Learning Transferable Visual Models From Natural Language Supervision

Alec Radford, Jong Wook Kim, Chris Hallacy, et al.
38th International Conference on Machine Learning (ICML), 2021
Outline

- Problem Statement
- Related Works
- Approach
- Experiments & Results
- Limitations, Societal Implications
- Summary of Strengths, Weaknesses, Relationship to Other Papers
Problem Statement

- Create robust vision models with natural language supervision
- Go beyond previous limitations on models with specific labels
- Enable zero-shot transfer to unseen tasks
- Involved training both a text encoder as well as an image encoder
- ResNets/models were relatively limited to ImageNet classifications
- Utilize a metric ton of image-text data available without retraining and dependency on task-specific datasets
Overcoming Traditional Approaches

• Not restricted to 1-of-N label space
  • Open-ended language descriptions
• Prior work issues
  • Data scale + compute (ResNet)
  • Restricted spaces (Inception-v3)
  • Dataset dependencies
• CLIP’s novelty
  • New categorization, even those not seen
Related Works

• **Visual N-Grams (2017)**
  - First zero-shot transfer methodology
  - CNN to predict relevant words and n-grams (adjacent order) from images
  - “Unsupervised” training on 30 mil Flickr images (used comments)

• **VirTex (2020)**
  - Transformer-based image captioning
  - CNN encoder + transformer decoder architecture
  - Caption generation for images lead to richer classifications, requires nuanced understanding
  - Better at downstream tasks like segmentation and object detection
Related Works

- VirTex continued (diagram was too good not to discuss)

Dual unidirectional Transformers for caption prediction bidirectional approach
Masked self-attention over captions and cross-attention with image features
Approach

- What is CLIP? Contrastive Language-Image Pre-training
- 400M (image, text) pairs collected from various internet sources
- Image encoder piece: Modified ResNet or Vision Transformer (ViT)
  - Picked based on performance
- Text encoder: Transformer with 63M parameters
Approach (Data Collection)

- Raw web pairs aren’t going to be perfect
  - Plenty of noise and even mismatches, abstract pairs
  - Either way → CLIP gets stronger with weird stuff

- CLIP filtering
  - 500,000 unique internet queries to cover all domains
  - Pulled in captions, descriptions, comments any kind of data paired with images
  - 1 query could produce max most relevant 20k image-text pairs, ensuring diversity

- De-duplication
  - Image text pairs underwent de-duplication which just ensures overlap is minimal
  - Each sample should ideally be unique
  - Also lowers overlap with benchmarking datasets, → real evaluation and generalization capabilities
Approach (Tokenization)

- **Text Tokenizer: Byte Pair Encoding (BPE)**
  - Relate each word in the text as character sequence
  - Words also get end of word tokens (e.g., “fork” → “f o r k</w>”)
- **Frequencies:** Count up common adjacency pairs
- **Merging:** Merge common pairs, add to vocabulary

**Why?**
- Common words and subwords are tokenized well
- Great for zero-shot tasks

**No non-linear projection**
- Other contrastive learning methods use a non-linear projection between the representation and embedded space

---

| word | count |
|------|-------|
| cat  | 4     |
| mat  | 5     |
| mats | 2     |
| mate | 3     |
| ate  | 3     |
| eat  | 2     |
Approach (Image Encodings)

- **ResNet encoder**
  - CNN architecture, conv layers + pooling → feature vector
  - Linear layer for final embedding, L2 normed for ease of similarity

- **ViT encoder**
  - Patches over image, flattened and projected into embedding (like with text)
  - Positional encodings for those patches, multi-head self attention + feedforward neural nets are strong
  - A classification token is added onto the patch embeddings sequence, then normalized too
Approach (Similarity)

- Cosine similarity explanation
Efficient Algorithm

• Contrastive objectives can outperform equivalent predictive objectives
  • Especially true in the visual space
  • CLIP only needs to compute embeddings once per image/text
  • Predictive methods need multiple forward passes
  • Each training example in CLIP provides N-1 negative examples "for free", learn better from a larger effective dataset.

