CSCI 566 - Project Report: Interpretability of multi-modal neural models trained with contrastive self-supervised learning

Zizhao Hu
University of Southern California
zizhaoh@usc.edu

Ravikiran Chanumolu
University of Southern California
chanumol@usc.edu

Shravan Vasista
University of Southern California
vasista@usc.edu

Sumeet Shirgure
University of Southern California
sshirgur@usc.edu

1 Introduction

Deep learning has seen tremendous growth in the last decade. At the core of this remarkable progress lies the rapid evolution of GPUs and the availability of large-scale datasets, which allows for accelerated training of deep models at scale. Traditional deep learning methodologies, often supervised, require large amounts of labelled samples. However, with recent models growing into trillions of parameters, even this amount of labelled data is not enough to mimic learning. With that in mind, self supervised learning (SSL) has experienced a rise in popularity in recent years.

SSL is a promising direction for the future of deep neural networks. In this paradigm, the fundamental idea is to obtain supervisory cues from the data itself. Since the structure of the data itself is a form of supervision, the model can make use of a variety of signals across correlated modalities like audio, video and text, while not relying on labels. The most common way to train such models is by occluding parts of the input and training a machine to predict the missing pieces.

As aptly noted in (fac, 2021), there is a unified view of thinking about self-supervised methods in terms of energy based models (EBMs). This generalizes the idea of a loss function to arbitrary inputs. The crux of EBMs is in learning an energy function $E(x, y)$ on pairs of inputs (x, y) where x is visible to the machine, while y isn't. $E(x, y)$ should be low for compatible predictions, y, and should be high for incompatible predictions. Only training the model on compatible pairs causes E to be small everywhere – a phenomenon known as collapse. The two most popular ways of avoiding collapse are (a) regularization, and (b) contrastive learning.

A familiar example of regularized self-supervised models are the variational autencoders (VAEs) (Kingma and Welling, 2013) that we encountered in our homework assignment. Penalizing large divergences between the latent distribution from the standard Gaussian is a form of regularization. What we're interested in in this paper are contrastive methods. The idea is to artificially construct incompatible pairs (x, y) and force the corresponding energy $E(x, y)$ to be large. The familiar example in this case is our other homework assignment, where we trained language models like BERT (Devlin et al., 2019) by randomly corrupting parts of input text and penalizing incorrect predictions.

CLIP (Contrastive Language–Image Pre-training)(Radford et al., 2021), a multi-modal network that is trained in a self-supervised fashion, is of special interest and important to this paper. Many multi-modal architectures have been introduced recently (Su et al., 2019) (Lu et al., 2019a). However, CLIP is the only multi-modal network where we can analyse the vision and text modules separately post pre-training on joint inputs. This gives us an opportunity to study the inductive bias of effcts of multi-modal training on single modality tasks. The authors of the paper show that the pre-trained vision encoder in CLIP shows high zero-shot capability on ImageNet (Deng et al., 2009) which is an image-only dataset. This inspired us to question the impact of image-text training on language-only tasks.

Our work also invloves interpretability of deep neural networks (DNN). While DNN's have exceptional performance, these models sometimes act like black-boxes. A lot of work is done on understanding how such models make their predictions. The difficulty in mathematically defining human interpretability itself makes the task more challenging. One way of understanding the models better is by studying the internal representations generated by these models on real world data like text and images. Recent work in this direction includes visualizing the internal representations for image data. See section 2.3 for more details. Another interesting approach taken by works like (Manning et al., 2020) is to probe the properties of neural networks like BERT by evaluating the model (maybe after augmenting it with a small perceptron, or even just a single linear layer) directly on downstream tasks in a zero-shot manner. The authors of CLIP (Radford et al., 2021) follow a similar approach. See section 2.4 for more details.

This report starts by discussing prior work most relevant to our project in section 2. We discuss experimental details in section 3. Conclusions and results are discussed in section 4.

2 Related works

2.1 Visual Language Intelligence

(Li et al., 2022) paper presents a comprehensive survey of the evolution of vision-language (VL) intelligence over time. The paper summarizes the progress in both computer vision and natural language processing, and recent trends shifting from single modality processing to multiple modality comprehension. The authors categorize the development into three time periods 1) task-specific 2) vision-language pre-training (VLP), and 3) larger models empowered by large-scale weakly-labeled data.

Real-world problems often involve multiple modalities, example, autonomous driving that require multi-modal perception. Furthermore, multimodality learning benefits single modality, for example, language learning needs perception which forms the basis of many semantic axioms (Bisk et al., 2020).

