BEIT: BERT Pre-Training of Image Transformers
(arXiv preprint, 2021)

Yonsoo Kim

Paper

BEIT: BERT Pre-Training of Image Transformers

Hangbo Bao[*], **Li Dong**, **Furu Wei**
Microsoft Research
{t-habao,lidong1,fuwei}@microsoft.com

Abstract

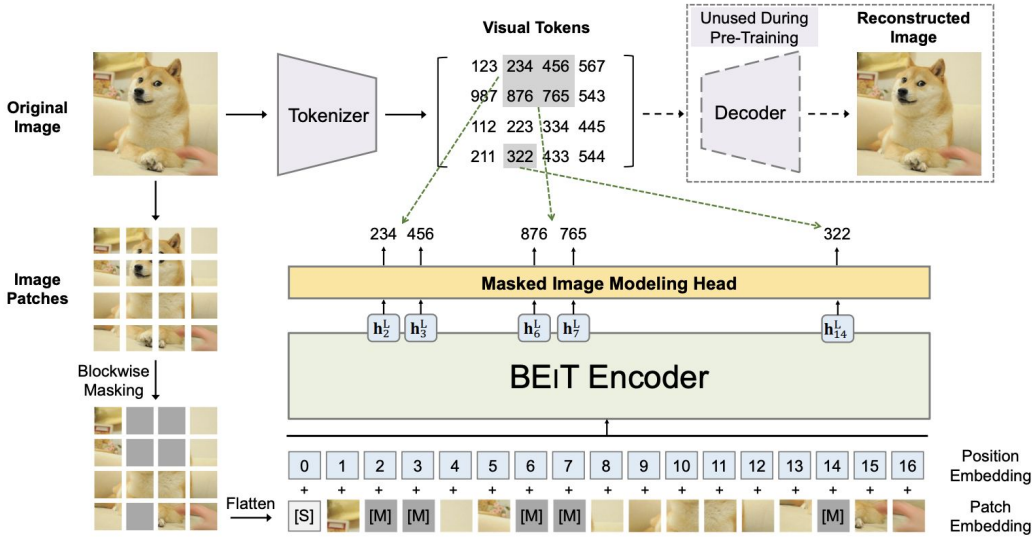
We introduce a self-supervised vision representation model **BEIT**, which stands for **B**idirectional **E**ncoder representation from **I**mage **T**ransformers. Following BERT (Devlin et al., 2019) developed in the natural language processing area, we propose a *masked image modeling* task to pretrain vision Transformers. Specifically, each image has two views in our pre-training, i.e, image patches (such as 16×16 pixels), and visual tokens (i.e., discrete tokens). We first "tokenize" the original image into visual tokens. Then we randomly mask some image patches and fed them into the backbone Transformer. The pre-training objective is to recover the original visual tokens based on the corrupted image patches. After pre-training BEIT, we directly fine-tune the model parameters on downstream tasks by appending task layers upon the pretrained encoder. Experimental results on image classification and semantic segmentation show that our model achieves competitive results with previous pre-training methods. For example, base-size BEIT achieves 83.2% top-1 accuracy on ImageNet-1K, significantly outperforming from-scratch DeiT training (81.8%; Touvron et al., 2020) with the same setup. Moreover, large-size BEIT obtains 86.3% only using ImageNet-1K, even outperforming ViT-L with supervised pre-training on ImageNet-22K (85.2%; Dosovitskiy et al., 2020). The code and pretrained models are available at `https://aka.ms/beit`.

Key Ideas

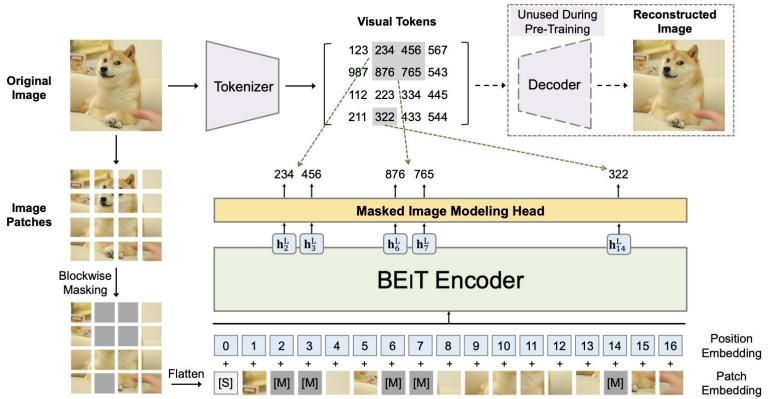
- self-supervised vision representation model
- propose **a masked image modeling(MIM)** task to pre-train vision Transformers
- Fine-tuning on downstream tasks by appending task layers upon the pre-trained encoder (image classification, semantic segmentation)

