

---

# **How does CLIP process negation? A multimodal interpretability study**

---



**Utrecht  
University**

Vincent Quantmeyer

1176153

13<sup>th</sup> February 2024

A thesis submitted for the degree of

MASTER OF SCIENCE

in Artificial Intelligence

First supervisor: Prof. Albert Gatt

Second supervisor: Prof. Pablo Mosteiro

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Related work</b>	<b>4</b>
2.1	Transformer architecture . . . . .	4
2.1.1	Text transformer . . . . .	5
2.1.2	Vision transformer . . . . .	6
2.2	Vision-and-language models . . . . .	6
2.3	Vision-and-language benchmarks . . . . .	8
2.4	Model interpretability and localisation . . . . .	9
<b>3</b>	<b>Methods</b>	<b>11</b>
3.1	Definitions . . . . .	11
3.2	Dataset pre-processing . . . . .	11
3.2.1	Rephrasing of labels . . . . .	12
3.2.2	Data segmentation . . . . .	12
3.3	Preliminary analyses . . . . .	12
3.4	Causal tracing in text encoder . . . . .	13
3.5	Negator-selective attention in text encoder . . . . .	15
3.5.1	Validation on the CANNOT dataset . . . . .	16
3.6	Image ablation in image encoder . . . . .	16
<b>4</b>	<b>Results</b>	<b>17</b>
4.1	Preliminary analyses . . . . .	17
4.2	Causal tracing in text encoder . . . . .	18
4.3	Negation-selective attention in text encoder . . . . .	20
4.3.1	Validation on the CANNOT dataset . . . . .	21
4.4	Image ablation in image encoder . . . . .	21
4.4.1	Multi-head attention (MHA) . . . . .	21
4.4.2	Multi-layer perceptron (MLP) . . . . .	23
4.4.3	Comparison of the role of MHA and MLP . . . . .	24
<b>5</b>	<b>Discussion</b>	<b>24</b>
5.1	Localisation patterns in text encoder . . . . .	24
5.2	Localisation patterns in image encoder . . . . .	26
5.3	Causes of CLIP's moderate classification performance . . . . .	27
<b>6</b>	<b>Conclusion</b>	<b>28</b>
<b>References</b>		<b>29</b>

## List of Figures

1	Examples from VALSE existence . . . . .	9
2	Original causal tracing methodology . . . . .	10
3	Distribution of CLIP’s classification score on VALSE existence . . . . .	13
4	Causal tracing methodology adopted for VALSE existence . . . . .	14
5	Image ablation methodology adopted for VALSE existence . . . . .	16
6	Caption-foil similarity vs. CLIP’s classification score . . . . .	17
7	Subject size vs. CLIP’s classification score . . . . .	18
8	Causal tracing results . . . . .	19
9	Negator-selective attention results . . . . .	20
10	Source of negator-selective attention in layer 4 . . . . .	21
11	Validation of negator-selective attention on the CANNOT dataset . . . . .	22
12	MHA image ablation results . . . . .	22
13	MLP image ablation results . . . . .	23
14	Comparison of MHA and MLP image ablation results . . . . .	24

## List of Tables

1	CLIP’s accuracy on the original and rephrased VALSE existence benchmark . .	13
2	Number of instances per segment in the VALSE existence dataset . . . . .	13
3	Correlation results from preliminary analyses . . . . .	18

## **Abstract**

Various benchmarks have measured linguistic capabilities of vision-and-language (VL) models, but do not provide insights into how models implement these capabilities. This thesis translates model interpretability techniques developed for large language models to the multimodal space in order to investigate the mechanisms involved in CLIP's processing of negation. In the text encoder, specific negator-selective attention heads are found that seem crucial in controlling the movement of negation-related information through the model. Early evidence suggests that these heads are dataset-independent. In the image encoder, MLPs seem more relevant than attention, particularly in early layers, but further research is needed to elucidate these processes. As for CLIP's imperfect ability to process negation correctly, multiple dataset features are identified that partly explain its performance, suggesting that benchmark performance isn't a direct indicator of linguistic understanding. Future research directions are discussed that refine our understanding of the discovered mechanisms and test their generalisability on other datasets and models.