In noticing that multi-modal perception helps in both multi-modal and single-modal tasks, there came a lot of research work. Within the field of multi-modality, the integration of vision and language gets much attention owing to the facts that vision is one of the most important perceptions for a human to understand the environment and language-aligned visual features greatly improves the performances of both vision and vision-language tasks. Additionally, the availability of abundant datasets and benchmarks in this field has increased the popularity of vision-language intelligence. Some of the problems are Image captioning, Visual Question Answering (VQA), image-text matching, etc. We discuss the deep learning models for these tasks in the next Section.

2.2 Multi-modal Machine Learning

Multi-modal machine learning has been a hot research area in recent years and among the possible combinations of modalities, image and language has been the most popular one. The first era VL methods were designed for specific tasks, (Vinyals et al., 2015) integrated a CNN image encoder and an RNN text decoder for image captioning. (Antol et al., 2015); (Yang et al., 2016); (Anderson et al., 2018) addressed the VQA task by mapping images and texts into the same latent space and predicting answers from the latent representations. (Kiros et al., 2014); (Karpathy et al., 2015); (Huang et al., 2021); (Lee et al., 2018) performed image-text matching by calculating the similarity between an image and a text either on sentence-level or token-level. Although the models for different tasks varied significantly, they all followed a similar trajectory shown in Fig.1. The main differences are the granularity of the visual representation and the way of fusing vision and language features.

The prevalence of pre-training and fine-tuning in both language (Devlin et al., 2018) and vision, and the advancement and success of transformer models(Vaswani et al., 2017) in both language and vision tasks, the interdisciplinary field of vision and language embraced a new era: to learn a joint representation of vision and language by pretraining on image-text pairs. Many recent studies (Li et al., 2019); (Lu et al., 2019b); (Zhang et al., 2020); (Yu et al., 2018); (Mikolov et al., 2013); (Yu et al., 2020); (Chen et al., 2020) adopted BERT-like (Devlin et al., 2019) architectures and training methods. The development of VL learning meets a serious challenge due to the lack of sufficiently large scale manually labeled data. Recently, (Radford et al., 2021); (Jia et al., 2021), (Wang et al., 2021), broke this limitation by adopting contrastive learning and making use of large-scale web-crawled data to learn visual-linguistic features which can be used for zero-shot learning.

In this paper, we focus on CLIP and examine its multi-modal learning capabilities through a set of experiments discussed in Sections 3 and 4. The core idea of CLIP is the training method. Instead of training to predict masked visual or textual tokens as in other VLP methods, CLIP learns to recognize paired image and text. Given a batch of N (image-text) pairs, the goal is to predict which of the N × N possible pairs are matched pairs (positive samples) and which are unmatched pairs (negative samples). After pre-training, CLIP can perform zero-shot image classification by using phrases such as ”a photo of” plus a category name as prompts to tell the model which categories an input image is the most similar to. Compared with fully supervised baselines, zero-shot CLIP outperforms the baseline on 16 of 27 datasets.

2.3 Understanding Internal Representations

Recent advances in deep learning have had a tremendous impact on common sense understanding of our world by machines. And since deep neural networks often have vaguely interpretable inner workings, a significant amount of research has been done on understanding it. One line of work in this regard has focused on visualization of parameters and/or inputs of these networks. E.g (Karpathy et al., 2015), (Olah et al., 2017), (Zeiler and Fergus, 2014) and (Simonyan et al., 2014). The work done in (Yosinski et al., 2015) (Nguyen et al., 2016) and others showed that neurons can also be multifaceted in the sense that they could fire in response to multiple types of features.

Such behaviour was also observed by (Quiroga et al., 2005) in neurons in the human brain. This idea has also been, to some extent, generalized by (Goh et al., 2021) to multi-modal settings, where the activation of neurons at an individual level has been studied under varying modes of stimuli ranging from vision to text.

2.4 Understanding Internal Representations Through Probing Tasks

Related to the works of (Manning et al., 2020), we looked at several studies on BERT’s internal representations that use structural probing. Some examples in this line of work are (Jawahar et al., 2019) and (Conneau et al., 2018b). We follow the methods introduced in (Jawahar et al., 2019) in our experiments for understanding the internal representations of transformer-based models. While said paper investigates BERT trained on just text corpora, we try to extend this work to transformers trained with multimodal media, like CLIP.

3 Experiments

Our experimental methods were two-fold. One direction was to qualitatively assess the internal representations of deep neural models trained with multi-modal self supervised training. The way we did that was by creating visualizations of these representations using standard methodologies. We are interested in the effect of multimodal training, and we believe that creating such visualizations for similar architectures, that only differ in training objectives, could be good way to gauge this effect. In this regard, because of the differences in their architecture, our experiments are divided into two subcategories – (a) visualizing features in convolutional models, and (b) visualizing attention maps in vision transformers.

