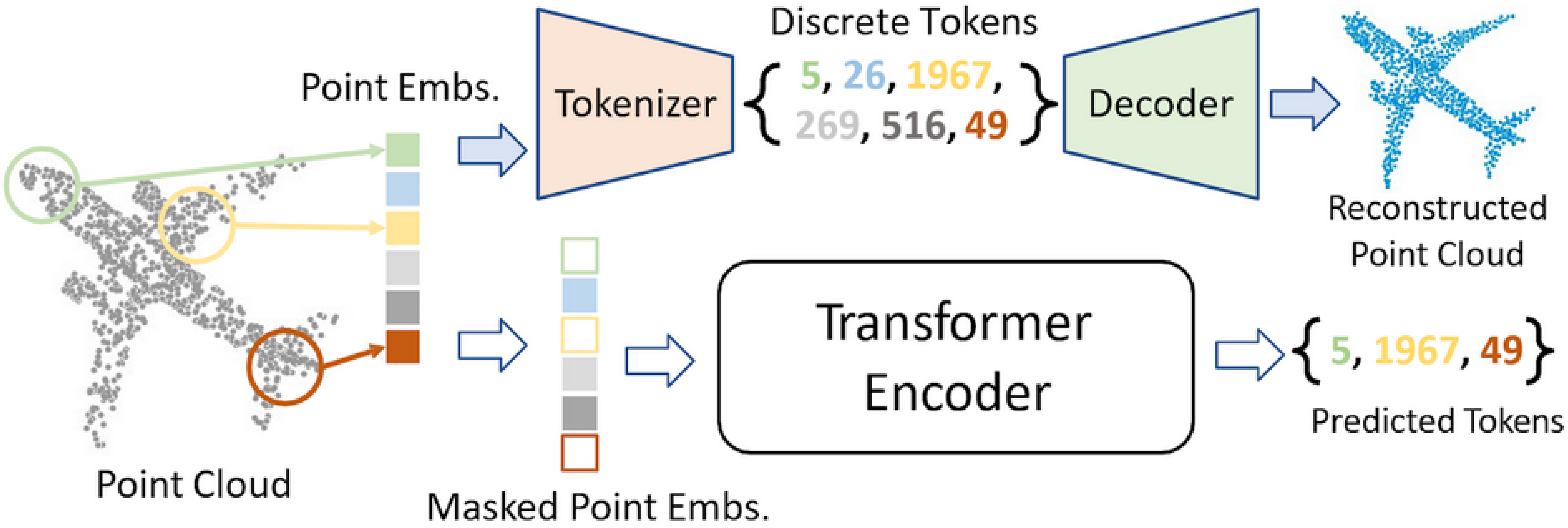
Yu, Tang, Rao, Huang, Zhou & Lu

Presented by

William Guimont-Martin

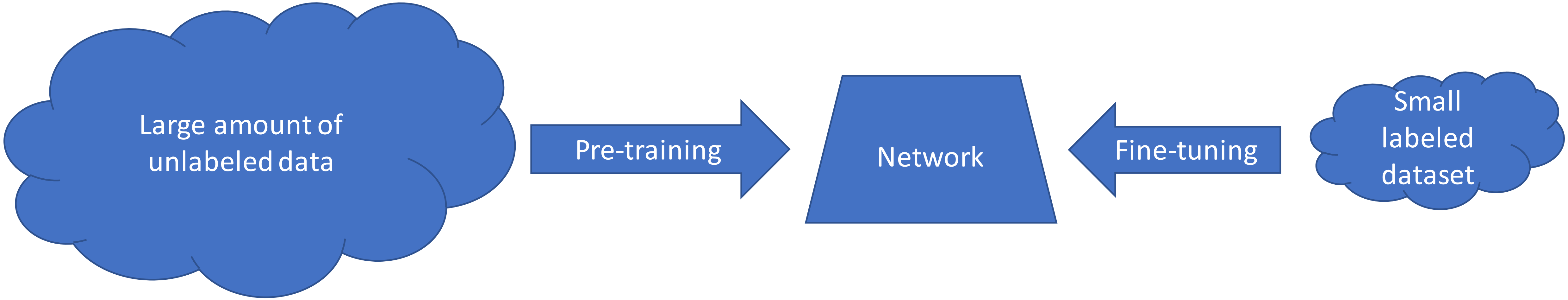
Master's Student

Point-BERT


Point Embs.
Tokenizer
Discrete Tokens
{ 5, 26, 1967, 269, 516, 49 }
Decoder
Reconstructed Point Cloud
Point Cloud
Masked Point Embs.
Transformer Encoder
{ 5, 1967, 49 }
Predicted Tokens

Self-Supervised Learning (SSL)

- Labeling data is hard and costly
- Generate its own supervision from the data
  - No label needed
- Pre-training tasks
- Self-supervised learning: The dark matter of intelligence by Yann LeCun



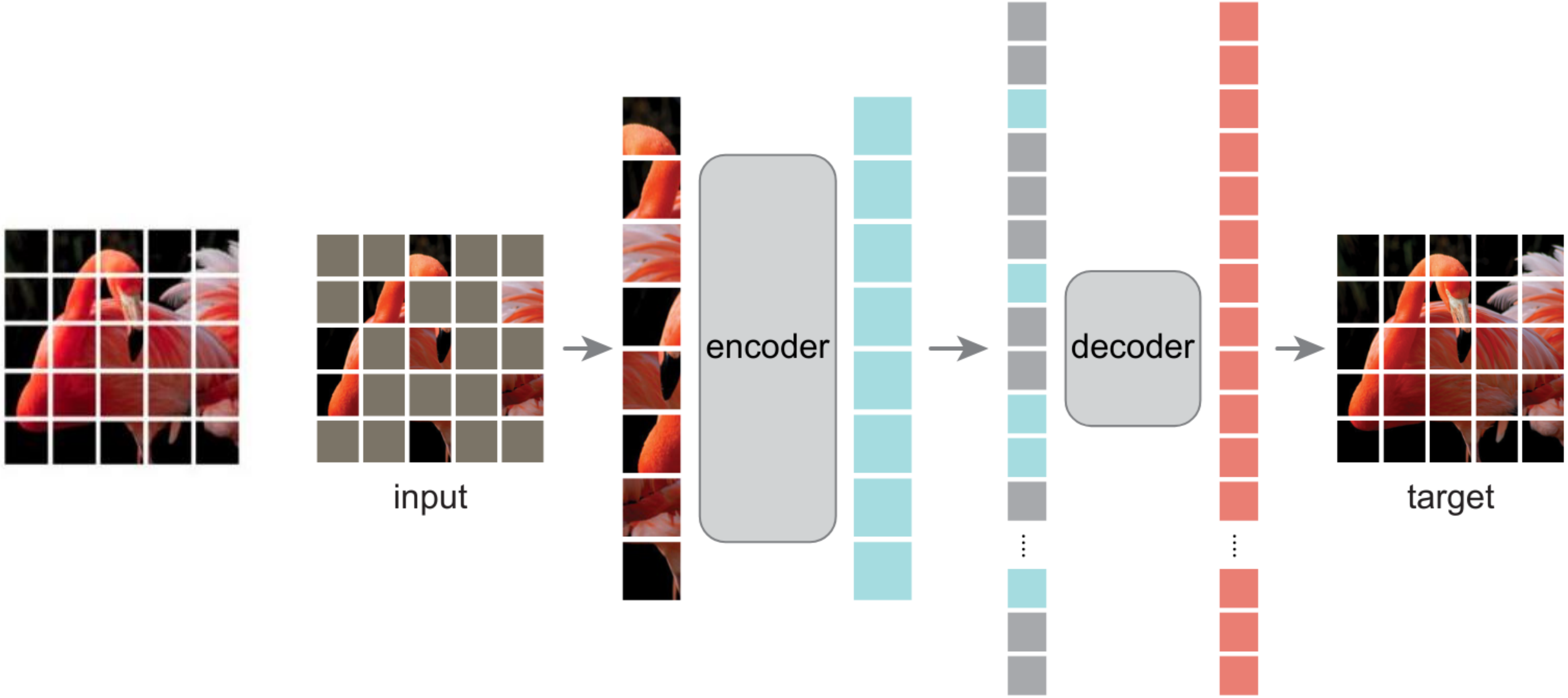
Pre-training:
BERT's Masked language modeling

The quick brown fox ~~jumps~~ over the lazy dog

The quick brown fox jumps over the lazy dog

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

Pre-training: Masked Autoencoder



Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. arXiv preprint arXiv:2111.06377, 2021.

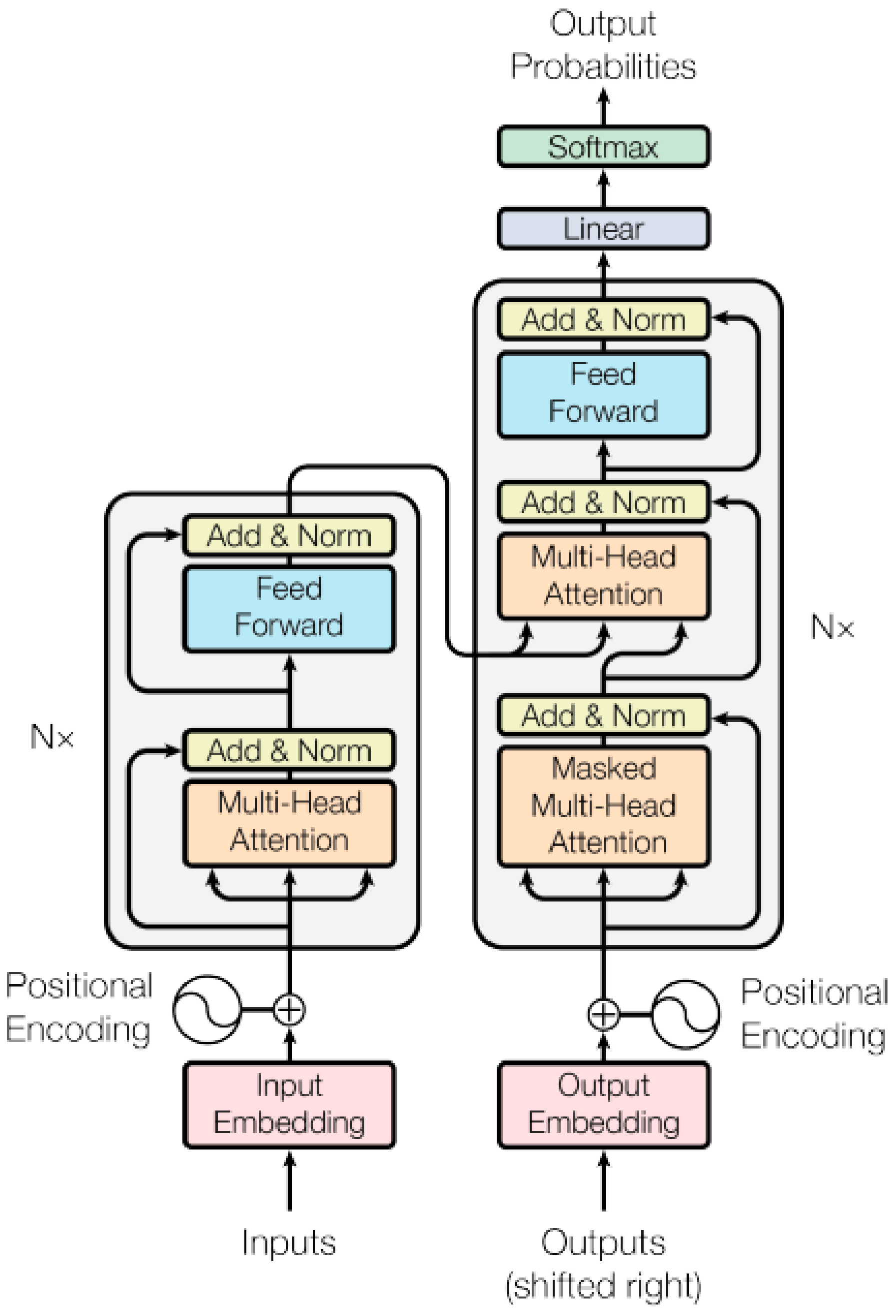


Figure 1: The Transformer - model architecture.

