Recent Progress on Transformer & SSL

Shengchao Liu, Jan 2022

Recent Progress on Transformer & SSL

1. **Vision**
   1. **ViT, ICLR’21**
   2. **DINO, ICCV’21**
   3. **MoCo-v3, ArXiv’21**
   4. **BEiT, ICLR’22**
   5. **MAE, CVPR’22**
2. **Graphs & Molecules**
3. **Tabular Data**

ViT: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR'21

Link

Scope of this paper:

- Previously:
    - Attention is applied in conjunction with CNN.
    - Attention is used to replace certain components of CNN.
- This work:
    - Pure Transformer is possible.

ViT: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR'21

Vision Transformer (ViT)

Three key steps:

1. Split an image into sequence of flattened patches
2. Add patch embeddings and position embeddings
3. Feed into Transformer

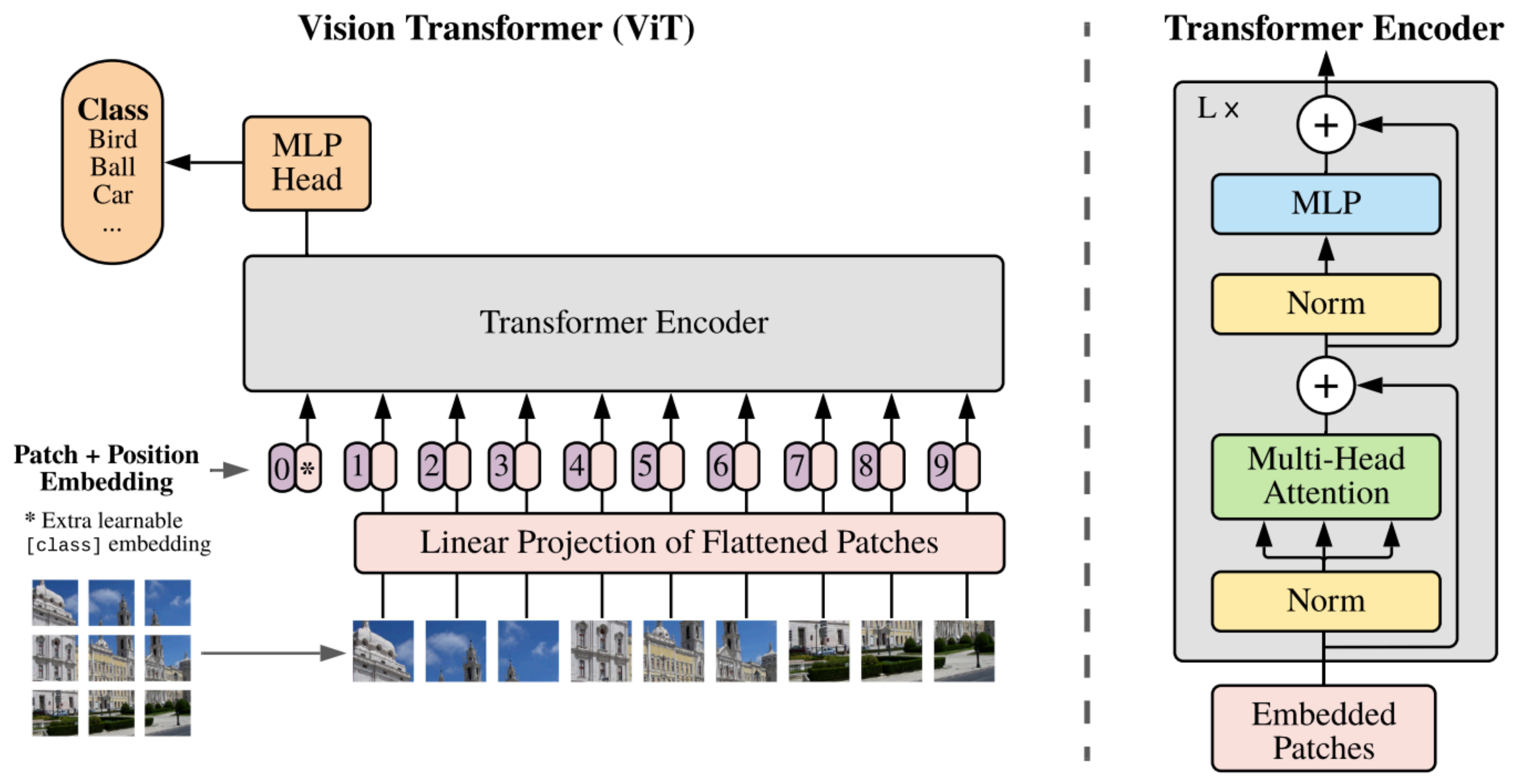


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable "classification token" to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).

ViT: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR'21

Vision Transformer (ViT)

Three key steps:

1. Split an image into sequence of flattened patches
2. Add patch embeddings and position embeddings
3. Feed into Transformer

Pros:

1. Comparative performance
2. Computationally efficient

Cons:

1. Unstable training

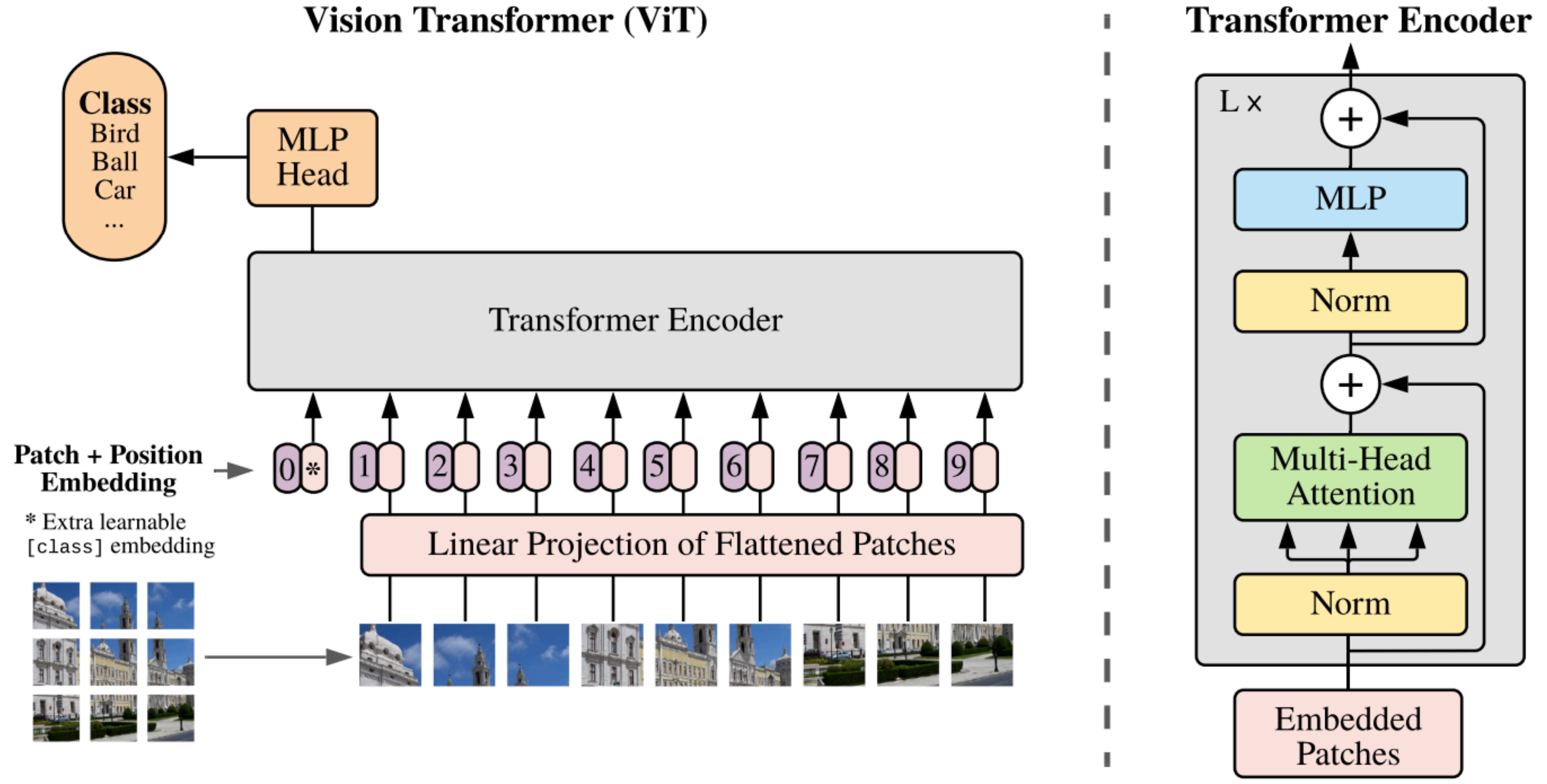


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable "classification token" to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).

ViT: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR'21

Observations:

1. ViT is worse on mid-sized dataset (with CNN)
2. ViT can reach or beat SOTA on larger-sized dataset

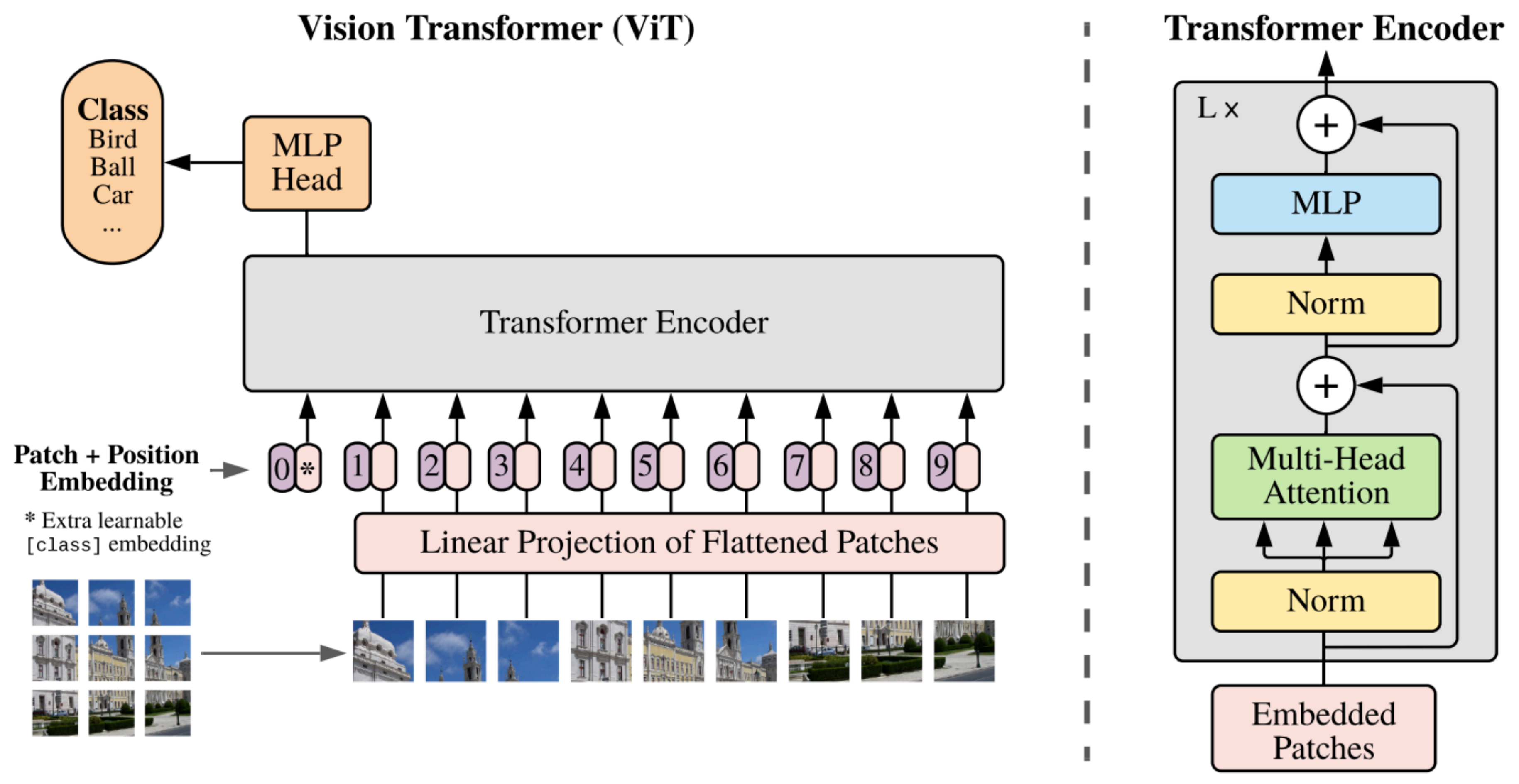


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable "classification token" to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).