More importantly, our other set of experiments are focused on quantitatively evaluating multimodal models on downstream tasks, through probing methods. The general idea here is to evaluate multi-modal models – namely CLIP encoders, on various tasks in a zero-shot manner. This is why CLIP is invaluable to us; it consists of a vision encoder and a text encoder that were trained jointly with contrastive learning. But the two models can be evaluated separately, as we do in the following.

3.1 Vision Encoder: Feature Visualization

This section explains the method that specially applies to the CLIP model with ResNet-50 vision encoder. This method is based on feature visualization method used in (Goh et al., 2021). This is done by sending a random generated image to the network and optimize the input image to maximize the activation of the neurons selected. We selected neurons in the same layers of the parallel models CLIP-ResNet-50 vision encoder and Vanilla ResNet-50 as the target neurons. The values of these neurons are added and back-propagated through the network to change the input image. The trained input image will be a visualization of the neurons we selected. We hope this parallel visualization comparison can give us some insights the difference between the two pretraining objectives.

3.2 Vision Encoder: Attention Map Visualization

This section explains the method that specially applies to the CLIP model with vision transformer vision encoder. Specifically, we looked at the attention maps generated on the same architecture introduced in (Kolesnikov et al., 2021), namely ViT-B/16, that are trained using different methodologies. (a) (Kolesnikov et al., 2021) train ViT for ImageNet classification in a supervised setting, (b) (Caron et al., 2021) train the network in a self-supervised manner for image representation learning (DINO) and (c) (Radford et al., 2021) train their transformer vision model using language-image pretraining (CLIP). The goal is to observe the similarities and the differences between similar architectures trained in supervised vs. self-supervised vs. multi-modal settings.

3.3 Vision Encoder: Probing

We look at 2 probing methods to understand the utility of multi-modal training on vision tasks. The first method is using a MLP probe to evaluate the zero-shot prediction accuracy of models on CIFAR-100 dataset, using 4 different intermediate learned representations across 3 models - CLIP, ResNet-50 and ViT. The second method is using a convolutional decoder probe to evaluate the reconstruction loss in the same manner.

3.4 Text Encoder: Probing

In this section we will look at probing methods used in (Jawahar et al., 2019) to understand the utility of multi-modal training on language only tasks. We evaluate the performance of CLIP against GPT2 and BERT on the probing tasks. GPT2 and BERT are both pretrained using language modelling with textual-only data. Therefore, to comment on the benefits of multi-modal training we use our experimental setup to evaluate if CLIP can outperform these models.

3.4.1 Phrase Level Probing

We start with phrase level probing using CoNLL 2000 chunking dataset. In this dataset for each sentence the constituent phrases are labeled with grammatical tags. Using this task, in (Jawahar et al., 2019) the authors find that in BERT, the lower layers are significantly better than the higher layers in capturing phrase level information. Here, higher layers are closer to the output. We repeat this task on CLIP to understand if CLIP objective shows a similar trend. Following (Jawahar et al., 2019), to find a representation for a phrase P tokenized as $(w1, w2, ..wj)$, we first find the hidden representations for each token $(h1, h2, ..hj)$. The final representation for the phrase P is obtained by concatenating $h1$, hj and their element product and difference. We then use these phrase level representations to to perform k-means clustering with k set to number of chunk tags in the dataset. Finally, we look at how well these clusters align with the actual chunk tags. The alignment is measured using the Normalised Mutual information (NMI) score between cluster labels and chunk tags.

3.4.2 Sentence Level Probing

We use the sentence level probing tasks introduced in (Conneau et al., 2018a) to compare BERT, GPT2 and CLIP in predicting a variety of linguistic properties. To achieve this, the embeddings of the three models are frozen and used to train an auxiliary classifier to predict the linguistic property of interest. We use an MLP as the auxilary classifier. The probing tasks are split into three categories described below.

Surface level probing: This category of tasks test if the model embeddings captures the surface level properties of the input text. Here, the tasks can be solved by simply looking at tokens in the input sentences while ignoring grammar and meaning. The task we look at in this category is to predict the sentence length i.e. count number of words in the input sentence. The task is framed as classification task by discretizing length in to 6 equal-width bins.

Syntactic probing: The second category of tasks test if the model embeddings captures the syntactic properties of the input text. The first task in this category, bigram shift (BShift), tests if the model is sensitive to the order of words in the sentence. In this task, the word order is corrupted by randomly swapping two adjacent words and the model must classify each sentence as correct or corrupted. The next task, tree depth (TreeDepth) tests if the embeddings capture any hierarchical information in the constituency parse of the sentence. The model needs to predict the length of the longest path in the constituency parse tree. This task is posed as a 8-way classification task with path lengths ranging from 5 to 12. The last task in this category, top constituents (TopConst), also looks at the constituency parse. TopConst requires the model to predict the top constituents immediately below the sentence (S) node in the parse tree. This task is framed as a 20-way classification problem. Each class represents a unique set of legal top constituents.