• 4x more efficient with zero-shot transfer to ImageNet compared to a bag-of-words prediction approach

• Alignment of text/image will always be easier than generating an entire caption
Experiments and Results (Linear Probe)

- Linear probe is a simple classifier (log reg) added to pre-trained features some labeled data
  - Beat logistic regression on ResNet50 features on 16/27 datasets – multimodal training power
  - Significance? ROBUST, no task-specific data or fine-tuning needed
  - Particularly good at general object recognition Food101, StanfordCars
  - Specific context-based understanding like EuroSAT and Satellite Imagery give CLIP more trouble
Experiments and Results (Few-Shot)

• Few-shot learning is training on a couple of examples per class
  • Kinda impossible with a huge dataset
• Outperforms 16-shot classifiers using features from other models
  • Embeddings learned by CLIP capture a plenty of transferable knowledge and can generalize to out of domain concepts
• Weird... CLIP vs CLIP???
  • Zero-shot directly specifies and communicates visual concepts more unambiguous
  • Few-shot learning is inferring from examples, (multiple interpretations)
  • ~4 examples for the linear probe to gather enough
Experiments and Results (Distribution Drift)

- Zero-Shot *unseen classification*:
  - Robust:
    - 75% reduction between ImageNet accuracy with distribution shift
    - Important for real-world scenarios, stuff is naturally noisy
    - What is distribution shift? Training data stats are different than test data → low model performance

| Dataset Examples | ImageNet | Zero-Shot | CLIP | Δ Score |
|------------------|----------|----------|------|---------|
| ImageNet         | 76.2     | 76.2     | 0%   |         |
| ImageNetV2       | 64.3     | 70.1     | +5.8%|         |
| ImageNet-R       | 37.7     | 88.9     | +51.2%|         |
| ObjectNet        | 32.6     | 72.3     | +30.7%|         |
| ImageNet Sketch  | 25.2     | 60.2     | +35.0%|         |
| ImageNet-A Sketch| 2.7      | 77.1     | +74.4%|         |
Experiments and Results (Scaling)

• Scaling:
  • ViT’s scale well with compute + data
    • ResNets... not so much
  • Learned representations that are not just specific to one type of data
  • Largest CLIP model (ViT-L/14@336px) outperforms existing models by a significant margin
  • 2.6% average improvement
    • CLIP benefits from larger models but strong architectures too which better capture complex relationships
Strengths, Weaknesses, Relationships (including limitations)

**Pros:**
- Pre-trained robust zero-shot learning with encoder-backed supervision
- Adaptable to distribution shifts
- Scales well with compute
- Many tasks learned and classifying classes without explicit supervision

**Cons:**
- Not SOTA on all tasks Satellite Imaging, (EuroSAT, RESISC45) etc
- 1000x more compute to reach SOTA?!
- “Prompt engineering” effects, like adding "child" to categories list reduced misclassification of young people into incorrect categories from 32.3% to 8.7%
- Societal issues – surveillance/privacy
- Data Overlap (3.2% testing dataset avg)

**CLIP vs CoCa:**
- Unified transformer architecture vs two distinct encoders
  - Shared multimodal space and decent for similarity matching versus better cross-modal relationships
- CoCa can also go text → image
  - Transformer can process and generate information in both directions
  - Wider task range
- Nuanced understanding and generation capabilities (VQA, detailed scenes)
CoCa: Contrastive Captioners are Image-Text Foundation Models

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, Yonghui Wu

Published in Transactions on Machine Learning Research (08/2022)
Background

• Rise of foundational models in computer vision
  • Convnets/ transformers pretrained in large datasets (ImageNet, JFT).
  • Self supervision (BEiT, MAE).
  • Problem: constrained to vision.

• Vision-Language pretraining: fusion models
  • Early work Fast RCNN to extract visual representations
  • ViLT, VLMo multimodal transformers trained from scratch.

