CLIP

Connecting
Text & Images

Nirvan Patil | BITS Pilani, Goa Campus

YouTube Presentation

CLIP

1. Introduction & Motivation
2. Architecture & Working
3. Zero-Shot Transfer
4. Miscellaneous Results
5. Limitations

Nirvan Patil | BITS Pilani, Goa Campus

Introduction & Motivation

Motivation

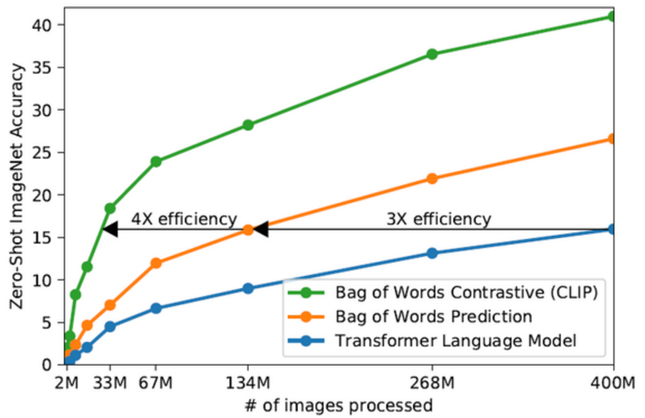
- ***Leveraging Web-Scale Natural Language Supervision***
  - Modern NLP pre-training on vast, web-scale text (e.g. GPT-3) outperforms high-quality, crowd-labelled datasets in many tasks.
  - Computer Vision still relies on pre-training with crowd-labelled image datasets such as ImageNet.
  - This raises the question: *Can CV similarly benefit from directly learning from raw web text paired with images?*

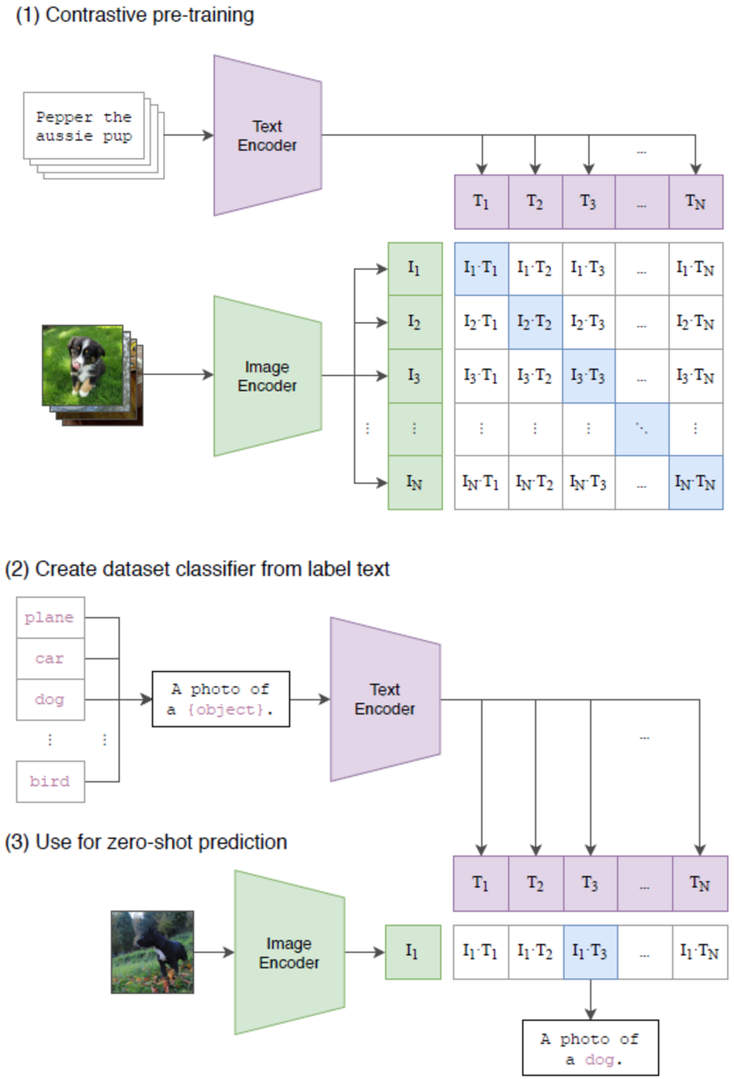
- ***Prior Work in Natural Language Supervision for Vision***
  - CNNs to predict *Bag-of-Words* from image captions
  - ConVIRT worked on contrastive objectives.
  - *Instagram hashtags*-based pre-training
  - This line of work represents the current pragmatic middle ground between learning from a limited amount of supervised