Transformers in NLP

- [Attention Is All You Need](#) (2017)
- Revolution in NLP
  - GPT-3 (Generative Pre-trained **Transformer** 3)
    - 175 billion parameters
    - 499 billion tokens
  - BERT (Bidirectional Encoder Representations from **Transformers**)
    - 110 million parameters

Transformers

The quick brown fox jumps over the lazy dog

- Can attend everywhere
  - Less inductive biases than CNN
- O(n^2)
  - Limits the number of tokens
- Set to Set
  - Positional encoding

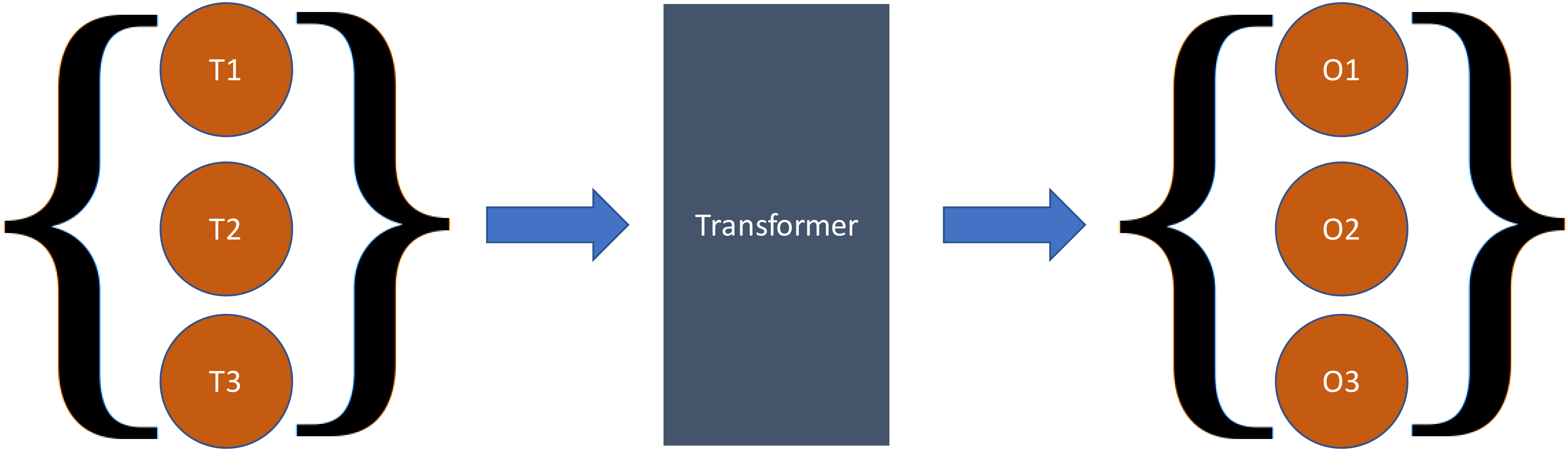
Input: The quick brown fox ~~jumps~~ over the lazy dog