• Image-Text foundational models subsuming both image and video pretraining (3 paradigms)
  • Dual Encoder
  • Single Encoder
  • Encoder decoder
**Image-Text foundational model paradigms**

- **Dual Encoder paradigm:**
  - Examples: CLIP and ALIGN
  - What is it: uses contrastive loss on noisy image-text to learn strong image and text representations.
  - Advantages: crossmodal alignment tasks and zero-shot image classification.
  - Disadvantages: lacks generative capabilities (VQA).

- **Single Encoder Paradigm:**
  - Examples: convnets
  - What: train single encoder using large image dataset and cross entropy.
  - Advantages: generic visual representations good for image classification etc.
  - Disadvantages: lacks multimodal capabilities.

- **Encoder-Decoder paradigm:**
  - Examples: SIMVL
  - What: trained with generative losses, cross-attention.
  - Advantages: competitive VL results, encoder good for visual tasks.
  - Disadvantages: lack of joint representation, poorer performance contrastive tasks (i.e. image text matching).

---

Figure 1: Illustration of the SimVLM model. This shows an example of training with PrefixLM of an image-text pair. For text-only corpora, it is straightforward to remove the image patches and utilize textual tokens only.
Image-Text foundational model paradigms

- Current models either excel at contrastive learning or generative tasks, but rarely both.

- CoCa focuses on training an image-text foundation model from scratch in a single pretraining stage to unify these approaches.

- Similar: ALBEF requires multiple pretraining stages of unimodal and multimodal modules to attain good performance.

- Research Question:

  Can we create a model that unifies contrastive and generative learning into a single, efficient architecture that achieves state-of-the-art performance across a wide variety of vision-language tasks?
CoCa Architecture

1. Image encoder produces unimodal image representation.
2. Unimodal decoder produces unimodal text representation.
3. Contrastive loss between unimodal image and text representations.
4. Unimodal representations get fed into multimodal decoder (cross attention).
5. Captioning loss between predicted caption and actual caption (autoregressive).
• ViT (or Convnet) create latent representation.

• Use single image embedding (global context, empirical) for contrastive loss.

• Use all embeddings for contrasting loss (fine grained).

• Via attentional pooling layer, customizes output through a learnable query parameter determining output length.

From: https://www.datacamp.com/tutorial/coca-contrastive-captioners-are-image-text-foundation-models-visually-explained
• Append CLS token to capture semantic meaning.

• Ignore vector representation of other input tokens and use representation of classification token to estimate the contrastive loss.
Dual Encoder Contrastive Learning

\[ \mathcal{L}_{\text{Con}} = -\frac{1}{N} \left( \sum_{i}^{N} \log \frac{\exp(x_i^T y_i / \sigma)}{\sum_{j=1}^{N} \exp(x_i^T y_j / \sigma)} \right) + \sum_{i}^{N} \log \frac{\exp(y_i^T x_i / \sigma)}{\sum_{j=1}^{N} \exp(y_i^T x_j / \sigma)} \), \]

- \( x_i \) and \( y_j \) are normalized embeddings of the image in the \( i \)-th pair and that of the text in the \( j \)-th pair.
- \( N \): batch size
- \( \sigma \): temperature

- Maximize similarity between matching image pairs.
- Minimize similarity between non-matching pairs.
Multimodal Text decoder

- Uses fine grain image representation (256 image tokens) + unimodal text representation.
- Ignores CLS.
- Uses cross attention.
- Obtain unified image-text representation used to predict probability distribution of the vocabulary through autoregression.
What is the word $y_t$ that you predict having predicted the rest given the current image?

- Maximize the conditional likelihood of the paired text $y$ under the forward autoregressive factorization

$$\mathcal{L}_{\text{Cap}} = - \sum_{t=1}^{T} \log P_\theta(y_t | y_{<t}, x).$$
\[ \mathcal{L}_{CoCa} = \lambda_{Con} \cdot \mathcal{L}_{Con} + \lambda_{Cap} \cdot \mathcal{L}_{Cap} \]
Experiments and Results

• Data:
  • JFT-3B
  • ALIGN

• Optimization:
  • GSPMD
  • \(N=65,536\)
  • 500k steps (5 epoch JFT, 10 ALIGN).
  • \(\lambda_{\text{Cap}} = 2.0\) and \(\lambda_{\text{Con}} = 1.0\).

