Contrastive Alignment of Vision to Language Through Parameter-Efficient Transfer Learning

Zaid Khan, Yun Fu
Northeastern University, Boston, USA
{khan.za, y.fu}@northeastern.edu

Abstract

Contrastive vision-language models (e.g. CLIP) are typically created by updating all the parameters of a vision model and language model through contrastive training. Can such models be created by a small number of parameter updates to an already-trained language model and vision model? The literature describes techniques that can create vision-language models by updating a small number of parameters in a language model, but these require already aligned visual representations and are non-contrastive, hence unusable for latency-sensitive applications such as neural search. We explore the feasibility and benefits of parameter-efficient contrastive vision-language alignment through transfer learning: creating a model such as CLIP by minimally updating an already-trained vision and language model. We find that a minimal set of parameter updates ($<7\%$) can achieve the same performance as full-model training, and updating specific components ($<1\%$ of parameters) can match 75% of full-model training. We describe a series of experiments: we show that existing knowledge is conserved more strongly in parameter-efficient training and that parameter-efficient scaling scales with model and dataset size. Where paired-image text data is scarce but strong multilingual language models exist (e.g. low resource languages), parameter-efficient training is even preferable to full-model training. Given a fixed compute budget, parameter-efficient training allows training larger models on the same hardware, achieving equivalent performance in less time. Parameter-efficient training hence constitutes an energy-efficient and effective training strategy for contrastive vision-language models that may be preferable to the full-model training paradigm for common use cases. Code and weights at https://github.com/codezakh/LilT.

1 Introduction

Advances in transfer learning within the field of natural language processing (Houlsby et al., 2019b; Ben Zaken et al., 2022) have shown that when adapting to a novel task, updates to a small percentage of neurons ($< 1\%$) in large, pretrained transformer-based language models can achieve nearly equivalent results to finetuning the entire model. Sung et al. (2021) showed that given the existence of already-aligned visual representations (e.g. CLIP's visual encoder) only a small number (4%) of parameters in a pretrained language model need to be updated for the language model to complete tasks such as visual question answering using the already-aligned visual representations. However, *the creation of aligned vision and language representations typically involves updating all the parameters* of a language model and a vision model, often randomly initialized (Radford et al., 2021). Zhai et al. (2021) find that if the weights of a pretrained vision model are used as an initialization, only the neurons of the language model need to be updated to align the visual and language representations and match or exceed the performance of full-model training, resulting in a 50% reduction in trainable parameters. We take this line of investigation to its natural conclusion, asking — *given that strong, pretrained vision and language models both exist, can we minimally update both of their parameters to align their representations?*

Answering this question is valuable for two reasons. From a practical perspective, contrastive vision-language alignment constitutes a form of large-scale pretraining and hence a heavy energy
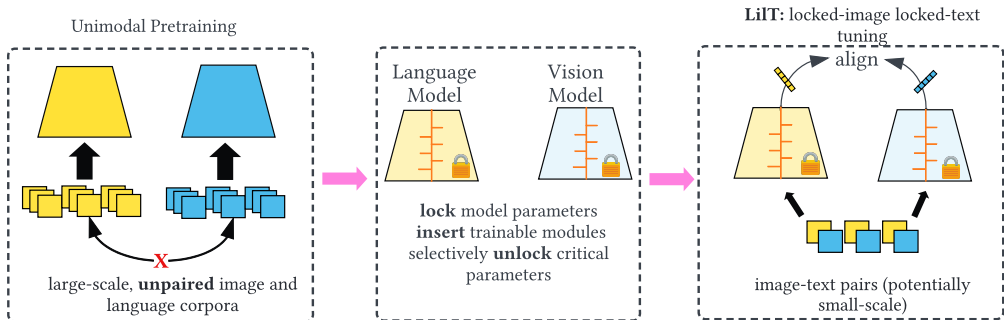
Figure 1: A conceptual diagram. After unimodal pretraining, parameter-efficient transfer to contrastive vision-language alignment is achieved by changing as few as 0.3% of the parameters from initialization, matching the performance of full model training.

expenditure. Methods for parameter-efficient transfer learning result in significantly reduced GPU memory requirements, and can therefore lower energy costs. Second, collecting millions of images with textual annotations is prohibitively expensive when millions of image-text pairs cannot be scraped from the internet, such as in the case of low resource languages or images from domains that require expert descriptions. In these cases, transfer learning by maximally preserving knowledge from strong, unimodal pretraining becomes compelling. Our contributions can be summarized as follows.

- We show contrastive vision-language models can be created by updates to a relatively small (<7%) set of parameters in pretrained vision and language models, which we dub **LilT** (Locked image-language tuning) for brevity.

- We conduct an detailed empirical study of combinations and interactions of various methods for parameter-efficient transfer learning.

- We show that contrastive vision-language models created with parameter-efficient transfer learning conserve useful existing knowledge from their initializations better than full model finetuning, and this has benefits in realistic scenarios.

Limitations Similar to Desai & Johnson (2021), we conduct most of our experiments on the COCO dataset, and conduct additional scaling experiments with a larger dataset of 1.5M pairs. There is a possibility that our conclusions may not hold beyond this range. Second, we choose to focus on zero-shot classification and information retrieval tasks. Our conclusions may not hold for other uses of image-text embeddings, such as using them as input for downstream vision-language tasks. Finally, we explicitly limit the scope of the study to transformer-based contrastive vision-language models. Thus, our conclusions may not apply to those based on other architectures. Despite these limitations, we believe our conclusions are useful because there are realistic situations in which there are much fewer than 1.5M image-text pairs (e.g. low resource languages) available.

Outline First, we cover background material (§2.1), then introduce our approach of parameter-efficient transfer learning for contrastive vision-language alignment (§2). We then describe experiments and a discussion of experimental results (§3), followed by related work (§4).

2 METHODS

The basic idea of our approach is to align a vision model and a language model by updating a small percentage of their parameters by gradient descent. This involves four main elements. First, the vision and language model must initialized from strong, pretrained vision and language models, rather than random initialization. Second, we lock all the parameters in each model. Third, we selectively unlock critical parameters. Fourth, we insert small trainable modules into each model to aid adaptation. There are multiple ways of implementing these strategies, which we cover in this section.