# 1 Introduction

In recent years, research in vision & language (VL) modelling has produced various pre-trained models that are capable of jointly processing image and text information by learning multimodal representations (e.g., Li et al., 2019; Lu et al., 2019; Radford et al., 2021; Jia et al., 2021; Li et al., 2021). This makes them applicable to a host of downstream tasks, such as visual question answering, image caption generation or zero-shot image classification. Various benchmarks have been proposed to test these models’ understanding of different linguistic features, such as word order (Akula et al., 2020), verb meaning (Hendricks and Nematzadeh, 2021), and compositionality (Thrush et al., 2022). The VALSE benchmark (Parcalabescu et al., 2022) was introduced to test these models’ ability to ground features such as existence, plurality, or spatial relations in images. An example of the existence piece would be an image of giraffes, based on which the model is tasked to distinguish between the true caption “There are giraffes.” and the incorrect foil “There are no giraffes.” (see Figure 1). As such, this piece can be used to test a model’s understanding of negation, which remains a weakness of even the most state-of-the-art large language models (Truong et al., 2023). In line with these findings, initial VALSE results reveal that models only achieve moderate performance in this (and other) linguistic categories. However, while VL benchmarks such as VALSE are useful for measuring current and future model performance, they do not reveal anything about how these models arrive at their outputs at such visio-linguistic tasks.

Meanwhile, a growing body of recent literature (Räuker et al., 2022) has proposed methods and generated insights into the inner workings of transformer-based architectures employed in language modelling. Crucially, such model interpretability research aims to explain model behaviour not as a function of model inputs (e.g., LIME and SHAP; Ribeiro et al., 2016; Lundberg and Lee, 2017), but of model internals, such as model weights, hidden activations, or subsections of the network. The present work aims to make use of this growing literature in order to explain the behaviour (and shortcomings) of VL models on the VALSE benchmark, specifically the existence task as a test of the model’s understanding of negation. To this end, different localisation techniques are used to quantify the roles different model components play in this task. This yields the following research question.

**Research question:** Which components of VL models are responsible for the model’s understanding of negation?

- How localised (vs. distributed) is the processing of negation in VL models?
- Are there high-level dataset features that explain VL models’ moderate performance on VALSE Existence?

Specifically, I approach these questions through an in-depth analysis of CLIP (Radford et al., 2021) as a representative pre-trained VL model. The contributions of this work are twofold: firstly, I demonstrate how methods from the language model interpretability literature can be translated to multimodal models and tasks; secondly, I provide concrete insights into how CLIP processes negation on the VALSE existence task.

## 2 Related work

### 2.1 Transformer architecture

Since this work focuses quite heavily on the internal processes of Transformer models, I briefly review the main architecture by Vaswani et al. (2017) as well as relevant modifications and

extensions in the subsequent literature. The Transformer was originally proposed as an encoder-decoder architecture (in the domain of machine translation) and prominent models adopting this architecture include T5 (Raffel et al., 2020) and BART (Lewis et al., 2020). By contrast, models such as BERT (Devlin et al., 2019) and GPT-3 (Devlin et al., 2019) use encoder-only or decoder-only versions of this architecture, respectively.

### 2.1.1 Text transformer

The Transformer is designed to process a sequence of input tokens. These tokens are represented as the sum of input embeddings and positional embeddings. The dimensionality of these embeddings is a model hyperparameter and also referred to as the model dimension  $d_{\text{model}}$ . Vaswani et al. (2017) use a sinusoid function to represent the positional embeddings whereas later models (e.g., Brown et al., 2020) use learned embeddings that were trained together with all other model parameters. The core of the Transformer architecture are a stack of so called Transformer layers, the exact number of which differs by model. Each layer applies multihead self-attention (MHA) and a fully-connected multilayer perceptron (MLP) to its input. The original architecture applies layer normalisation (Ba et al., 2016) after each MHA and MLP block, whereas later architectures moved this normalisation step in front of these blocks. In addition, both the MHA and the MLP blocks are surrounded by residual connections. Elhage et al. (2021) therefore propose viewing the residual stream as the central object of the Transformer which both the MHA and MLP read from and write to.