• Results
  • Visual Recognition
  • Crossmodal alignment
  • Image Captioning and Multimodal Understanding Tasks

Zero-shot, frozen-feature or finetuning
Figure 4: Comparison of CoCa with other image-text foundation models (without task-specific customization) and multiple state-of-the-art task-specialized models.
Results: Visual Recognition

| Model               | ImageNet |
|---------------------|----------|
| ALIGN\(^a\)         | 88.6     |
| Florence\(^b\)      | 90.1     |
| MetaPseudoLabels\(^c\) | 90.2     |
| CoAtNet\(^d\)       | 90.9     |
| ViT-G\(^e\)         | 90.5     |
| + Model Soups\(^f\) | 90.9     |
| CoCa (frozen)       | 90.6     |
| CoCa (finetuned)    | 91.0     |

Table 2: Image classification and video act reference: \(^a\)(Jia et al., 2021) \(^b\)(Yuan et al., 20\(g\))(Wortsman et al., 2022) \(^g\)(Arnab et al., 2021) \(^1\)(Zhang et al., 2021).

Is visual encoder finetuning that relevant?

(a) Finetuned ImageNet Top-1 Accuracy.
Results: cross-modal alignments

- Use contrastive loss embeddings, discard multimodal decoder.
- **Zero-Shot Image retrieval:**

![Image](image.png)

**Figure 5:** Image classification scaling performance of model sizes.

| Model                  | Flickr30K (1K test set) | MSCOCO (5K test set) |
|------------------------|--------------------------|-----------------------|
|                        | Image → Text | Text → Image | Image → Text | Text → Image |
| CLIP (Radford et al., 2021) | R@1 | R@5 | R@10 | R@1 | R@5 | R@10 | R@1 | R@5 | R@10 |
| ALIGN (Jia et al., 2021)   | 88.0 | 98.7 | 99.4 | 68.7 | 90.6 | 95.2 | 58.4 | 81.5 | 88.1 |
| FLAVA (Singh et al., 2021) | 88.6 | 98.7 | 99.7 | 75.7 | 93.8 | 96.8 | 58.6 | 83.0 | 89.7 |
| FILIP (Yao et al., 2021)   | 67.7 | 94.0 | -    | 65.2 | 89.4 | -    | 42.7 | 76.8 | -    |
| Florence (Yuan et al., 2021) | 89.8 | 99.2 | 99.8 | 75.0 | 93.4 | 96.3 | 61.3 | 84.3 | 90.4 |
| CoCa-Base               | 90.9 | 99.1 | -    | 76.7 | 93.6 | -    | 64.7 | 85.9 | -    |
| CoCa-Large              | 89.8 | 98.8 | 99.8 | 76.8 | 93.7 | 96.8 | 63.8 | 84.7 | 90.7 |
| CoCa                   | 91.4 | 99.2 | 99.9 | 79.0 | 95.1 | 97.4 | 65.4 | 85.6 | 91.4 |
|                         | 92.5 | 99.5 | 99.9 | 80.4 | 95.7 | 97.7 | 66.3 | 86.2 | 91.8 |

Table 3: Zero-shot image-text retrieval results on Flickr30K (Plummer et al., 2015) and MSCOCO (Chen et al., 2015) datasets.
Results: cross-modal alignments

- Use contrastive loss embeddings, discard multimodal decoder.
- **Zero-Shot Image classification:**

| Model                  | ImageNet | ImageNet-A | Image |
|------------------------|----------|------------|-------|
| CLIP (Radford et al., 2021) | 76.2     | 77.2       | 85    |
| ALIGN (Jia et al., 2021)    | 76.4     | 75.8       | 92    |
| FILIP (Yao et al., 2021)    | 78.3     | -          | -     |
| Florence (Yuan et al., 2021) | 83.7     | -          | -     |
| LiT (Zhai et al., 2021b)    | 84.5     | 79.4       | 93    |
| BASIC (Pham et al., 2021a)  | 85.7     | 85.6       | 95    |
| CoCa-Base                | 82.6     | 76.4       | 93    |
| CoCa-Large               | 84.8     | 85.7       | 95    |
| CoCa                     | **86.3** | **90.2**   | **96**|