ViT: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR'21

Observations:

1. ViT is worse on mid-sized dataset (with CNN)
2. ViT can reach or beat SOTA on larger-sized dataset

Conjectures:

1. CNN inherently possess inductive biases (locality and translation equivalence).
2. Transformer lacks these inductive biases, thus generalizes poorly.

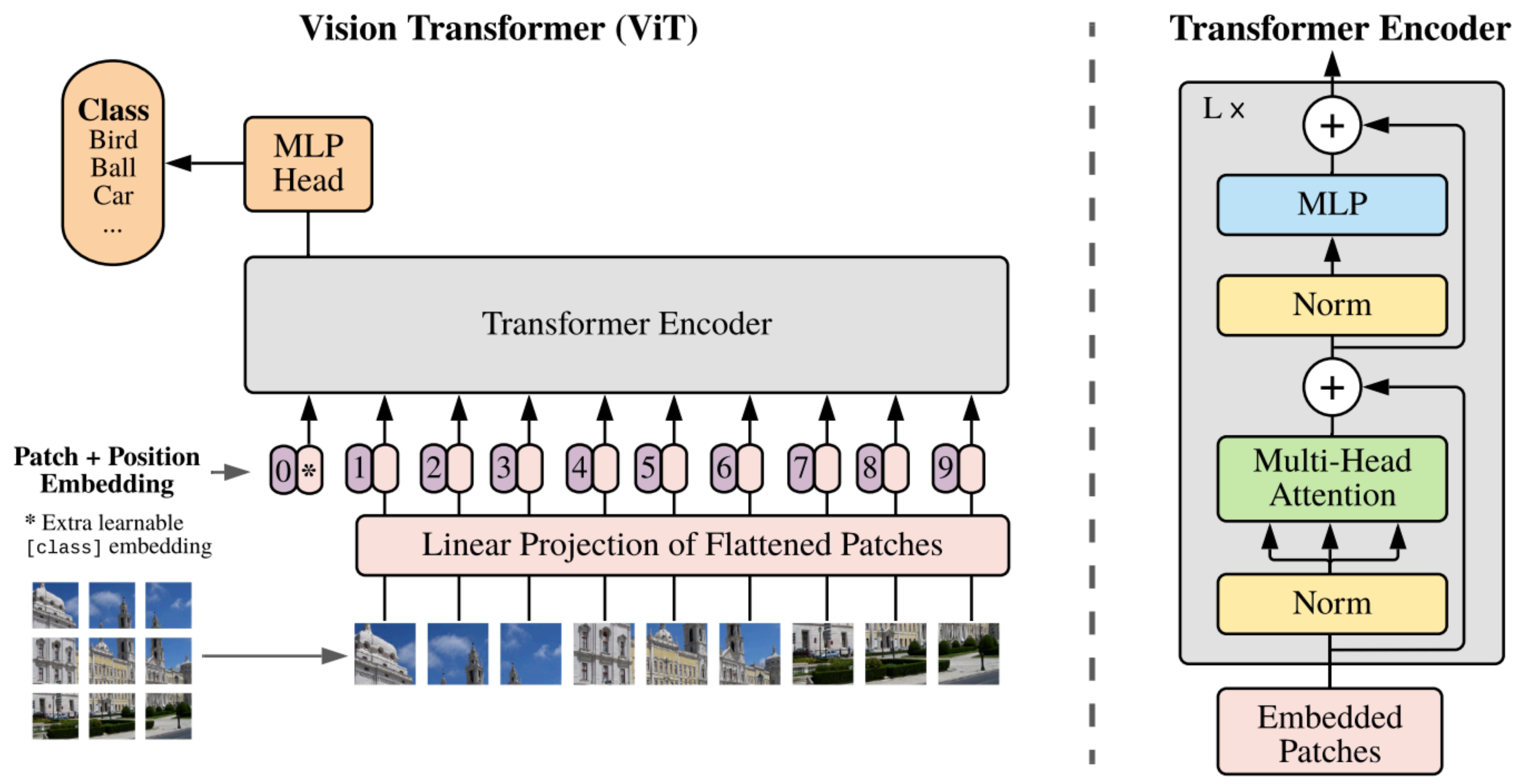


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable "classification token" to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).

ViT: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR'21

A more recent work on image representation ConvNeXt [1].

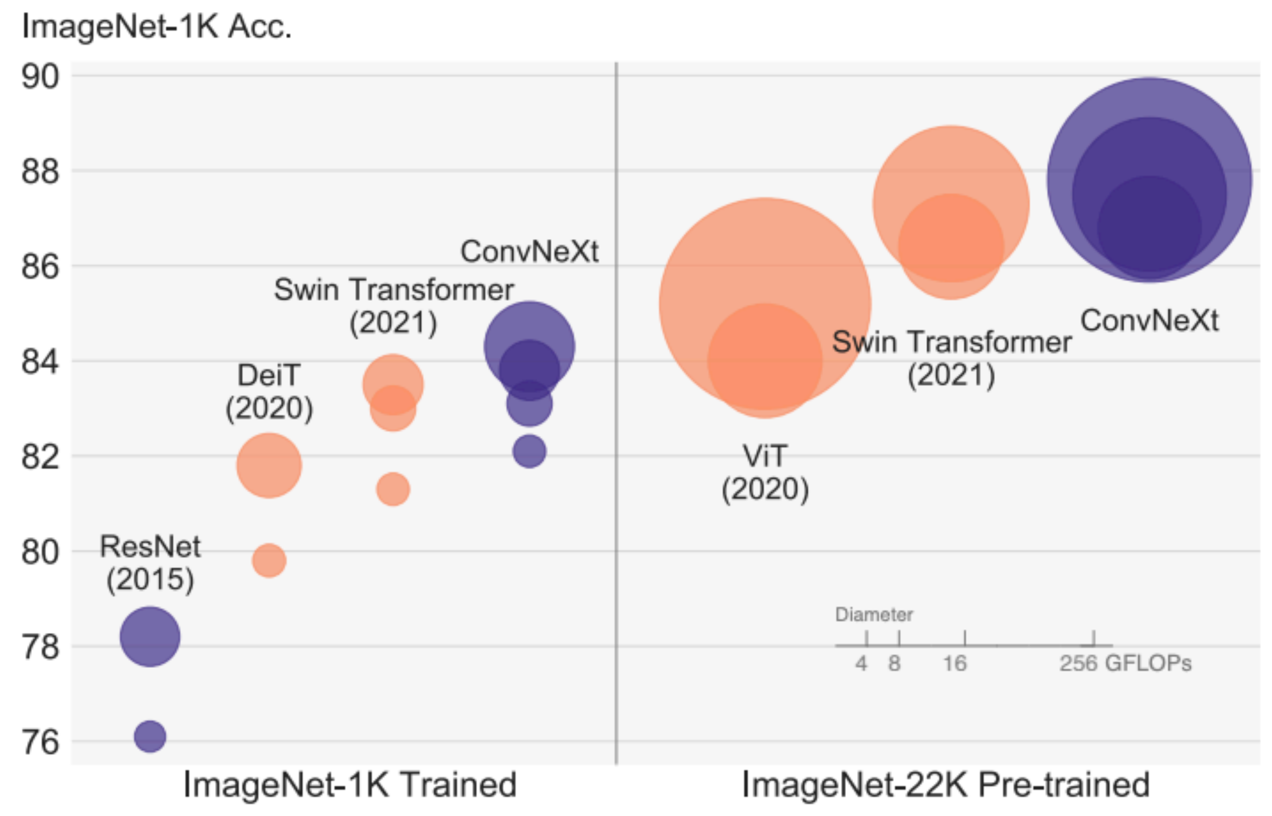


Figure 1. **ImageNet-1K classification** results for ● ConvNets and ○ vision Transformers. Each bubble's area is proportional to FLOPs of a variant in a model family. ImageNet-1K/22K models here take $224^2/384^2$ images respectively. We demonstrate that a standard ConvNet model can achieve the same level of scalability as hierarchical vision Transformers while being much simpler in design.

[1] Liu, Zhuang, et al. "A ConvNet for the 2020s." *arXiv preprint arXiv:2201.03545* (2022).