2.1 Background

In this section, we briefly cover the mechanics of contrastive language image alignment as used by (Radford et al., 2021), as well as the common "two-tower" (Zhai et al., 2021), dual transformer encoder architectures employed by CLIP-style models. Contrastive language image alignment pulls representations of matched image-text pairs together, while pushing those of unmatched pairs apart. The goal is to learn an image encoder f_θ and a text encoder g_ϕ such that given an image-text pair $(\mathbf{x}^I, \mathbf{x}^T)$, the encoded representations $f_\theta\left(\boldsymbol{x}^I\right)$ and $g_\phi\left(\boldsymbol{x}^T\right)$ are close under a distance metric if they are semantically similar and far apart if not. Let $\left\{x_k^I, x_k^T\right\}_{k=1}^b$ be a batch of b image-text pairs. For each image $\boldsymbol{x}_k^I$ in an image-text pair $\left\{x_k^I, x_k^T\right\}$, the matched text $\boldsymbol{x}_k^T$ is the positive, while all other texts within the batch are used as negatives. The image-to-text contrastive loss $\mathcal{L}_k^I$ for $\boldsymbol{x}_k^I$ is then

$$\mathcal{L}_k^I\left(\boldsymbol{x}_k^I, \left\{\boldsymbol{x}_j^T\right\}_{j=1}^b\right) = -\frac{1}{b}\log\frac{\exp\left(s_{k,k}^I\right)}{\sum_j \exp\left(s_{k,j}^I\right)},$$

where $s_{k,j}^I$ is the similarity of the k-th image to the j-th text. The similarity function is usually taken to be the cosine similarity, which can be easily computed as $f_\theta\left(\boldsymbol{x}^I\right) \cdot g_\phi\left(\boldsymbol{x}^T\right)$ if the representations are normalized to unit length. Conversely, the text-to-image contrastive loss for $\boldsymbol{x}_k^T$ is

$$\mathcal{L}_k^T\left(\boldsymbol{x}_k^T, \left\{\boldsymbol{x}_j^I\right\}_{j=1}^b\right) = -\frac{1}{b}\log\frac{\exp\left(s_{k,k}^T\right)}{\sum_j \exp\left(s_{j,k}^T\right)}.$$

The complete training loss then becomes

$$\mathcal{L} = \frac{1}{2}\sum_{k=1}^b \left(\mathcal{L}_k^I + \mathcal{L}_k^T\right). \tag{1}$$

Architectures for contrastive language image alignment must encode both texts and images to vector representations. This is usually implemented using separate text encoder and image encoders. A variety of choices are possible for these encoders, but we restrict ourselves to the popular (Radford et al., 2021; Li et al., 2021a;b; Yao et al., 2021; Khan et al., 2022; Zhai et al., 2021; Yang et al., 2022; Wang et al., 2021) choice of transformer (Vaswani et al., 2017) architectures, specifically, the BERT (Devlin et al., 2019) family of language models for the text encoder, and the ViT (Dosovitskiy et al., 2021) family for the image encoder. Let $t(\cdot)$ denote an arbitrary architecture from one of the above families. After consuming an input $\mathbf{x}$, the transformer $t(\cdot)$ produces a sequence of vectors $t(\mathbf{x}) = \{\mathbf{z}_{\text{cls}}, \mathbf{z}_1, \ldots, \mathbf{z}_N\}$, where $\mathbf{z}_{\text{cls}}$ is the embedding of the `[CLS]` token, which is taken to be the representation of the input $\mathbf{x}$ following dimensionality reduction by a trainable linear projection.

2.2 Adding Adapters

Aligning the representations of a language transformer and a vision transformer is typically done by updating 100% of the parameters in one (Zhai et al., 2021) or both (Radford et al., 2021) of the transformers. By freezing the transformers, we exclude full-model training, and must use an alternative strategy to align the image and text representations. A promising approach is inserting a small (relative to each transformer), trainable module into the frozen, pretrained transformers that can learn to modify the internal representations of the transformer it is placed within, such that the representation spaces of the frozen vision and language transformers become aligned while leaving the pretrained parameters untouched. We explore two such modules: layerwise adapters (Houlsby et al., 2019a; He et al., 2021) and "deep" adapters.

Layerwise adapters (Houlsby et al., 2019a) have been used to adapt pretrained transformer-based language models to new tasks while only updating $2-3\%$ of model parameters. A layerwise adapter is inserted before each layer normalization (Ba et al., 2016) layer in a transformer, and consists of a weight matrix that downsamples the input, followed by an activation function (we use GELU (Hendrycks & Gimpel, 2016)) and a weight matrix that restores the input to the original dimensionality, and finally, a residual connection. We depict the architecture / placement of layerwise adapters in Fig 3.
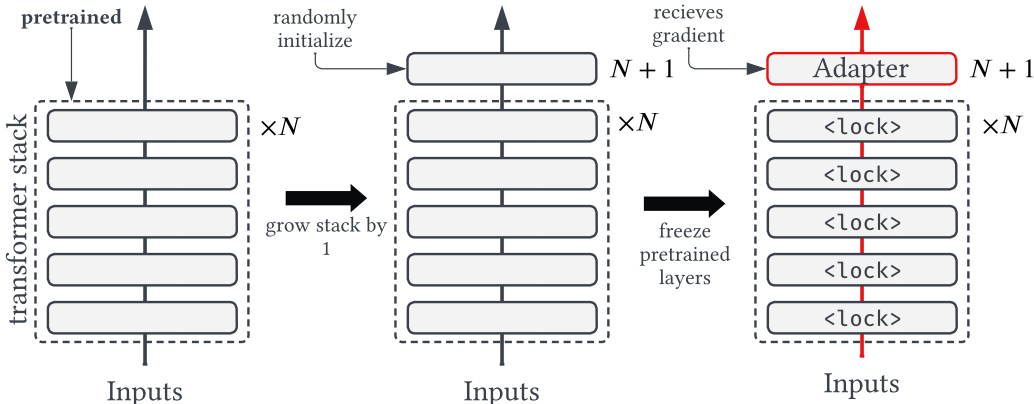
Figure 2: Growing the transformer encoder stack to add a trainable deep adapter to a locked model. The deep adapter is architecturally identical to a layer from the encoder stack.

Another solution is to treat the frozen encoders as feature extractors, and learn trainable adapters that align the frozen image and text features. Transformer architectures can be seen as a stack of identically structured transformer encoder layers, so a natural solution to the problem of designing a trainable adapter atop a stack of frozen transformer encoder layers is to grow the stack, and keep the newly added layers trainable. This yields a generic approach (Fig. 2) to add a trainable adapter to a frozen transformer from any of the standardized families (e.g. BERT (Devlin et al., 2019), ViT (Dosovitskiy et al., 2021)) that only requires a small number of parameters to recieve gradients ($\approx 7\%$ for `bert-base`).

2.3 UNLOCKING PARAMETERS

We try two strategies for selectively unlocking parameters in a frozen transformer: unlocking the layer normalization (Ba et al., 2016) parameters, and BitFit (Ben Zaken et al., 2022). Standard transformers (Vaswani et al., 2017) have two layer normalization (Ba et al., 2016) modules for each transformer encoder layer, and these are known to play an important role (§4). Each layer normalization layer has learnable scale γ and bias parameters β that apply an elementwise scale and shift to the input of the layer normalization layer. In the first strategy, we allow the layer normalization layers to remain unlocked and receive gradient updates. In BitFit (Ben Zaken et al., 2022), (Bias-term Finetuning), the additive bias terms of *every* module in a transformer encoder layer are allowed to remain unlocked and receive gradient updates. Both of these strategies unlock a small percentage (0.24% and 0.31% of the parameters in a 12-layer `base` transformer respectively).
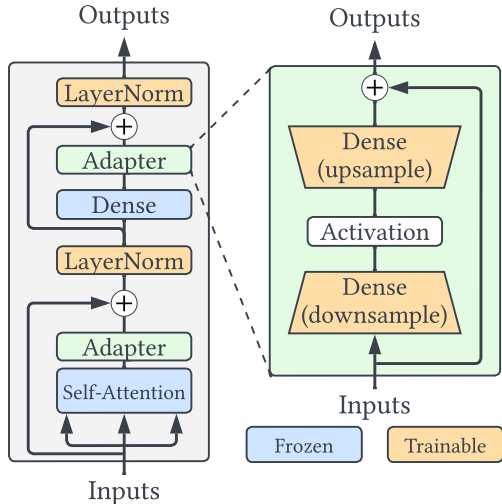

Figure 3: The architecture and placement of layerwise adapters combined with a layernorm unlocking strategy.