- ***Poor Zero Shot Transfer***
  - Static softmax outputs curtail flexibility and limit "zero-shot" capabilities.

CLIP!

- A simplified version of ConVIRT trained from scratch, which we call CLIP, for Contrastive Language-Image Pre-training
- CLIP closes the gap, *studying the behaviours of image classifiers trained with natural language supervision at a large scale*
- CLIP, similar to the GPT family, learns to perform a wide set of tasks during pre-training, including OCR, geo-localisation, etc.
- Benchmarking, comparison with human performance and zero-shot transfer capabilities





Approach CLIP

Motivation

At the core approach: learning perception from supervision contained in natural language.

- ***Why Natural Language Supervision?***
  - Large quantities of data of this form are available publicly on the internet.
  - *A single label "compresses" the meaning of the image to a single word*. Natural language label captures a wider spread of context in the image.

Dataset!

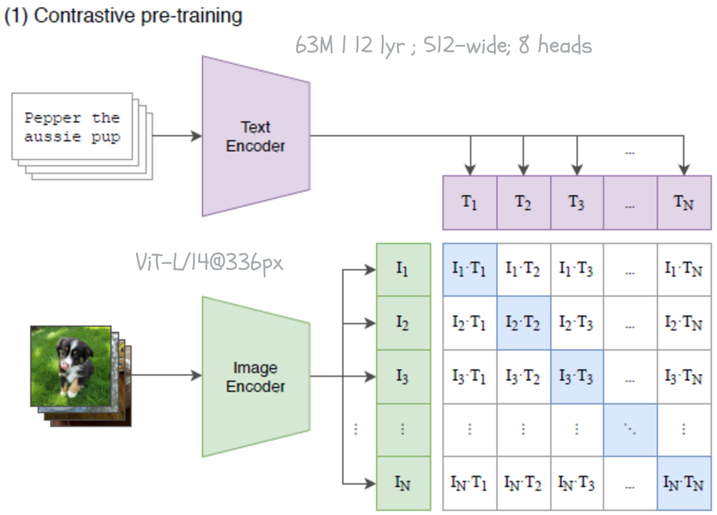
- *Existing datasets are not adequately large*
- The authors created a new dataset of 400 million (image, text) pairs collected from a variety of publicly available sources on the Internet.

Pre-Training

- Existing methods try to *predict the exact words* of the text accompanying each image. This is a *difficult task* due to the wide variety of descriptions.
- Contrastive objectives can learn better representations than their equivalent predictive objective.

Pre-Training

A training system to solve the easier proxy task of predicting only which text as a whole is paired with which image, and not the exact words of that text.



Maximise the cosine similarity of the image and text embeddings of the N real pairs in the batch while minimising the cosine similarity of the embeddings of the N^2-N incorrect pairings.