Output: The quick brown fox jumps over the lazy dog

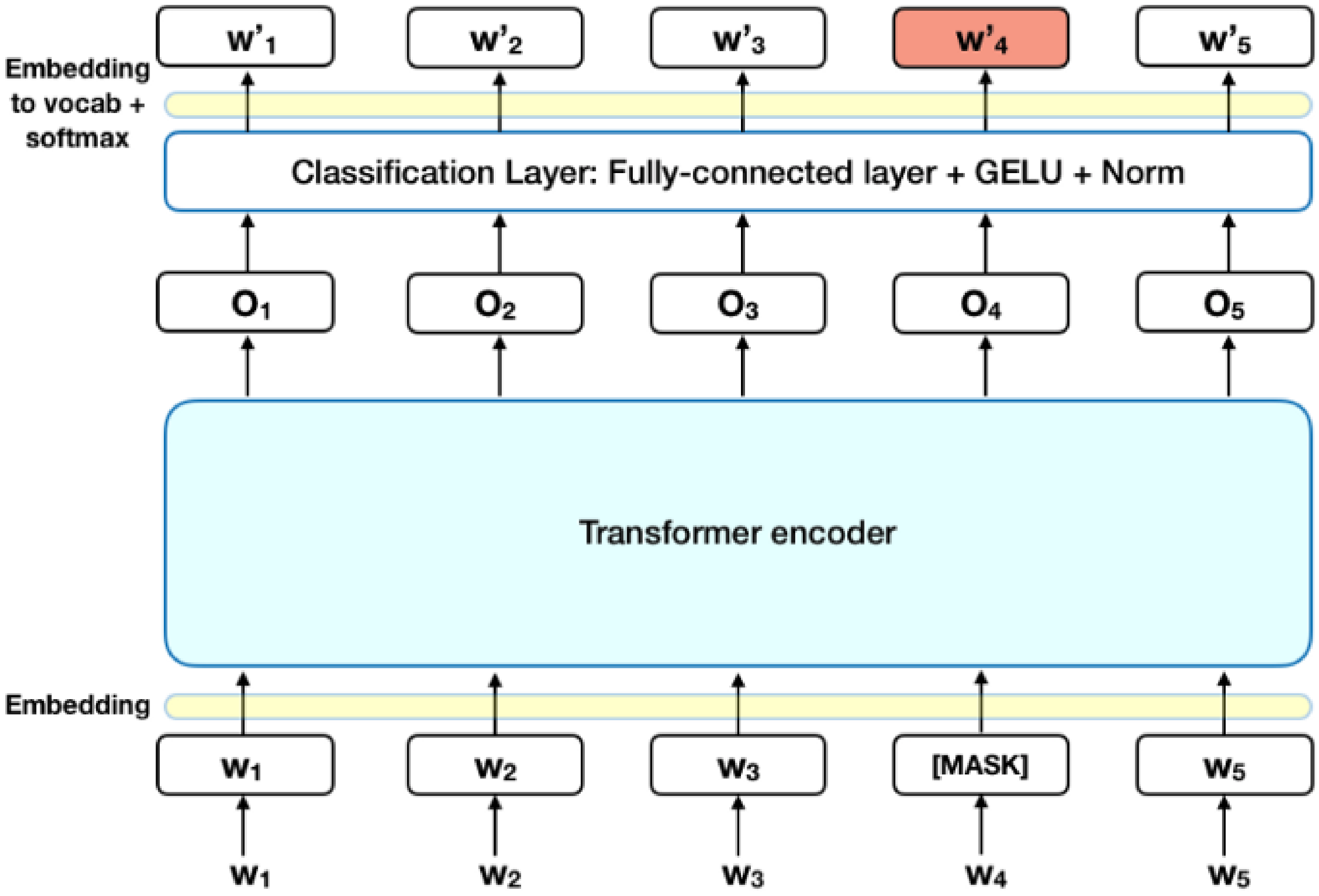
Transformers

The quick brown fox jumps over the lazy dog

The lazy dog jumps over the quick brown fox

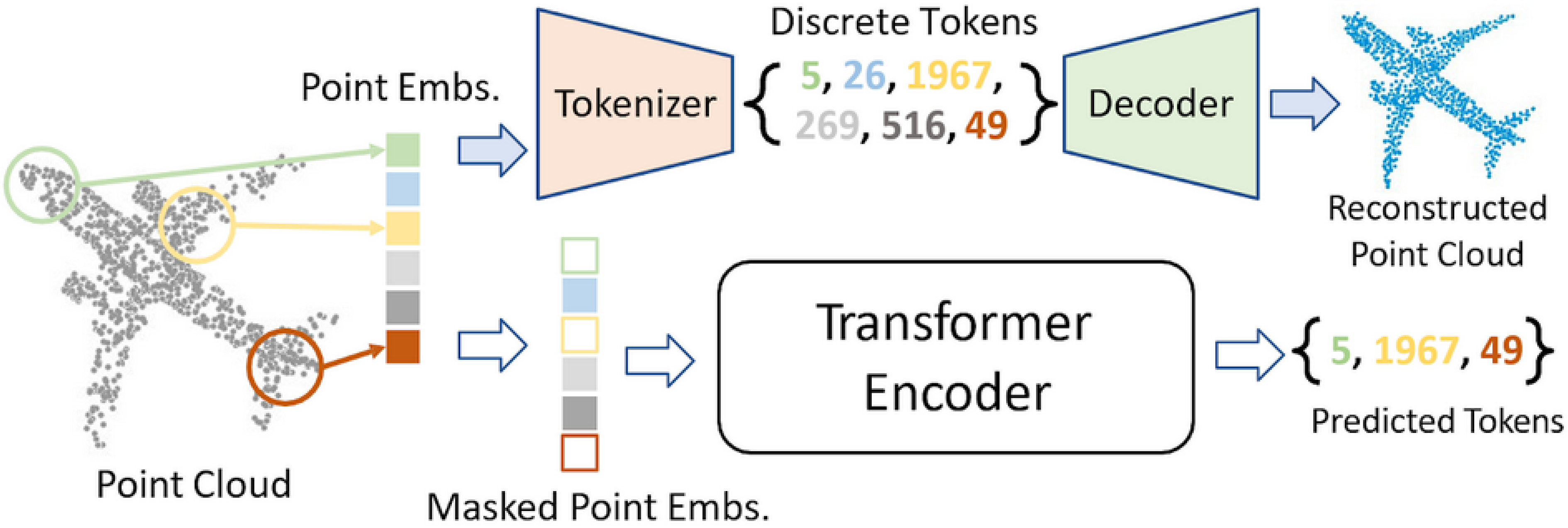


BERT



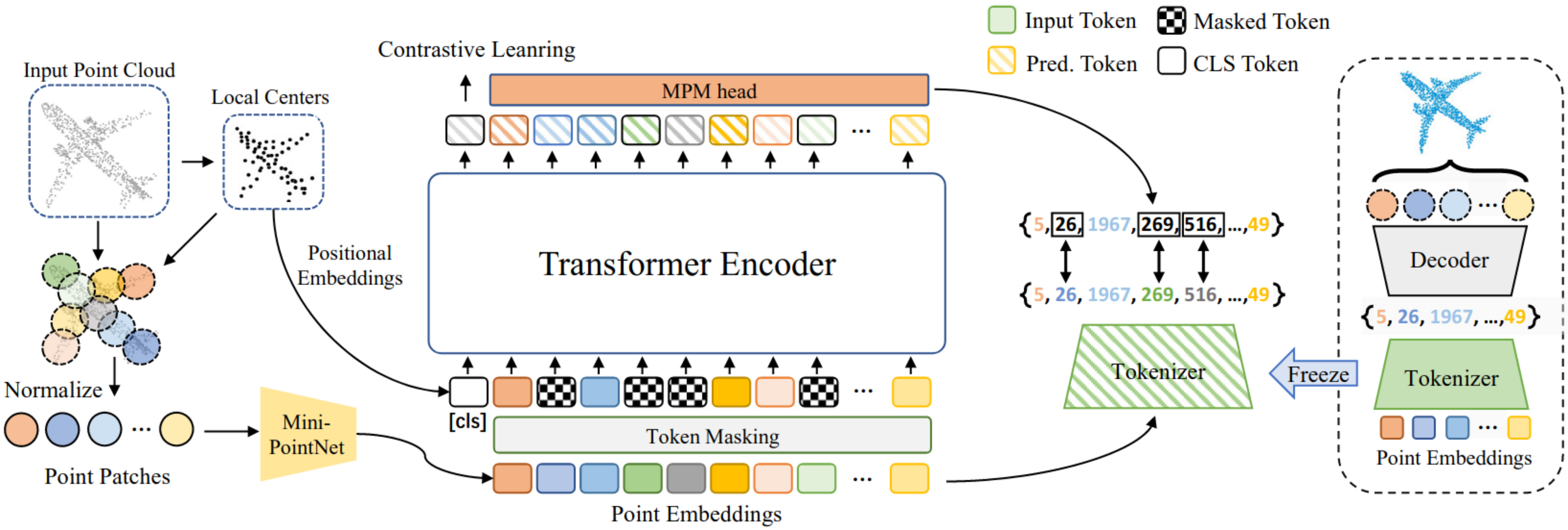
https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

Point-BERT


Point Embs.
Tokenizer
Discrete Tokens
{ 5, 26, 1967, 269, 516, 49 }
Decoder
Reconstructed
Point Cloud
Transformer
Encoder
{ 5, 1967, 49 }
Predicted Tokens
Point Cloud
Masked Point Embs.

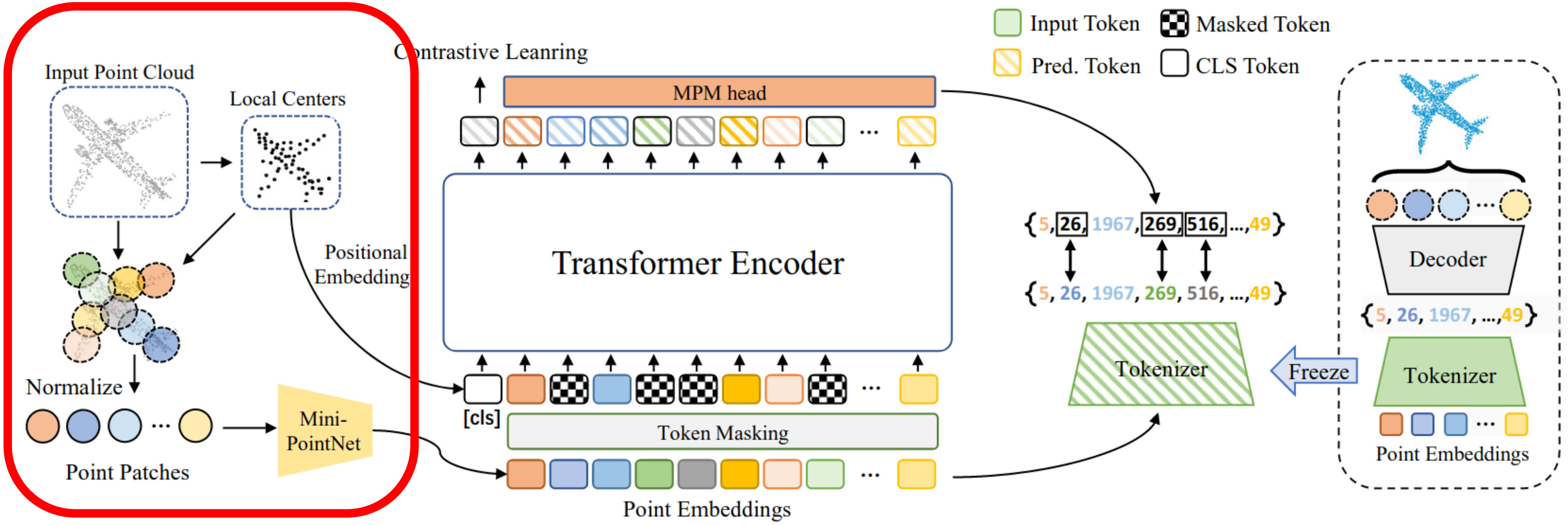
Point-BERT

Pre-training 3D Point Cloud Transformers with Masked Point Modeling



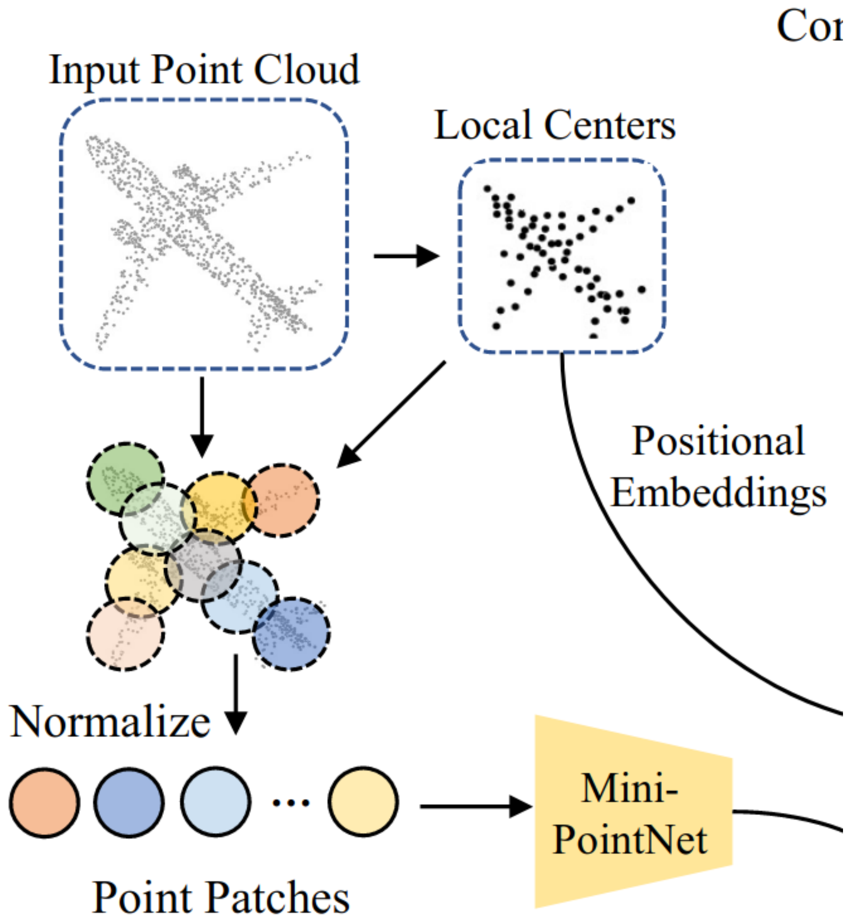
Point-BERT

Pre-training 3D Point Cloud Transformers with Masked Point Modeling



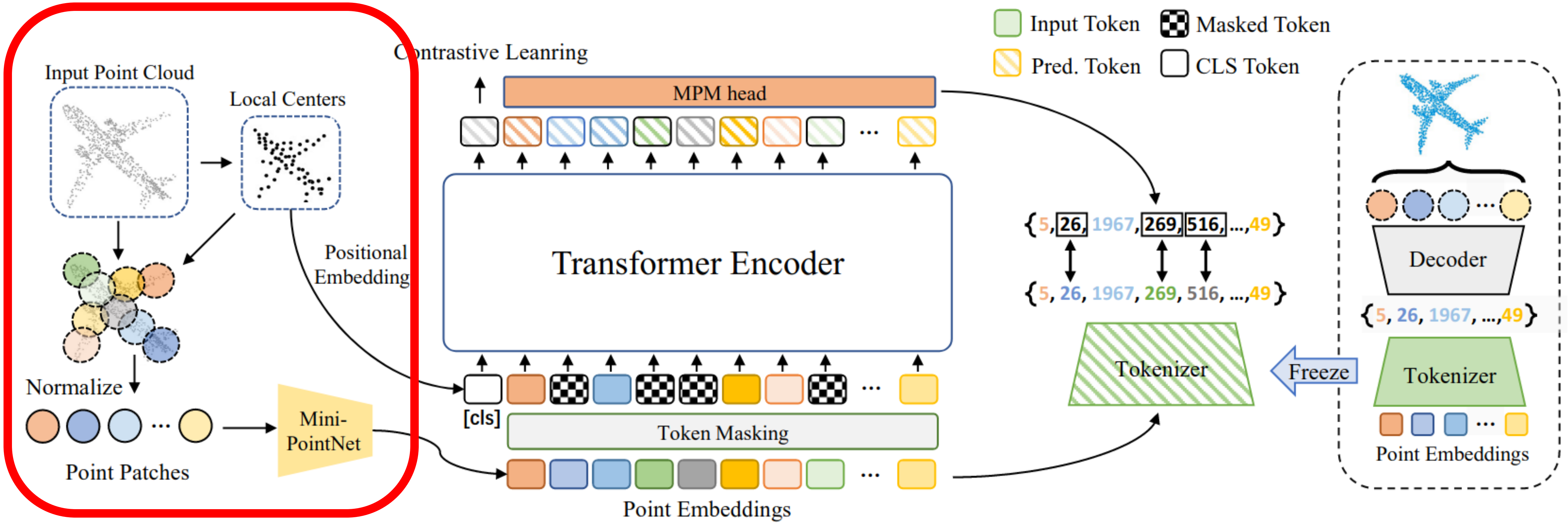
Tokenization

- Naive approach
  - O(n^2)
- Patches
  - Farthest Point Sampling
    - Like in PointNet++
  - kNN
  - Center the patch
  - Mini-PointNet