Importantly, throughout the model, the shape of the input is maintained. That is, the input and output to and from each layer is of shape  $(t, d_{\text{model}})$ , where  $t$  is the number of tokens in the input sequence.<sup>1</sup> The MHA output at each position can be viewed as a weighted sum of information from all positions (incl. the current one). If the model uses *masked* self-attention, then each position can only “look” backwards in the sequence, i.e., the information represented at a given position cannot be influenced by later positions in the sequence. The weights with which each position in the input influences each position in the output of the MHA are called attention weights and are the normalised product of *keys* and *queries*, which in turn are learned linear projections of the inputs. The attention weights are multiplied not with the inputs themselves, but with the so called *values*, which is another linear projection of the input. This process of self-attention runs in each head of the MHA in parallel, where the number of heads  $h$  is a model hyperparameter. Here, the model dimensions are effectively distributed across the heads such that each head processes inputs of dimension  $d_{\text{head}} = d_{\text{model}}/h$ . This allows a single MHA layer to employ various distinct attention patterns, for example to capture different linguistic connections between input tokens. Lastly, the results from all heads are stacked and multiplied with a learned *output* matrix.

The MLP inside each attention layer is a simple feed-forward network with one hidden layer, whose size is a model hyperparameter (often chosen to be a multiple of the model dimension).

The output from the final transformer layer is then used depending on the model type and training task. In auto-regressive decoder-only models (e.g., Brown et al., 2020), the output at the last token position is projected linearly from the model embedding space back to the vocabulary space such that a next token can be predicted via softmax. Conversely, encoder-only models (e.g., Devlin et al., 2019) append a classifier token to the input and use the output at this position as input into a linear layer that is used to generate an encoding of the input.

---

<sup>1</sup>In reality, models often process multiple sequences in parallel, adding a batch dimension  $b$ . However, this is only done for computational reasons and has no impact on the processing of each sequence.

### 2.1.2 Vision transformer

Dosovitskiy et al. (2021) proposed the Vision Transformer (ViT), adopting the Transformer architecture for image encoders. Here, the input image is first divided into square patches which are the equivalent of input tokens in the text model. The patch size is a model hyperparameter. The 3-dimensional pixel values (colour, width, height) from each patch are then flattened into a 1-dimensional array, which is subsequently linearly projected into the model’s embedding space. Learned positional embeddings are added to the output of this linear projections and this sum, obtained at each input position, is fed into the first transformer layer. Similar to text encoder models (e.g., Devlin et al., 2019), a classification token position is prepended to the sequence of embedded image patches. The operations inside the transformer layers are identical to the text version described above. The output of the last layer at the classification token position is fed into a final linear layer whose shape depends on model configuration and training task. For example, in an image classification task, the number of output nodes of this layer would correspond to the number of classes.

## 2.2 Vision-and-language models

A major group of pre-trained VL models fundamentally follow the architecture of BERT, a Transformer-based encoder-only language model (Devlin et al., 2019). Their architectures can broadly be divided into single-stream and dual-stream encoders. Single-stream encoders include VisualBERT (Li et al., 2019) and VL-Bert (Su et al., 2020) and consist of a single transformer stack that jointly encodes image and text as one concatenated input. Conversely, dual-stream encoders include VILBERT (Lu et al., 2019) and LXMERT (Tan and Bansal, 2019) and encode image and text inputs in two separate transformer blocks before fusing these blocks to allow for cross-modal attention in subsequent transformer layers. Bugliarello et al. (2021) review these architectures and unify them under a single framework called VOLTA. Besides these architectural differences, individual BERT-like VL models also differ in their training objectives, which include multimodal masked modelling, where the model learns to predict masked words in the text or regions in the image, and image-sentence alignment, where the model learns to predict whether a text and an image correspond. In contrast to the encoder-only architectures described above, Oscar (Li et al., 2020) and VinVL (Zhang et al., 2021) are examples of VL encoder-decoder architectures.