Semantic probing: These tasks also require the model to capture meaning in input text. These include tense which is to predict the tense of the main clause verb. The two other tasks are

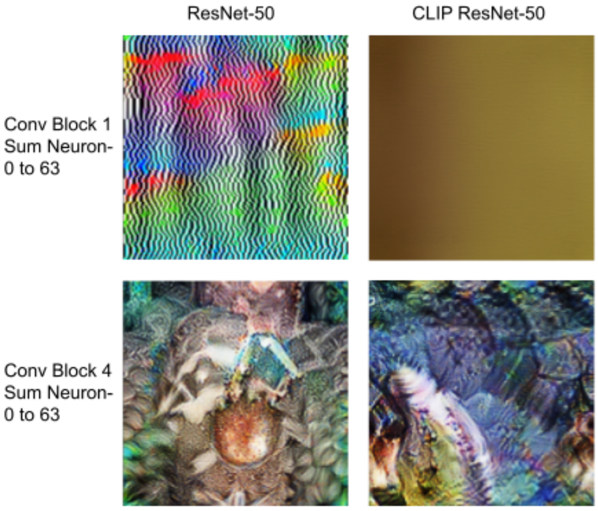


Figure 1: Rows 1 and 2 contain layer level visualisations for convolution blocks 1 and 4. Columns 1 and 2 correspond to ResNet-50 and CLIP ResNet-50 respectively.

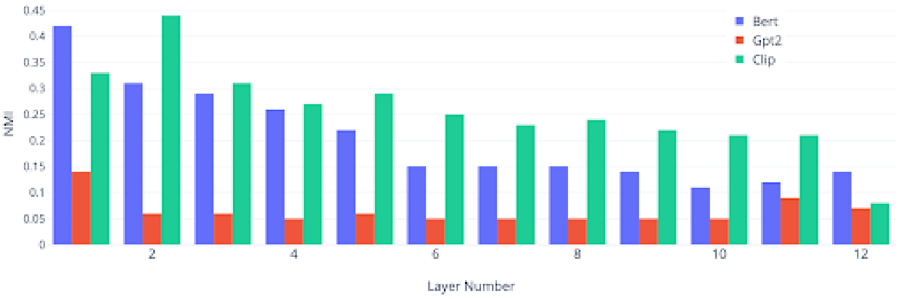


Figure 2: Bar graph summarising the results on the phrase level probing task. The x-axis shows the layer used for getting the embeddings. The y-axis shows the NMI score.

subject number (SubjNum) and object number (ObjNum) which require the model to predict whether the subject and object in a sentence is singular or plural. Another task in this category is the semantic odd man out (SOMO) task, where the authors corrupt a sentence by replacing a noun n and verb v pair with a different (n, v) pair. The model must differentiate corrupted sentences from correct sentences. And the last task in this category is coordination inversion (CoordInv), where the authors corrupt the sentences by inverting the clauses in a sentence made of two coordinate clauses. All the tasks in this category are binary classification tasks.

3.5 Text Encoder: Evaluating On Superglue tasks

The probing tasks described above are very useful as a diagnostic tool to test a variety of lingustic phenomena. However, they are not representative of the real-world tasks which are significantly more complex. Therefore, in addition to the probing tasks, we also evaluated CLIP on SuperGlue tasks. More specifically, we focus on following 2 tasks in SuperGlue.

3.5.1 BoolQ

BoolQ (Clark et al., 2019) is a question answering dataset where there are only two possible answers yes/no for all questions. Each example in BoolQ is a triplet (question, passage and answer). Therefore using the passage as the context, the model must output either yes/no based on the question. Actual queries to the Google search engine are anonymized and used as questions in the dataset. Heuristic methods are used to identify queries that are likely to be yes/no questions. [1]

3.5.2 Words in Context (WIC)

Words can have different meaning based on the context. A model's task on WIC (Pilehvar and

[1] Click here for sample rows in the BoolQ dataset.

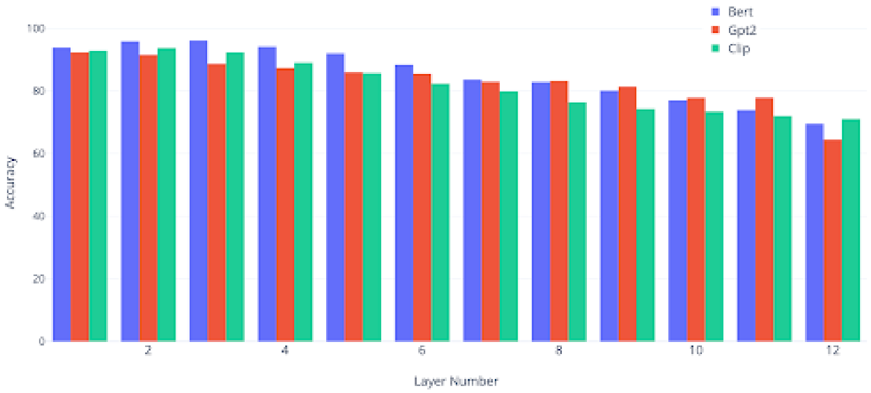


Figure 3: Bar graph summarising the results on the surface level probing task (SentLen). The x-axis shows the layer used for getting the embeddings. The y-axis shows the accuracy.