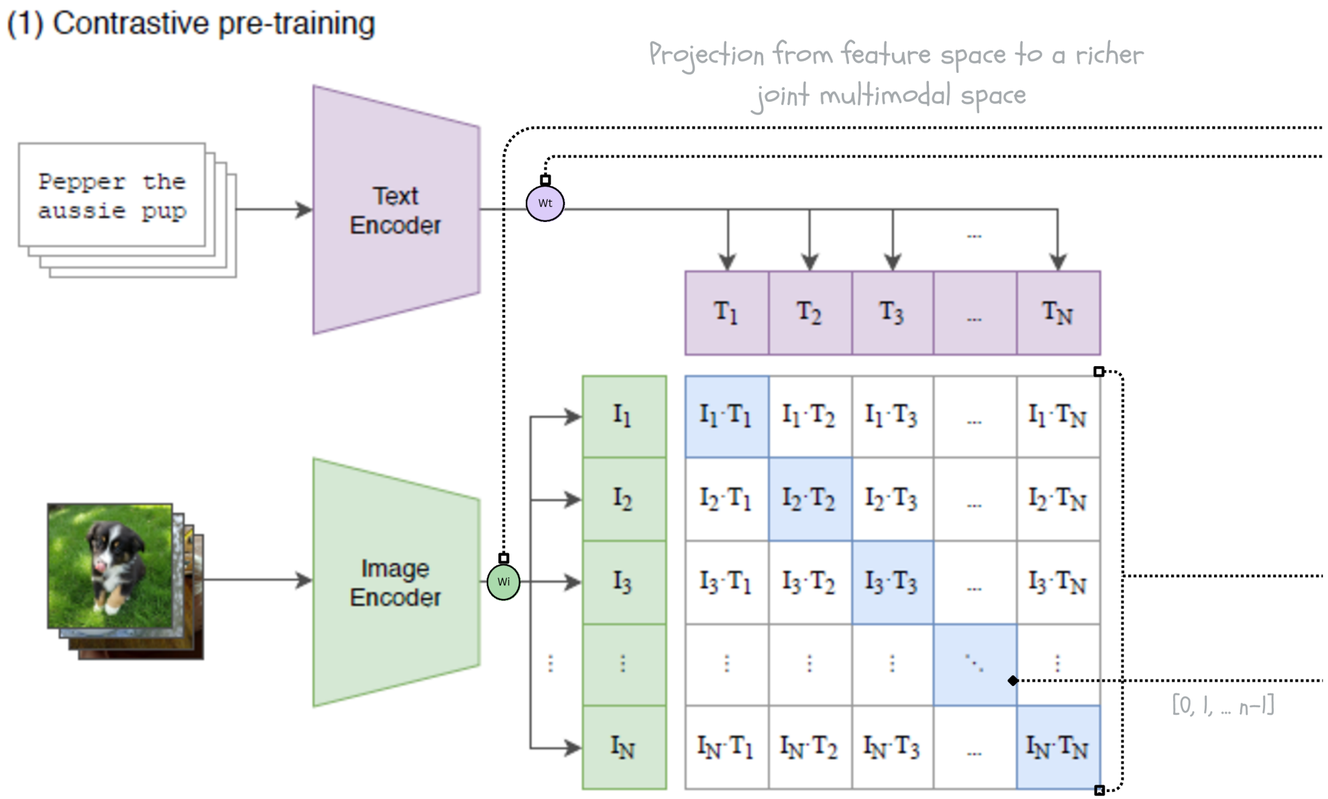
Psuedo-Code

```
# image_encoder - ResNet or Vision Transformer
# text_encoder  - CBOW or Text Transformer
# I[n, h, w, c] - minibatch of aligned images
# T[n, l]       - minibatch of aligned texts
# W_i[d_i, d_e] - learned proj of image to embed
# W_t[d_t, d_e] - learned proj of text to embed
# t             - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T)  #[n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss   = (loss_i + loss_t)/2
```

No noticeable difference between linear and non-linear projections

Working CLIP

Pre-Training



Along each row (Image → Text) : Treat each row as a prob. dist.
For first row ground truth is index 0, for second it is 1 and so on i.e. np.arange(n)