CLIP (Radford et al., 2021) uses an architecture different from the models outlined above. It consists of an image encoder and a text encoder that process their respective inputs completely separately from each other, i.e., CLIP makes no use of multimodal cross-attention. Different variants of CLIP have been proposed which differ in size and model dimensions as well as in the choice of model for the image encoder. One set of variants is built upon a modified version of the ResNet50 architecture (He et al., 2015) while the other uses the Vision Transformer architecture (Dosovitskiy et al., 2021). Radford et al. (2021) found largely superior performance for the vision transformer which is why this variant will be used in the present study.<sup>2</sup> The text encoder is a Transformer (Vaswani et al., 2017) with masked self-attention. Importantly, both encoders project their inputs into the same latent space, in which CLIP uses a contrastive training objective that maximises the similarity of matching text-image pairs and minimises the similarity of all other non-matching text-image pairs within a training batch. During inference, CLIP takes a pair of text and image and outputs a scalar, which is the scaled dot product of the image embedding and the text embedding in the model’s multimodal latent space. This value can be interpreted as the similarity between text and image. Research into the geometry of word embeddings resulting from such a contrastive training objective (Wolfe and Caliskan,

---

<sup>2</sup>Specifically, I will use ViT-B-32, the same version that was used in the VALSE benchmark evaluation.

2022) has shown that they exhibit lower intra-layer self-similarity than embeddings produced by autoregressive training in language models such as GPT-2 (Radford et al., 2019). Intuitively, this means that the individual word embeddings are more dispersed across the model’s embedding space. This has been shown to be beneficial for semantic representation of embeddings as it improves performance on semantic intrinsic evaluation tasks (Mu et al., 2018).

Subsequent to CLIP, many VL models have adopted its contrastive training objective. Perhaps most similar to CLIP is ALIGN (Jia et al., 2021) which uses the same dual encoder contrastive learning approach, but different encoder architectures and training data. ALBEF (Li et al., 2021) combines contrastive training of separate image and text encoders (similar to CLIP) with later fusion into one multimodal encoder that allows for cross-attention between the modalities. Besides the contrastive objective that aligns the unimodal image and text encoders, it uses masked language modelling and image-text matching as joint training objectives. FLAVA (Singh et al., 2022) is comparable to ALBEF in that it combines unimodal encoders with later multimodal fusion and notably uses the very same Vision Transformer architecture for both image and text encoder. X-VLM (Zeng et al., 2022) also builds on the architecture of ALBEF, making use of a text, image, and crossmodal encoder with a contrastive objective between the two unimodal encoders. In addition, it learns to locate objects in the image by training a separate head that predicts an object’s bounding box coordinates. BLIP (Li et al., 2022) is notable in its use of image-grounded text encoder and decoders, in addition to contrastively trained unimodal encoders. It aims to resolve the problem of noisy image labels by fine-tuning a captioner that produces synthetic image captions and a filter that learns to detect noisy captions. The follow-up model BLIP-2 (Li et al., 2023) addresses the problem of high training cost and utilises existing image encoder and large language models, whose parameters remain frozen. It trains a querying transformer that learns to extract useful visual features from the image encoder and feed those to the LLM in a way that enables image captioning and visual question answering capabilities among others. Related to BLIP-2 is Flamingo (Alayrac et al., 2022) which also uses pre-trained frozen image encoders and LLMs and trains new cross-attention layers that are conditioned by the image encoder and interleaved with the existing LLM layers.