2.4 IMPLEMENTATION DETAILS

Datasets We draw $591,753$ image-text pairs from the training set of COCO2014Lin et al. (2014), following the split of Karpathy & Fei-Fei (2017). The weights of the vision encoders are initialized from DeiT Touvron et al. (2021), and the text encoders are initialized from SimCSE (Gao et al., 2021). We train each model with a batch size of 512 on 4x NVIDIA A6000 GPUs for 15 epochs, using the AdamW optimizer (Loshchilov

Table 1: An ablation study with `bert-base` as the text encoder and a ViT-B/16 as the image encoder. An 🔒 indicates the component is locked and does not recieve gradient updates, while 🔶 indicates the opposite. LN($\square_T$/$\square_I$) indicates the layer normalization weights in the text encoder were locked while those of the image encoder recieved gradient updates, and vice versa for LN($\square_T$/$\square_I$). θ is the trainable linear projection. TR and IR is mean text retrieval and image retrieval scores across Rank-1,5,10. Deep (Fig 3) and Layerwise (Fig. 2) adapters are detailed in §2.2, and BitFit in §2.3.

		Components						Flickr		ImageNet V2
		TE	IE	θ	Unlock Strategy	Adapter	% Trained	TR	IR	Acc-1
(a)	Frozen	🔒	🔒	🔒	LN($\square_T$/$\square_I$)	-	0.00 %	0.8	1.3	0.2
(b)	LN Only	🔒	🔒	🔒	LN($\square_T$/$\square_I$)	-	0.04 %	24.3	21.6	4.3
(c)	Projection Only	🔒	🔒	🔶	LN($\square_T$/$\square_I$)	-	0.20%	38.7	31.8	6.7
(d)	LilT$_{LN}$	🔒	🔒	🔶	LN($\square_T$/$\square_I$)	-	0.24%	62.3	51.74	12.5
(e)	LilT$_{BF}$	🔒	🔒	🔶	BitFit	-	0.31%	62.6	52.1	12.6
(f)	LilT$_{DA}$ w/o LN	🔒	🔒	🔶	LN($\square_T$/$\square_I$)	Deep	6.96 %	57.5	47.8	9.02
(g)	LilT$_{DA}$	🔒	🔒	🔶	LN($\square_T$/$\square_I$)	Deep	6.99 %	68.6	58.5	12.9
(h)	LilT$_{LwA}$ w/o LN	🔒	🔒	🔶	LN($\square_T$/$\square_I$)	Layerwise	6.97 %	74.8	63.9	12.0
(i)	LilT$_{LwA}$	🔒	🔒	🔶	LN($\square_T$/$\square_I$)	Layerwise	7.01 %	75.4	64.4	12.2
(j)	LilT$_{LwA}$(BitFit)	🔒	🔒	🔶	BitFit	Layerwise	7.09%	75.3	64.4	12.2
(k)	LilT$_{DA}$ (BitFit)	🔒	🔒	🔶	BitFit	Deep	7.06%	68.7	58.4	13.2
(l)	LiT	🔶	🔒	🔶	LN($\square_T$/$\square_I$)	-	56.01 %	66.1	53.5	15.0
(m)	LiT (reversed)	🔒	🔶	🔶	LN($\square_T$/$\square_I$)	-	43.99 %	53.7	46.22	8.8
(n)	LiT + LilT$_{DA}$	🔶	🔒	🔶	LN($\square_T$/$\square_I$)	Deep	65.87 %	84.2	75.2	13.6
(o)	LiT + LilT$_{LwA}$	🔶	🔒	🔶	LN($\square_T$/$\square_I$)	Layerwise	57.57%	76.7	64.9	13.84
(p)	CLIP	🔶	🔶	🔶	LN($\square_T$/$\square_I$)	-	100.0 %	75.8	65.8	12.3

& Hutter, 2017) optimizer with a weight decay of 0.02. The learning rate is warned up to $1e^{-4}$ in the first 10 epochs, and then decayed to $1e^{-5}$. We use random crops of resolution 256×256 with RandAugment(Cubuk et al., 2020), with colors transformations removed following Li et al. (2021a).

3 EXPERIMENTS

We conduct experiments on zero-shot multimodal classification, image-text retrieval, and multilingual image text retrieval to investigate the following research questions.

1. Can contrastive vision language models be created through parameter-efficient transfer learning?

2. How do different methods for parameter efficient transfer learning interact with each other?

3. Do contrastive vision language models created through parameter-efficient transfer learning conserve useful knowledge from their initializations better than full-model finetuning?

4. Does parameter-efficient transfer learning scale with respect to model size and dataset size?

We evaluate all models on five tasks: zero-shot natural-language guided image classification (Radford et al., 2021), image-to-text retrieval (TR), text-to-image retrieval (IR), and 0-shot TR/IR. For zero-shot classification, we use the ImageNetV2 (Recht et al., 2019) test set. For IR/TR, we use the COCO2014 test split of Karpathy & Fei-Fei (2017), containing 5k images and 25k captions. For zero-shot IR/TR, we use the test set of Flickr30k(Plummer et al., 2015), containing 1k images and 5k captions.

3.1 ABLATION STUDY

The results of the study are displayed in Table 1. After updating only 0.24% of parameters, parameter unlocking methods achieve equivalent zero-shot classification performance to full-model training: compare (d) & (e) to (p). However, parameter unlocking alone is insufficient to achieve the image-text retrieval abilities of full-model training, but adapter-based methods (f-k) can match full-model training (p) in both zero-shot classification and image-text retrieval. BitFit and layer normalization unlocking are interchangeable as parameter unlocking strategies ($< 0.2\%$ difference between (f/j) and (h/i)). LilT$_{LwA}$ (h), with the layerwise adapters, is substantially better ($\approx 7\%$) at image text retrieval than LilT$_{DA}$ (f), and only slightly worse at classification. LilT and LiT are complimentary (m/n), and

Table 2: Cross-lingual zero-shot retrieval. A multilingual `bert-base` model is aligned with a ViT-B/16 on English image-text pairs from COCO, and evaluated on image-text pairs in languages unseen during alignment.