Table 4: Zero-shot image classification results on ImageNet (2021b), ImageNet-R (Hendrycks et al., 2021a), ImageNet-A (Hendrycks et al., 2019) and ObjectNet (Barbu et al., 2019).
Results: Multimodal Understanding Tasks

- Multimodal Understanding: Visual question answering, visual reasoning and visual entailment. Train classifier on top of decoder.

| Model                | VQA test-dev | VQA test-std | SNLI-VE dev | SNLI-VE test | NLVR2 dev | NLVR2 test-p |
|----------------------|--------------|--------------|-------------|--------------|-----------|--------------|
| UNITER (Chen et al., 2020) | 73.8         | 74.0         | 79.4        | 79.4         | 79.1      | 80.0         |
| VinVL (Zhang et al., 2021b) | 76.6         | 76.6         | -           | -            | 82.7      | 84.0         |
| CLIP-VIL (Shen et al., 2021) | 76.5         | 76.7         | 80.6        | 80.2         | -         | -            |
| ALBEF (Li et al., 2021)   | 75.8         | 76.0         | 80.8        | 80.9         | 82.6      | 83.1         |
| BLIP (Li et al., 2022)    | 78.3         | 78.3         | -           | -            | 82.2      | 82.2         |
| OFA (Wang et al., 2022)   | 79.9         | 80.0         | 90.3†       | 90.2†        | -         | -            |
| VLMo (Wang et al., 2021a) | 79.9         | 80.0         | -           | -            | 85.6      | 86.9         |
| SimVLM (Wang et al., 2021b) | 80.0        | 80.3         | 86.2        | 86.3         | 84.5      | 85.2         |
| Florence (Yuan et al., 2021) | 80.2        | 80.4         | -           | -            | -         | -            |
| METER (Dou et al., 2021)  | 80.3         | 80.5         | -           | -            | -         | -            |
| CoCa                   | **82.3**     | **82.3**     | **87.0**    | **87.1**     | **86.1**  | **87.0**     |

Table 5: Multimodal understanding results comparing vision-language pretraining methods. †OFA uses both image and text premises as inputs while other models utilize the image only.
Results: Image Captioning

- No further adaptation

|               | MSCOCO          | NoCaps          |
|---------------|-----------------|-----------------|
|               | B@4  | M   | C    | S    | Valid C | S | Test C | S |
| CLIP-ViL (Shen et al., 2021) | 40.2  | 29.7 | 134.2 | 23.8 | -       | - | -      | - |
| BLIP (Li et al., 2022)        | 40.4  | -   | 136.7 | -    | 113.2   | 14.8 | -      | - |
| VinVL (Zhang et al., 2021b)   | 41.0  | 31.1 | 140.9 | 25.4 | 105.1   | 14.4 | 103.7  | 14.4 |
| SimVLM (Wang et al., 2021b)   | 40.6  | 33.7 | 143.3 | 25.4 | 112.2   | -   | 110.3  | 14.5 |
| LEMON (Hu et al., 2021)       | **41.5** | 30.8 | 139.1 | 24.1 | 117.3   | 15.0 | 114.3  | 14.9 |
| LEMON_{SCST} (Hu et al., 2021)† | 42.6  | 31.4 | 145.5 | 25.5 | -       | -   | -      | - |
| OFA (Wang et al., 2022)†      | 43.5  | 31.9 | 149.6 | 26.1 | -       | -   | -      | - |
| CoCa                       | 40.9  | **33.9** | **143.6** | 24.7 | **122.4** | **15.5** | **120.6** | **15.5** |