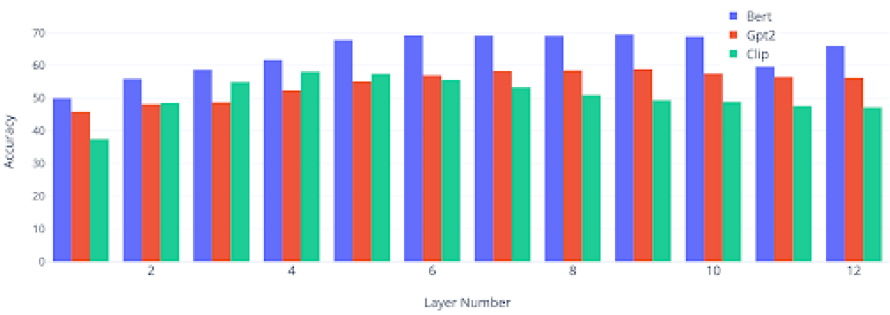


Figure 4: Bar chart summarising the results on the syntactic probing tasks. Here we show the average accuracy across the syntactic probing tasks. The x-axis shows the layer used for getting the embeddings. The y-axis shows the average accuracy across all syntactic probing tasks.

Camacho-Collados, 2018) is to find the intended meaning of the words based on the provided context. WIC is a binary classification task. Each example in WIC contains two sentences and a target word. The model need to output a binary label based on whether the meaning of the word in the two sentences is same or different. [2] In order to have a better coverage of different task categories we select the above two datasets. BoolQ is a sentence level classification dataset while WIC is a token level classification dataset. We tried to evaluate CLIP model on more superGlue tasks. However, CLIP limits the input sequence length to 77 as it was trained using short image captions. This lead to heavy overfitting on tasks where input sequences were very long as truncating the sequences resulted in significant loss of information.

4 Results

4.1 Vision Encoder: Feature Visualization

Figure 1 shows the feature visualization. As expected lower layer(Row1) has more general visualizations compared to high layer(Row 2). We also found there are significant difference between CLIP ResNet and vanilla ResNet in lower layers. Since CLIP ResNet tends to learn only simple color gradient, while Vanilla ResNet learns texture and more complex color patterns. In higher layers CLIP ResNet tends to have less sharp edges and more gradient colors, while Vanilla ResNet learns more distinct objects and colors.

4.2 Vision Encoder: Attention Map Visualization

As we can see in figure 9, the self-supervised image-only training have good, interpret-able attention maps in the form of segmentation maps. Note that the attention maps in figure 9(b) are generated by averaging over the heads in a transformer layer, while the rest of them correspond to a *single attention head* in a given layer. This suggests better, more-interpretable internal representations being generated by self-supervised methods in essentially the same architecture. As for the difference between multi-modal and image-only training, we find that there is no reasonable difference in the

[2] Click here for sample rows in the WIC dataset.

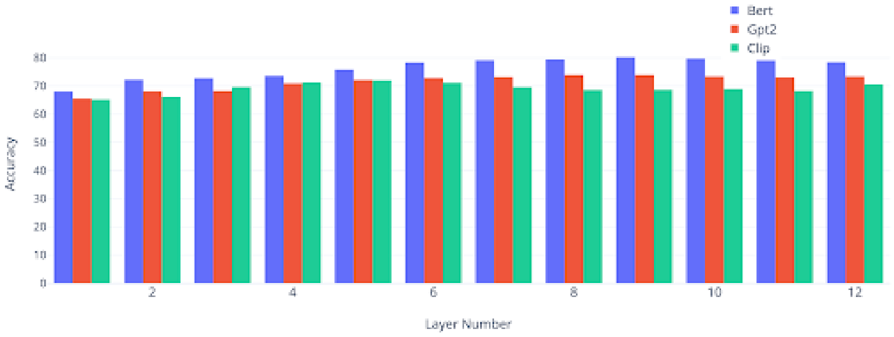


Figure 5: Bar chart summarising the results on the semantic probing tasks. Here we show the average accuracy across the semantic probing tasks. The x-axis shows the layer used for getting the embeddings. The y-axis shows the average accuracy across all semantic probing tasks.