	RU		PL		TR		ZH		KO		IT		ES	
	TR	IR	TR	IR	TR	IR	TR	IR	TR	IR	TR	IR	TR	IR
LiT	45.17	40.17	44.0	41.83	24.17	23.33	64.67	61.0	34.17	29.67	60.17	56.0	65.67	62.33
CLIP	57.67	53.17	59.17	54.83	33.33	29.83	79.0	**74.0**	42.33	35.33	71.0	65.33	75.67	69.5
LilT$_{DA}$	58.5	51.33	60.33	55.33	42.33	35.0	74.17	67.67	44.67	35.67	74.5	68.83	77.0	74.17
LilT$_{LwA}$	**61.83**	**57.0**	**63.0**	**56.5**	**46.5**	**41.0**	**79.0**	72.83	**50.0**	**43.67**	**77.67**	**72.17**	**79.17**	**74.5**
Δ	↑4.17	↑3.83	↑3.83	↑1.67	↑13.17	↑11.17	↑0.0	↑-1.17	↑7.67	↑8.33	↑6.67	↑6.83	↑3.5	↑5.0

it is possible to only align only *one* of the encoders in a parameter-efficient manner. While LiT (k) excels at image classification, it suffers from a similar problem as parameter unlocking strategies: it is relatively poor at image text retrieval.

Discussion First, it is clear that creating contrastive vision-language models through parameter-efficient transfer learning is feasible, and there are clear differences between model capabilities induced by different parameter-efficient transfer learning methods. Layerwise adapters stand out as the parameter-efficient transfer learning strategy capable of matching or exceeding full-model training. However, in cases where the language distribution is sufficiently simple (e.g. a list of singular words), parameter unlocking is sufficient, and easier to implement. Deep adapters stand out for their ability to achieve *better* performance than full-model training when combined with LiT (m).

3.2 CONSERVATION OF KNOWLEDGE FROM INITIALIZATION

We hypothesize that parameter efficient transfer learning preserves more knowledge from initialization than full model finetuning, and this is beneficial in some realistic scenarios. Low-resource languages likely do not have large-scale image-text pairs available to train a multimodal CLIP-like model for that language. However, *unimodal*, multilingual language models that have been trained on a dataset containing sentences from a given low-resource language often exist. A possible solution in this situation is to train a CLIP-like model on available image-text pairs from a high-resource language, while using a multilingual language model as the text encoder. The resulting model may be able to generalize to image-text retrieval tasks in a language unseen during vision-language alignment due to the multilinguality of the pretrained text encoder. We simulate this setting by aligning a pretrained multilingual `BERT-base` model with an ImageNet-pretrained ViT-B/16 on English-only image-text pairs, and evaluate it on image-text pairs in six different languages that the model was never provided paired images for. If parameter-efficient training preserves more knowledge from initialization, and that knowledge is useful, we expect that the retrieval model created through parameter efficient transfer learning should retain more of its multilingual language ability, and hence display greater accuracy on non-English languages.

We reuse the English training data from §2.4, and evaluate each model on the test set of Aggarwal & Kale (2020), which contains 1400 image-text pairs, split equally between Russian, Polish, Turkish, Chinese, Korean, Italian, and Spanish. We summarize results in Table 2. LilT$_{LwA}$ outperforms CLIP on $12/14$ tasks (5.3% absolute improvement), while LilT$_{DA}$ achieves better performance than CLIP on 11/14 tasks (1.4% absolute improvement). This suggests that parameter-efficient transfer learning conserves more information from initialization, and that information is useful for multimodal tasks.

3.3 SCALING WITH RESPECT TO DATA AND MODEL SIZE

Can parameter-efficient transfer learning take advantage of larger models and larger amounts of data? We test the the performance of parameter-efficient transfer learning as the amount of image-text pairs is increased to 1500k from 591k (Table 4) and as model size is increased (Table 3) from `base` ($\approx$ 200M params) to `large` ($\approx$ 700M params). When the amount of training pairs available triples, parameter-efficient transfer learning continues to match the performance of full-model training: (b) vs (d) in Table 4. Similarly, the performance of parameter-efficient transfer learning improves as model size increases: (a) vs (b) & (c) vs (d) in Table 3.

Table 3: Zero-shot task performance of `base/large` models after parameter-efficient training. LwA/DA indicates adapter types, corresponding to (rows h/f in Table 1).

	Model (591k Training Pairs)			Flickr				ImageNet V2	
	Configuration	# Trainable	% Trained	TR@1	IR@1	TR@5	IR@5	Acc-1	Acc-5
(a)	LilT$_{DA}$-base	14.65 M	7.51%	47.6	34.46	74.1	64.92	12.94	28.39
(b)	LilT$_{DA}$-large	25.92 M	4.06%	57.6	42.18	82.2	72.38	13.97	30.89
(c)	LilT$_{LwA}$-base	14.67 M	7.01%	56.8	41.7	81.1	70.74	12.18	27.78
(d)	LilT$_{LwA}$-large	51.18 M	7.43%	63.5	50.7	88.5	79.14	14.05	31.31
(e)	LiT-base	109.28 M	56.01%	44.1	29.64	72.1	59.94	15.0	29.44
(f)	CLIP-base	195.13 M	100.0%	56.1	44.3	81.7	71.98	12.29	28.44

Table 4: Zero-shot performance of `base` models after larger-scale pretraining (1.5M pairs).

	Model (1.5M Pairs)			Flickr				ImageNet V2	
	Configuration	# Trainable	% Trained	TR@1	IR@1	TR@5	IR@5	Acc-1	Acc-5
(a)	LiT-base	109.28 M	56.01%	48.8	32.72	78.1	63.02	**20.63**	**38.12**
(b)	CLIP-base	195.13 M	100.0%	60.5	43.8	84.7	72.16	16.61	35.14
(c)	LilT$_{DA}$-base	14.65 M	7.51%	50.4	35.66	78.2	65.3	16.98	35.53
(d)	LilT$_{LwA}$-base	14.67 M	7.01%	**61.1**	**44.5**	**85.6**	**72.9**	15.83	35.31

