DALL-E

Authors: Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever

Open AI (ICML 2021)

Presenters: Adam Kutchak, George Lu, Fernando Treviño, and Sarah Wilson

(CAP6412, Spring 2022)

https://www.youtube.com/watch?v=ArPTcWpVCZw
Introduction

- Generate Images from text captions
- 12 billion parameters version of GPT-3
- Dataset comprised of 3.3 million text - image pairs
- Combine unrelated concepts
(a) a tapir made of accordion. a tapir with the texture of an accordion.
(b) an illustration of a baby hedgehog in a christmas sweater walking a dog
(c) a neon sign that reads “backprop”. a neon sign that reads “backprop”. backprop
(d) the exact same cat on the top as a sketch on the bottom
Related Works

- Autoencoder - (encoder - decoder)
- Variational Autoencoders (continuous state space)
- VQ-VAE (discrete quantized state space)
Related Work - Autoencoder

Input-Image $X$ → Encoder $G_\theta$ → Latent-Vector $Z$ → Decoder $F_\phi$ → Predicted-Image $\hat{X}$

$P(Z|X)$ → $P(\hat{X}|Z)$
Related Work - Autoencoder problem
Related Work - Variational Autoencoder

\[ z = \mu + \sigma \odot \epsilon \]
Related Work - Autoencoder vs. VAE
Related Work - Variational Autoencoder problem

Latent space distribution after training

Latent space is regularized. Vectors sampled from latent space can generate valid data.

Vectors sampled from overlapping distribution generates morphed data.
Related Work - VAE problem

Latent space distribution after training

Latent space is regularized. Vectors sampled from latent space can generate valid data.

Vectors sampled from overlapping distribution generates morphed data.
Related Work - VAE problem

Latent space distribution after training

Latent space is regularized. Vectors sampled from latent space can generate valid data.

Vectors sampled from overlapping distribution generates morphed data.
Related Work - VQ-VAE
DALL-E Model

- Transformer to model text and image tokens as single stream of data
  - 2 stage training!
Stage One: Learning the Visual Codebook

- Discrete Variational Autoencoder (dVAE)
  - Similar to VQ-VAE (in VQ-GAN) but uses distribution instead of nearest neighbor
Stage One: Learning the Visual Codebook

- Discrete Variational Autoencoder (dVAE) encoder
Stage One: Learning the Visual Codebook

- Discrete Variational Autoencoder (dVAE) decoder
Stage Two: Learning Prior Distribution

- Transformer
  - Predict distribution for next token
  - Sample distribution and repeat until 1024 image tokens
Stage Two: Transformer Characteristics

- Transformer
  - BPE-encode lowercase captions into 256 text tokens
    - Vocab size of 16,384
  - 32x32 image tokens
    - Vocab size of 8192
  - 64 attention layers
    - 62 attention heads
  - 12 Billion parameters
Training: Dataset

- Training Dataset
  - Wikipedia images
  - YFCC100M++

- Filter removed:
  - Small Captions
  - Non-English
  - Dates
  - Extreme Aspect Ratios
Testing Datasets

- **MS-COCO**
  - 328k images
  - object detection, segmentation, key-point detection, captioning

- **CUB-200**
  - 200 bird species
  - 11,788 images
Example Results

- A very cute cat laying by a big bike.
- A China Airlines plane on the ground at an airport with baggage cars nearby.
- A table that has a train model on it with other cars and things.
Sample Generation

- CLIP
  - Pre-trained contrastive model
  - Ranks DALLE’s generated images
  - Input = image + caption
  - Output = score
  - More images to rank = better quality of best
Learning Transferable Visual Models from Natural Language Supervision

Alec Radford  JongWook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, Iya Sutskever

ICML-2021; 3,131 Citations

Presented by: Moazam Soomro, Fatemah Najafali, Alec Kerrigan, and Connor Malley

https://www.youtube.com/watch?v=t5MPdf8NG1g
Contrastive Language Image Pre-training (CLIP)