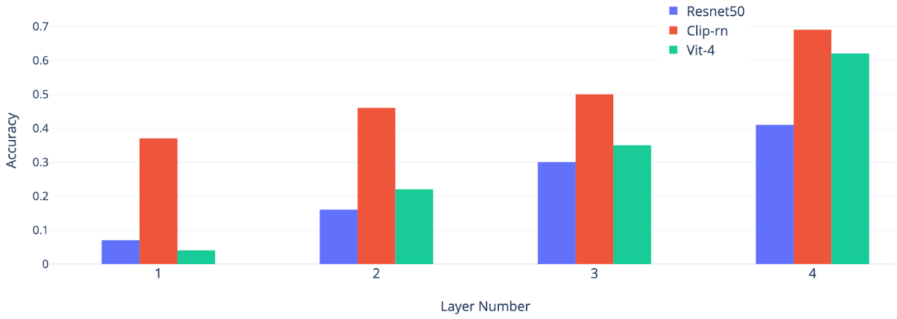


Figure 6: Prediction Accuracy on 4 intermediate layers in 3 models

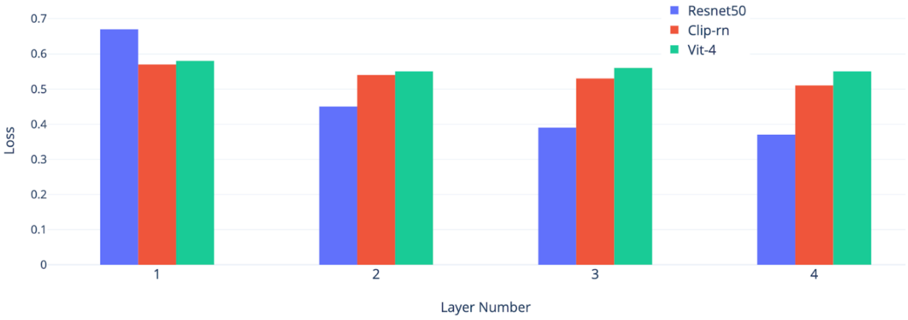


Figure 7: Reconstruction loss on 4 intermediate layers in 3 models

kinds of segmentation maps.

4.3 Vision Encoder: Probing

We probed a vanilla ResNet-50 and a ViT with similar amount of parameters and building blocks to provide parallel comparison with the CLIP ResNet-50. From figure 6, we see an increase in classification accuracy in all three models when we use deeper embedding representations. But clip has significantly higher accuracy in lower layers. This is in-line with the clip zero-shot capability found in the original paper. From figure 7, we can see the reconstruction capability of vanilla ResNet-50 is higher than both CLIP and ViT. Although CLIP is using ResNet-50 for its encoder, it shows no obvious reconstruction capability. This implies some general information loss during the contrastive learning process.

4.4 Text Encoder: Probing

4.4.1 Phrase Level Probing

Figure 2 shows the results for phrase level probing task for BERT, GPT2 and CLIP. As discussed in (Jawahar et al., 2019), we see that the performance in the case of BERT degrades from lower to higher layers. We observe a similar trend in GPT2. However, the NMI score across all layers is lower than BERT. One explanation for this observation could be that BERT uses bidirectional context while GPT2 only uses left context for generating token representations. We see that the NMI score for CLIP is higher across all layers compared to GPT2 and even higher than BERT except the last layer where see a sharp drop. Nevertheless, CLIP has the overall best performance showing that even without the LM pretraining the performance of CLIP text encoder is comparable to that of BERT.

4.4.2 Sentence Level Probing

From figure 3 we see that for surface level tasks the accuracy of CLIP is comparable to GPT2 and BERT and the trend of the performance of all the three models is similar. For syntactic and semantic probing tasks we only show the average performance across all tasks in the respective category for concise presentation of results. In figure 4 we see that in the case of syntactic probing tasks the performance of CLIP is lower than BERT across all layers and lower than GPT2 in the later layers. Further the trend in the performance of CLIP deviates from other models as there is no improvement in accuracy after the 6th layer. From figure 5 the trend of the results on semantic tasks are very similar to the sytanctic tasks. So we can conclude that the performance of CLIP is significantly lower than GPT2 and BERT in higher layers on syntactic and semantic sentence level probing tasks.

4.5 Text Encoder: Downstream task performance

From the table 1, we see that the performance of CLIP is approximately 4 points lower than GPT2 and 7 points lower than BERT. This further bolsters our previous findings using probing setup that CLIP is not suitable for sentence level tasks. However, in table 1 it is interesting to see that performance of CLIP is better in comparison to GPT2 when evaluated on WIC dataset which as discussed in a previous section is a token level classification dataset. This observation is inline with the results we obtained on the phrase level probing task where CLIP outperformed GPT2. But, BERT beats CLIP on WIC by a large margin.