Table 6: Image captioning results on MSCOCO and NoCaps (B@4: BLEU@4, M: METEOR, C: CIDEr, S: SPICE). †Models finetuned with CIDEr optimization.
Ablation Analysis

| loss  | LE  | FT |
|-------|-----|----|
| $\mathcal{L}_{\text{Cls}}$ | 81.0 | 85.1 |
| $\mathcal{L}_{\text{Cap}}$ | 82.1 | 84.9 |

(a) Encoder-decoder vs. single-encoder models (trained on JFT).

| loss  | ZS  | VQA | TPU cost |
|-------|-----|-----|----------|
| $\mathcal{L}_{\text{Con}}$ | 70.7 | 59.2 | 1x |
| $\mathcal{L}_{\text{Cap}}$ | - | 68.9 | 1.17x |
| $\mathcal{L}_{\text{CoCa}}$ | **71.6** | **69.0** | **1.18x** |

(b) Training objectives ablation.

| $\lambda_{\text{Cap}} : \lambda_{\text{Con}}$ | ZS  | VQA |
|---------------------------------|-----|-----|
| 1:1                             | 71.5 | 68.6 |
| 1:2                             | 71.0 | 68.1 |
| **2:1**                         | **71.6** | **69.0** |

(c) Training objectives weights.

| $n_{\text{uni}}$ | ZS  | VQA |
|------------------|-----|-----|
| 3                | 70.2 | 69.0 |
| **6**            | **71.6** | **69.0** |
| 9                | 71.4 | 68.8 |

(d) Unimodal decoder layers.

| variant | AE  | MSCOCO |
|---------|-----|--------|
| 1 [CLS] | **80.7** | **41.4** |
| + text tokens | 80.3 | 40.2 |
| 8 [CLS] | 80.3 | 36.9 |
| + text tokens | 80.4 | 40.3 |

(e) Contrastive text embedding design ablation.

| variant | ZS  | VQA |
|---------|-----|-----|
| parallel | 71.2 | 68.7 |
| **cascade** | **71.6** | **69.0** |
| $n_{\text{query}} = 0$ | 71.5 | 69.0 |
| $n_{\text{query}} = 1$ | 69.3 | 64.4 |
| $n_{\text{query}} = 32$ | 71.2 | 68.2 |

(f) Attentional pooler design ablation.

Table 7: CoCa ablation experiments. On ImageNet classification, we report top-1 accuracy for: zero-shot (ZS), linear evaluation (LE), attentional evaluation (AE) using pooler on frozen feature, and finetuning (FT). On MSCOCO retrieval, we report the average of image-to-text and text-to-image R@1. On VQA, we report the dev-set vqa score. The default CoCa setting is bold.
### Summary of Strengths, Weaknesses, Relationships

| Strengths                                                                 | Weaknesses                                                                                                                                   |
|---------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| • Strong at both generative and contrastive tasks.                        | • Poor scalability?                                                                                                                           |
| • Single pretraining stage.                                               | • it can have trouble with zero-shot captioning because the noisy web text it was trained on isn’t as rich as unimodal text data.         |
| • Single backprop.                                                        | • Truly foundational?                                                                                                                         |
|                                                                           | • ...?                                                                                                                                       |
Limitations

- Why not also add a multimodal image decoder?
Conclusion

• CoCa successfully leverages 3 paradigms to create a single efficient model.

• CoCa is able to excel at both generative and contrastive tasks.
BEiT (Bidirectional Encoder representation from Image Transformers) is a self-supervised learning framework designed for vision tasks, inspired by the success of BERT in natural language processing. BEiT pretrains Vision Transformers (ViTs) by treating images as sequences of visual tokens, where each token represents a patch of the image. The model learns to predict masked visual tokens in an image, similar to how BERT predicts masked words in a sentence, enabling it to capture rich visual features.