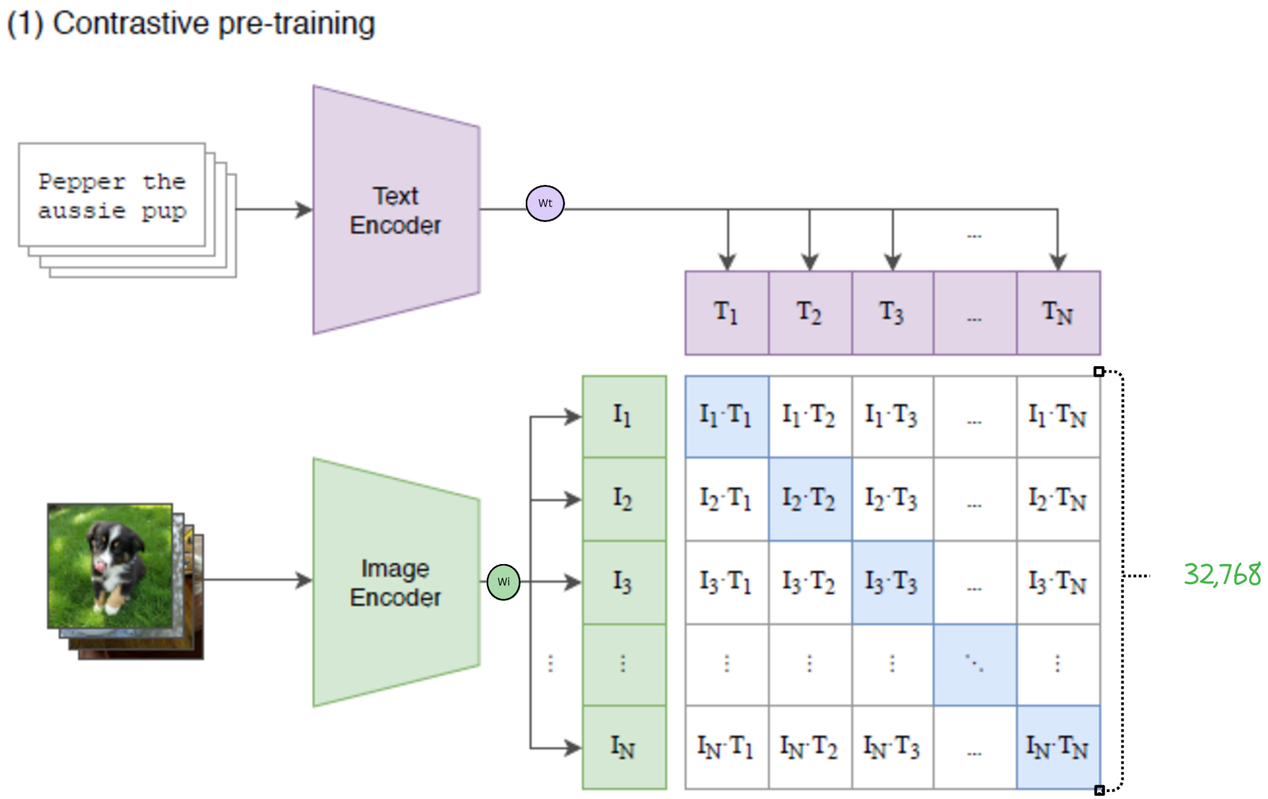
Psuedo-Code

```
# image_encoder - ResNet or Vision Transformer
# text_encoder  - CBOW or Text Transformer
# I[n, h, w, c] - minibatch of aligned images
# T[n, l]       - minibatch of aligned texts
# W_i[d_i, d_e] - learned proj of image to embed
# W_t[d_t, d_e] - learned proj of text to embed
# t             - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T)  #[n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)                                    T → I i.e. along each col.
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss   = (loss_i + loss_t)/2                             I → T i.e. along each row
```

Symmetric loss i.e. equal importance to Image → Text and Text → Image
"Two way" Classification

Working CLIP

Context Size



Key Concept

Every step of CLIP pre-training can be viewed as optimising the performance of a randomly created proxy to a computer vision dataset, which contains 1 example per class and has 32,768 total classes defined via natural language descriptions.

Context Window = 32,768, i.e. 32,768 image classes, each with 1 example
These 32,768 images form "the proxy" to a Computer Vision Dataset
Each image is described by a textual/natural language desc.