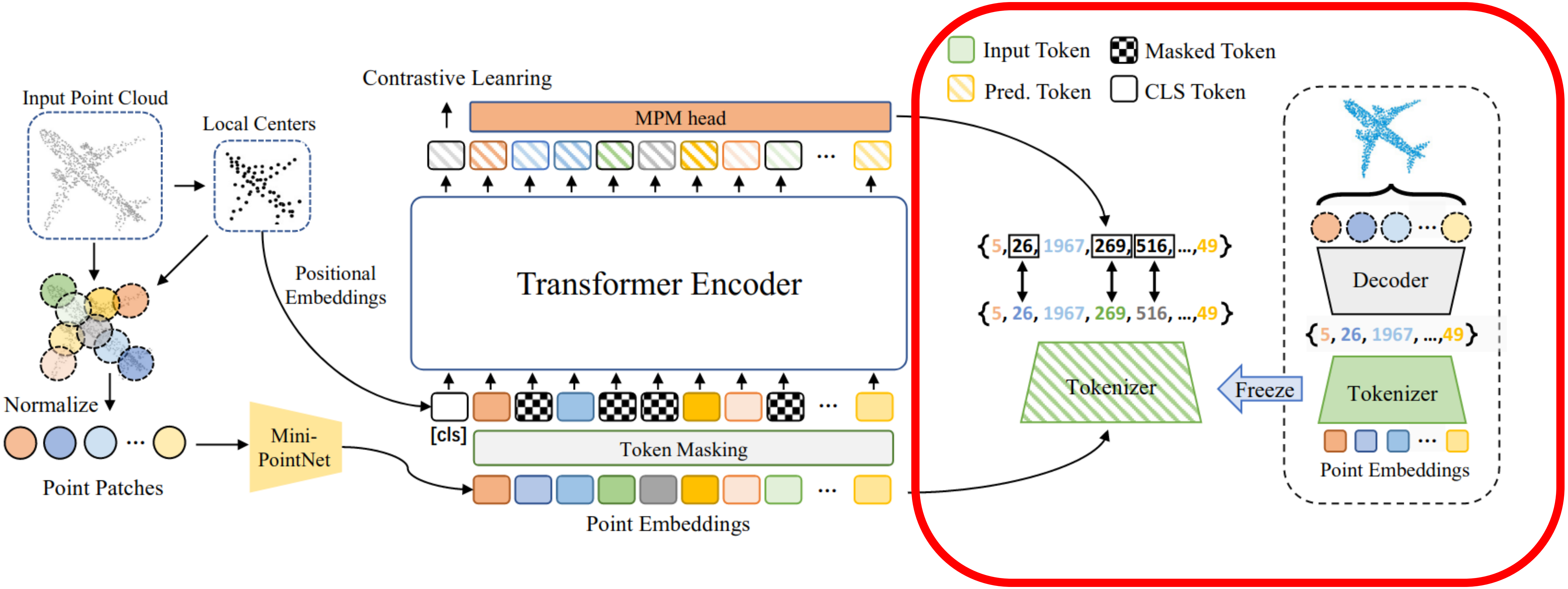
Point-BERT

Pre-training 3D Point Cloud Transformers with Masked Point Modeling



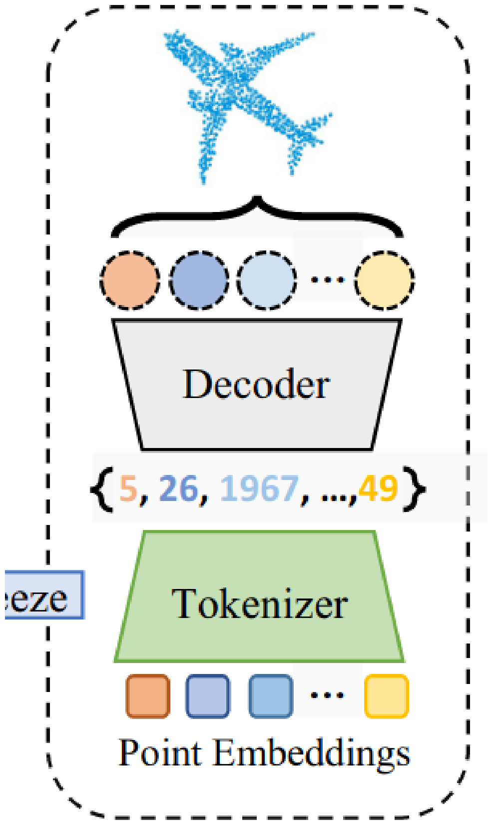
Point-BERT

Pre-training 3D Point Cloud Transformers with Masked Point Modeling

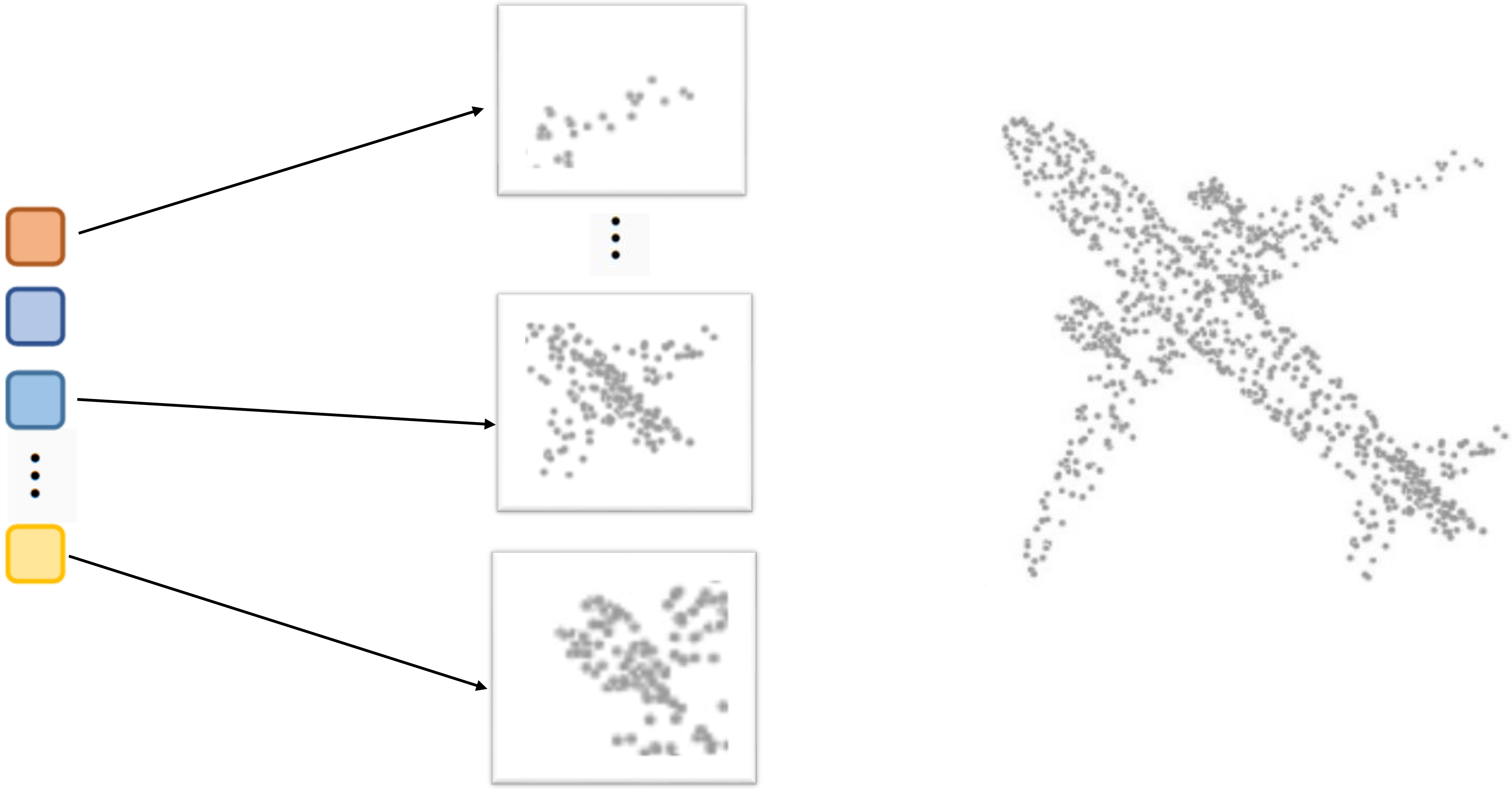


Discrete Variational Autoencoder

- Variational autoencoder
  - Discrete tokens
- Patches -> Tokens
  - Fixed vocabulary of geometric "words"
  - Points embeddings form a "sentence"