The analyses in the present study will focus on CLIP over other VL models chiefly for three key reasons. First, its architecture is comparatively simple. For example, unlike other models it does not use separately trained bounding box detectors in the visual backbone or other auxiliary models. It also relies purely on the Transformer architecture, as opposed to convolutions or recurrence, which makes its architecture more streamlined. This is expected to be conducive in applying interpretability techniques, especially those that were specifically designed for Transformer architectures. Second, prior work by Parcalabescu and Frank (2023) has provided evidence that CLIP makes balanced use of text and image input and avoids so called unimodal collapse. In unimodal collapse, a model learns to exploit one modality (e.g., text) while largely ignoring another (e.g., vision), thus effectively reducing itself to a unimodal model, which greatly affects model reliability. Since the present study aims to translate interpretability techniques from language modelling to the multimodal domain, avoiding unimodal collapse is a welcome property of the model. Third, CLIP is a prominent model within research on multimodal learning with a host of downstream applications. Its architecture and training paradigm allow it to be used in zero-shot image classification settings where the classes of interest are entered as individual text prompts and the prompt with the highest embedding similarity to the image is taken as the model’s classification output (Radford et al., 2021). CLIP also serves as the backbone for the CLIPSeg image segmentation model (Lüdddecke and Ecker, 2022) and is used as an encoder for the CLIPCap image captioning model (Mokady et al., 2021). Lastly, CLIP’s multimodal embedding space also serves as the backbone of generative diffusion models such as DALL-E (Ramesh et al., 2022) and Stable Diffusion (Rombach et al., 2022).

## 2.3 Vision-and-language benchmarks

Early benchmarks for VL models have evaluated them based on performance in particular tasks. This includes the task of answering a natural language question about a pair of images (VQA; Goyal et al., 2017) or determining whether a natural language statement is reflected in a pair of photographs (NLVR2; Suhr et al., 2019). These can be argued to test a model’s ability to recognise objects and their relative position to one another in an image or to reason visually about semantic information. However, while the captions in NLVR2 do contain various linguistic phenomena, such as quantifiers, coreference, and negation, this benchmark isn’t designed to analyse a model’s understanding of these phenomena.

In contrast to such task-based benchmarks, other benchmarks have focused specifically on linguistic understanding. Akula et al. (2020) test models’ sensitivity to word order in recognising objects in images. They develop adversarial examples that only change the word order in a prompt, but not the set of words, such that it refers to a different object within the same image. The Winoground benchmark (Thrush et al., 2022) partly builds on this idea of pairs of captions that differ only in word order. Here, two such prompts are presented alongside two images and the model has to correctly assign prompts to images, a task at which most models perform barely better than chance. CREPE (Ma et al., 2023) and ARO (Attribution, Relation and Order; Yuksekgonul et al., 2023) are further benchmarks focusing on models’ ability to reason compositionally. SVO-Probes (Hendricks and Nematzadeh, 2021) focuses on verb understanding in models by having them predict whether a sentence and image correspond to one another. Crucially, for each sentence this benchmark contains a matching image and another image that depicts the same subject and object but a different activity, i.e., verb. VSR (Visual Spatial Reasoning; Liu et al., 2023) is a benchmark designed to test models’ spatial reasoning abilities. It consists of caption-image pairs with either true or false labels, where the caption always describes the spatial relation of two objects in the image.

VALSE (Parcalabescu et al., 2022) is a benchmark for VL models designed to test their ability to ground different linguistic phenomena in the visual domain. Specifically, VALSE prompts a model with an image along with both its correct caption and a foiled caption and tests a model’s ability to distinguish the caption from foil. Foiling, as a method to design VL model inputs, was introduced by Shekhar et al. (2017). Crucially, foils differ from true image captions only with respect to a particular phenomenon of interest. This minimal difference allows for targeted testing of specific model capabilities and clearer attribution of model errors. It also represents a step towards precluding models from relying on language priors (the simple fact that certain word sequences are more likely to occur, and thus be a correct image caption, than others) and ignoring their visual input (e.g., Jabri et al., 2016). VALSE tests six different linguistic phenomena, referred to as pieces, namely existence, plurality, counting, relations, actions, and coreference (Figure 1 shows examples from the existence piece). It is applied to a number of different VL models, including VisualBERT, ViLBERT, LXMERT, and CLIP. Overall, these models only perform moderately well with average accuracy scores across linguistic phenomena ranging from 46% to 75% (note that random guessing would yield an accuracy of 50%). Specifically, CLIP only achieved an accuracy of 66.9% on the existence piece.