ViT: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR'21

SSL: Masked patch prediction

- Inputs: masked/corrupted patches
  - Replace embeddings with [mask] embedding (80%)
  - Replace with a random other patch embedding (10%)
  - Keep them as is (10%)

ViT: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR'21

SSL: Masked patch prediction

- Inputs: masked/corrupted patches
  - Replace embeddings with [mask] embedding (80%)
  - Replace with a random other patch embedding (10%)
  - Keep them as is (10%)
- Outputs, three options:
  - **Mean of the raw patches (only report this one)**
  - 4*4 downsized version of the 16*16 patches
  - Regression on the full patch with L2
    - slightly worse, which seems to conflict with MAE
    - main difference: decoder

DINO: Emerging Properties in Self-Supervised Vision Transformers, ICCV'21

Link

Scope of this paper:

- In NLP, the success of Transformers comes from SSL pre-training, like BERT or GPT
- This work studies ViT in SSL pre-training

DINO: Emerging Properties in Self-Supervised Vision Transformers, ICCV'21

DINO: self-distillation with no labels
is essentially BYOL, wrapped in teacher-student framework

Local and global views use cropping for each image:

- Global view:
  - Large resolution covering a large area (>50%) of original image
  - To teacher network
- Local view:
  - Small resolution covering a small area (<50%) of original image
  - To student network

DINO: Emerging Properties in Self-Supervised Vision Transformers, ICCV'21

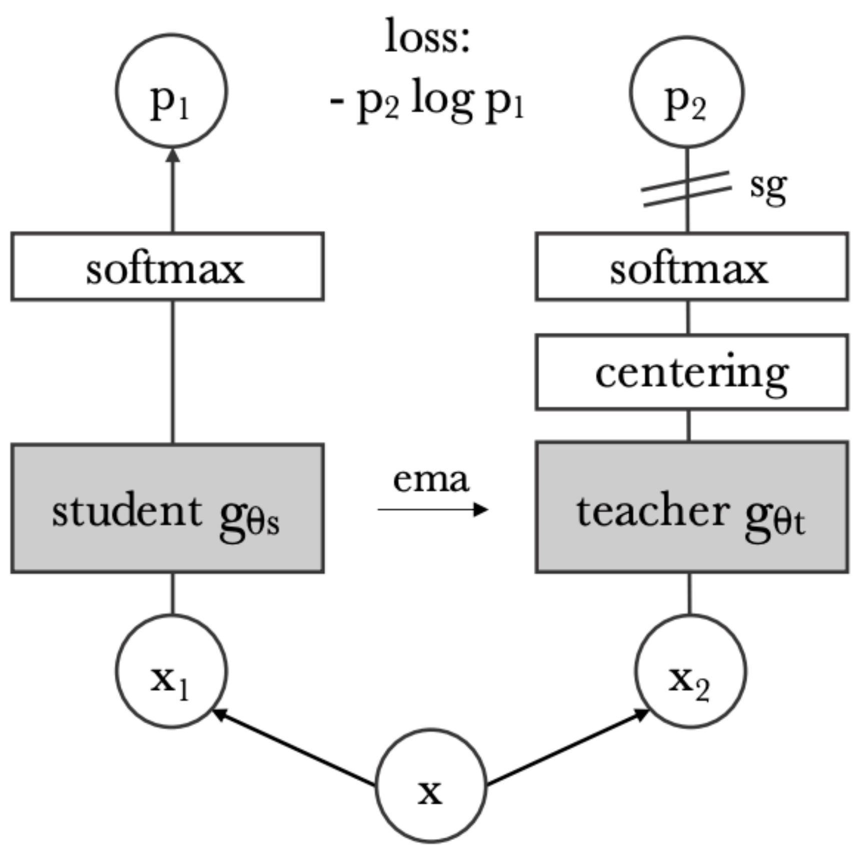


Figure 2: **Self-distillation with no labels.** We illustrate DINO in the case of one single pair of views (x_1, x_2) for simplicity. The model passes two different random transformations of an input image to the student and teacher networks. Both networks have the same architecture but different parameters. The output of the teacher network is centered with a mean computed over the batch. Each networks outputs a K dimensional feature that is normalized with a temperature softmax over the feature dimension. Their similarity is then measured with a cross-entropy loss. We apply a stop-gradient (sg) operator on the teacher to propagate gradients only through the student. The teacher parameters are updated with an exponential moving average (ema) of the student parameters.

SG: Stop-Gradient

EMA: Exponential Moving Average $\theta_t = \lambda\theta_t + (1-\lambda)\theta_s$

DINO: Emerging Properties in Self-Supervised Vision Transformers, ICCV'21

Observations:

- SSL ViT features/embeddings explicitly contain the scene layout and object boundaries.

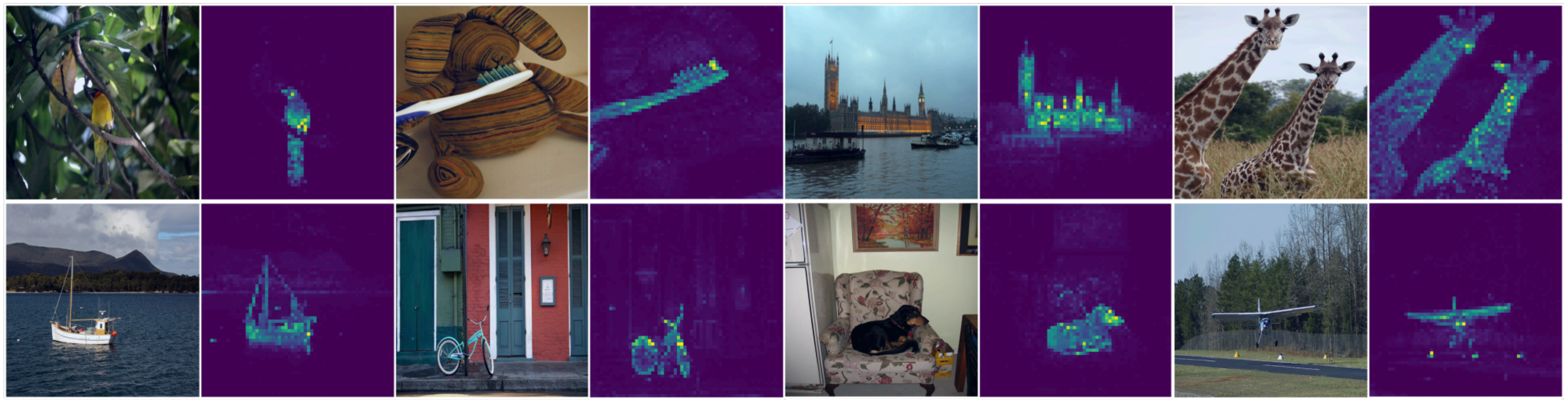


Figure 1: **Self-attention from a Vision Transformer with 8×8 patches trained with no supervision.** We look at the self-attention of the [CLS] token on the heads of the last layer. This token is not attached to any label nor supervision. These maps show that the model automatically learns class-specific features leading to unsupervised object segmentations.

- SSL ViT features/embeddings perform particularly well with k-NN *w/o fine-tuning, linear classifier nor data augmentation*, achieving 78.3% top-1 acc on ImageNet.

MoCo-v3: An Empirical Study of Training Self-Supervised Vision Transformers, ArXiv’21

Link

Scope of this paper:

- Not a novel method.
- A straightforward, incremental, yet must-known baseline: contrastive SSL for ViT

MoCo-v3: An Empirical Study of Training Self-Supervised Vision Transformers, ArXiv'21

Contrastive SSL using ViT:

1. Take two augmentations for each image as two views
2. ViT as encoder
3. Train with InfoNCE

$$\mathcal{L}_q = -\log \frac{\exp(q \cdot k^+/\tau)}{\exp(q \cdot k^+/\tau) + \sum_{k^-} \exp(q \cdot k^-/\tau)}. \quad (1)$$

MoCo-v3: An Empirical Study of Training Self-Supervised Vision Transformers, ArXiv'21

Contrastive SSL using ViT:

1. Take two augmentations for each image as two views
2. ViT as encoder
3. Train with InfoNCE

$$\mathcal{L}_q = -\log \frac{\exp(q \cdot k^+/\tau)}{\exp(q \cdot k^+/\tau) + \sum_{k^-} \exp(q \cdot k^-/\tau)}. \quad (1)$$

Other details:

Encoder

Momentum Encoder

- Use two encoders for two views: f_q and f_k
- SGD to update f_q
- EMA to update f_k: $f_k = m \cdot f_k + (1 - m) \cdot f_q$

BEiT: BERT Pre-Training of Image Transformers, ICLR’22

Link

Scope of this paper:
A SSL method on ViT

BEiT: BERT Pre-Training of Image Transformers, ICLR’22

Two views for each image:

- image patches
- visual tokens: tokenize the image into discrete visual tokens, by the latent of the discrete VAE (given/well-trained)

Prediction task: (no motivation/intuition)

- reconstruct the visual tokens, instead of raw pixels of masked patches

BEiT: BERT Pre-Training of Image Transformers, ICLR’22

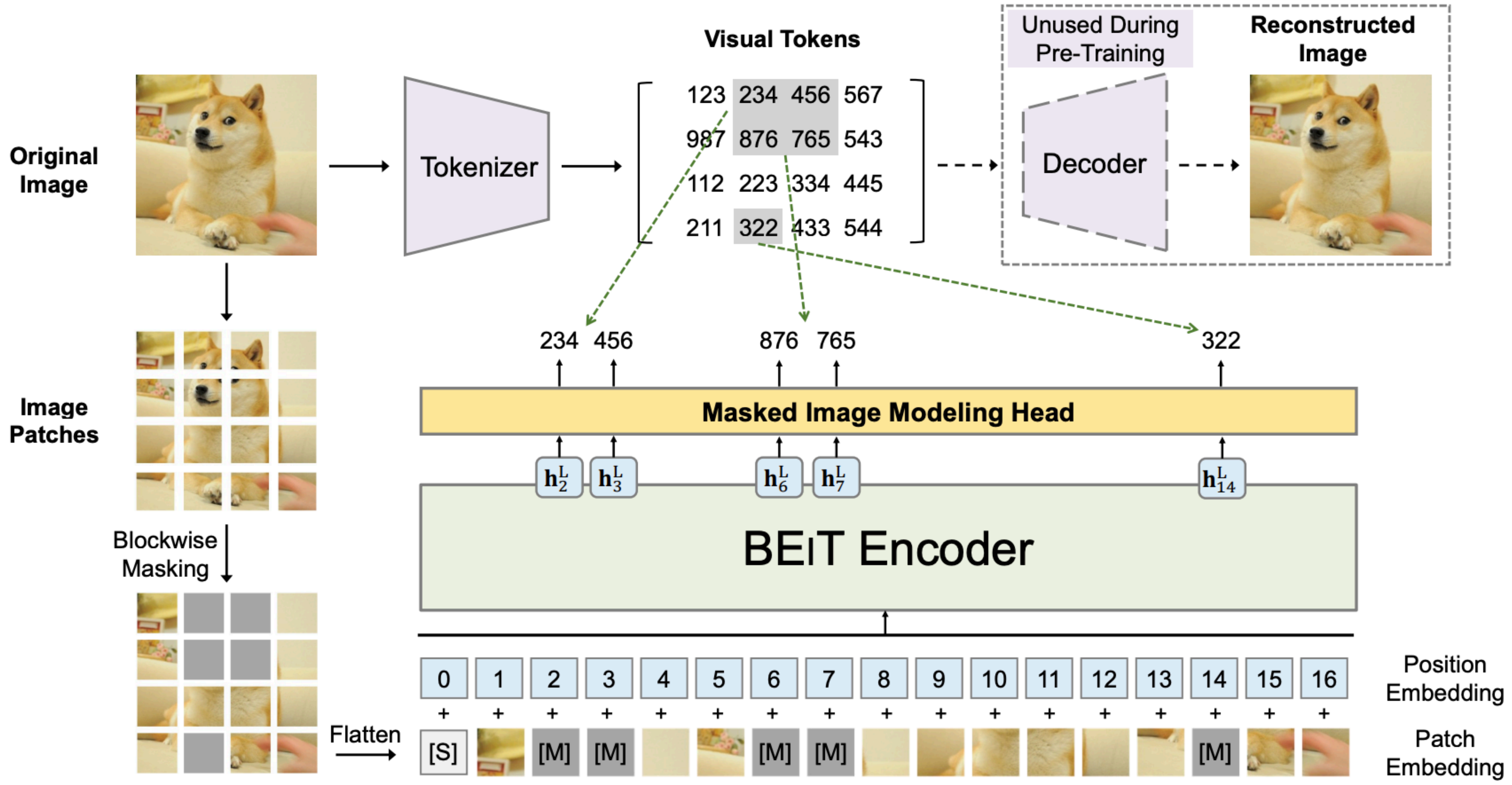


Figure 1: Overview of BEIT pre-training. Before pre-training, we learn an “image tokenizer” via autoencoding-style reconstruction, where an image is tokenized into discrete visual tokens according to the learned vocabulary. During pre-training, each image has two views, i.e., image patches, and visual tokens. We randomly mask some proportion of image patches (gray patches in the figure) and replace them with a special mask embedding `[M]`. Then the patches are fed to a backbone vision Transformer. The pre-training task aims at predicting the visual tokens of the *original* image based on the encoding vectors of the *corrupted* image.