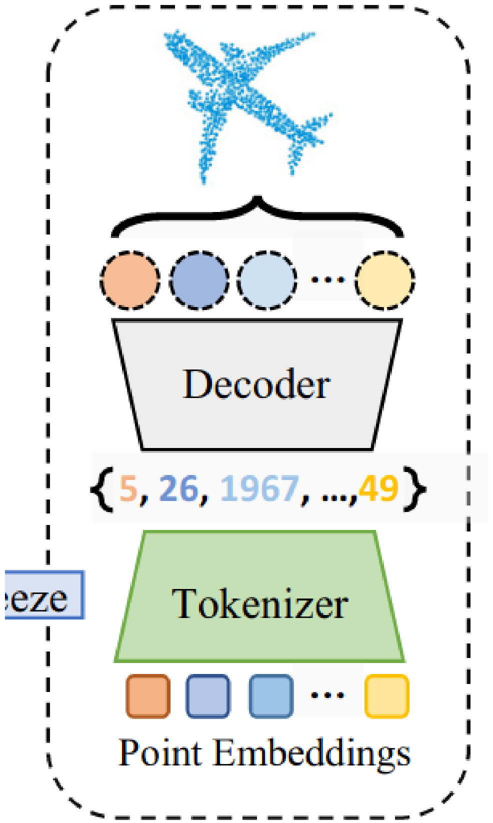


The Tokenizer {5, 26, 1967, …,49}



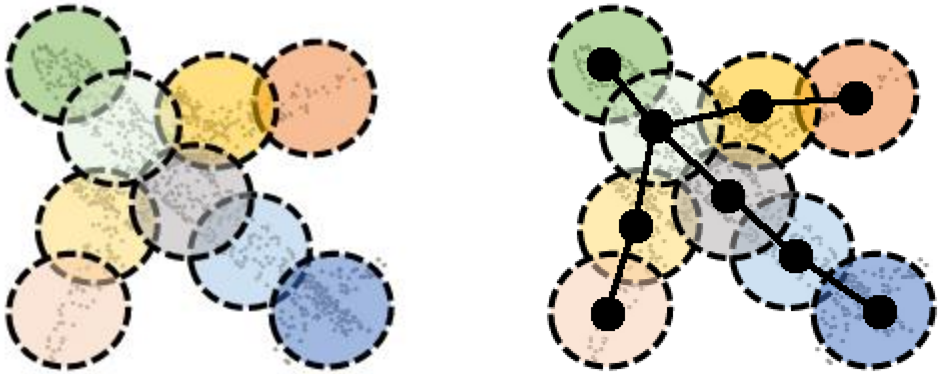
Discrete Variational Autoencoder

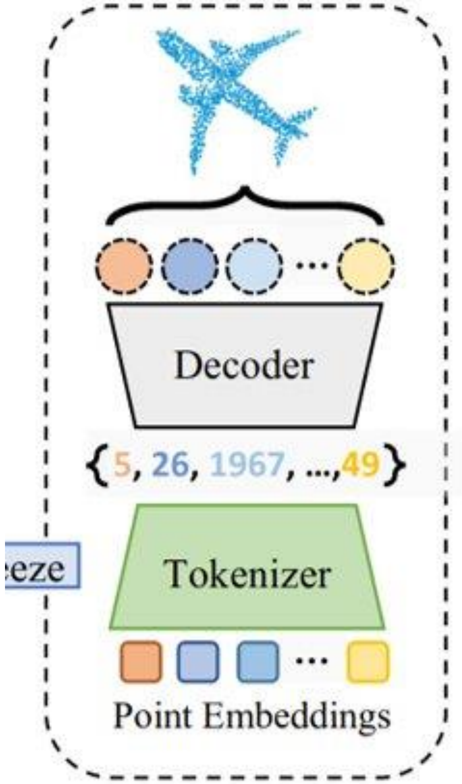
- Variational autoencoder
  - Discrete tokens
- Patches -> Tokens
  - Fixed vocabulary of geometric "words"
  - Points embeddings form a "sentence"

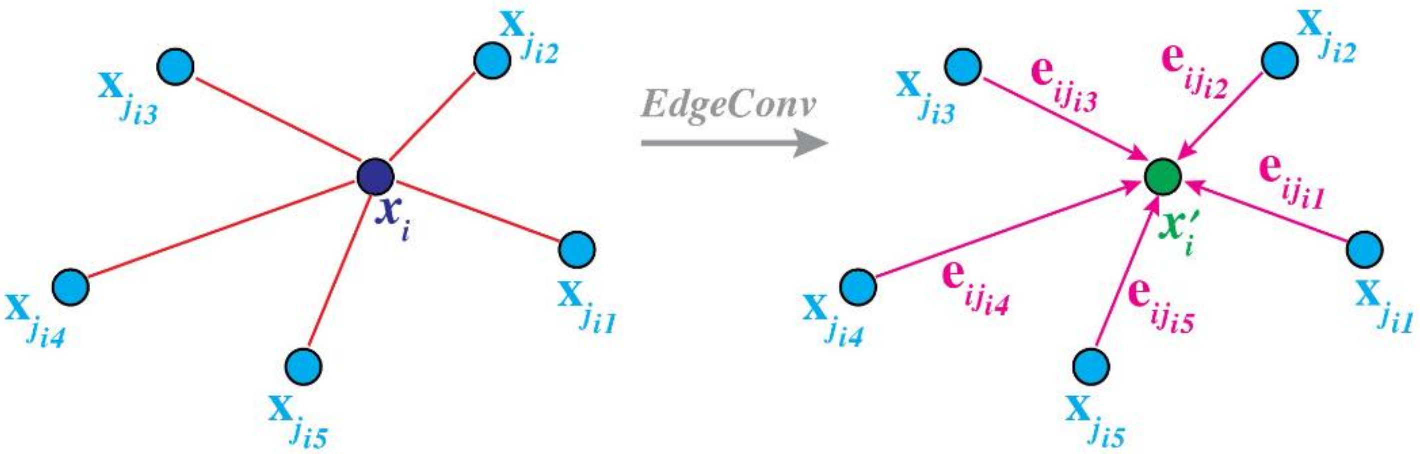


The Tokenizer

- DGCNN
  - Build a kNN graph
  - EdgeConv
  - Rince and repeat in feature space
- Mix local and global information



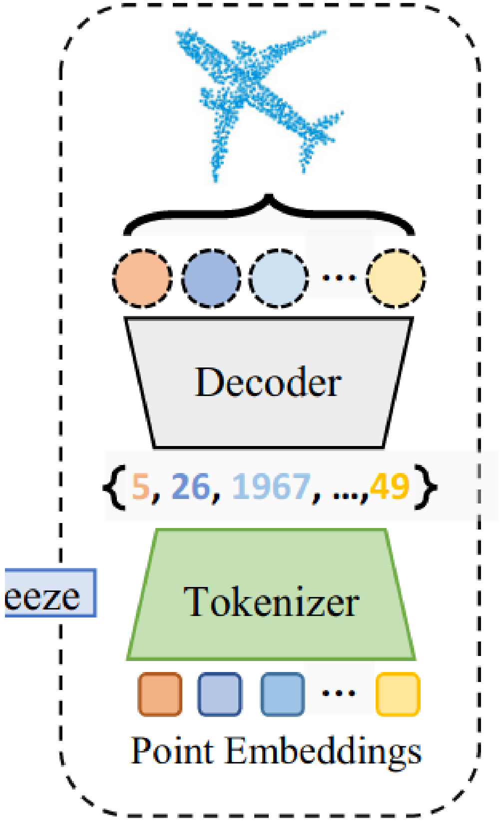




Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. TOG, 2019.

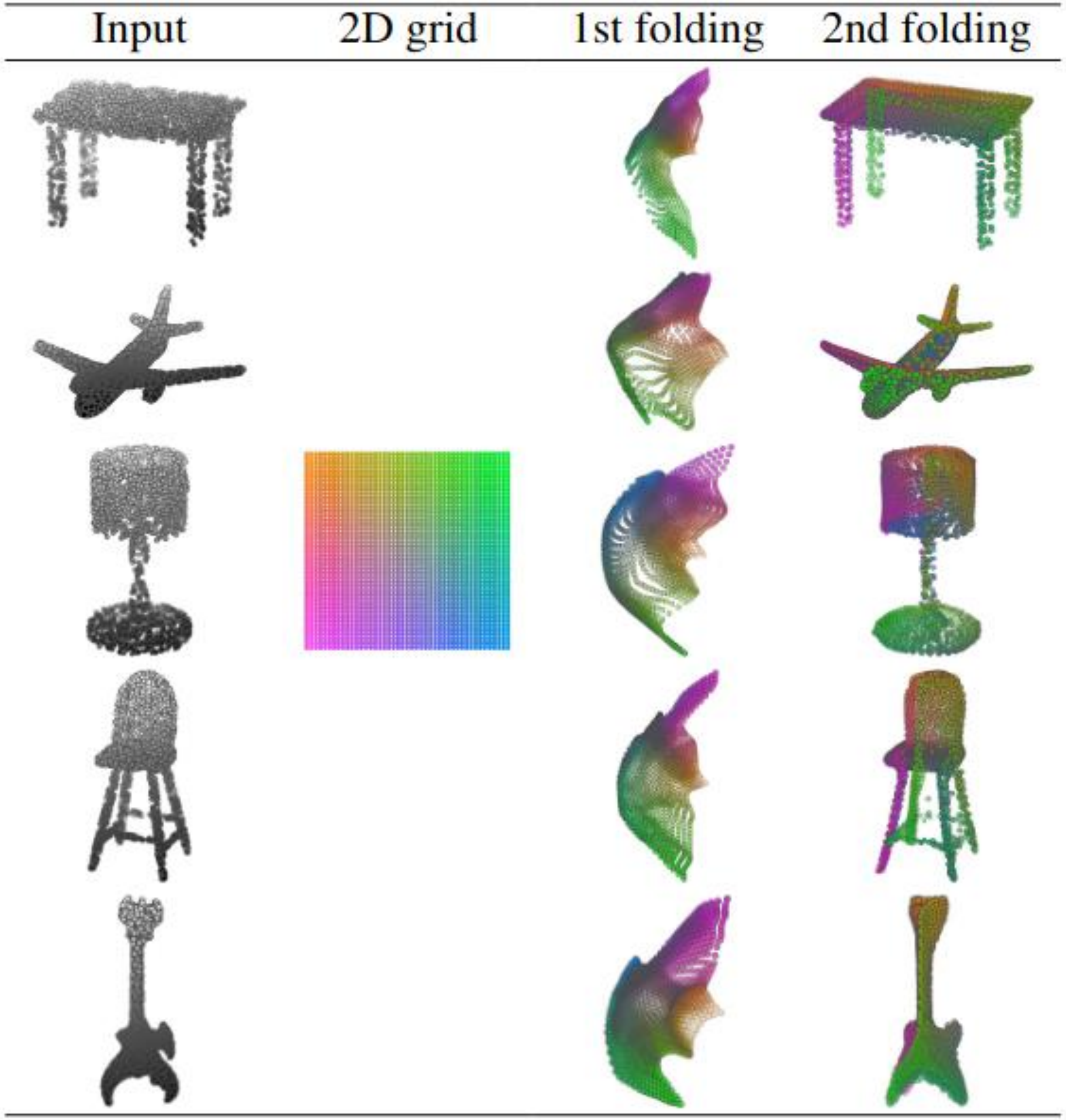
The Decoder

- Tokens -> Patches
- DGCNN between tokens
  - Take the whole "sentence" into account
- FoldingNet to reconstruct the point cloud