A recent benchmarking study by Bugliarello et al. (2023) brings together SVO-Probes, VALSE, VSR, and Winoground to test the visio-linguistic capabilities of a host of recent VL models. They find that X-VLM (Zeng et al., 2022) outperforms many other models (including CLIP (Radford et al., 2021) and ALBEF (Li et al., 2021)) which highlights the value of explicitly training VL models on object recognition (by learning to predict the coordinates of an object’s bounding box). In particular, they find that adding this training objective leads to greater performance boosts than increasing the training data volume.

Type	Image	Caption	Foil
Negation in foil		There are giraffes	There are no giraffes
Negation in caption		There are no people	There are people

**Figure 1:** Examples from VALSE existence (Parcalabescu et al., 2022). Caption and foil only differ in the presence or absence of the negator “no”. The negator is either in the caption or the foil.

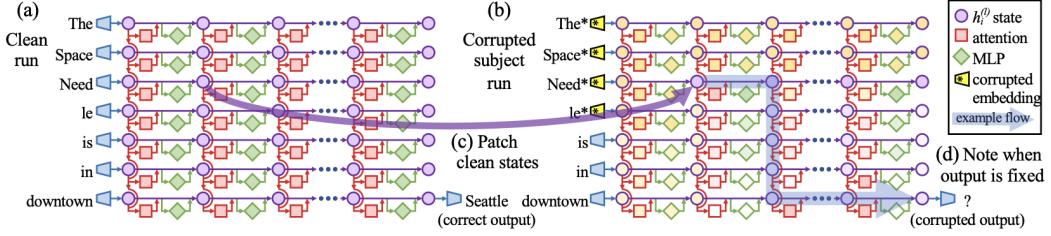
## 2.4 Model interpretability and localisation

Räuker et al. (2022) define inner interpretability methods as those that help understand a model’s internal structures and activations. Their taxonomy divides such methods based on the component of a model’s architecture that the method tries to explain: weights, neurons, subnetworks or circuits, and latent representations. They furthermore divide methods into intrinsic methods which are used during training and post-hoc methods which are applied after training. This particular dichotomy is also adopted by other interpretability surveys (e.g., Das and Rad, 2020). The focus of the present study is on existing pre-trained models (namely CLIP) and therefore on post-hoc interpretability techniques.

One recurring strategy in such techniques is to analyse the effect of perturbations or ablations on the model’s behaviour and output. For example, some studies aim to characterise the function of individual neurons in a network by testing whether and how ablating their activations changes the model’s output on a particular task (e.g., Zhou et al., 2018). Ghorbani and Zou (2020) refine this approach using Shapley values, as originally studied in cooperative game theory (Shapley, 1951), that quantify a neuron’s contribution to a model’s output by systematically ablating coalitions of other neurons in the network. Other studies apply ablation not to neurons, but to weights in neural networks with the goal of identifying modular subnetworks responsible for specific tasks (Csordás et al., 2021).

The choice of a suitable level of granularity at which to apply ablation is largely dictated by the model’s size and complexity. Smaller networks lend themselves more easily to ablating individual neurons. By contrast, modern pre-trained language models are so large that an attempt to explain their behaviour at the neuron level would likely be futile. In light of this, transformer model interpretability methods often operate at the level of attention heads, MHA modules, MLPs, or even full Transformer layers.<sup>3</sup>

<sup>3</sup>However, note that previous research (Goh et al., 2021) has also produced neuron-level interpretations of CLIP’s image encoder, albeit the ResNet and not the ViT variant.



**Figure 2:** Causal tracing (reproduced from Meng et al., 2023). The activation of a particular component from the clean run is restored in the corrupted run to measure its effect on the model output.

In this line of research, Meng et al. (2023) introduced the causal tracing methodology to localise factual associations in a model. A running example for this is a model’s ability to complete the prompt “The space needle is in downtown” with the correct response “Seattle”. Their method is built upon three subsequent model runs. In the first run, the model is prompted with the input that is designed to elicit the model behaviour of interest and the hidden activations at all layers and positions in the transformer from this forward pass are stored. In the second run, the input is corrupted at the positions that are relevant for the task at hand, such as the subject tokens of “The space needle” in the example above. This will predictably corrupt the model’s output. The third run is designed to quantify a specific model component’s ability to restore the correct output in the model. To this end, the hidden activation of a particular model component from the second (corrupted) run is replaced by its corresponding activation from the first (clean) run. Intuitively, if this component is crucial to the model’s ability to produce the correct output, then restoring its correct activation should also restore the correct model output. Figure 2 illustrates this method graphically (reproduced from Meng et al., 2023).