Masked Autoencoders Are Scalable Vision Learners, CVPR'22

Link

Scope of this paper:

1. Masked autoencoding
2. Insights of comparison between images and languages.

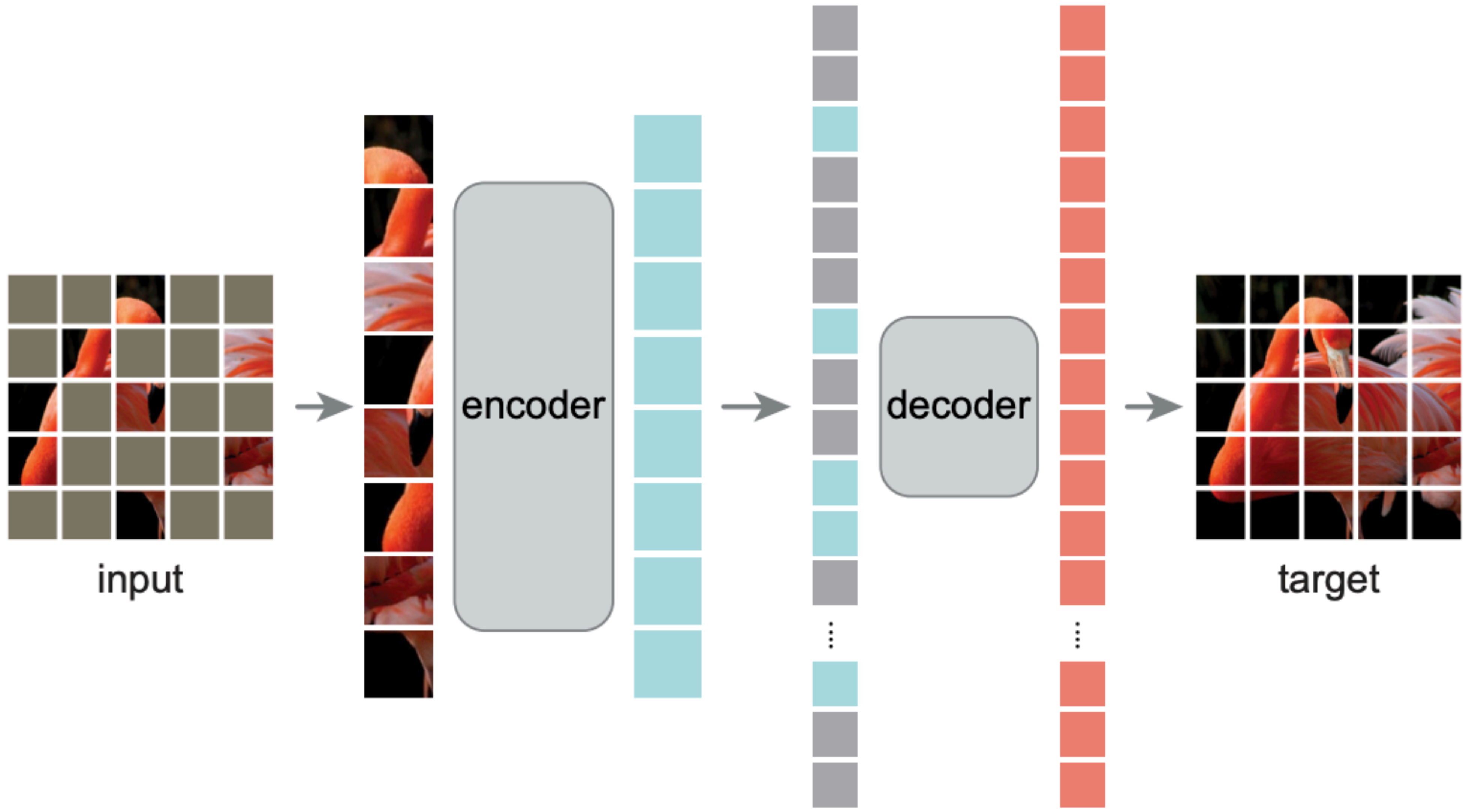


Figure 1. **Our MAE architecture**. During pre-training, a large random subset of image patches (*e.g.*, 75%) is masked out. The encoder is applied to the small subset of *visible patches*. Mask tokens are introduced *after* the encoder, and the full set of encoded patches and mask tokens is processed by a small decoder that reconstructs the original image in pixels. After pre-training, the decoder is discarded and the encoder is applied to uncorrupted images (full sets of patches) for recognition tasks.

Masked Autoencoders Are Scalable Vision Learners, CVPR'22

Question: *what makes masked autoencoding different between vision and language?*

Masked Autoencoders Are Scalable Vision Learners, CVPR'22

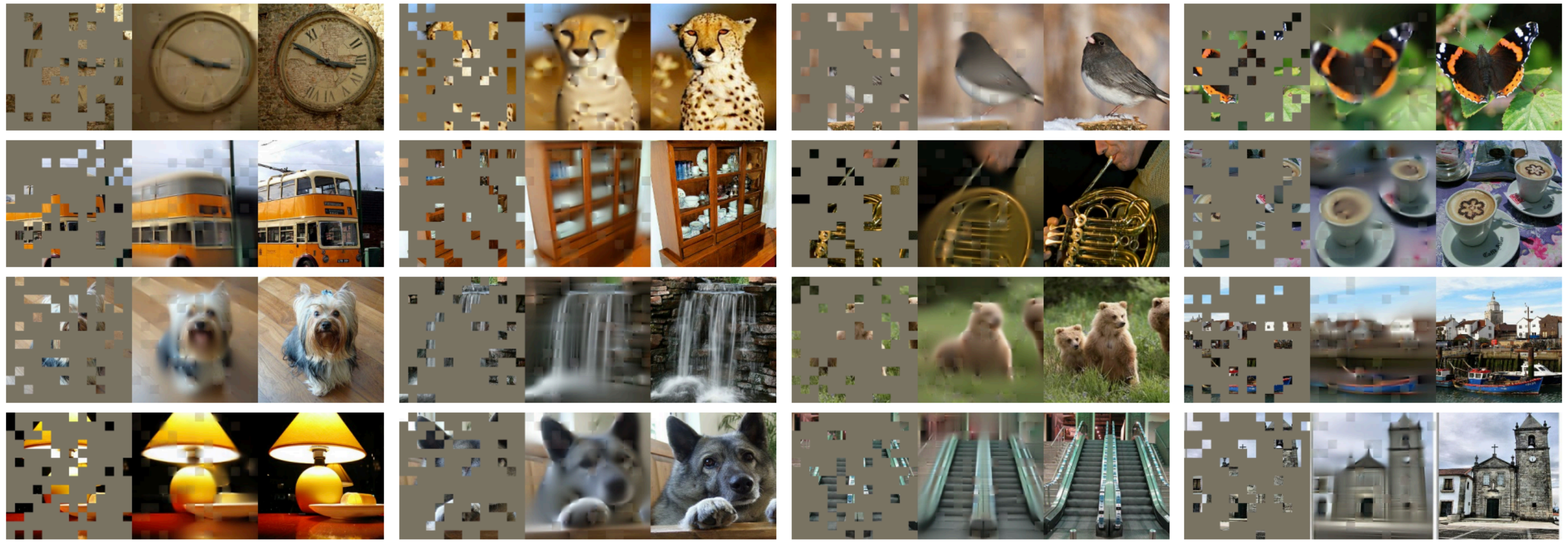
From following perspectives:

- Architectures are different.
    - In NLP, Transformer has been the dominant model.
    - In vision, CNN were dominant over the last decade.
    - —> this architecture gap has been addressed by ViT

Masked Autoencoders Are Scalable Vision Learners, CVPR'22

From following perspectives:

- Architectures are different.
  - In NLP, Transformer has been the dominant model.
  - In vision, CNN were dominant over the last decade.
  - —> this architecture gap has been addressed by ViT.
- Information density is different between language and vision.
  - In NLP, languages are highly semantic and information-dense.
  - In vision, images are natural signals with heavy spatial redundancy.
  - —> high masking ratio: reduce redundancy and makes pre-text tasks more challenging.



Masked Autoencoders Are Scalable Vision Learners, CVPR'22

From following perspectives:

- Architectures are different.
    - In NLP, Transformer has been the dominant model.
    - In vision, CNN were dominant over the last decade.
    - —> this architecture gap has been addressed by ViT.
- Information density is different between language and vision.
    - In NLP, languages are highly semantic and information-dense.
    - In vision, images are natural signals with heavy spatial redundancy.
    - —> high masking ratio: reduce redundancy and makes pre-text tasks more challenging.
- The antoencoder's decoder plays a different role between reconstructing text and images.
    - In vision, the decoder reconstructs pixels — output is of a lower semantic level than common recognition tasks.
    - In NLP, the decoder reconstructs missing words — contain rich semantic information.
    - —> in vision, the decoder is more important; while in NLP, the decoder can be trivial (as MLP).
        - MAE decoder has another series of Transformer blocks, and only used during SSL pre-training.

Masked Autoencoders Are Scalable Vision Learners, CVPR'22

Results on ImageNet-1K.

method	pre-train data	ViT-B	ViT-L	ViT-H	ViT-H_{448}
scratch, our impl.	-	82.3	82.6	83.1	-
DINO [5]	IN1K	82.8	-	-	-
MoCo v3 [9]	IN1K	83.2	84.1	-	-
BEiT [2]	IN1K+DALLE	83.2	85.2	-	-
MAE	IN1K	83.6	85.9	86.9	**87.8**

Table 3. **Comparisons with previous results on ImageNet-1K**. The pre-training data is the ImageNet-1K training set (except the tokenizer in BEiT was pre-trained on 250M DALLE data [50]). All self-supervised methods are evaluated by end-to-end fine-tuning. The ViT models are B/16, L/16, H/14 [16]. The best for each column is underlined. All results are on an image size of 224, except for ViT-H with an extra result on 448. Here our MAE reconstructs normalized pixels and is pre-trained for 1600 epochs.