FoldingNet

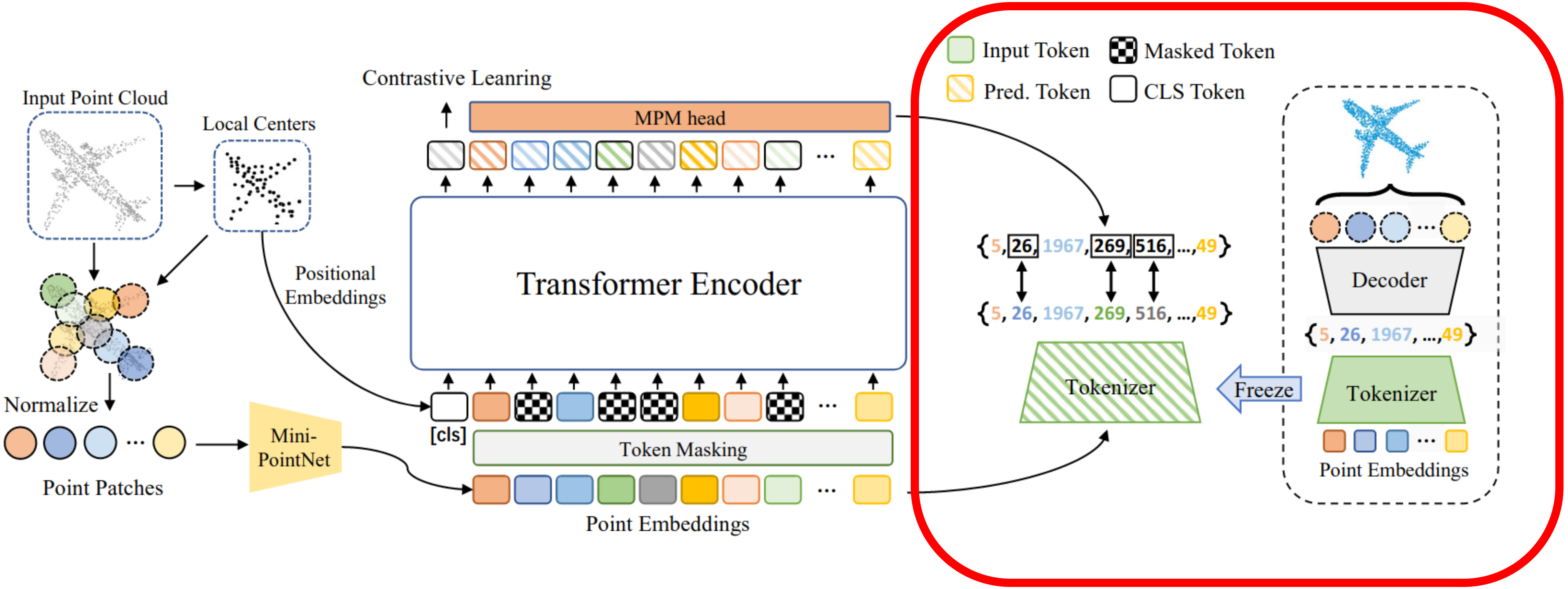
- Point clouds as folded plane



Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In CVPR, 2018.

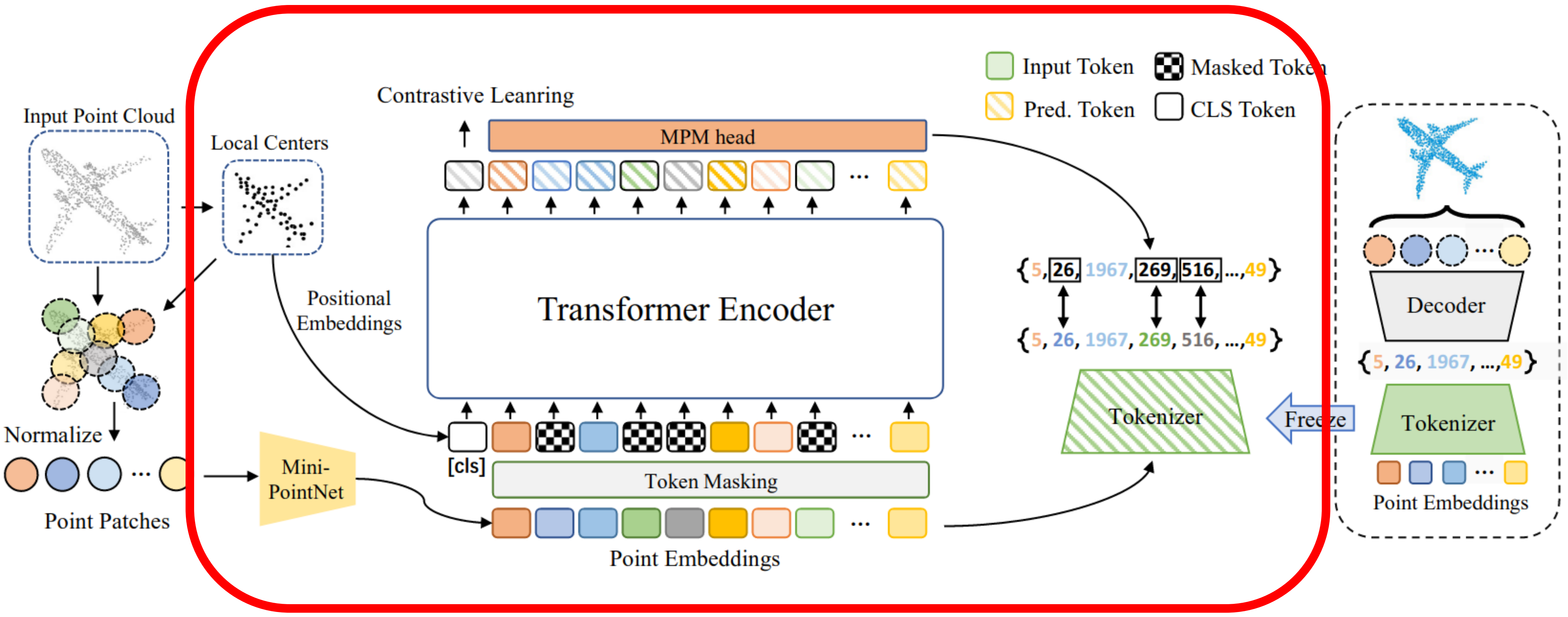
Point-BERT

Pre-training 3D Point Cloud Transformers with Masked Point Modeling



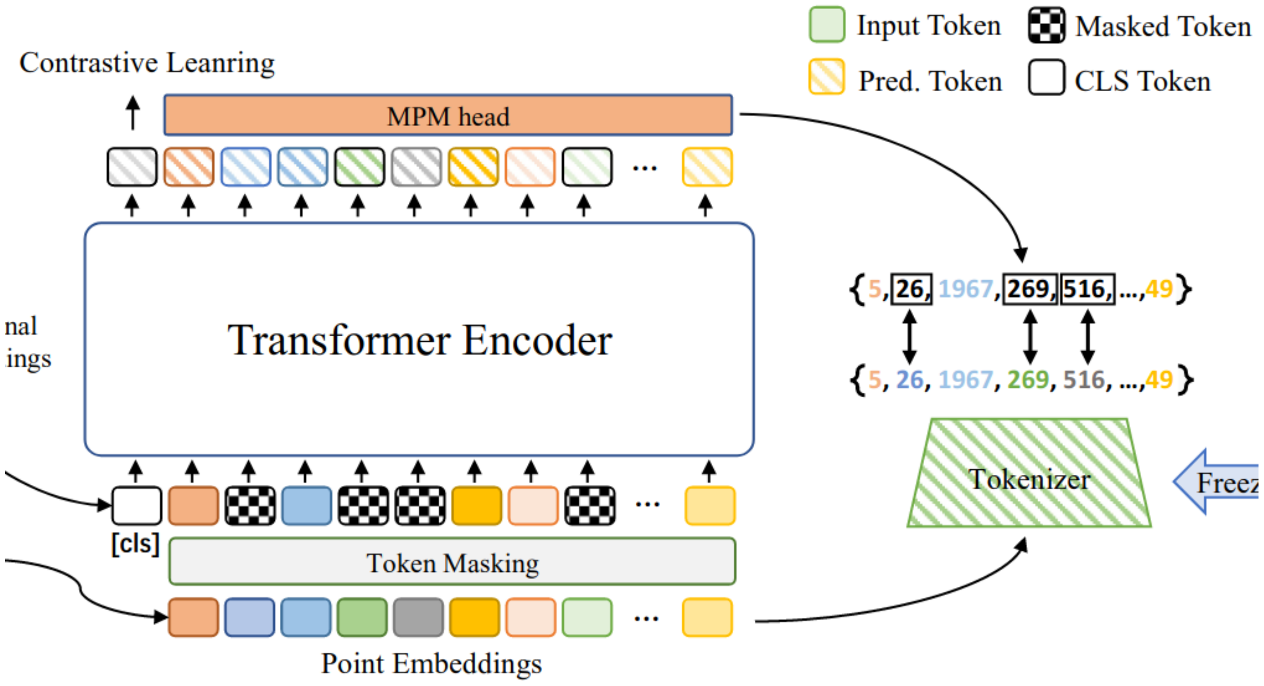
Point-BERT

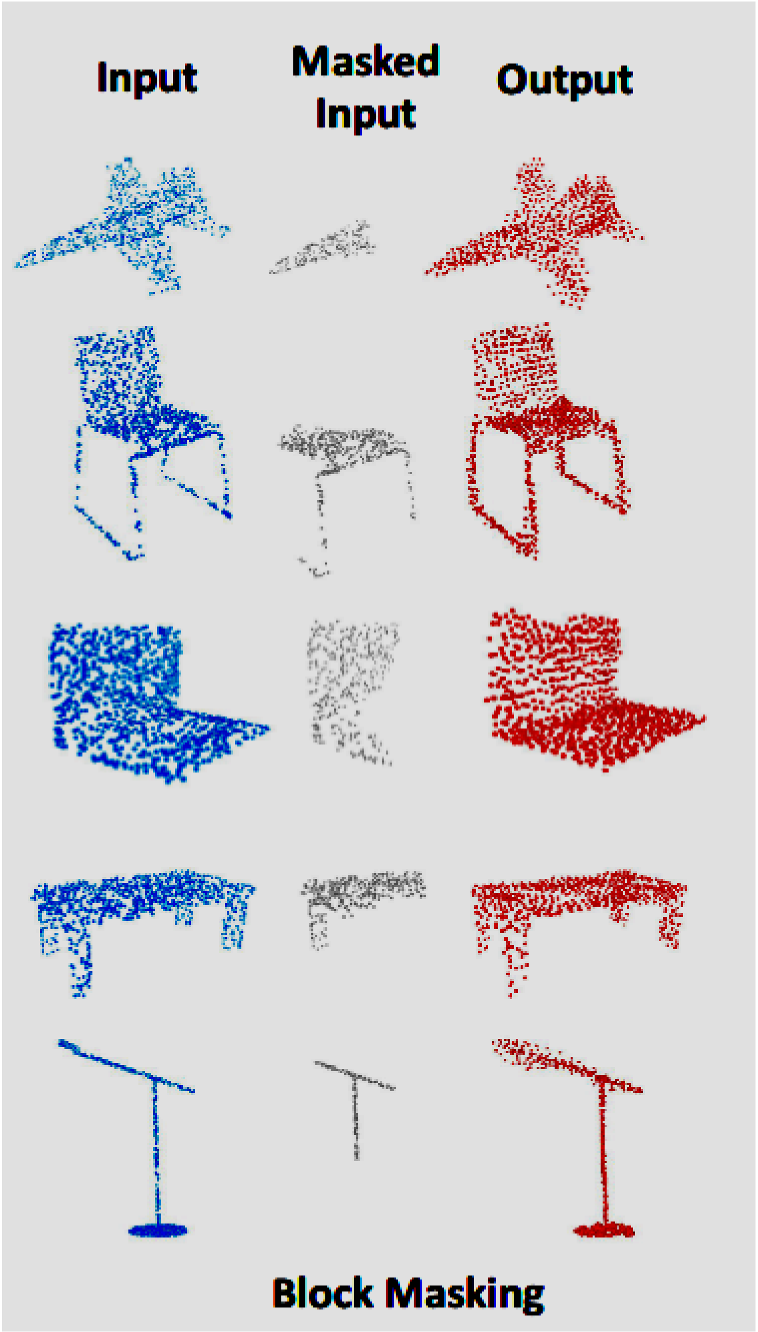
Pre-training 3D Point Cloud Transformers with Masked Point Modeling