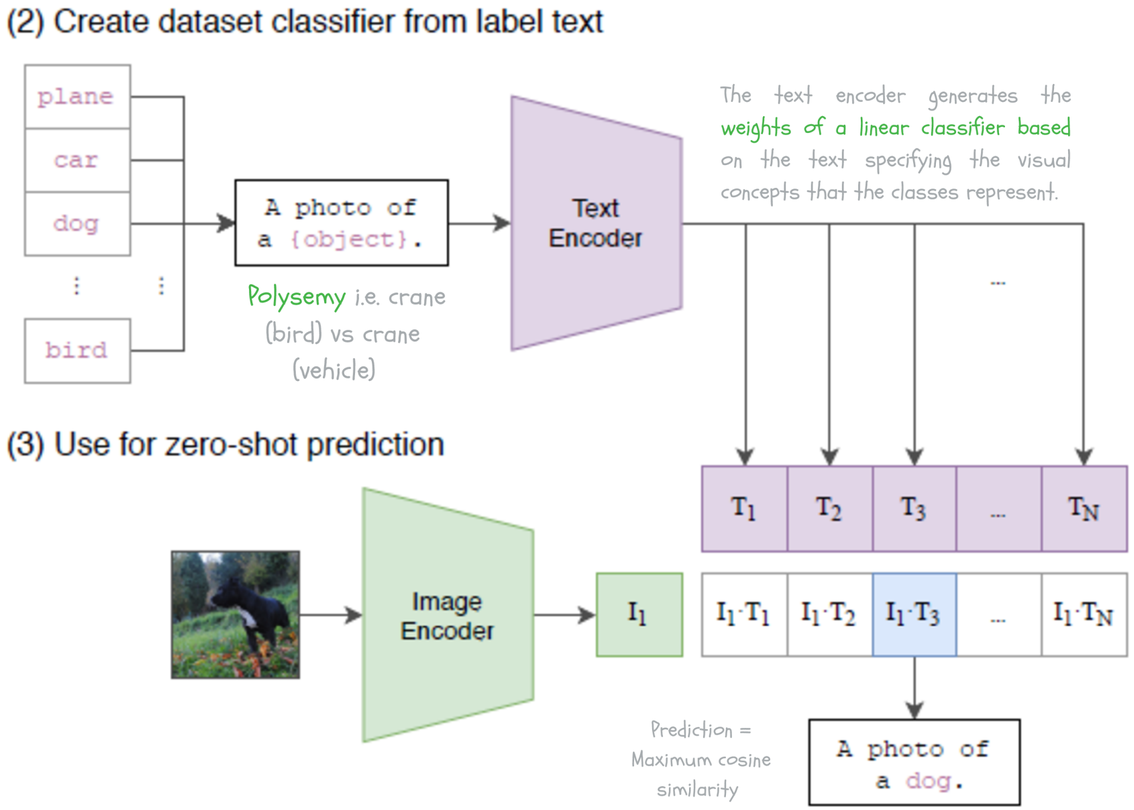
Importance

A large context window enhances the model's ability to generalise over vast amounts of data and learn richer image representations through natural language.

It does so by not only "understanding" the correct (ground truth) text associated with an image, but also by discerning how the other candidate texts are not related to the image.