Summary

	View Construction	SSL Objective	Contrastive or Generative
ViT (SSL part)	Masked patches Mean of patches	Reconstruction to the mean of patches	Generative
DINO	Global: larger patch Local: smaller patch	Teacher-student (BYOL)	Generative
MoCo-v3	Two random augmentations as two views	InfoNCE	Contrastive
BEiT	Masked patches Visual Tokens: latent from discrete VAE	Reconstruction to visual tokens	Generative
MAE	Masked patches Raw patches	Reconstruction to raw patches	Generative

Recent Progress on Transformer & SSL

1. **Vision**
2. **Graphs & Molecules**
   1. **Graphormer, NeurIPS'21**
   2. **Keep it Simple, ArXiv'21**
   3. **ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction, NeurIPS'20 ML4M Workshop**
3. **Tabular Data**

Graphormer: Do Transformers Really Perform Bad for Graph Representation?, NeurIPS'21

Link

Scope of this paper:
A GNN algorithm.

Graphormer: Do Transformers Really Perform Bad for Graph Representation?, NeurIPS’21

Three key components claimed in this paper:

1. Centrality Encoding
2. Spatial Encoding
3. Edge Encoding in the Attention

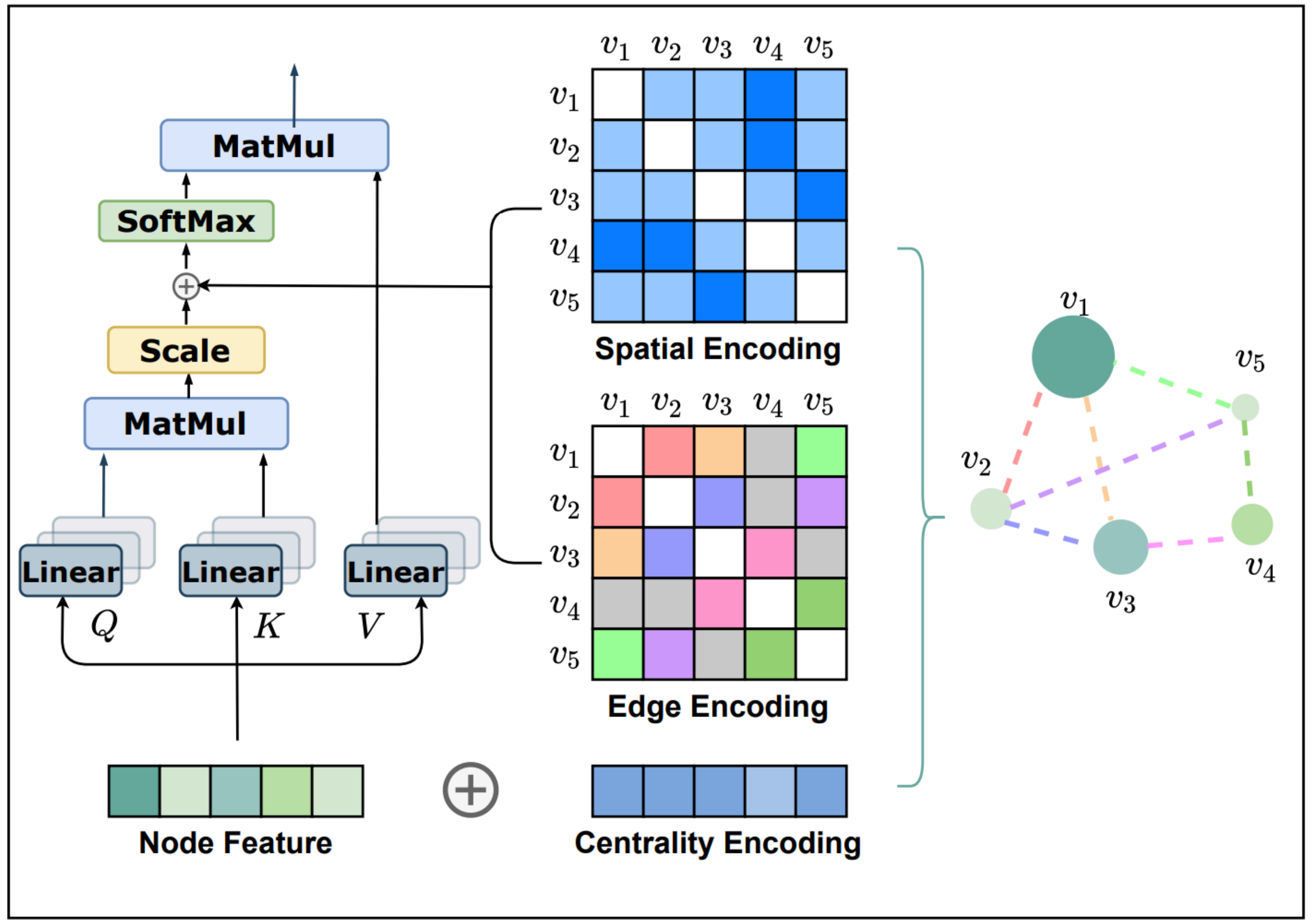


Figure 1: An illustration of our proposed centrality encoding, spatial encoding, and edge encoding in Graphormer.

Graphormer: Do Transformers Really Perform Bad for Graph Representation?, NeurIPS'21

1. Centrality Encoding
    - Node centrality measures how important a node is in the graph.
    - Should be added into the model.
    - Degree as node centrality, and should be added into the node feature. (But most of the existing GNN models already done this?)

$$h_i^{(0)} = x_i + z^-_{\deg^-(v_i)} + z^+_{\deg^+(v_i)}, \tag{5}$$

where $z^-, z^+ \in \mathbb{R}^d$ are learnable embedding vectors specified by the indegree $\deg^-(v_i)$ and outdegree $\deg^+(v_i)$ respectively. For undirected graphs, $\deg^-(v_i)$ and $\deg^+(v_i)$ could be unified to $\deg(v_i)$. By using the centrality encoding in the input, the softmax attention can catch the node importance signal in the queries and the keys. Therefore the model can capture both the semantic correlation and the node importance in the attention mechanism.

Graphormer: Do Transformers Really Perform Bad for Graph Representation?, NeurIPS'21

2. Spatial Encoding
   - Embed node pairwise spatial information.
   - Use 2D topology graph distance, i.e., shortest path distance.
   - Assign each output a learnable scalar, which serves as a bias term in self-attention module.

$$A_{ij} = \frac{(h_i W_Q)(h_j W_K)^T}{\sqrt{d}} + b_{\phi(v_i, v_j)}, \tag{6}$$

where $b_{\phi(v_i,v_j)}$ is a learnable scalar indexed by $\phi(v_i, v_j)$, and shared across all layers.

Graphormer: Do Transformers Really Perform Bad for Graph Representation?, NeurIPS'21

3. Edge Encoding in the Attention
   - For each node pair, find a shortest path.
   - Path encoding: the average of the dot-products of the edge feature and a learnable embedding along the path.

$$A_{ij} = \frac{(h_i W_Q)(h_j W_K)^T}{\sqrt{d}} + b_{\phi(v_i,v_j)} + c_{ij}, \text{ where } c_{ij} = \frac{1}{N}\sum_{n=1}^{N} x_{e_n}(w_n^E)^T, \tag{7}$$

where x_{e_n} is the feature of the n-th edge e_n in SP_{ij}, $w_n^E \in \mathbb{R}^{d_E}$ is the n-th weight embedding, and d_E is the dimensionality of edge feature.

Graphormer: Do Transformers Really Perform Bad for Graph Representation?, NeurIPS'21

Results need benchmarking.

(PCBA & HIV results are using pre-training.)

Table 2: Results on MolPCBA.

method	#param.	AP (%)
DeeperGCN-VN+FLAG [30]	5.6M	28.42±0.43
DGN [2]	6.7M	28.85±0.30
GINE-VN [5]	6.1M	29.17±0.15
PHC-GNN [29]	1.7M	29.47±0.26
GINE-APPNP [5]	6.1M	29.79±0.30
GIN-VN[54] (fine-tune)	3.4M	29.02±0.17
Graphormer-FLAG	119.5M	**31.39**±0.32

Table 3: Results on MolHIV.

method	#param.	AUC (%)
GCN-GraphNorm [5, 8]	526K	78.83±1.00
PNA [10]	326K	79.05±1.32
PHC-GNN [29]	111K	79.34±1.16
DeeperGCN-FLAG [30]	532K	79.42±1.20
DGN [2]	114K	79.70±0.97
GIN-VN[54] (fine-tune)	3.3M	77.80±1.82
Graphormer-FLAG	47.0M	**80.51**±0.53

RF + Fingerprints: 80.60

Table 4: Results on ZINC.

method	#param.	test MAE
GIN [54]	509,549	0.526±0.051
GraphSage [18]	505,341	0.398±0.002
GAT [50]	531,345	0.384±0.007
GCN [26]	505,079	0.367±0.011
GatedGCN-PE [4]	505,011	0.214±0.006
MPNN (sum) [15]	480,805	0.145±0.007
PNA [10]	387,155	0.142±0.010
GT [13]	588,929	0.226±0.014
SAN [28]	508, 577	0.139±0.006
$\text{Graphormer}_{\text{SLIM}}$	489,321	**0.122**±0.006

Graphormer in PCQM4M: FIRST PLACE SOLUTION OF KDD CUP 2021 & OGB LARGE-SCALE CHALLENGE GRAPH PREDICTION TRACK

Key differences:

1. An ensemble of Graphormer & ExpC
2. For featurization: use 3D euclidean distance instead 2D topology distance in Graphormer.

Type	Attribute type	Description
Atom	Atomic number	Number of protons
	Degree	With Hydrogens and without Hydrogens
	Number of Hydrogens	
	Hybridization	Sp, sp2 or sp3 etc.
	Aromatic atom	a part of an aromatic ring
	Is in ring	
	Valence	Explicit valence, implicit valence, total valence
	Radical electrons	
	Formal charge	
	Gasteiger charge	
	Periodic table features	rvdw, default valence, outer electrons, rb0 and etc.
	Chirality	Is chiral center
	Donor or accepter	donate electron or accept electron
Bond	Bond type	Single, double, triple, aromatic bond, etc.
	Bond stereo	Z, E, cis, trans double bond, etc.
	Bond direction	Bond's direction (for chirality)
	Is conjugated	
	Is in ring	
	Euclidean distance	Using MMFF optimizer (RDKit[2]) to obtain the coordinates of a molecule
Atom Pair	Euclidean distance	Using MMFF optimizer to obtain the coordinates of a molecule

Table 1: Atomic and bond attributes used to construct graph inputs.