3.4 WHAT HAPPENS DURING ALIGNMENT?

We attempt to understand how alignment changes the language and vision model by studying the layer normalization layers of each model. Let f_θ be an image encoder g_ϕ be a text encoder. We initialize f_θ with weights from DEiTTouvron et al. (2021), and g_ϕ with weights from SimCSE Gao et al. (2021). We then lock all parameters except the layer normalization layers (configuration (c) in Tab. 1), and train the model following the standard CLIP training procedure, resulting in a pair of aligned encoders $(\bar{f}_\theta, \bar{g}_\phi)$. In total, we have four different models: the unaligned and aligned image encoders $(f_\theta, \bar{f}_\theta)$ and the unaligned and aligned text encoders $(g_\phi, \bar{g}_\phi)$. Without loss of generality, we describe our procedure for the text encoder pair $(g_\phi, \bar{g}_\phi)$. Let $\mathrm{LN}_i^1(\gamma, \beta)$ and $\mathrm{LN}_i^2(\gamma, \beta)$, denote the two normalization sublayers of the i-th layer in the transformer encoder stack. For layer $i \in 1, 2, \ldots N$, we plot the L^1 norm of the difference between the trainable layer normalization parameters γ, β of the aligned and unaligned encoders. We plot the results in Fig 4. Surprisingly, the text and image encoders display clearly opposite patterns (negative Pearson's r). In the text encoder, the difference between the aligned and unaligned layer normalization parameters decreases with depth — layer normalization parameters in the deeper layers of the text encoder change less as a result of alignment training. This is the *opposite* of the image encoder. In the image encoder, the layer normalization
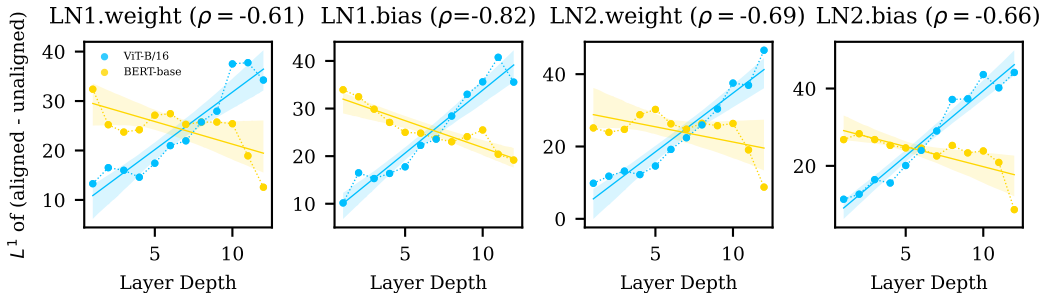
Figure 4: The depth of the layer normalization layers affects how much they are changed by alignment training, and the pattern is reversed between the image and text encoders. ρ is the Pearson correlation coefficient, and the translucent blue/yellow shading indicates 95% confidence intervals.
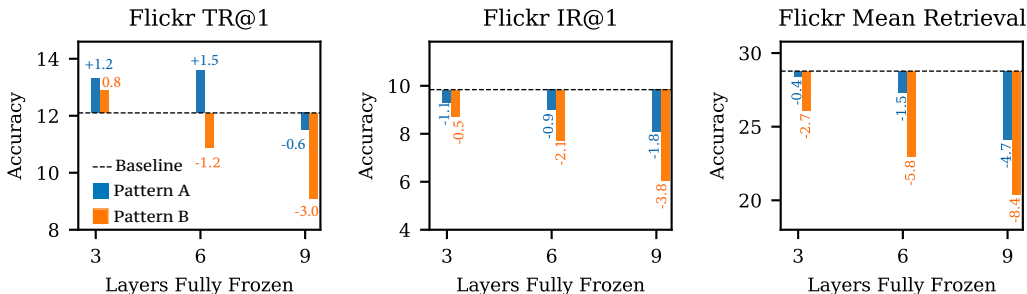
Figure 5: We freeze all parameters except for the LN parameters, then progressively lock LN parameters by layer. Fig 4 suggests that freezing the LN parameters in the deepest layers of the language model and the shallowest layers of the vision model (Pattern A) should have a smaller effect on performance than the opposite pattern (Pattern B), relative to the baseline (LNs in every layer unlocked) which we observe.

parameters which shift the most as a result of training are the deepest. We conduct another experiment with 50k pairs (Fig 5) to test the consequences of this pattern.

Discussion The patterns in the layer normalization layers may indicate that during alignment, the language and image modalities undergo changes at different semantic levels. The shallowest three layer normalization layers of the ViT-B/16 experience a $\approx 70\%$ lower magnitude shift than the deepest three layers. The shallow layers of a vision transformer attend more to local information (Raghu et al., 2021), while the deeper layers attend more to global context. Intuitively, this makes sense – we should expect an asymmetry between the amount of information in a short image caption compared to a dense image. Simple natural language concepts are often visually complex. Interestingly, this has already been exploited by certain vision-language models — (Khan et al., 2022; Li et al., 2021a) align *the lower half* of their text encoder to the visual encoder, while using the top half for a different purpose. This makes sense, given that the lower layers of the text encoder seem to change the most during alignment.

4 RELATED WORK

Vision-Language Pretraining The dual-encoder CLIP (Radford et al., 2021) (400m pairs) and ALIGN (Jia et al., 2021) (1b+ pairs) architectures were the first attempts at large-scale contrastive image-language alignment using the InfoNCE (van den Oord et al., 2018) loss to maximize the mutual information between matched image and text pairs. Subsequent work (Pham et al., 2021; Li et al., 2021b; Yao et al., 2021; Cui et al., 2022; Yang et al., 2022; Khan et al., 2022; Li et al., 2021a) has improved on the training tasks, dataset, and architecture of CLIP. While systems utilizing a multimodal encoder and cross attention Li et al. (2022); Khan et al. (2022); Wang et al. (2022); Lu et al. (2022); Zhu et al. (2021) perform better on benchmarks, their multimodal encoder makes them unsuitable for latency-sensitive search application, because rather than learning separate but aligned image and text embeddings, they learn a single multimodal embedding for an image-text pair. Thus, neural search remains the domain of contrastive vision-language models.

Frozen Language Models Tsimpoukelli et al. (2021) demonstrated that pretrained large language models are capable of quickly adapting to image understanding. They use an autoregressive transformer-based language model, which is frozen. A trainable ResNet (He et al., 2016) is then trained to transform images into input the frozen transformer can understand, by backpropagating the loss through the frozen transformer. MAGMA Eichenberg et al. (2021), FROMAGE Koh et al. (2023) and FLAMINGO Alayrac et al. (2022) scaled the conceptual approach of Tsimpoukelli et al. (2021) to billions of parameters, and recently, Merullo et al. (2022) have shown that a simple linear mapping is enough to allow a frozen large language model to (roughly) understand visual input, as long as the visual encoder has been trained to represent visual concepts aligned to language (e.g. CLIP). However, emerging approaches such as BLIP-2 Li et al. (2023) show that by combining soft prompting with a frozen LLM and a trainable visual encoder, a LLM can achieve state-of-the-art

accuracy on visuolinguistic understanding tasks such as visual question answering. Lu et al. (2021) propose the idea that transformers trained on language are capable of a form of universal computation, and can adapt to new tasks even if they are frozen, and do so better than fine-tuned models. However, Rothermel et al. (2021) find the findings may be reversed under certain hyperparameter settings. Interestingly, both note that the normalization layers seem to play an important role in this adaptation.

Parameter-Efficient Finetuning Many forms of adapters (Houlsby et al., 2019b; Karimi Mahabadi et al., 2021; Mahabadi et al., 2021) have been explored in natural language processing. VL-Adapter (Sung et al., 2021) investigate adapters in vision-language, but assume aligned visual representations. Lester et al. (2021) find that for very large language models, parameter-efficient adaptation approaches such as soft prompting are equivalent to finetuning the large language model. Liu et al. (2021) extend this finding, showing that combining soft prompting with adapters can often *exceed* finetuning on a given downstream task. Both prefix (Li & Liang, 2021) and prompt (Lester et al., 2021) tuning can also be understood as exploiting the knowledge in frozen transformers, as their optimization loops involve freezing the language model, effectively turning it into a part of the loss. Zhang & He (2020) develop a training scheme that progressively unfreezes / freezes layers of a transformer language model, and see significant improvements in training speed. Progressive growth approaches (Gu et al., 2021) slowly increase the depth of a transformer as training proceeds.

Layer Normalization in Transformers Kovaleva et al. (2021) find that the representations of transformers contain outlier dimensions that disrupt the quality of the learned embedding, and point to high-magnitude parameters in the layer normalization layers. A variety of techniques targeting layer normalization in transformers have been proposed, with various benefits. Xiong et al. (2020) prove that the placement of layer normalization layers relative to the residual connection in the transformer block contributes to learning instability under large learning rates, and propose an alternate placement. In contrast, FixUp (Huang et al., 2020) develops a novel initialization scheme for transformers that enables removing the normalization layers entirely. ReZero (Bachlechner et al., 2021) adds a learnable gate parameter to each residual connection before layer normalization, and demonstrate training extremely deep transformers quickly.