Masked Point Modeling

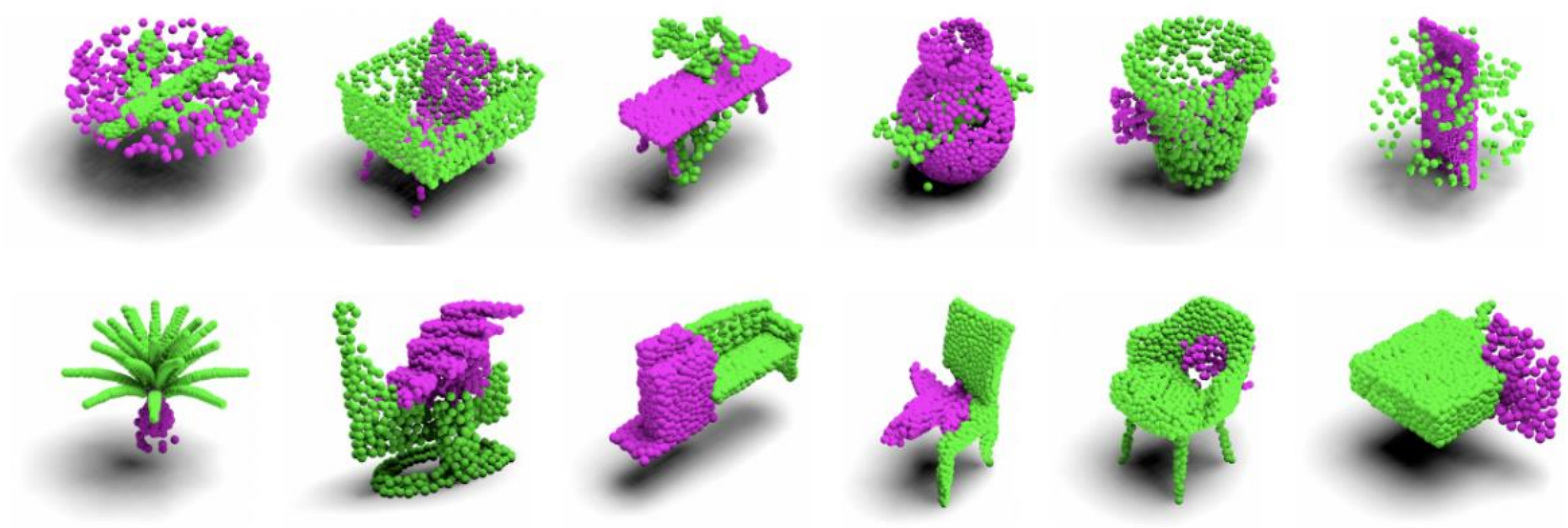
- Block Masking
- Mask 25%-45% of tokens





Auxiliary pre-training: Point Patch Mixing

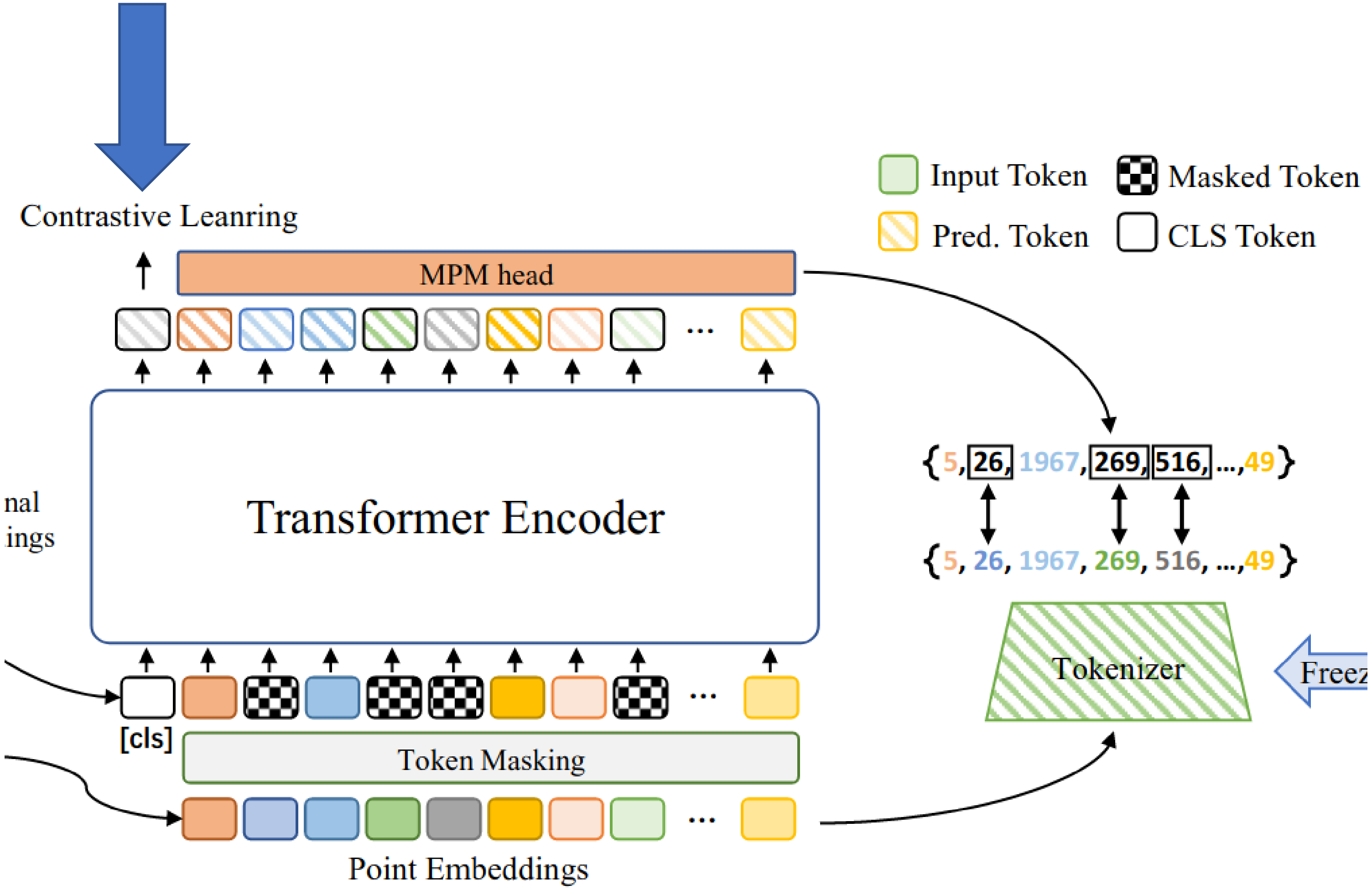
- Similar to CutMix
- Cut and past point patch from different models
- Predict the original token



Jinlai Zhang, Lyujie Chen, Bo Ouyang, Binbin Liu, Jihong Zhu, Yujing Chen, Yanmei Meng, and Danfeng Wu. Point-cutmix: Regularization strategy for point cloud classification. arXiv preprint arXiv:2101.01461, 2021.