Graphormer: FIRST PLACE SOLUTION OF KDD CUP 2021 & OGB LARGE-SCALE CHALLENGE GRAPH PREDICTION TRACK

$$A_{ij} = \frac{(h_i W_Q)(h_j W_K)^T}{\sqrt{d}} + \boxed{b_{\phi(v_i,v_j)}} + \boxed{c_{ij}}, \text{ where } c_{ij} = \frac{1}{N}\sum_{n=1}^{N} x_{e_n}(w_n^E)^T,$$

1. For spatial encoding:

Use RBF on the Euclidean distance as $\phi(v_i, v_j)$

2. For path encoding:

$$X_{bonddist} = X_{bonddist} + \text{Lapalce}(\mu, b),$$

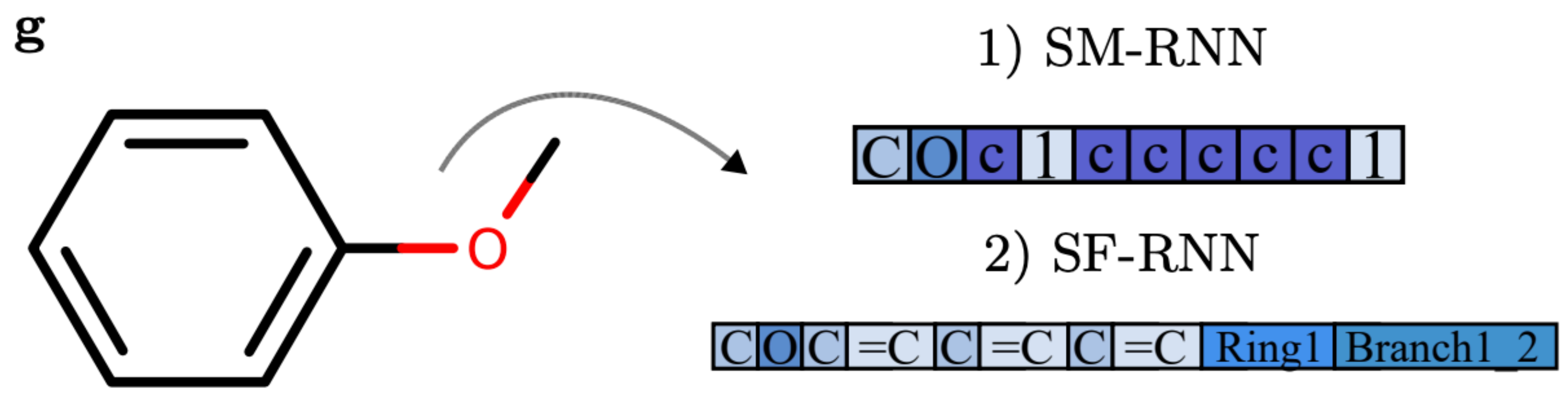
where $\mu = 0.001994, b = 0.031939$. We choose μ and b by fitting the difference between the calculated results of RDKit and DFT, on another dataset called QM9 [Ramakrishnan et al., 2014], which provides the DFT-calculated 3D molecular structures.

Keeping it Simple: Language Models can learn Complex Molecular Distributions, ArXiv'21

Link

Scope of this paper:

Re-exploration of RNN (2-layer LSTM) + string representation: SMILES & SELFIES

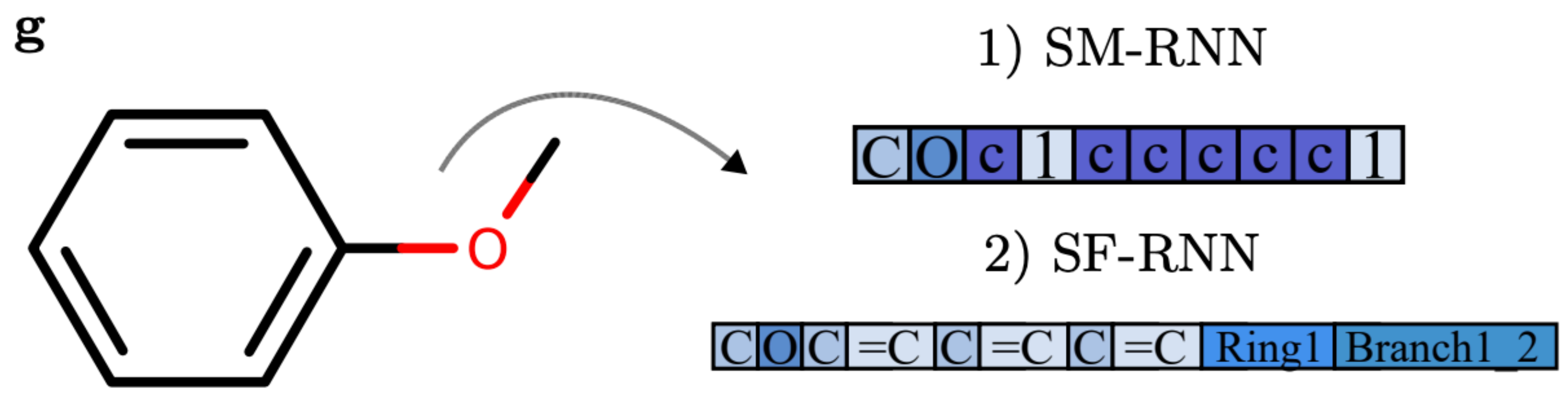


Keeping it Simple: Language Models can learn Complex Molecular Distributions, ArXiv'21

Link

Scope of this paper:

Re-exploration of RNN (2-layer LSTM) + string representation: SMILES & SELFIES



Comparable with JTVAE & CGVAE.

ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction

Link

Input: SMIELS or SELFIES (similar performance)

Backbone model: ChemBERTa, built on RoBERTa [1]

Pre-training task: masked language model (MLM)

		BBBP 2,039		ClinTox (CT_TOX) 1,478		HIV 41,127		Tox21 (SR-p53) 7,831	
		ROC	PRC	ROC	PRC	ROC	PRC	ROC	PRC
w/ SSL	ChemBERTa 10M	0.643	0.620	0.733	0.975	0.622	0.119	**0.728**	0.207
w/o SSL	D-MPNN	**0.708**	0.697	**0.906**	**0.993**	0.752	0.152	0.688	**0.429**
	RF	0.681	0.692	0.693	0.968	**0.780**	**0.383**	0.724	0.335
	SVM	0.702	**0.724**	0.833	0.986	0.763	0.364	0.708	0.345

Table 1: Comparison of ChemBERTa pretrained on 10M PubChem compounds and Chemprop baselines on selected MoleculeNet tasks. We report both ROC-AUC and PRC-AUC to give a full picture of performance on class-imbalanced tasks.

[1] Liu, Yinhan, et al. "Roberta: A robustly optimized bert pretraining approach." *arXiv preprint arXiv:1907.11692* (2019).

Recent Progress on Transformer & SSL

1. **Vision**
2. **Graphs & Molecules**
3. **Tabular Data**
    1. **TabNet, ArXiv’19 / AAAI’21**
    2. **TabTransformer, ArXiv’20**
    3. **VIME (Value Imputation and Mask Estimation), NeurIPS’20**

Tabular Data

Problem formulation:

Age	Cap. gain	Education	Occupation	Gender	Relationship
60	200000	Bachelors	Exec-managerial	M	Husband
23	0	High-school	Farming-fishing	M	Unmarried
45	5000	Doctorate	Prof-specialty	M	Husband
23	0	High-school	Handlers-cleaners	F	Wife
56	300000	Bachelors	Exec-managerial	M	Husband
38	10000	Bachelors	Prof-specialty	F	Wife
23	0	High-school	Armed-Forces	M	Husband

Income > $50k
True
False
True
False
True
True
False

TabNet, AAAI'21

Link

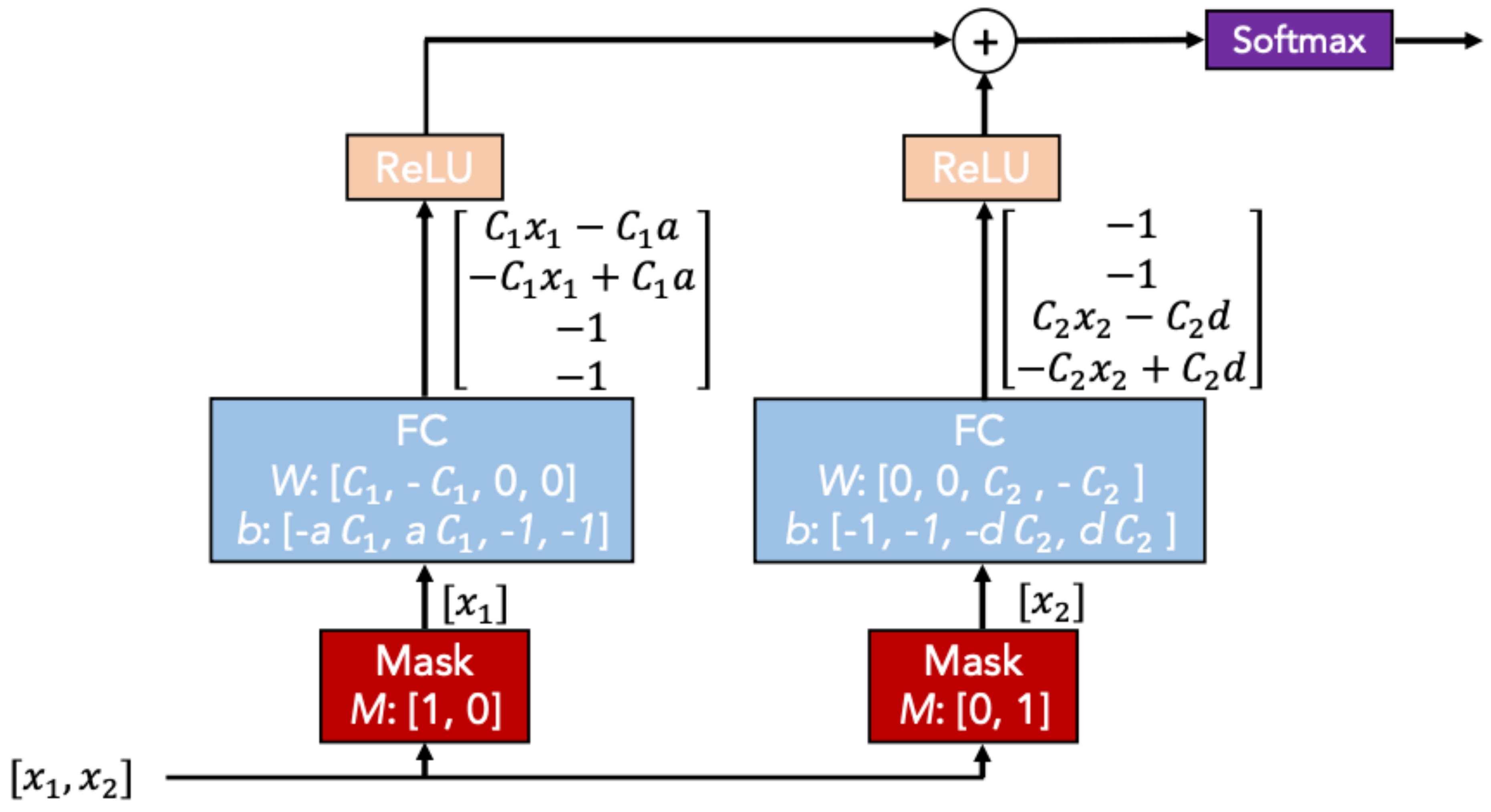
Scope of this paper:

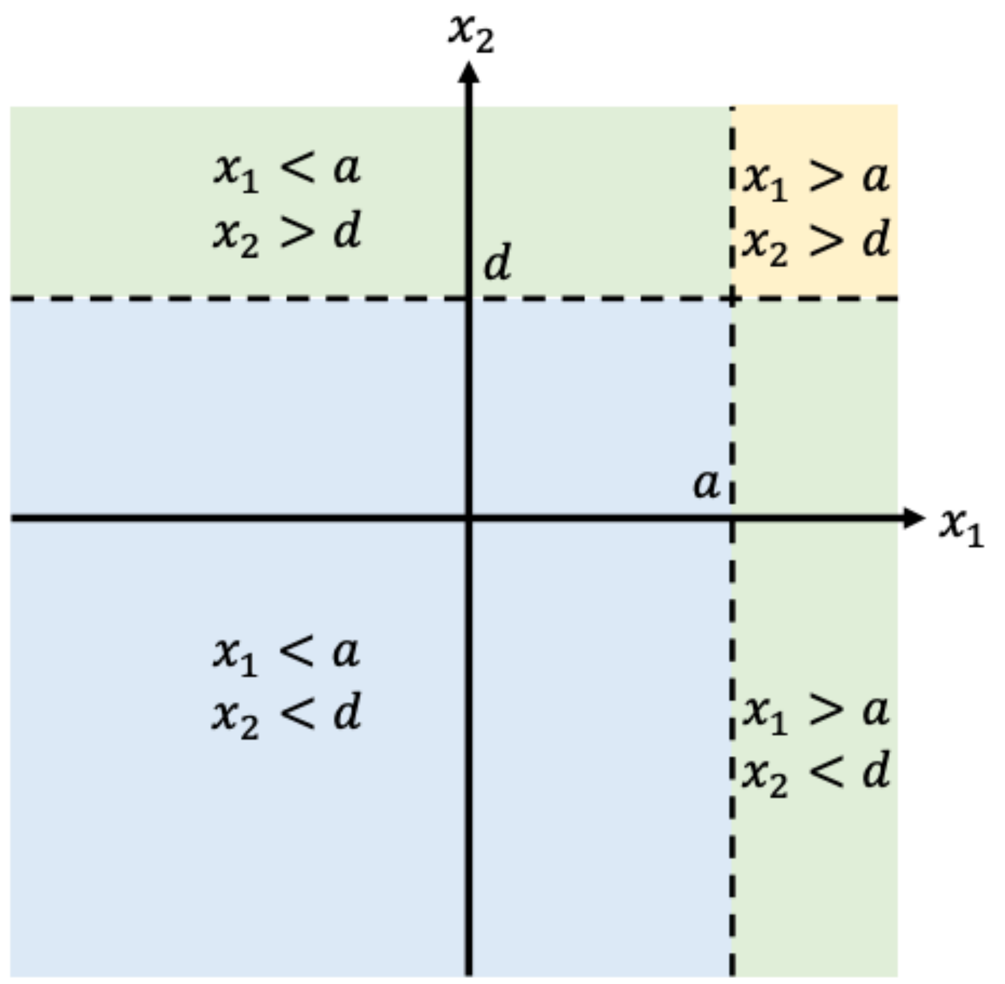
- High-level pipeline for supervised learning
- High-level pipeline for self-supervised learning
- Model architecture

TabNet, AAAI'21

- Supervised learning: decision tree (DT)-like classification using DNN.
- Or, using DNN for the decision making in DT-like algorithm (instead of the entropy, etc.)
  - End-to-end learning
  - Explicit representation
  - Larger model capacity

- An example:





TabNet, AAAI'21

- Self-supervised learning: masked auto-encoding.

Unsupervised pre-training

Age	Cap. gain	Education	Occupation	Gender	Relationship
53	200000	?	Exec-managerial	F	Wife
19	0	?	Farming-fishing	M	?
?	5000	Doctorate	Prof-specialty	M	Husband
25	?	?	Handlers-cleaners	F	Wife
59	300000	Bachelors	?	?	Husband
33	0	Bachelors	?	F	?
?	0	High-school	Armed-Forces	?	Husband

TabNet encoder

TabNet decoder

Age	Cap. gain	Education	Occupation	Gender	Relationship
		Masters			
		High-school			Unmarried
43					
	0	High-school		F	
			Exec-managerial	M	
			Adm-clerical		Wife
39				M	

Supervised fine-tuning

Age	Cap. gain	Education	Occupation	Gender	Relationsh
60	200000	Bachelors	Exec-managerial	M	Husbanc
23	0	High-school	Farming-fishing	M	Unmarrie
45	5000	Doctorate	Prof-specialty	M	Husbanc
23	0	High-school	Handlers-cleaners	F	Wife
56	300000	Bachelors	Exec-managerial	M	Husbanc
38	10000	Bachelors	Prof-specialty	F	Wife
23	0	High-school	Armed-Forces	M	Husbanc

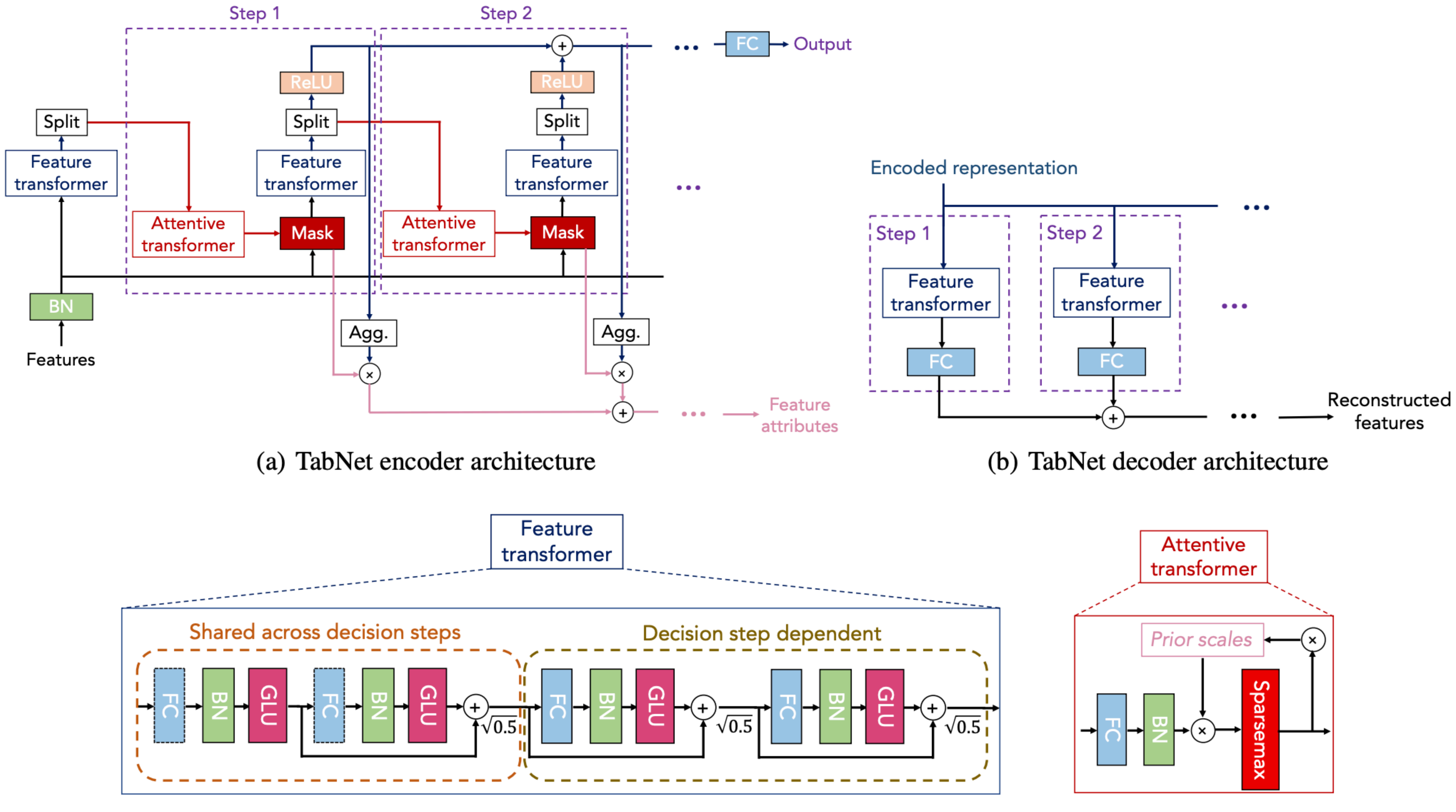
TabNet encoder

Decision making

Income > $50k
True
False
True
False
True
True
False

TabNet, AAAI'21

- Fig (b) is for SSL only.
- The transformer here is not the Transformer
  - Feature transformer
  - Attentive transformer (mask)



(a) TabNet encoder architecture
(b) TabNet decoder architecture
(c)
(d)

TabTransformer, ArXiv'20

Link

Scope of this paper:

- Model architecture
- Supervised learning
- Self-supervised learning

TabTransformer, ArXiv’20

- Two key components:
  - Transformer
  - Column embedding for categorical feature

for i-th column, the j-th categorical value,

embedding is $e_{\phi_i}(j) = [c_{\phi_i}, w_{\phi_{ij}}]$

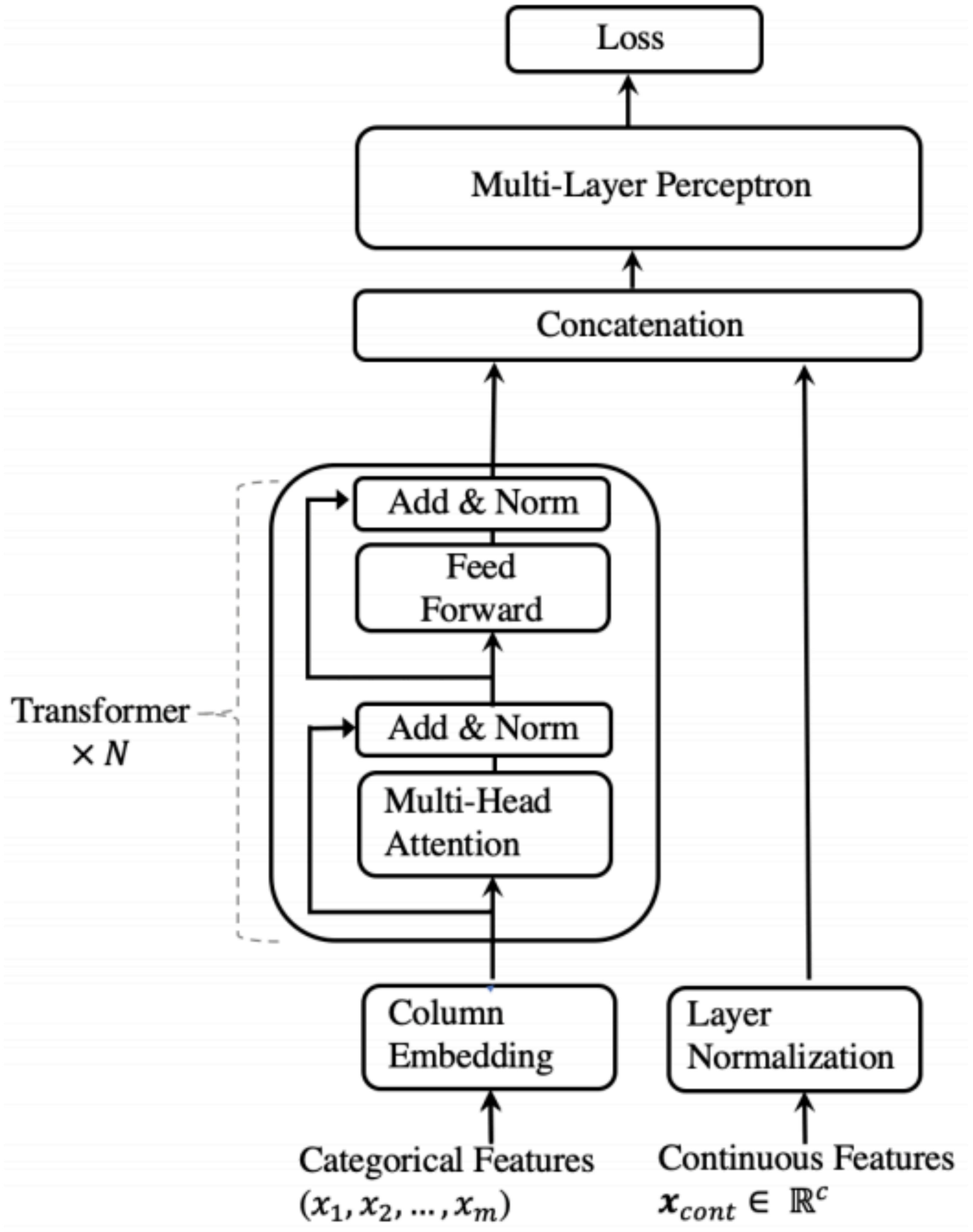


Figure 1: The architecture of TabTransformer.