5 CONCLUSION & FUTURE WORK

We show that the performance of full model training for contrastive vision language alignment can be matched by updating a small number of parameters in existing vision models and language models, followed by an insertion of trainable modules. This suggests that the current paradigm of full-model training for contrastive vision language alignment involves significant unnecessary computation, and can be replaced by parameter-efficient transfer learning when the downstream use cases are natural-language classification or image-text retrieval. Current alignment strategies align representations from the top of each encoder stack. We find that in the text encoder, alignment changes the normalization parameters in the shallowest layers the most, while it is the opposite for the image encoder. Investigating and exploiting the asymmetry between vision and language could yield further benefits for multimodal understanding or more efficient training strategies. For future work, it would be interesting to analyze whether CLIP-like models created through parameter-efficient transfer learning are similar to CLIP in ways other than performance — for example, are they more or less biased? Or more or less robust to distribution shift? Another useful line of investigation would be probing vision-language models further to understand how alignment training effects the ability of the model to understand language. In summary, we believe that existing training methods are not fully exploiting the knowledge that exists in their initializations. Our approach presents one simple but effective way to use that knowledge.

REFERENCES

Pranav Aggarwal and Ajinkya Kale. Towards zero-shot cross-lingual image retrieval, 2020.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan

Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning. *ArXiv*, abs/2204.14198, 2022.

Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *ArXiv*, abs/1607.06450, 2016.

Thomas C. Bachlechner, Bodhisattwa Prasad Majumder, Huanru Henry Mao, G. Cottrell, and Julian McAuley. Rezero is all you need: Fast convergence at large depth. In *UAI*, 2021.

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1–9, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.1. URL `https://aclanthology.org/2022.acl-short.1`.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

Ekin D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. Randaugment: Practical automated data augmentation with a reduced search space. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pp. 3008–3017. Computer Vision Foundation / IEEE, 2020. doi: 10.1109/CVPRW50498.2020.00359. URL `https://openaccess.thecvf.com/content_CVPRW_2020/html/w40/Cubuk_Randaugment_Practical_Automated_Data_Augmentation_With_a_Reduced_Search_Space_CVPRW_2020_paper.html`.

Yufeng Cui, Lichen Zhao, Feng Liang, Yangguang Li, and Jing Shao. Democratizing contrastive language-image pre-training: A clip benchmark of data, model, and supervision. *ArXiv*, abs/2203.05796, 2022.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Karan Desai and Justin Johnson. VirTex: Learning Visual Representations from Textual Annotations. In *CVPR*, 2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423. URL `https://doi.org/10.18653/v1/n19-1423`.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL `https://openreview.net/forum?id=YicbFdNTTy`.

Constantin Eichenberg, Sid Black, Samuel Weinbach, Letitia Parcalabescu, and Anette Frank. Magma - multimodal augmentation of generative models through adapter-based finetuning. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

Xiaotao Gu, Liyuan Liu, Hongkun Yu, Jing Li, Chen Chen, and Jiawei Han. On the transformer growth for progressive bert training. In *NAACL*, 2021.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. *ArXiv*, abs/2110.04366, 2021.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Dan Hendrycks and Kevin Gimpel. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *ArXiv*, abs/1606.08415, 2016.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *ICML*, 2019a.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *ICML*, 2019b.

Xiaoshan Huang, Felipe Pérez, Jimmy Ba, and Maksims Volkovs. Improving transformer optimization through better initialization. In *ICML*, 2020.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.

Rabeeh Karimi Mahabadi, Sebastian Ruder, Mostafa Dehghani, and James Henderson. Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. In *Annual Meeting of the Association for Computational Linguistics*, 2021.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):664–676, 2017. doi: 10.1109/TPAMI.2016.2598339. URL https://doi.org/10.1109/TPAMI.2016.2598339.

Zaid Khan, B Vijaykumar, Xiang Yu, Samuel Schulter, Manmohan Chandraker, and Yun Raymond Fu. Single-stream multi-level alignment for vision-language pretraining. *ArXiv*, abs/2203.14395, 2022.

Jing Yu Koh, Ruslan Salakhutdinov, and Daniel Fried. Grounding Language Models to Images for Multimodal Generation, January 2023.

Olga Kovaleva, Saurabh Kulshreshtha, Anna Rogers, and Anna Rumshisky. Bert busters: Outlier dimensions that disrupt transformers. In *FINDINGS*, 2021.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *ArXiv*, abs/2104.08691, 2021.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 9694–9705. Curran Associates, Inc., 2021a. URL https://proceedings.neurips.cc/paper/2021/file/505259756244493872b7709a8a01b536-Paper.pdf.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *ArXiv*, abs/2201.12086, 2022.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *ArXiv*, abs/2301.12597, 2023.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, abs/2101.00190, 2021.

Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *ArXiv*, abs/2110.05208, 2021b.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. URL http://arxiv.org/abs/1405.0312.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *ArXiv*, abs/2110.07602, 2021.

Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *ArXiv*, abs/1711.05101, 2017.

Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. *ArXiv*, abs/2206.08916, 2022.

Kevin Lu, Aditya Grover, Pieter Abbeel, and Igor Mordatch. Pretrained transformers as universal computation engines. *arXiv preprint arXiv:2103.05247*, 2021.

Rabeeh Karimi Mahabadi, James Henderson, and Sebastian Ruder. Compacter: Efficient low-rank hypercomplex adapter layers. In *NeurIPS*, 2021.

Jack Merullo, Louis Castricato, Carsten Eickhoff, and Ellie Pavlick. Linearly Mapping from Image to Text Space, September 2022.

Yuval Netzer, Tao Wang, Adam Coates, A. Bissacco, Bo Wu, and A. Ng. Reading digits in natural images with unsupervised feature learning. 2011.

Hieu Pham, Zihang Dai, Golnaz Ghiasi, Kenji Kawaguchi, Hanxiao Liu, Adams Wei Yu, Jiahui Yu, Yi-Ting Chen, Minh-Thang Luong, Yonghui Wu, Mingxing Tan, and Quoc V. Le. Combined scaling for open-vocabulary image classification. 2021.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 2641–2649, 2015. doi: 10.1109/ICCV.2015.303.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL http://proceedings.mlr.press/v139/radford21a.html.

Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? In *NeurIPS*, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? *ArXiv*, abs/1902.10811, 2019.

Dan Rothermel, Margaret Li, Tim Rocktaschel, and Jakob N. Foerster. Don't sweep your learning rate under the rug: A closer look at cross-modal transfer of pretrained transformers. *ArXiv*, abs/2107.12460, 2021.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2015.

Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. *ArXiv*, abs/2112.06825, 2021.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herv'e J'egou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021.

Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 200–212. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/01b7575c38dac42f3cfb7d500438b875-Paper.pdf.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *ArXiv*, abs/1807.03748, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Jianfeng Wang, Xiaowei Hu, Zhe Gan, Zhengyuan Yang, Xiyang Dai, Zicheng Liu, Yumao Lu, and Lijuan Wang. Ufo: A unified transformer for vision-language representation learning. *ArXiv*, abs/2111.10023, 2021.

Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, 2022.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the transformer architecture. *ArXiv*, abs/2002.04745, 2020.

Jinyu Yang, Jiali Duan, S. Tran, Yi Xu, Sampath Chanda, Liqun Chen, Belinda Zeng, Trishul M. Chilimbi, and Junzhou Huang. Vision-language pre-training with triple contrastive learning. *ArXiv*, abs/2202.10401, 2022.

Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *ArXiv*, abs/2111.07783, 2021.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. *ArXiv*, abs/2111.07991, 2021.

Minjia Zhang and Yuxiong He. Accelerating training of transformer-based language models with progressive layer dropping. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/a1140a3d0df1c81e24ae954d935e8926-Abstract.html.

Xizhou Zhu, Jinguo Zhu, Hao Li, Xiaoshi Wu, Xiaogang Wang, Hongsheng Li, Xiaohua Wang, and Jifeng Dai. Uni-perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16783–16794, 2021.

6 APPENDIX

6.1 ADDITIONAL DATASETS

We conduct zero-shot classification experiments on three further datasets (Table 5): CIFAR-100 Krizhevsky (2009), SVHNNetzer et al. (2011), and ImageNet-AHendrycks et al. (2021). As CIFAR-100 and SVHN are both standard datasets, we only briefly describe them here. The CIFAR-100 dataset consists of 60k 32x32 colour images divided into 100 classes containing 600 images per class. Each class has 500 training and 100 test images, for a total of 50k training and 10k test images. We use the CIFAR-100 test set for the evaluations. SVHN is a harder version of MNIST Deng (2012), consisting of natural images of digits cropped from street-level pictures. We use the 26k test images for evaluation. ImageNet-A consists of *natural adversarial examples* from the ImageNet1k distribution, which are natural, correctly labeled images that classifiers incorrectly classify with high confidence. We use the 7k test images.

Table 5: Evaluation on additional zero-shot classification tasks. **First place** is in bold and second place is in red. LilT models are boxed in green. Acc-1 stands for top-1 accuracy, and Acc-5 is top-5 accuracy. Higher is better.

	Model			CIFAR100		SVHN		ImageNet-A	
	Configuration	# Trainable	% Trained	Acc-1	Acc-5	Acc-1	Acc-5	Acc-1	Acc-5
(a)	LilT-tiny	736.45 K	7.37	16.98	37.49	13.0	57.39	2.77	9.15
(b)	LiT-tiny	4.45 M	44.57	18.33	39.14	12.47	55.02	3.39	11.03
(c)	LilT-small	5.19 M	10.28	27.52	50.28	11.95	54.15	4.79	13.8
(d)	CLIP-tiny	9.99 M	100.0	18.74	41.1	**14.97**	**63.18**	2.73	10.49
(e)	LilT-base	14.65 M	7.51	29.9	53.77	11.84	57.08	5.11	15.8
(f)	LilT-large	25.92 M	4.06	**31.33**	**57.93**	7.39	42.21	**7.61**	**23.44**
(g)	LiT-small	28.73 M	56.98	26.88	47.17	12.3	59.17	5.37	16.01
(h)	CLIP-small	50.42 M	100.0	26.43	49.54	7.18	54.41	4.41	14.45
(i)	LiT-base	109.28 M	56.01	26.15	48.69	11.51	55.75	5.92	18.13
(j)	CLIP-base	195.13 M	100.0	25.25	50.93	9.47	53.33	4.68	16.41
(k)	VGG-19Hendrycks et al. (2021)	143M M	100.0	-	-	-	-	2.72	-
(l)	ResNet-50Hendrycks et al. (2021)	23 M	100.0	-	-	-	-	2.17	-
(m)	ResNet-101 Hendrycks et al. (2021)	44.7 M	100.0	-	-	-	-	4.9	-
(n)	ResNet-152Hendrycks et al. (2021)	60.4 M	100.0	-	-	-	-	5.2	-

6.2 NATURAL ADVERSARIAL EXAMPLES

Vision language models display impressive performance on ImageNet-A. ImageNet-A can be considered a "hard slice" of the ImageNet distribution, containing samples which are problematic for supervised classifiers. Suprisingly, the zero-shot classification performance of self-supervised vision-language models on ImageNet-A matches and is sometimes greater than the performance of supervised classifiers (ResNet-50 He et al. (2016) and VGG-19 Simonyan & Zisserman (2015)). This may be partially due to the parameter count — there are more total parameters in most of the vision-language models compared to the supervised CNNs. However, considering that the vision-language models are facing a harder problem (performing zero-shot classification), their performance relative to supervised CNNs is surprising.

6.3 WHERE DO THE MODELS FAIL?

On the SVHN dataset, performance is poor. The large models perform worse than random chance ($< 10\%$), and the smaller the model, the better it performs. One explanation could be that there is no way for the models to learn a correspondence between images of digits and the name of each digit, as nothing similar appears in the COCO training distribution, which only contains common objects.
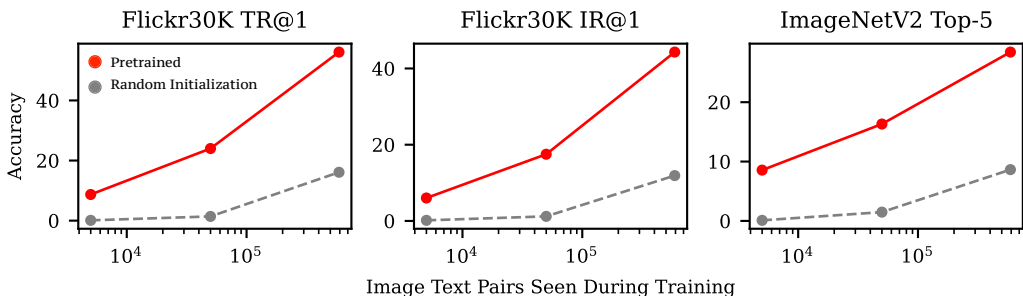
Figure 6: The effect of pretraining on model performance.

6.4 DOES PRETRAINING MATTER?

6.4.1 PRETRAINING VS. RANDOM INITIALIZATION

We follow the standard training procedure (§2.4) and train a CLIP-base model where both of the encoders are initialized randomly, instead of using weights initialized from unimodally pretrained models (DeIT Touvron et al. (2021) and SimCSE Gao et al. (2021)). We train three models, one for each dataset size. The results can be seen in Fig 6. Compared to the randomly initialized model, the pretrained model is substantially better across all three datasets and all 3 model sizes. However, it is likely that the benefit of unimodal pretraining will be diminished as the number of training pairs available for multimodal vision-language pretraining increases, although we do not explore this.