- Mechanism for natural language supervision
- Pair an image with its caption using contrastive learning
- Beats fully supervised learning baseline on many datasets
- Can be used as a zero-shot classifier
Contrastive Language Image Pre-training (CLIP)
Zero-shot CLIP is much more robust

<table>
<thead>
<tr>
<th>DATASET</th>
<th>IMAGENET RESNET101</th>
<th>CLIP VIT-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageNet</td>
<td>76.2%</td>
<td>76.2%</td>
</tr>
<tr>
<td>ImageNet V2</td>
<td>64.3%</td>
<td>70.1%</td>
</tr>
<tr>
<td>ImageNet Rendition</td>
<td>37.7%</td>
<td>88.9%</td>
</tr>
<tr>
<td>ObjectNet</td>
<td>32.6%</td>
<td>72.3%</td>
</tr>
<tr>
<td>ImageNet Sketch</td>
<td>25.2%</td>
<td>60.2%</td>
</tr>
<tr>
<td>ImageNet Adversarial</td>
<td>2.7%</td>
<td>77.1%</td>
</tr>
</tbody>
</table>
Motivation

- Image classification models are limited:
  - Fixed number of labels
  - Generalization

- CLIP overcomes these limitations.
What is Contrastive Learning?

- **Classification task:**

- **Contrastive Learning:**

  $N$ negative samples
What is Zero-Shot Learning?

- To train on one dataset and generalizing on unseen categories.
WebImageText Dataset

- Motivation for using natural language is the vast amounts of data
- Previous datasets did not have enough natural language descriptions (YFCC100M)
- Authors searched for (image, text) pairs which contained one of 500,000 text queries
- Used for pre-training CLIP

WebImageText (WIT)

400M (image, text) pairs
Up to 20,000 pairs per query
Contrastive Learning Objective - similar (image, text) pair

Input Image

A dog lying in grass

Input Text

maximize\left(\frac{\vec{H}_i \cdot \vec{H}_t}{\|\vec{H}_i\| \times \|\vec{H}_t\|}\right)

Image Representation

Text Representation
Contrastive Learning Objective - dissimilar (image, text) pair

Image Representation

Text Representation

minimize\left(\frac{\vec{H}_i \cdot \vec{H}_t}{||\vec{H}_i|| \times ||\vec{H}_t||}\right)

Input Image

A dog lying in grass

Input Text
* Authors also tested many other ResNet/ViT variants, but found this ViT to perform the best
CLIP Pre-training

(1) Contrastive pre-training

Pepper the aussie pup
CLIP Pre-training

1. Encode batch of text samples
2. Encode batch of image samples
3. Maximize cosine similarity between correct matches
4. Minimize cosine similarity between incorrect matches
Computing Loss

\[ m_i = \text{one-hot encoded label vector for the } i\text{-th image sample} \]
\[ y_i^m = \text{cosine similarities vector for } i\text{-th image sample} \]
\[ t_i = \text{one-hot encoded label for the } i\text{-th text sample} \]
\[ y_i^t = \text{cosine similarities vector for } i\text{-th text sample} \]
\[ \phi = \text{cross entropy loss} \]
Computing Loss

\[ m_i = \text{one-hot encoded label vector for the i-th image sample} \]
\[ y_i^m = \text{cosine similarities vector for i-th image sample} \]
\[ t_i = \text{one-hot encoded label for the i-th text sample} \]
\[ y_i^t = \text{cosine similarities vector for i-th text sample} \]
\[ \phi = \text{cross entropy loss} \]

\[ L_m = \frac{\sum_{i=1}^{N} \phi(y_i^m, m_i)}{N} \]
\[ L_t = \frac{\sum_{i=1}^{N} \phi(y_i^t, t_i)}{N} \]

\[ \mathcal{L} = \frac{L_m + L_t}{2} \]
Some CLIP details

Training
- Trained on 400M image-text pairs from the internet
- Batch size of 32,768
- 32 epochs over the dataset
- Cosine learning rate decay