To quantify a component’s contribution, the authors compare the total effect with the indirect effect of a particular component, following the methodology of causal mediation analysis (Pearl, 2001). The measure of interest in the context of generative language models is the model’s probability of outputting the correct answer token (“Seattle” in the example above). They define  $P[o]$ ,  $P_*[o]$ , and  $P_{*,i,l}$  as the model’s probability of outputting the correct token  $o$  under the clean, corrupted, and restored condition, respectively, where the restoration was applied at position  $i$  in layer  $l$ .<sup>4</sup> This then leads to the definition of the total effect  $TE = P[o] - P_*[o]$  and the indirect effect  $IE = P_{*,i,l} - P_*[o]$ , both of which can be averaged over input samples to obtain the average total and indirect effect, respectively.

Meng et al. (2023) apply this technique in such a way to allow them to separate the effect of MLPs from attention blocks in each transformer layer and input position. Their key finding is that MLP modules at the last subject token position in early layers of the model appear to contain factual associations such as the example above.

In Meng et al. (2023), this localisation step serves as the basis for subsequent model editing. Here, the authors introduce ROME, a technique to alter a factual association in the MLP layers of a transformer encoder. To reuse the above example, after localising a particular layer that contains the relevant knowledge using causal tracing, such a model edit could be used to make the model output, for example, “Paris” in response to the prompt “The space needle is in downtown”. However, this editing methodology has been called into question in at least two places. Hase et al. (2023) demonstrated that while model editing is an effective technique in itself, the causal tracing effect and editing success of a layer are virtually uncorrelated. This means that the ability to edit knowledge in a particular layer does not imply that this knowledge

<sup>4</sup>For simplicity, the notation is slightly altered compared to the original in Meng et al. (2023)

is localised in this layer. As such, model editing success is not a suitable method for verifying localisation results. Furthermore, Hoelscher-Obermaier et al. (2023) showed that editing results from the ROME method are in fact not as specific as proclaimed by Meng et al. (2023) but introduce systematic unwanted side effects. For example, editing the model to complete the prompt “The Louvre is in ” with “Rome” will also make it falsely output “Rome” in response to the prompt “The Louvre is cool. Obama resides in ”. Given these uncertainties surrounding model editing techniques, the present focuses on localisation only.

Gandelsman et al. (2023) propose a related approach to identify relevant components in the vision transformer that CLIP uses as its image encoder. They perform mean ablation (see also Wang et al., 2022; Nanda et al., 2023) on both the MHA and MLP modules inside the vision transformer to analyse their relevance to CLIP’s zero-shot classification performance on the ImageNet validation set (Russakovsky et al., 2015). Mean ablation replaces the activation of a particular network component (e.g., a particular MLP) with its average activation over a dataset. If the component is relevant to the task at hand (in this case, image classification), a significant decrease in model performance (in this case, classification accuracy) would be expected. Conversely, components that are irrelevant to the task should produce no meaningful change in performance. Gandelsman et al. (2023) find that only the MHA blocks in the late layer of CLIP’s image encoder produce a significant drop in classification accuracy, whereas ablating the MLPs and early MHA blocks does not affect performance.

A final line of interpretability literature relevant to the present study comes from research that directly analyses attention patterns in large Transformer models. Clark et al. (2019) analyse the attention patterns found in BERT (Devlin et al., 2019) and report a number of attention heads that seem to exhibit specific syntactic functions, for example attending to direct objects of verbs, objects of prepositions, or determiners of nouns. Kovaleva et al. (2019) further show that BERT uses only a limited set of attention patterns that are repeated across heads. Vig and Belinkov (2019) run a similar analysis on GPT-2 (Radford et al., 2019) and find attention heads attending to various specific part-of-speech tags. Furthermore, all of these studies converge on the finding that these pre-trained Transformer language models allocate significant attention to tokens that do not carry inherent semantic meaning, such as the separator token in BERT or the start-of-sequence token in GPT-2.