**Figure 1:** Overview of BEiT pre-training. Before pre-training, an image is tokenized to the learned vocabulary. During pre-training, each image is modeled as a sequence of visual tokens. We randomly mask some proportion of image patches and replace them with a special mask embedding. Then the Transformer tries to predict the visual tokens of the original image based on the encoding vectors of the corrupted image.

| Models                  | Model Size | Resolution | ImageNet |
|-------------------------|------------|------------|----------|
| VITb34-B [DBK+20]       | 86M        | 384^2      | 77.9     |
| VITb34-L [DBK+20]       | 307M       | 384^2      | 76.5     |
| DeciT-B [TCD+20]        | 86M        | 224^2      | 81.8     |
| DeciT384-B [TCD+20]    | 86M        | 384^2      | 83.1     |

| Supervised Pre-Training on ImageNet-22K (using labeled data) |
|---------------------------------------------------------------|
| VITb34-B [DBK+20]                                          | 86M  | 384^2 | 84.0 |
| VITb34-L [DBK+20]                                          | 307M | 384^2 | 85.2 |

| Self-Supervised Pre-Training on ImageNet-1K (without labeled data) |
|---------------------------------------------------------------------|
| iGPT-1.36B [CRC+20]                                                | 1.36B | 224^2 | 66.5 |
| VITb34-B-JFT300M [DBK+20]                                         | 86M  | 384^2 | 79.9 |
| MoCo v3-B [CXLH21]                                                | 86M  | 224^2 | 83.2 |
| MoCo v3-L [CXLH21]                                                | 307M | 224^2 | 84.1 |
| DINO-B [CTM+21]                                                  | 86M  | 224^2 | 82.8 |
| BEiT-B (ours)                                                    | 86M  | 384^2 | 83.2 |
| BEiT-L (ours)                                                   | 86M  | 384^2 | 84.6 |
| BEiT384-B (ours)                                               | 307M | 384^2 | 85.2 |
| BEiT384-L (ours)                                              | 307M | 384^2 | 86.3 |

Table 1: Top-1 accuracy on ImageNet-1K. We evaluate base- ("B") and large-size ("L") models at resolutions 224 × 224 and 384 × 384. ¹: iGPT-1.36B contains 1.36 billion parameters, while others are base-size models. ²: VITb34-B-JFT300M is pretrained with the “masked patch prediction” task on Google’s in-house 300M images, while others use ImageNet.
ViLT (Vision-and-Language Transformer) is a model that integrates visual and textual information using a minimalist approach by directly processing image patches and text tokens through a shared transformer architecture. Unlike other models that rely on separate, complex visual backbones (e.g., CNNs), ViLT uses a simplified design where images are divided into patches and jointly processed with text in a single transformer, enabling efficient and effective vision-language tasks such as image-text retrieval, visual question answering, and image captioning. This streamlined architecture reduces computational overhead while maintaining competitive performance on various multimodal benchmarks.
VLMo (Vision-Language Model with Mixture-of-Modality-Experts) is a unified model that combines vision and language pre-training by using a mixture-of-experts approach. The model integrates modality-specific experts (for both vision and language) within a single transformer architecture, allowing it to dynamically select the most relevant expert for processing different types of input. This flexible design enables VLMo to efficiently handle both unimodal and multimodal tasks, such as image-text retrieval, visual question answering, and image captioning, by leveraging the strengths of specialized experts while maintaining a unified framework. The mixture-of-experts mechanism enhances the model's ability to learn from and generalize across diverse vision-language tasks.
Cross Attention

• Cross-attention is a mechanism used in transformer-based models to integrate information from different modalities (e.g., images and text) or different sources of data. In the context of vision-language models like SimVLM, cross-attention is crucial for combining visual and textual information in a meaningful way.

• How Cross-Attention Works:
  - Attention Mechanism: In a transformer, the attention mechanism allows the model to weigh different parts of the input sequence differently when making predictions. This is done using three components: queries (Q), keys (K), and values (V).
  - Cross-Attention Setup: In a cross-attention layer, the queries, keys, and values come from different sources:
    - Queries (Q): Typically come from one modality (e.g., text).
    - Keys (K) and Values (V): Come from another modality (e.g., image).
  - The attention mechanism computes how much each part of the text (queries) should focus on different parts of the image (keys/values) when making predictions or generating output.