Architecture
- ResNet-based or ViT-based image encoder
- Transformer-based text encoder
Testing

- Linear Probe

- Zero-shot Prediction
Linear Probe CLIP

- Train a linear classifier on another dataset using CLIP features
Kornblith et al.'s 12 datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Classes</th>
<th>Train size</th>
<th>Test size</th>
<th>Evaluation metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>Food-101</td>
<td>102</td>
<td>75,750</td>
<td>25,250</td>
<td>accuracy</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>10</td>
<td>50,000</td>
<td>10,000</td>
<td>accuracy</td>
</tr>
<tr>
<td>CIFAR-100</td>
<td>100</td>
<td>50,000</td>
<td>10,000</td>
<td>accuracy</td>
</tr>
<tr>
<td>Birdsnap</td>
<td>500</td>
<td>42,283</td>
<td>2,149</td>
<td>accuracy</td>
</tr>
<tr>
<td>SUN397</td>
<td>397</td>
<td>19,850</td>
<td>19,850</td>
<td>accuracy</td>
</tr>
<tr>
<td>Stanford Cars</td>
<td>196</td>
<td>8,144</td>
<td>8,041</td>
<td>accuracy</td>
</tr>
<tr>
<td>FGVC Aircraft</td>
<td>100</td>
<td>6,667</td>
<td>3,333</td>
<td>mean per class</td>
</tr>
<tr>
<td>Pascal VOC 2007 Classification</td>
<td>20</td>
<td>5,011</td>
<td>4,952</td>
<td>11-point mAP</td>
</tr>
<tr>
<td>Describable Textures</td>
<td>47</td>
<td>3,760</td>
<td>1,880</td>
<td>accuracy</td>
</tr>
<tr>
<td>Oxford-IIIT Pets</td>
<td>37</td>
<td>3,680</td>
<td>3,669</td>
<td>mean per class</td>
</tr>
<tr>
<td>Caltech-101</td>
<td>102</td>
<td>3,060</td>
<td>6,085</td>
<td>mean-per-class</td>
</tr>
<tr>
<td>Oxford Flowers 102</td>
<td>102</td>
<td>2,040</td>
<td>6,149</td>
<td>mean per class</td>
</tr>
</tbody>
</table>
## Extended 27 Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Classes</th>
<th>Train size</th>
<th>Test size</th>
<th>Evaluation metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>Food-101</td>
<td>102</td>
<td>75,750</td>
<td>25,250</td>
<td>accuracy</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>10</td>
<td>50,000</td>
<td>10,000</td>
<td>accuracy</td>
</tr>
<tr>
<td>CIFAR-100</td>
<td>100</td>
<td>50,000</td>
<td>10,000</td>
<td>accuracy</td>
</tr>
<tr>
<td>Birdsnap</td>
<td>500</td>
<td>42,283</td>
<td>2,149</td>
<td>accuracy</td>
</tr>
<tr>
<td>SUN397</td>
<td>397</td>
<td>19,850</td>
<td>19,850</td>
<td>accuracy</td>
</tr>
<tr>
<td>Stanford Cars</td>
<td>196</td>
<td>8,144</td>
<td>8,041</td>
<td>accuracy</td>
</tr>
<tr>
<td>FGVC Aircraft</td>
<td>100</td>
<td>6,667</td>
<td>3,333</td>
<td>mean per class</td>
</tr>
<tr>
<td>Pascal VOC 2007 Classification</td>
<td>20</td>
<td>5,011</td>
<td>4,952</td>
<td>11-point mAP</td>
</tr>
<tr>
<td>Describable Textures</td>
<td>47</td>
<td>3,760</td>
<td>1,880</td>
<td>accuracy</td>
</tr>
<tr>
<td>Oxford-IIIT Pets</td>