TabTransformer, ArXiv'20

- Supervised learning

- Self-supervised learning: explore 2 methods
  1. Masked auto-encoding / Masked language modeling (MLM)
  2. Replaced token detection (RTD)

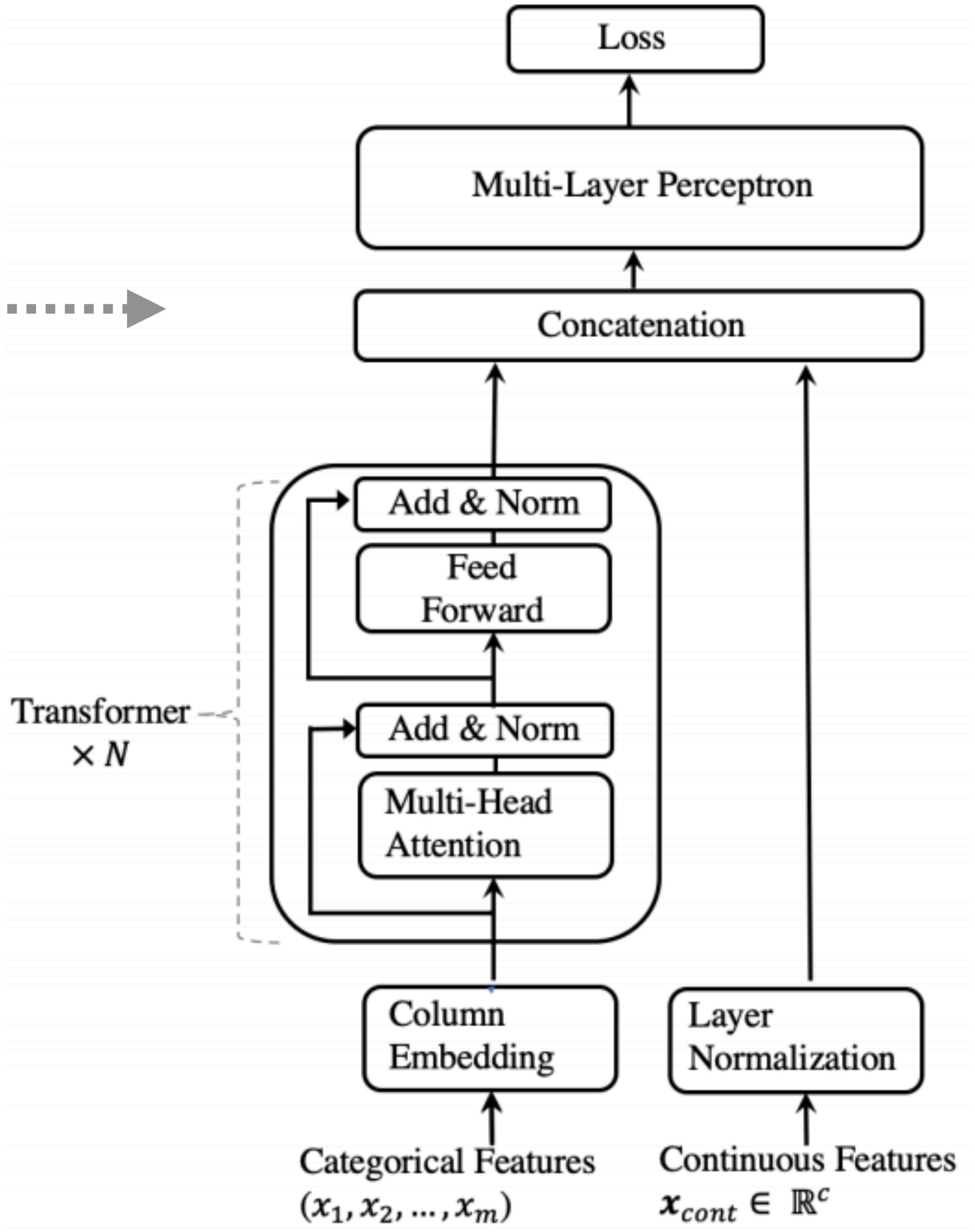


Figure 1: The architecture of TabTransformer.

VIME, NeurIPS’20

Link

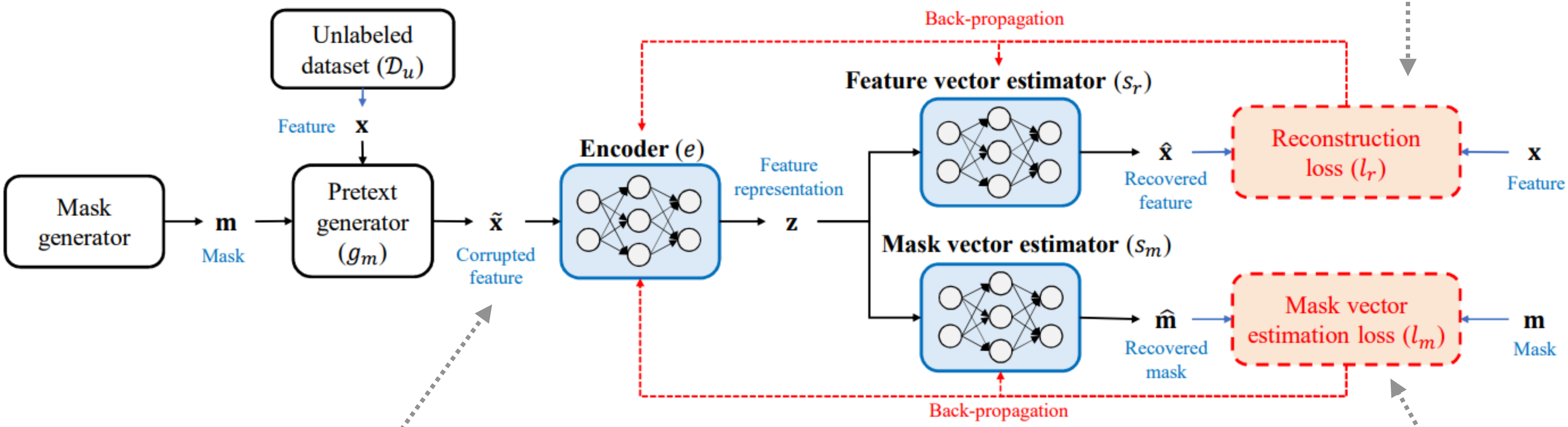
Scope of this paper:

- Self-supervised learning
- Semi-supervised learning

VIME, NeurIPS’20

- Self-supervised learning

MAE/ MLM



RTD

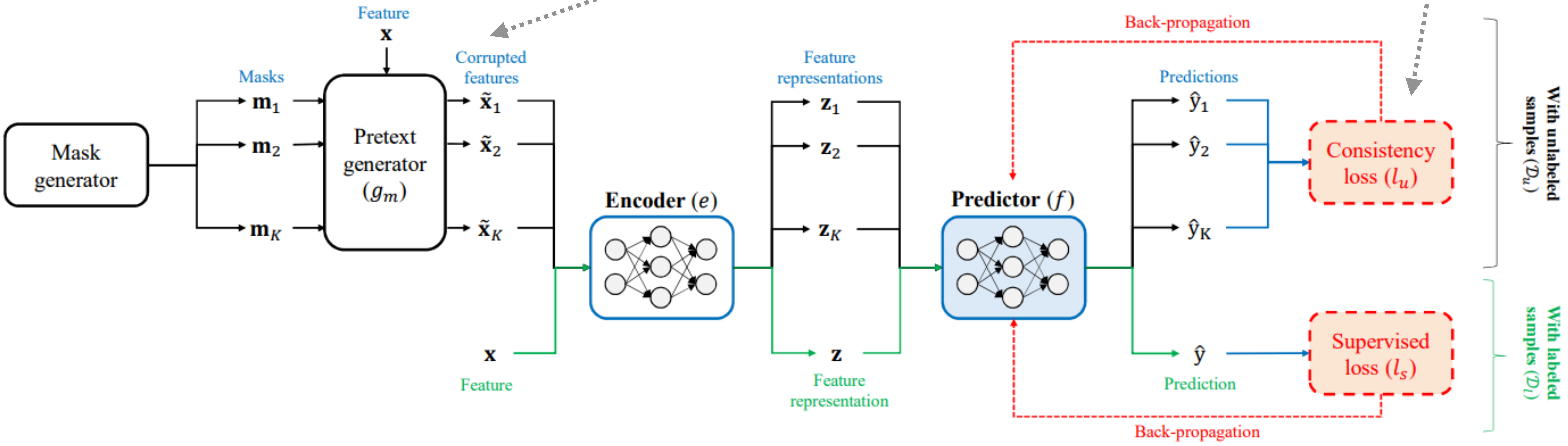
$$\tilde{\mathbf{x}} = g_m(\mathbf{x}, \mathbf{m}) = \mathbf{m} \odot \bar{\mathbf{x}} + (1 - \mathbf{m}) \odot \mathbf{x}$$

VIME, NeurIPS'20

- Semi-supervised learning

$$\tilde{\mathbf{x}} = g_m(\mathbf{x}, \mathbf{m}) = \mathbf{m} \odot \bar{\mathbf{x}} + (1 - \mathbf{m}) \odot \mathbf{x}$$

$$\left(f_e(\tilde{\mathbf{x}}) - f_e(\mathbf{x})\right)^2$$



Conclusion

- Transformer has been advancing from NLP to many different fields: vision, graph applications, tabular data, etc.
- Self-supervised learning, on the other hand, provides a powerful yet model-agnostic framework for unsupervised representation learning.