Model	BoolQ	WIC
BERT	69.24	68.51
GPT2	66.47	52.68
CLIP	62.34	58.04

Table 1: Comparing model accuracy on BoolQ and WIC datasets.

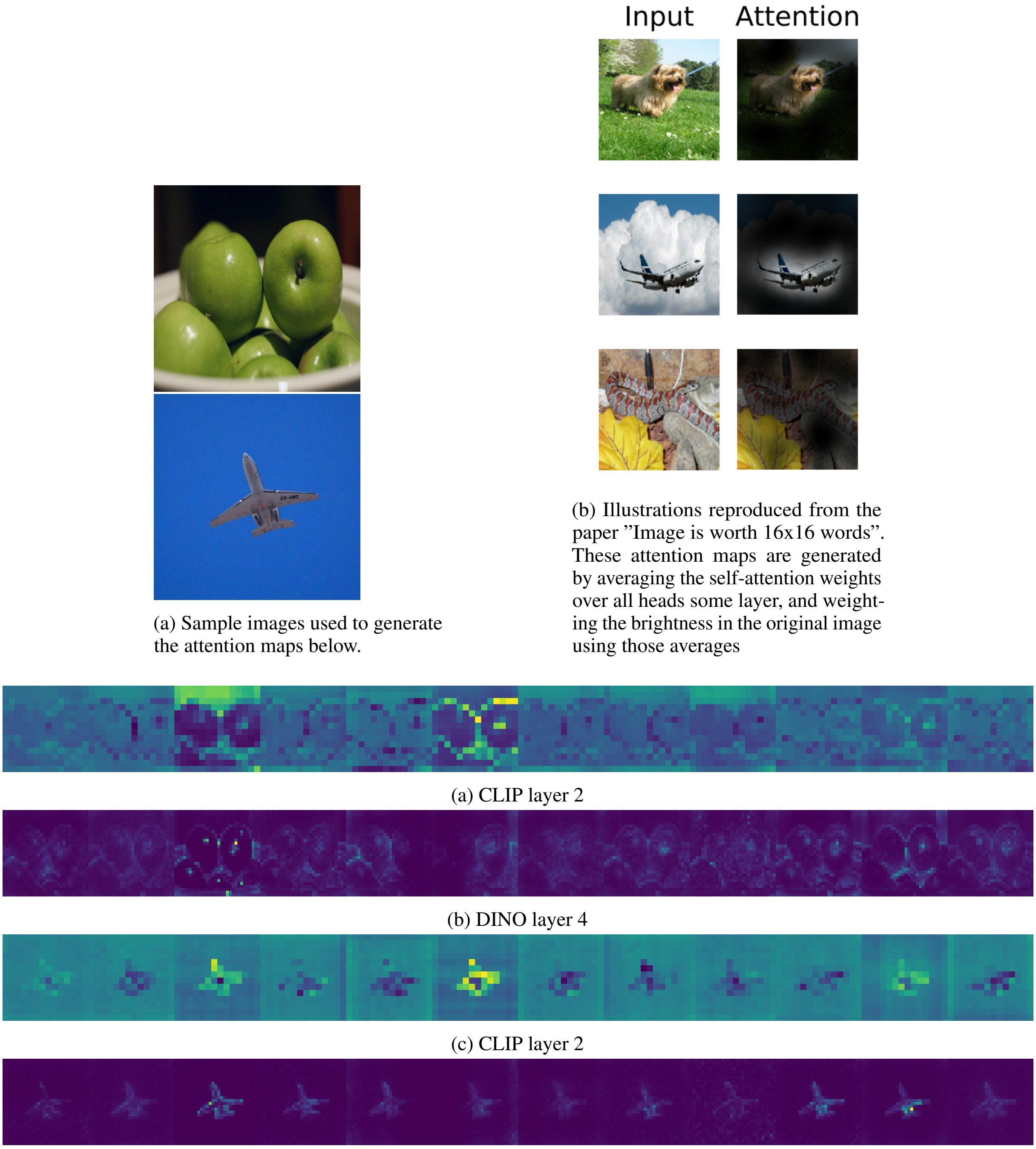
5 Conclusion

From the experiments results we have seen from the previous section, we have several observations on the single-modal performance of CLIP, compared with other popular single-modal models. CLIP was first developed as a potential architecture that can bridge language and vision domains and boost the performance of tasks in both domains, and at the same time take advantage of the zero-shot capability provided by prompt engineering in the language domain. However, we observed that to achieve this, CLIP does not learn better or more general representations in either the language or the vision domain. For image classification and phrase Level probing, the tasks have classification nature. And contrastive learning objective suits this type of task well since the best classification models tend to maximize the cross-class discrepancies rather than generalize a single class well. This explained the better performance in all of CLIP's intermediate layers compared to the single-modal counter parts in these tasks. However, when we look at more complex tasks that require the model to have more general knowledge of the domain, such as regeneration task in vision domain and sentence level understanding in language domain, CLIP fails to learn a better representation compared to BERT, GPT, and ResNet. The reason behind this might be that CLIP lacks the single-modal pretraining and focuses only on the cross-modal contrastive pretraining.