<td>37</td>
<td>3,680</td>
<td>3,669</td>
<td>mean per class</td>
</tr>
<tr>
<td>Caltech-101</td>
<td>102</td>
<td>3,060</td>
<td>6,085</td>
<td>mean-per-class</td>
</tr>
<tr>
<td>Oxford Flowers 102</td>
<td>102</td>
<td>2,040</td>
<td>6,149</td>
<td>mean per class</td>
</tr>
<tr>
<td>MNIST</td>
<td>10</td>
<td>60,000</td>
<td>10,000</td>
<td>accuracy</td>
</tr>
<tr>
<td>Facial Emotion Recognition 2013</td>
<td>8</td>
<td>32,140</td>
<td>3,574</td>
<td>accuracy</td>
</tr>
<tr>
<td>STL-10</td>
<td>10</td>
<td>10000</td>
<td>8000</td>
<td>accuracy</td>
</tr>
<tr>
<td>EuroSAT</td>
<td>10</td>
<td>10,000</td>
<td>5,000</td>
<td>accuracy</td>
</tr>
<tr>
<td>RESISC45</td>
<td>45</td>
<td>3,150</td>
<td>25,200</td>
<td>accuracy</td>
</tr>
<tr>
<td>GTSRB</td>
<td>43</td>
<td>26,640</td>
<td>12,630</td>
<td>accuracy</td>
</tr>
<tr>
<td>KITTI</td>
<td>4</td>
<td>6,770</td>
<td>711</td>
<td>accuracy</td>
</tr>
<tr>
<td>Country211</td>
<td>211</td>
<td>43,200</td>
<td>21,100</td>
<td>accuracy</td>
</tr>
<tr>
<td>PatchCamelyon</td>
<td>2</td>
<td>294,912</td>
<td>32,768</td>
<td>accuracy</td>
</tr>
<tr>
<td>UCF101</td>
<td>101</td>
<td>9,537</td>
<td>1,794</td>
<td>accuracy</td>
</tr>
<tr>
<td>Kinetics700</td>
<td>700</td>
<td>494,801</td>
<td>31,669</td>
<td>mean(top1, top5)</td>
</tr>
<tr>
<td>CLEVR Counts</td>
<td>8</td>
<td>2,000</td>
<td>500</td>
<td>accuracy</td>
</tr>
<tr>
<td>Hateful Memes</td>
<td>2</td>
<td>8,500</td>
<td>500</td>
<td>ROC AUC</td>
</tr>
<tr>
<td>Rendered SST2</td>
<td>2</td>
<td>7,792</td>
<td>1,821</td>
<td>accuracy</td>
</tr>
<tr>
<td>ImageNet</td>
<td>1000</td>
<td>1,281,167</td>
<td>50,000</td>
<td>accuracy</td>
</tr>
</tbody>
</table>
Results - Efficiency - Kornblith

- Kornblith 12 dataset evaluation suite, standard for most works
- CLIP’s ResNet based model underperforms EfficientNet
- ViT based CLIP outperforms everything
Results - Efficiency - Extended

- On the extended testing suite, both CLIP versions outperform all other models.
- Performance gap increases with GFLOPS.
Results - Low-Shot

- CLIP scales well
- Linear-Probe CLIP climbs
- ResNet and other methods flatten
- Zero-Shot CLIP outperforms all non-CLIP methods up until 16 shot
Contrastive Language Image Pre-training (CLIP)
Results - Accuracy

- Zero-shot CLIP using ResNet50 backbone is compared to off the shelf ResNet50
- CLIP outperforms on a wide variety of popular datasets
- For video, a single frame was sampled
Results - Accuracy

- Underperforms on many other datasets
- Mostly on specialized/complex datasets
- EuroSAT for satellite images, Tumor classification
- Makes intuitive sense, Zero-shot CLIP is highly generalized
- Not suited for hyper specific tasks unless fine-tuned
Thank You