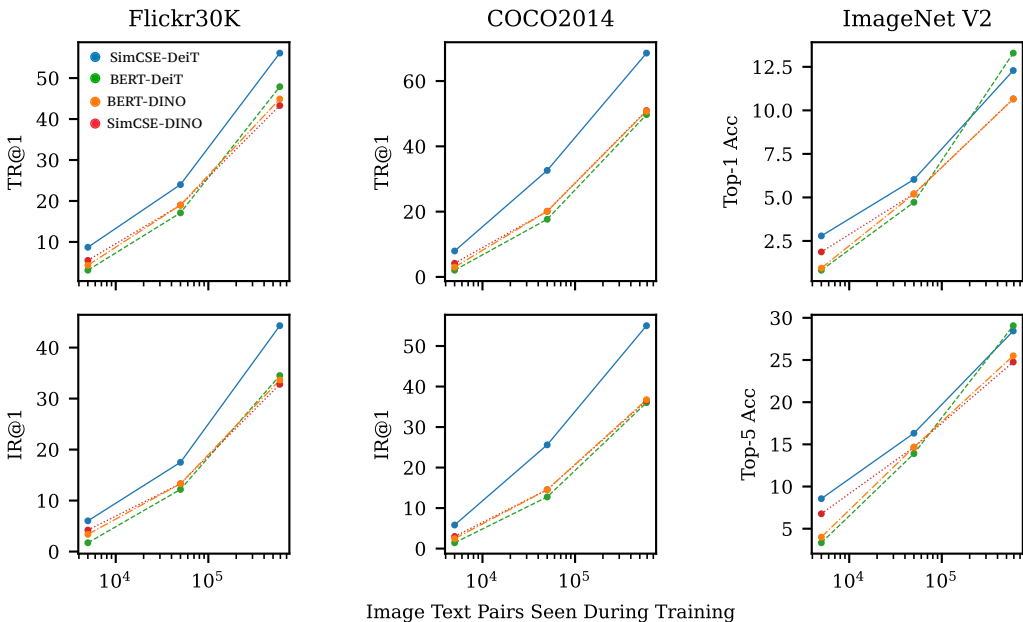### 6.4.2 DOES THE KIND OF UNIMODAL PRETRAINING MATTER?

Figure 7: A comparison of different kinds of pretraining on LilT performance. Each model is trained on 591k pairs.

We train LilT-base models with encoders initialized from different kinds of pretraining methods. For the text encoder, we choose between `bert-base-uncased` Devlin et al. (2019) and SimCSE Gao et al. (2021). For the image encoder, we choose between DeiTTouvron et al. (2021) and DINO Caron
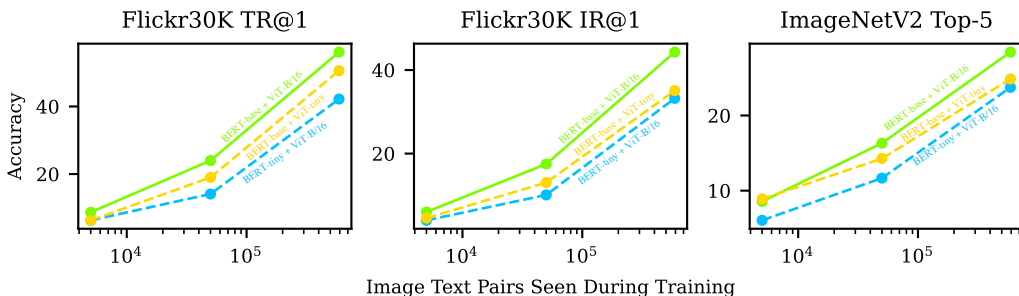
Figure 8: CLIP appears to be more sensitive to the size of the text encoder than the size of the image encoder.

et al. (2021). We train all models on 591k pairs following §2.4. The unimodal pretraining methods chosen do have an effect on the performance on the vision-language model. The combination of SimCSE and DeiT appears to be consistently better than other combinations, although on ImageNetV2, BERT-DeiT performs better.

6.5 ZERO-SHOT PROMPTS

Although CLIPRadford et al. (2021) uses a prompt ensemble, we use only a single prompt for all datasets except SVHN: `a photo of { }`. For SVHN, we use the prompt `a photo of the number { }`.

6.6 ENCODER SYMMETRY

Which encoder matters more? We train three configurations of CLIP on 5k, 50k, 591k pairs (Fig. 8). One is the symmetric CLIP-base, while the two asymmetric configurations have their text encoder and image encoder respectively replaced with the "tiny" version. Across all three dataset scales, the model with the smaller text encoder performs worse. Zhai et al. (2021) find that on large scale data (10m+ pairs), the opposite holds true — a larger image encoder is better than a larger language model.

6.7 DOES LILT WORK WITH SMALLER MODELS AND LESS DATA?

We test LilT and full-model training on smaller versions of transformers, corresponding to 'bert-base', 'bert-small', 'bert-tiny', and with decreasing amounts of image-text pairs (5k, 50k). The results are depicted in Figure 9 and Figure 10 for LilT$_D A$. There are no idiosyncratic results — as model size is decreased, performance decreases for both full model training and parameter efficient transfer learning. Similarly, as the amount of data decreases, performance also decreases. This holds true for all tested combinations of dataset size and model size.
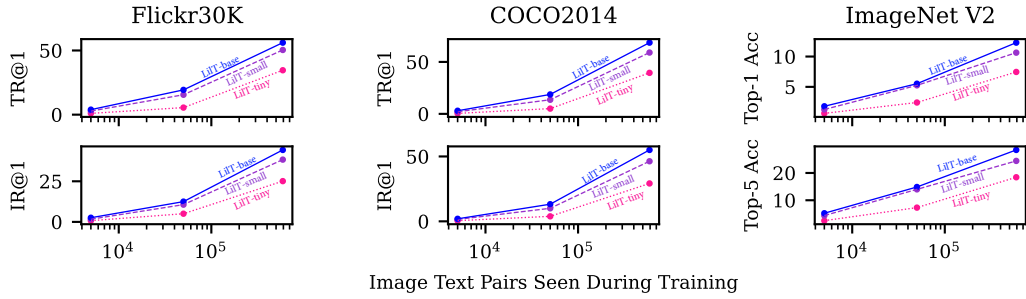
16

Figure 9: LilT's performance scales with increasing model size and dataset size — it is not limited to a specific model size or dataset size. LilT$_{DA}$ is pictured.
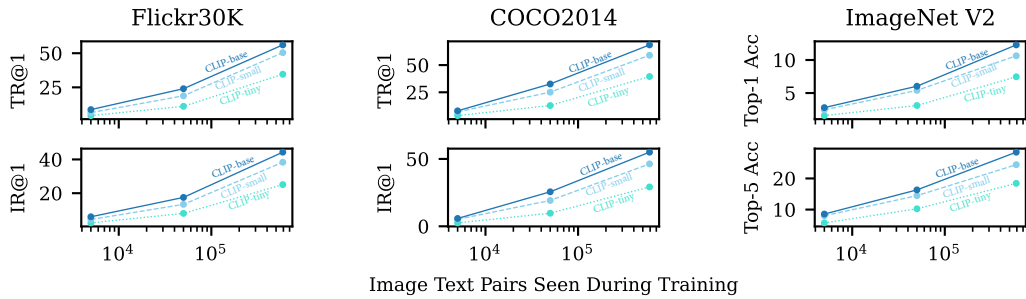

Figure 10: The performance of full-model training on smaller models and with less data.