These observations suggest that multi-modal/cross-modal contrastive learning gives models cross-modal zero-shot capability, but fails to achieve better single-modal generalization performance. For future design of more general multi-modal architectures, finding a way to balance the cross-modal constrastive objective, and single-modal pretraining objective might give us more capable and general multi-modal models.

6 Work Distribution

Zizhao: Vision Task Experiments - Visualization, Probing. **Ravikiran**: Language Task Experiments - Probing and Downstream tasks. **Shravan**: Language Task Experiment - Probing and Downstream tasks. **Sumeet**: Vision Task Experiments - Visualization, Method Proposing. The four of us had equal contribution in paper presentation, proposal, midterm report, final presentation and report.



(a) Sample images used to generate the attention maps below.

(b) Illustrations reproduced from the paper ”Image is worth 16x16 words”. These attention maps are generated by averaging the self-attention weights over all heads some layer, and weighting the brightness in the original image using those averages

(a) CLIP layer 2

(b) DINO layer 4

(c) CLIP layer 2

(d) DINO layer 4

Figure 9: Corresponding self attention maps in transformers. The 12 different maps correspond to the attention heads at the earlier layers of the aforementioned models. Specific heads at certain layers seem to behave like edge detectors. As explained in the paper that introduces DINO, the attention heads in the later layers also seem to attend to different parts of the input image.

References

2021. Self-supervised learning: The dark matter of intelligence. *Self-supervised learning: The dark matter of intelligence*.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2425–2433.

Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, et al. 2020. Experience grounds language. *arXiv preprint arXiv:2004.10151*.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018a. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. *arXiv preprint arXiv:1805.01070*.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018b. What you can cram into a single vector: Probing sentence embeddings for linguistic properties.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Gabriel Goh, Nick Cammarata †, Chelsea Voss †, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. 2021. Multimodal neurons in artificial neural networks. *Distill*. Https://distill.pub/2021/multimodal-neurons.

Zhicheng Huang, Zhaoyang Zeng, Yupan Huang, Bei Liu, Dongmei Fu, and Jianlong Fu. 2021. Seeing out of the box: End-to-end pre-training for vision-language representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12976–12985.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does bert learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR.

Andrej Karpathy, Justin Johnson, and Li Fei-Fei. 2015. Visualizing and understanding recurrent networks.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Alexander Kolesnikov, Alexey Dosovitskiy, Dirk Weissenborn, Georg Heigold, Jakob Uszkoreit, Lucas Beyer, Matthias Minderer, Mostafa Dehghani, Neil Houlsby, Sylvain Gelly, Thomas Unterthiner, and Xiaohua Zhai. 2021. An image is worth 16x16 words: Transformers for image recognition at scale.

Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. 2018. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216.

Feng Li, Hao Zhang, Yi-Fan Zhang, Shilong Liu, Jian Guo, Lionel M Ni, PengChuan Zhang, and Lei Zhang. 2022. Vision-language intelligence: Tasks, representation learning, and large models. *arXiv preprint arXiv:2203.01922*.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019a. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, pages 13–23.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019b. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Christopher D. Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Anh Mai Nguyen, Jason Yosinski, and Jeff Clune. 2016. Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks. *CoRR*, abs/1602.03616.

Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. 2017. Feature visualization. *Distill*. Https://distill.pub/2017/feature-visualization.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2018. Wic: the word-in-context dataset for evaluating context-sensitive meaning representations. *arXiv preprint arXiv:1808.09121*.

R Quian Quiroga, Leila Reddy, Gabriel Kreiman, Christof Koch, and Itzhak Fried. 2005. Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. 2021. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 21–29.

Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. 2015. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*.

Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie-vil: Knowledge enhanced vision-language representations through scene graph. *arXiv preprint arXiv:2006.16934*.

Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. 2018. Mattnet: Modular attention network for referring expression comprehension. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1307–1315.

Matthew D. Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *Computer Vision – ECCV 2014*, pages 818–833, Cham. Springer International Publishing.

Chao Zhang, Zichao Yang, Xiaodong He, and Li Deng. 2020. Multimodal intelligence: Representation learning, information fusion, and applications. *IEEE Journal of Selected Topics in Signal Processing*, 14(3):478–493.