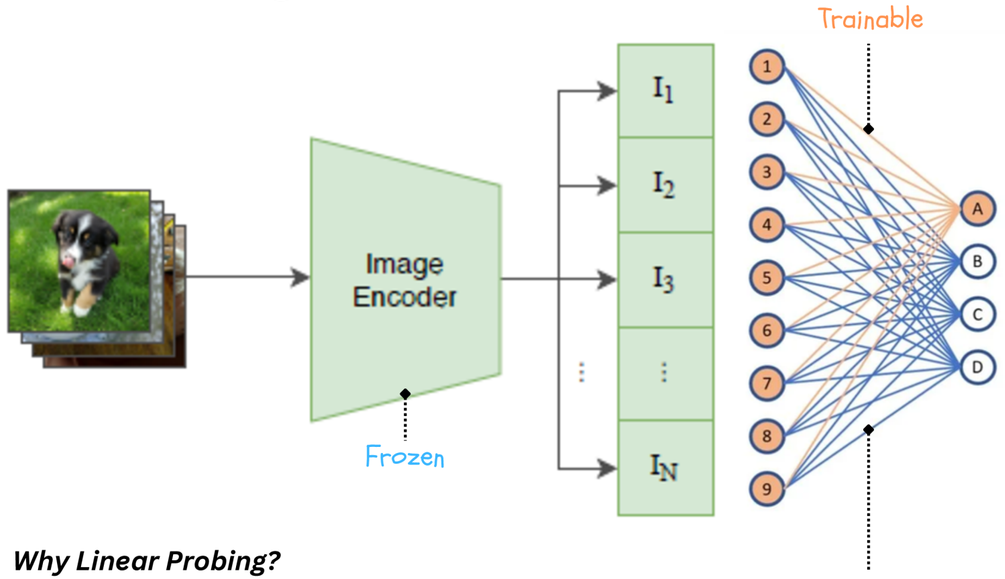
Working CLIP

Inference



Prompt Engineering to overcome polysemy. Prompts might contain the answer

Linear Probing



- ***Why Linear Probing?***
  - CLIP pre-training produces a rich Image Encoder
  - Fairer comparison with the classification models trained on ImageNet
  - A measure of the "quality" of feature representation

$$\hat{p}(y = k \mid x) = \frac{\exp(w_k^\top x + b_k)}{\sum_j \exp(w_j^\top x + b_j)}$$

Single Linear Layer + Softmax

Zero-Shot CLIP

Motivation

Zero-shot learning usually refers to the study of generalising to unseen object categories

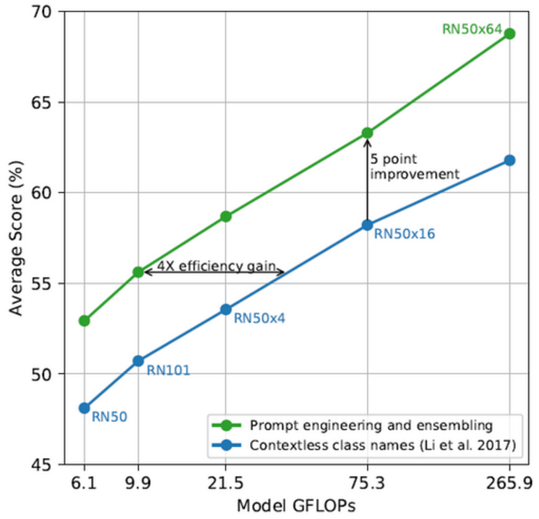
- ***Why pursue Zero Shot Transfer?***
  - Most research focused on unsupervised learning focuses on representation learning
  - The authors study zero-shot transfer as a way of measuring the task learning (or domain shift) capabilities of machine learning systems