Overview

- before pre-training, learn "image tokenizer" via auto-encoding-style reconstruction (dVAE)
  - image is tokenized into discrete visual tokens according to the learned vocab.
- randomly mask some image patches (gray patches) ⇒ special mask embedding [M]
- **pre-training task : predicting the visual tokens of the original image based on the encoding vectors of the corrupted image.**
  - instead of the raw pixels of masked patches



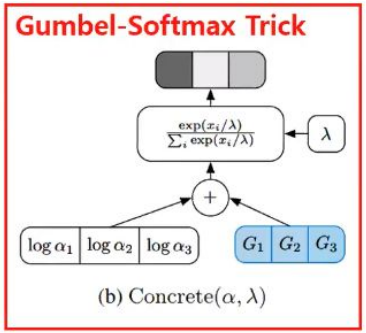
Overview - workflow



1. Image is divided into patches.
2. Patches are masked randomly.
3. Flatten the image patch into a vector.
4. Positional embeddings and embeddings are learned for the patches.
5. Now these embeddings are passed through BeIT Encoder
6. For masked part, model has to predict image token.
7. These tokens come from image tokenizer.
8. Finally, image can be reconstructed using tokens.

Tokenizer & Decoder

- learn the image tokenizer via discrete VAE(dVAE)
- tokenizer
  - $q_\phi(z|x)$ maps image patches x into discrete tokens z according to a visual codebook(i.e., vocab.)
  - **of visual tokens == # of image patches**
- decoder
  - $p_\psi(x|z)$ learns to reconstruct the input image *x* based on the visual tokens *z*
  - reconstruction objective → $\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\psi(x|z)]$
  - latent visual tokens are discrete → the model training is non-differentiable!
    ⇒ Gumbel-softmax relaxation



Pre-Training BEIT : Masked Image Modeling(MIM)

- What to do : randomly mask some % of image patches and then **predict the visual tokens** that are corresponding to the masked patches
- randomly mask
  - masking 40% of image patches
  - blockwise masking
- predict the corresponding visual tokens

  $p_{MIM}(z'|x^{\mathcal{M}}) = softmax_{z'}(W_c h_i^L + b_c)$
  - **pre-training objective** -> to maximize the log-likelihood of the correct visual tokens z_i given the corrupted image
  - $max \sum_{x \in \mathcal{D}} \mathbb{E}_{\mathcal{M}} [\sum_{i \in \mathcal{M}} \log p_{MIM}(z_i|x^{\mathcal{M}})$

    D : training corpus

    M -> randomly masked positions

    x^M -> corrupted image

Pre-Training BEIT : Masked Image Modeling(MIM)

- **pixel-level** auto-encoding for vision pre-training → pushes the model to focus on short-range dependencies and high-frequency details
- BEIT : predicting discrete **visual tokens** which summarizes the details to high-level abstractions

Variational Autoencoder

- The BEIT pre-training can be viewed as VAE training.
- x : original image, \tilde{x} : masked image, z : visual tokens
- $q_\phi(z|x)$: the image tokenizer that obtains visual tokens
- $p_\psi(x|z)$: decodes the original image given input visual tokens
- $p_\theta(z|\tilde{x})$: recovers the visual tokens based on the masked image (MIM pre-training task)

$$\sum_{(x_i,\tilde{x}_i)\in\mathcal{D}} \Big(\underbrace{\mathbb{E}_{z_i\sim q_\phi(z|x_i)}[\log p_\psi(x_i|z_i)]}_{\text{Stage 1: Visual Token Reconstruction}} + \underbrace{\log p_\theta(\hat{z}_i|\tilde{x}_i)}_{\text{Stage 2: Masked Image Modeling}} \Big) \quad (3)$$

6p

- decoder
  - $p_\psi(x|z)$ learns to reconstruct the input image *x* ba
  - reconstruction objective → $\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\psi(x|z)]$

Experiments

- Top-1 accuracy on CIFAR-100 and ImageNet-1K

Models	CIFAR-100	ImageNet
Training from scratch (i.e., random initialization)		
ViT_{384} (Dosovitskiy et al., 2020)	48.5*	77.9
DeiT (Touvron et al., 2020)	n/a	81.8
Supervised Pre-Training on ImageNet-1K (using labeled data)		
ViT_{384} (Dosovitskiy et al., 2020)	87.1	77.9
DeiT (Touvron et al., 2020)	90.8	81.8
Self-Supervised Pre-Training on ImageNet-1K (without labeled data)		
iGPT-1.36B† (Chen et al., 2020a)	n/a	66.5
ViT_{384}-JFT300M‡ (Dosovitskiy et al., 2020)	n/a	79.9
DINO (Caron et al., 2021)	91.7	82.8
MoCo v3 (Chen et al., 2021)	87.1	n/a
BEiT (ours)	90.1	**83.2**
Self-Supervised Pre-Training, and Intermediate Fine-Tuning on ImageNet-1K		
BEiT (ours)	**91.8**	**83.2**