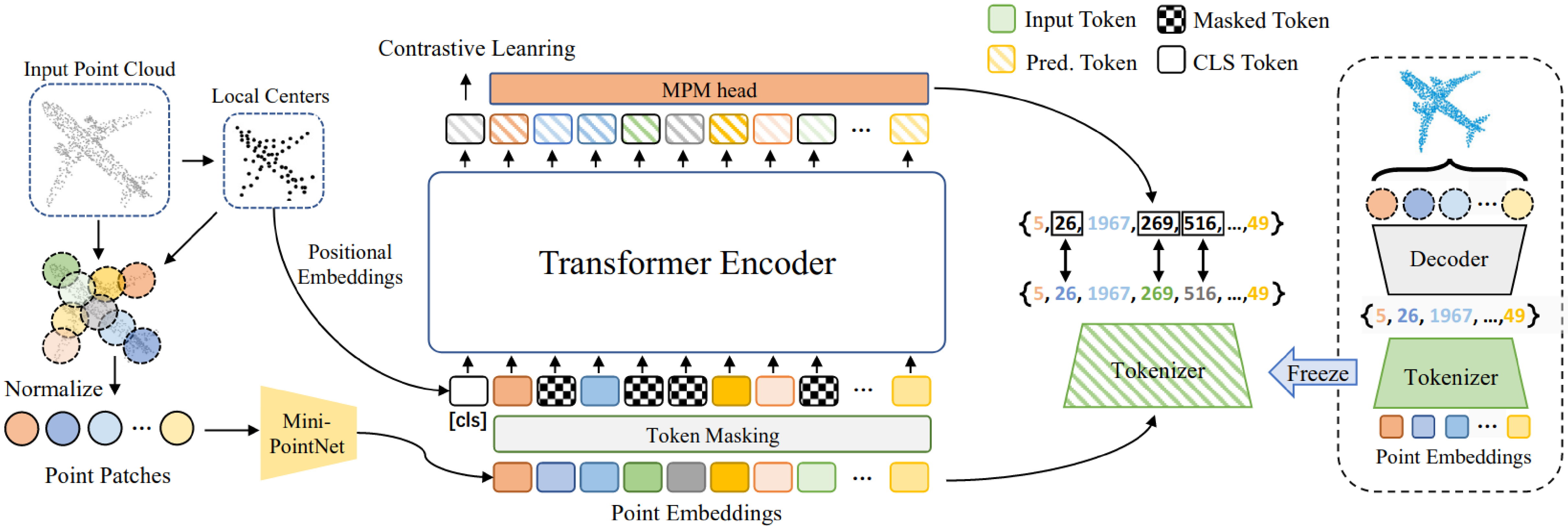
Contrastive Learning

- Another kind of SSL
- MoCo

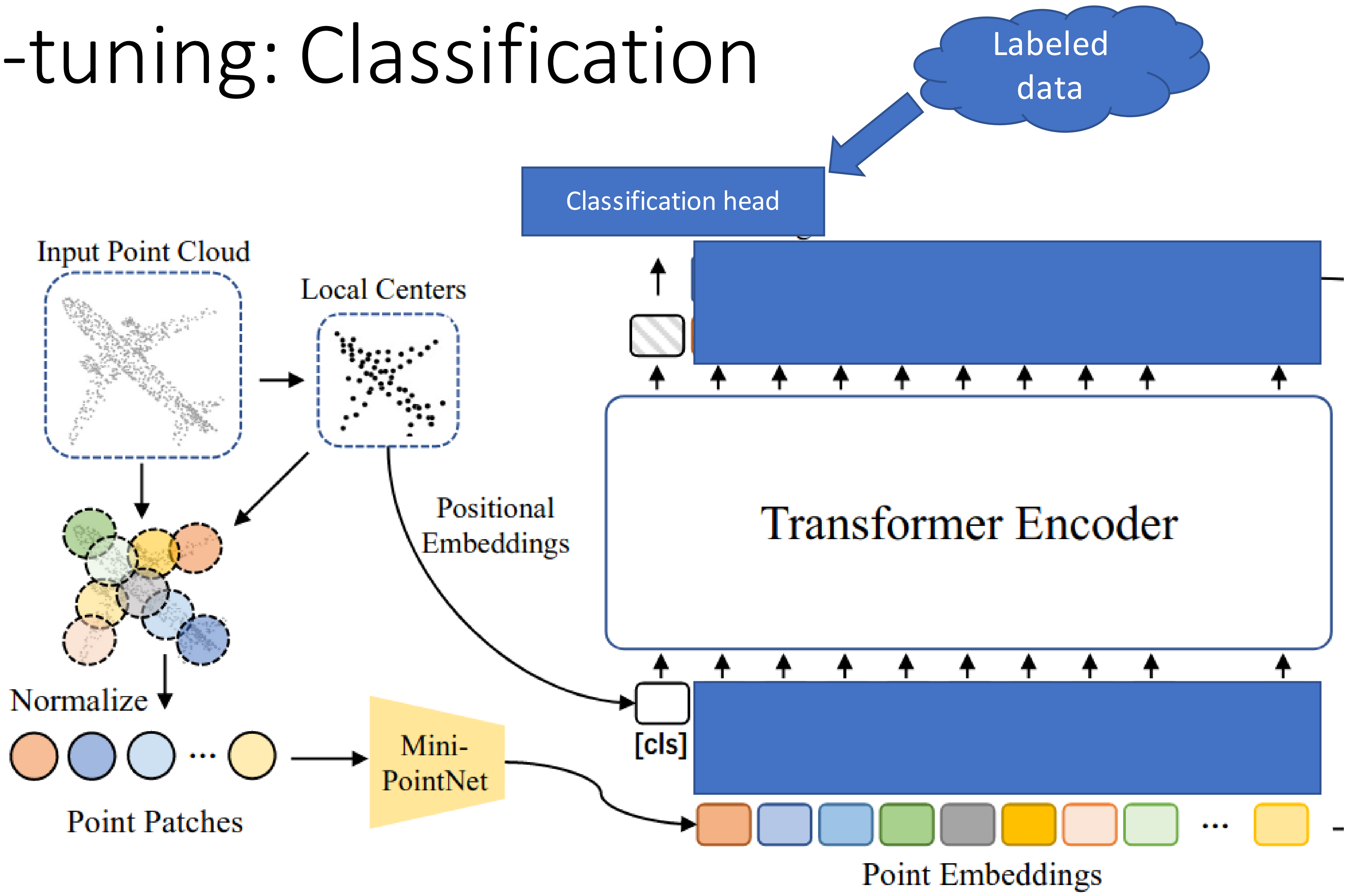


Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In CVPR, 2020.

Pre-training with Point-BERT



Fine-tuning: Classification



Classification

- Pre-training on ShapeNet
- Test on ModelNet40
- OcCo = completion of occlusion

Table 1. **Comparisons of Point-BERT with of state-of-the-art models on ModelNet40.** We report the classification accuracy (%) and the number of points in the input. [ST] and [T] represent the standard Transformers models and Transformer-based models with some special designs and more inductive biases, respectively.

Method	#point	Acc.
PointNet [39]	1k	89.2
PointNet++ [40]	1k	90.5
SO-Net [24]	1k	92.5
PointCNN [25]	1k	92.2
DGCNN [60]	1k	92.9
DensePoint [28]	1k	92.8
RSCNN [45]	1k	92.9
[T] PTC [11]	1k	93.2
[T] PointTransformer [72]	–	93.7
[ST] NPTC [11]	1k	91.0
[ST] Transformer	1k	91.4
[ST] Transformer + OcCo [58]	1k	92.1
[ST] Point-BERT	1k	93.2
[ST] Transformer	4k	91.2
[ST] Transformer + OcCo [58]	4k	92.2
[ST] Point-BERT	4k	93.4
[ST] Point-BERT	8k	**93.8**

Few-Shot Classification

- K-way N-shot
  - K classes
  - N samples
- Fine-tuning

Table 2. **Few-shot classification results on ModelNet40.** We report the average accuracy (%) as well as the standard deviation over 10 independent experiments.

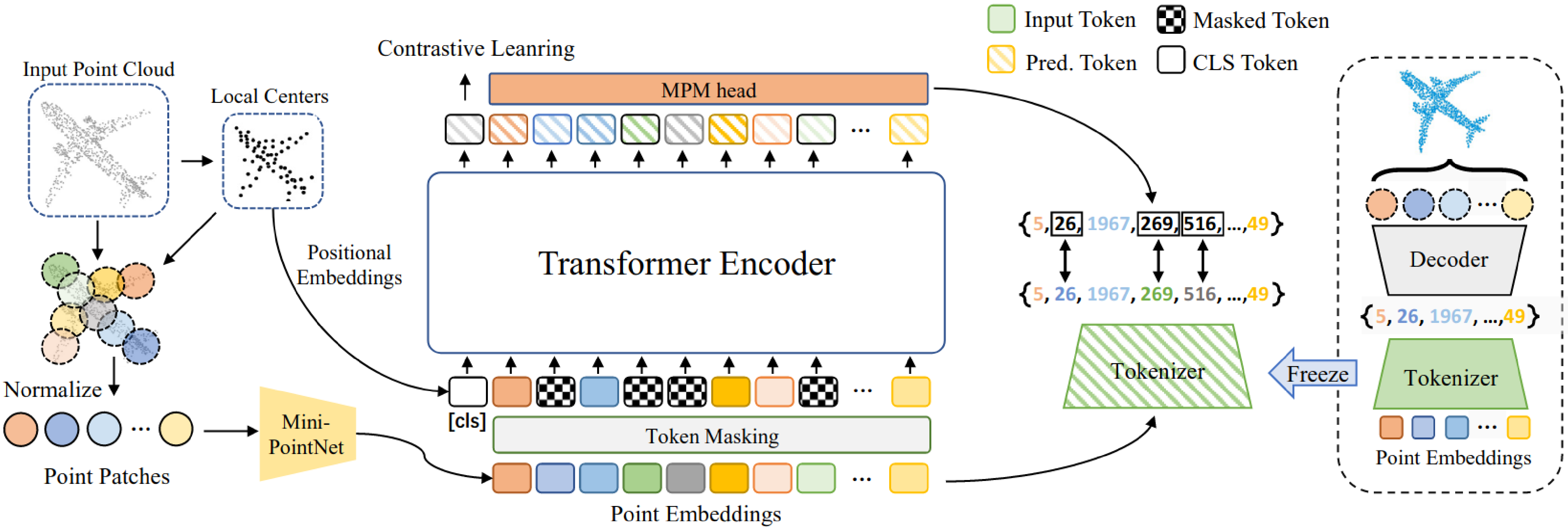
	5-way		10-way	
	10-shot	20-shot	10-shot	20-shot
DGCNN-rand [58]	31.6 ± 2.8	40.8 ± 4.6	19.9 ± 2.1	16.9 ± 1.5
DGCNN-OcCo [58]	90.6 ± 2.8	92.5 ± 1.9	82.9 ± 1.3	86.5 ± 2.2
DGCNN-rand*	91.8 ± 3.7	93.4 ± 3.2	86.3 ± 6.2	90.9 ± 5.1
DGCNN-OcCo*	91.9 ± 3.3	93.9 ± 3.1	86.4 ± 5.4	91.3 ± 4.6
Transformer-rand	87.8 ± 5.2	93.3 ± 4.3	84.6 ± 5.5	89.4 ± 6.3
Transformer-OcCo	94.0 ± 3.6	95.9 ± 2.3	89.4 ± 5.1	92.4 ± 4.6
Point-BERT	$\mathbf{94.6 \pm 3.1}$	$\mathbf{96.3 \pm 2.7}$	$\mathbf{91.0 \pm 5.4}$	$\mathbf{92.7 \pm 5.1}$

Ablation Study

Table 5. **Ablation study.** We investigate the effects of different designs and report the classification accuracy (%) after fine-tuning on ModelNet40. All models are trained with 1024 points.

Pretext tasks	MPM	Point Patch Mixing	Moco	Acc.
Model A				91.41
Model B	✓			92.58 ↑
Model C	✓	✓		92.91 ↑
Model D	✓	✓	✓	93.24 ↑
Augmentation	**mask type**	**mask ratio**	**replace**	**Acc.**
Model B	block mask	[0.25, 0.45]	No	92.58
Model B	block mask	[0.25, 0.45]	Yes	91.81 ↓
Model B	rand mask	[0.25, 0.45]	No	92.34 ↓
Model B	block mask	[0.55, 0.85]	No	92.52 ↓
Model D	block mask	[0.25, 0.45]	No	93.16
Model D	block mask	[0.25, 0.45]	Yes	92.58 ↓
Model D	rand mask	[0.25, 0.45]	No	92.91 ↓
Model D	block mask	[0.55, 0.85]	No	92.59 ↓

Questions?


Input Point Cloud
Local Centers
Normalize
Point Patches
Mini-PointNet
Positional Embeddings
Contrastive Leanring
MPM head
Transformer Encoder
[cls]
Token Masking
Point Embeddings
Input Token
Masked Token
Pred. Token
CLS Token
{5, 26, 1967, 269, 516, ..., 49}
{5, 26, 1967, 269, 516, ..., 49}
Tokenizer
Freeze
Decoder
{5, 26, 1967, ..., 49}
Tokenizer
Point Embeddings

Want more transformers?

- Talk I gave last semester on Transformers in Computer Vision (French)

A Quick Review on QKV Attention

- Query
- Key
- Value