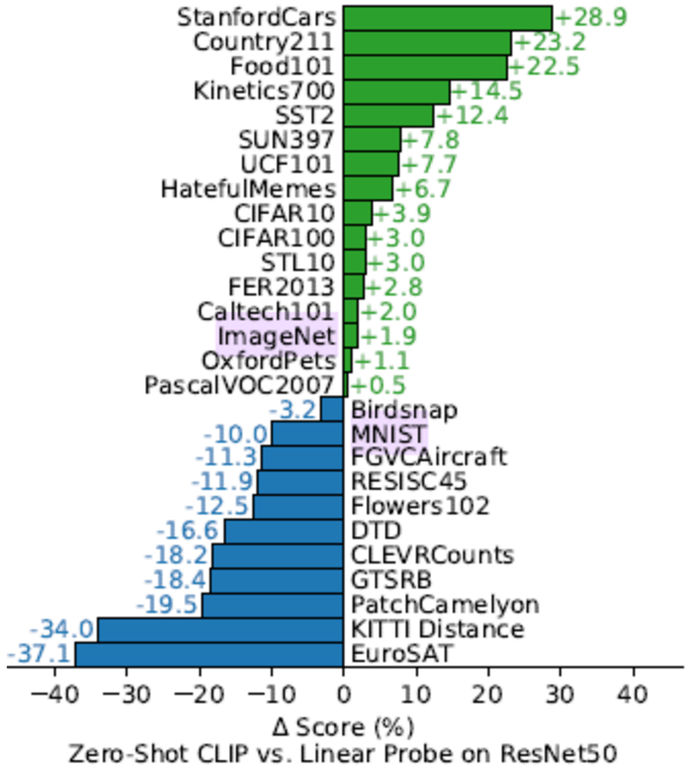
- ***What does it signify for CLIP?***
  - Zero-shot CLIP outperforms a supervised alternative (Linear Probe on ResNet-50) on multiple datasets.
  - Most notably, zero-shot CLIP outperforms ResNet-50 on ImageNet, the very dataset ResNet-50 was directly trained on.
    - We speculate this is due to natural language providing wider supervision for visual concepts involving verbs, compared to the noun-centric object supervision in ImageNet.
  - Interestingly, CLIP lags behind on the trivial MNIST dataset.

Prompt Engineering

The authors observed that zero-shot performance can be significantly improved by customising the prompt text to each task.

- ***How it helps***
  - Prompts provide targeted contextual reference
  - E.g. Image containing a photo of {label} vs Image containing a photo of {label}, a pet.
  - E.g. For OCR tasks use “” i.e. The word “{label}”





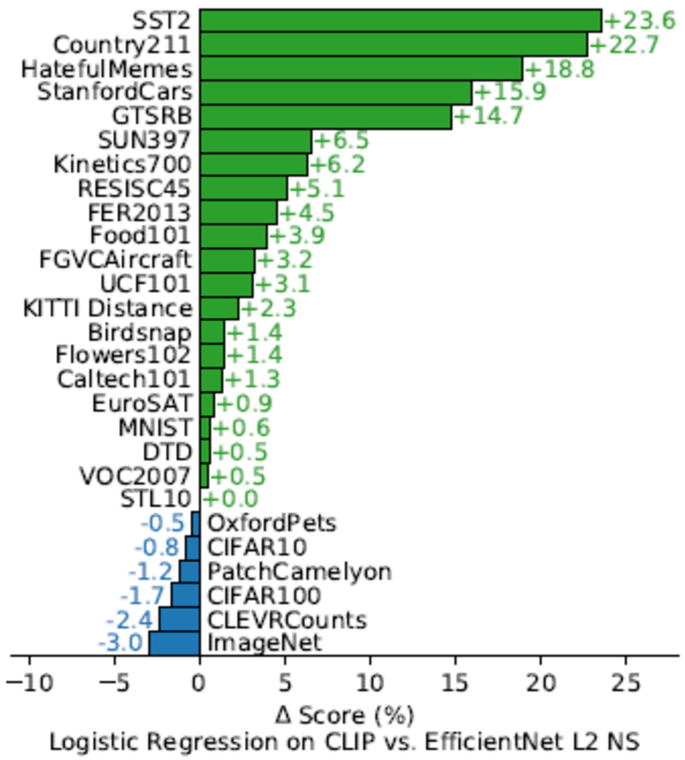
Results CLIP

Representation Learning

Representation Learning corresponds to the model's capability to uncover meaningful underlying patterns from input data (images, text, etc).

- ***Why pursue Representation Learning?***
  - The authors extensively studied zero-shot transfer via task learning capabilities; it is more common to study the representation learning abilities of a model.

