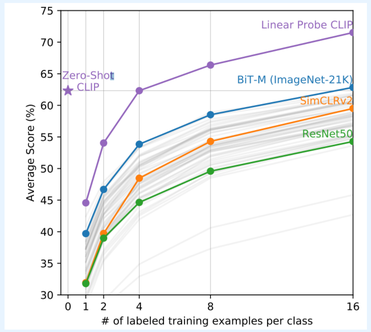
CLIP: Learning transferable visual models from natural langage supervision

Mathis Wauquiez

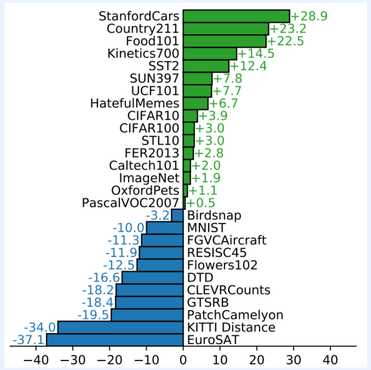
MVA - 2025

Objectives

High-quality image features. The original objective of CLIP (Contrastive Language-Image Pre-training) was to create high-quality features leveraging natural langage supervision. This is a hot topic in vision, called few-shot learning, which allows one to do classification and other tasks on very small datasets.



(a) **Linear probe evaluation** of CLIP and other publicly available models on 20 datasets with at least 16 samples per class.



(b) Zero-shot vs Few-shot on several datasets.

Figure 1. Comparison of linear probe evaluation and zero-shot vs few-shot results.

By-product: A zero-shot classifier. But the main contribution of CLIP is its zero-shot classifier ability: in under 50 lines of code, anyone can do almost SOTA classification on really small datasets, or even with one sample per class! Without any form training, its accuracy matches that of a good ResNet50 model on the ImageNet dataset.

Textual features. The textual embeddings produced by CLIP's textual encoder are aligned with the corresponding images. As such, it makes excellent embeddings for text-guided image generation, and CLIP's textual embeddings are used in a variety of applications, such as **Stable Diffusion** and **DALLE-2**. They also serve for image retrieval purposes, or for content moderation

CLIP's Approach

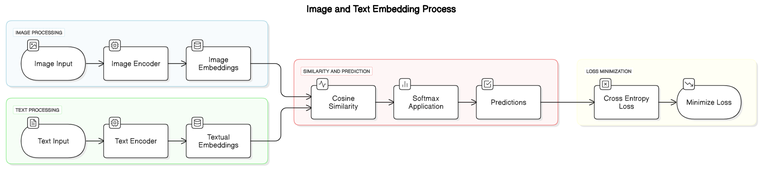


Figure 2. Summary of the approach

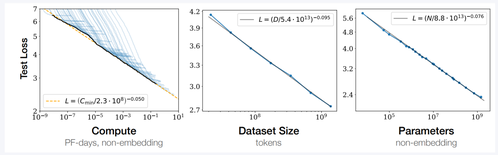
CLIP is an example of contrastive learning. Contrastive learning is a type of learning where the objective is to predict wether a pair of data is positive (similar) or negative (dissimilar). In the case of CLIP, it tries to predict wether a given text corresponds to an image or not, by maximizing the cosine similarity between similar items and minizing it for dissimilar items (InfoNCE loss).

Representation learning has gained attention since the paper "A Simple Framework for Contrastive Learning of Visual Representations" (SimCLR), was published. This type of learning is very powerful, as it does not need any form of label, and yet gives powerful representations. It is part of the representation learning techniques (sometimes, also called "self-supervised learning", or even "unsupervised learning").
Other representation learning methods exists, that achieves results comparable to that of CLIP, such as DINO, DINOv2, Masked AutoEncoders, BYOL (Bootstrap Your Own Latent), ... Several models built on and improved CLIP, aiming to enhance its performance, scalability, or adaptability (Florence/Flaming/OpenCLIP/...).