Experiments

- Top-1 accuracy on CIFAR-100 and ImageNet-1K
  + Fine-tuning to 384x384 resolution

Models	Model Size	Image Size	ImageNet
Training from scratch (i.e., random initialization)			
ViT_{384}-B (Dosovitskiy et al., 2020)	86M	384^2	77.9
ViT_{384}-L (Dosovitskiy et al., 2020)	307M	384^2	76.5
DeiT-B (Touvron et al., 2020)	86M	224^2	81.8
DeiT_{384}-B (Touvron et al., 2020)	86M	384^2	83.1
Supervised Pre-Training on ImageNet-22K (using labeled data)			
ViT_{384}-B (Dosovitskiy et al., 2020)	86M	384^2	84.0
ViT_{384}-L (Dosovitskiy et al., 2020)	307M	384^2	85.2
Self-Supervised Pre-Training on ImageNet-1K (without labeled data)			
iGPT-1.36B† (Chen et al., 2020a)	1.36B	224^2	66.5
ViT_{384}-B-JFT300M‡ (Dosovitskiy et al., 2020)	86M	384^2	79.9
DINO-B (Caron et al., 2021)	86M	224^2	82.8
BEIT-B (ours)	86M	224^2	83.2
BEIT_{384}-B (ours)	86M	384^2	84.6
BEIT-L (ours)	307M	224^2	85.2
BEIT_{384}-L (ours)	307M	384^2	**86.3**

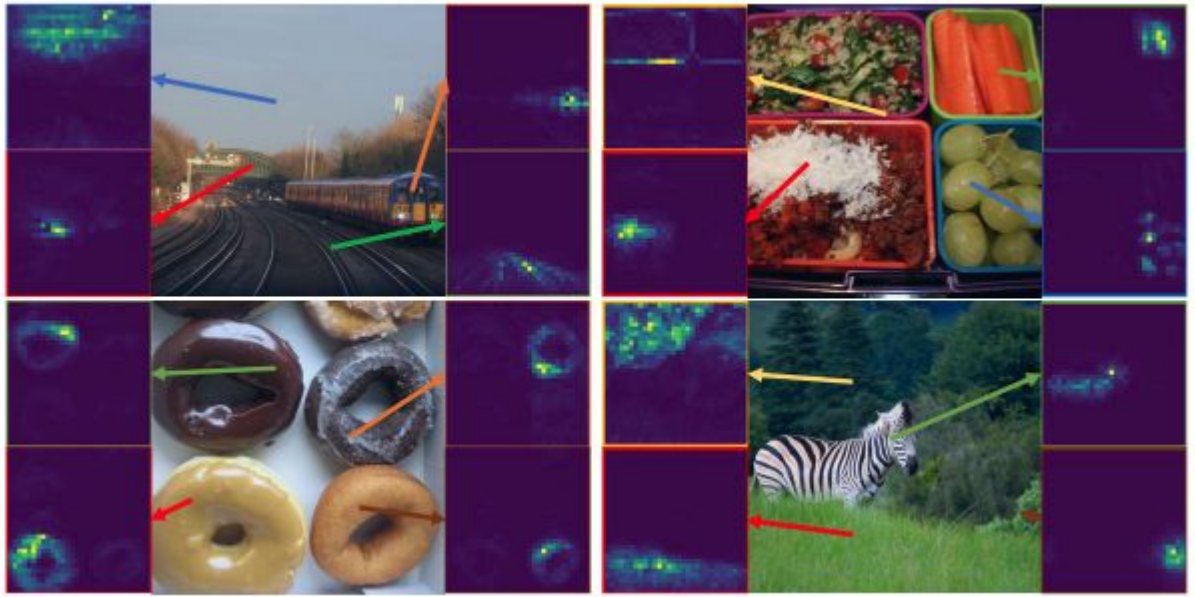
Experiments

- Ablation studies

Models	ImageNet	ADE20K
BEIT (300 Epochs)	82.86	44.65
– Blockwise masking	82.77	42.93
– Visual tokens (i.e., recover masked pixels)	81.04	41.38
– Visual tokens – Blockwise masking	80.50	37.09
+ Recover 100% visual tokens	82.59	40.93
Pretrain longer (800 epochs)	83.19	45.58

Experiments

- Self-attention map for different reference points
  → BEIT is able to separate objects, although self-supervised pre-training does not use manual annotations



Conclusion

- BERT-like pre-training(i.e., auto-encoding with masked input)
- achieves strong fine-tuning on downstream tasks