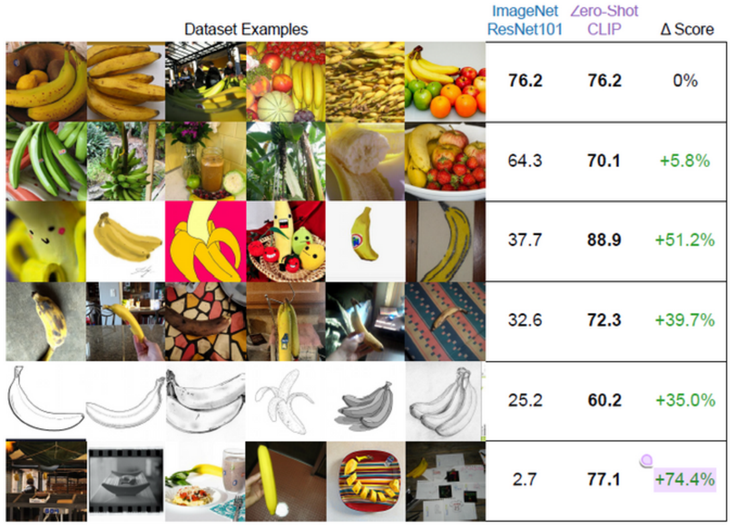
- ***Methodology***
  - End-to-end fine-tuning of a model is a strong way to measure representation learning. However, it poses some drawbacks:
    - It can compensate for and potentially mask failures to learn general and robust representations during the pre-training
    - Fine-tuning is orthogonal to CLIP's goal of developing a high-performing task and dataset-agnostic pre-training approach.
  - Although Linear-Probing provides less flexibility than fine-tuning, its efficiency and resonance with CLIP's goals make it an ideal choice for measuring Representation Learning in this case.



CLIP improves the most on tasks requiring OCR (SST2, hateful Memes)

CLIP also beats SOTA models at all levels in terms of compute efficiency

Distribution Shift



- ***Key Takeaways***
  - This showcases CLIPS effective representation learning. E.g., it understands the shape and structure of a banana compared to ImageNet models relying heavily on colour.
- ***An "~~un~~fair" advantage?***
  - ZS Clip is fed input prompts, one of which contains the "answer," while ImageNet models are only given the image.

Limitations CLIP

Human Comparison

Zero-shot learning usually refers to the study of generalising to unseen object categories

- ***Similarities***
  - We see that the hardest problems for CLIP are also hard for humans. To the extent that errors are consistent.
- ***Disimilarities***
  - The findings suggest that humans possess an awareness of their own uncertainty; they "know what they don't know."
  - Humans can effectively update their prior beliefs based on a single example, especially for images they are most uncertain about.
  - CLIP is not able to update priors similarly, and there remains a large gap between zero-shot vs one-shot improvement in humans versus CLIP.

	Accuracy	Majority Vote on Full Dataset	Accuracy on Guesses	Majority Vote Accuracy on Guesses
Zero-shot human	53.7	57.0	69.7	63.9
Zero-shot CLIP	**93.5**	**93.5**	**93.5**	**93.5**
One-shot human	75.7	80.3	78.5	81.2
Two-shot human	75.7	85.0	79.2	86.1

Truly Out of Distribution Data

CLIP performs poorly on truly OOD data, which was not present as part of its web-scale pre-training

- ***Optical Character Recognition (OCR)***
  - CLIP learns a high-quality semantic OCR representation
    - It performs well on digitally rendered text, which is common in its pre-training dataset.
    - Evident by performance on SST2.
  - However, CLIP only achieves 88% accuracy on the handwritten digits of MNIST
    - Which remains a trivial dataset for even a simple MLP

Social / Ethical Concerns

- ***Side effects of Web-scale pre-training***
  - CLIP is trained on text paired with images from the internet.
  - These image-text pairs are unfiltered and uncurated
  - As a result, CLIP models learning many pre-concieved human social biases on the internet

Miscellaneous

- ***Poor adaptation to few-shot learning***
  - CLIP does not optimise for few-shot performance
  - In the work, the authors fall back on fitting linear classifiers on top of CLIP's features

- ***Complex visual concepts***
  - The authors emphasise that using natural language to specify image classifiers offers a flexible and general interface.
  - However, this approach also comes with limitations.
  - Certain complex tasks and visual concepts are difficult to accurately express using text alone.

THANK YOU

Connecting
Text & Images