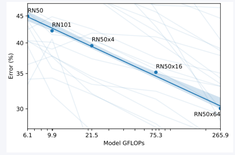
Training

Dataset. The dataset designed for CLIP is Web Image Text (WIT): an open sourced dataset containing 400 millions of (image, text) pairs, scrapped from the internet and accessible to all.

Trained Models and scaling laws. Recent work suggest that transfer performance is a smoothly predictable function of compute. To verify this hypothesis, CLIP trained 8 different models spanning 2 orders of magnitude. They trained five Resnet-like models, and Vision Transformer models (B/32, B/16, L/14 (+ L/14 @ 336px)). But nowadays 300+ different models trained using CLIP methodology are available on huggingface, each available through huggingface's `'open_clip'` library.



(a) Scaling laws for langage modeling performance. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two [2].



(b) CLIP Zero-shot performance as a function of compute

Contrastive objective. The final methodology uses a contrastive objective. But the first approach of the paper was to predict a caption! This approach failed to scale, as it tried to predict the *exact* words of the caption, which is difficult due to the wide variety of captions. As such, contrastive objectives can learn better representations than their predictive counterparts.

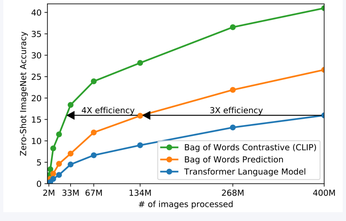


Figure 4. Contrastive vs predictive vs transformer zero-shot transfer

Hyperparameter search for massive models. Training the RN50x64 model required 18 days using 592 V100 GPUs. Therefore, a good methodology for hyperparameters optimization is crucial. To minimize training expenses, they found the hyperparameters for ResNet50 as a baseline, and scaled them heuristically. The hyperparameters for ResNet50 were found using a combination of grid search, random search and manual tuning.

Practical training optimization. They used large batch sizes because of the contrastive objective (see [1]). They used mixed precision and gradient checkpointing to save on memory and accelerate the training (available through a single line of pytorch code). They also used half-precision Adam statistics and half-precision stochastically rounded text encoder weights. These are simple but significant improvements.

Zero-shot experiments

We investigated the zero-shot performance of CLIP-Vit-L/14 across several datasets. The hypothesis we made is that the model will have a high zero-shot accuracy for datasets with "popular" topics, as they are more present in the training dataset. A broader investigation would be needed to confirm this hypothesis, but it seems coherent with the results we obtained.

Dataset	Accuracy	Prompt
Minerals	0.339	*"a close-up photo of a {} mineral"*
Food	0.967	*"the {} I got for lunch"*
Birds	0.993	*"a picture of a {} bird."*
Brain tumor	0.234	*"a MRI scan of an healthy patient"/"a MRI scan of a patient with {}"*
Sports	0.810	*"someone playing {}"*

Table 1. Performance of models on various datasets.

As expected, we had very low accuracy on complex datasets like the brain tumor dataset, and high accuracy for birds and for the food dataset. We suspect that the model was trained on the birds dataset or a subset of it.

CLIP linear probe performance

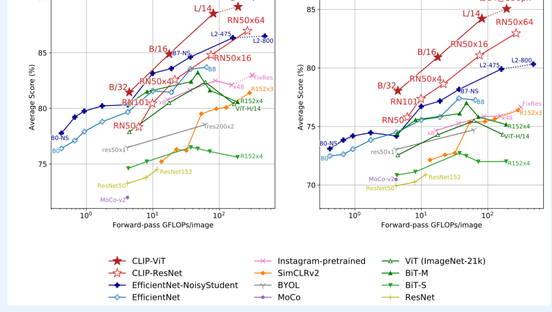


Figure 5. Linear probe performance of CLIP models in comparison with state-of-the-art computer vision models

References

[1] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020.

[2] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020.

Figure 6. Read the CLIP article!