## 3 Methods

### 3.1 Definitions

A forward pass in CLIP of a single VALSE existence instance consists of a text caption, a text foil, and an image. This produces one similarity score for caption and image and one for caption and foil. I denote these two scores  $S_{(c,i)}$  and  $S_{(f,i)}$ , respectively.

CLIP is said to correctly classify a caption-foil-image triple when it assigns a higher similarity to the caption-image pair than to the foil-image pair, i.e., when  $S_{(c,i)} > S_{(f,i)}$ . We can quantify CLIP’s classification performance using the difference between the two similarities. I denote the classification score  $d = S_{(c,i)} - S_{(f,i)}$ . Therefore, an instance is classified correctly, if and only if  $d > 0$  and the absolute size of  $d$  can be seen as an indicator of CLIP’s confidence in the classification.

### 3.2 Dataset pre-processing

The VALSE existence dataset consists of 505 image-caption-foil triples. It was created using examples from the Visual7W visual question answering dataset (Zhu et al., 2016) with

“how many” questions. These were transformed into existential statements that form a correct statement about what the image shows, e.g., “There are giraffes.” or “There are no people.”. The dataset is thus divided into instances where the negation is in the foil (249) and instances where the negation is in the caption (256).

### 3.2.1 Rephrasing of labels

Since caption and foil in the VALSE existence dataset differ only in the presence of the negator, they sometimes have a different number of tokens. Concretely, this is the case in “bare plural” sentences where there is no article or other qualifier in the non-negated sentence (e.g., “There are tennis players.” vs “There are *no* tennis players.”). Identifying differences in how CLIP processes negated vs. non-negated labels is a core facet of the present study and such comparisons are greatly facilitated if caption and foil have the same number of tokens. Therefore, labels were rephrased to achieve equal sequence length. Concretely, the qualifier “some” was inserted into the non-negated plural sentences right before the subject (i.e., at the same position as the negator in negated sentences). For example, “There are tennis players” was rephrased to “There are *some* tennis players”. In labels with singular subjects (e.g., “There is a giraffe shown” vs. “There is *no* giraffe shown”), such rephrasing was not needed and these instances remained in their original form in the dataset. 15 instances (0.03%) from the original dataset have labels that do not follow the simple “There is/are no [subject] ...” structure and therefore aren’t amenable to the rephrasing rule described above. For reasons of simplicity, these were omitted from the rephrased dataset.

Importantly, rephrasing the dataset in this way only led to minor changes in CLIP’s classification accuracy on this dataset (see Table 1). All subsequent analyses are based on the rephrased dataset, unless denoted otherwise.

### 3.2.2 Data segmentation

CLIP achieves an only moderate accuracy of 0.686 on VALSE existence (see Table 1), suggesting that it correctly processes negation in some instances but not in others. To be able to identify patterns of processing in the model that gives rise to correct classification of negation (and ideally distinguish it from those that lead to incorrect classification) it is necessary to analyse correctly classified instances separately from incorrectly classified ones. To do this in a consistent manner across different analyses, the dataset was divided into three segments (correct, ambiguous, incorrect) based on the classification score  $d$ . Table 2 describes these segments in more detail. Figure 3 shows the distribution of classification scores and resulting segments.

## 3.3 Preliminary analyses

The purpose of these initial analyses is to identify high-level features that correlate with CLIP’s classification score on VALSE existence. The following features are analysed:

1. Sequence length: number of tokens in the image caption
2. Text embedding similarity: cosine similarity of caption and foil embedding in CLIP’s multimodal embedding space
3. Relative subject size: proportion of the image taken up by the caption’s subject

The motivation for analysing subject size is that determining the existence of a subject in an image likely becomes easier the larger and more prominent its depiction in the image. Relative subject size is determined using CLIPSeg (Lüdecke and Ecker, 2022). CLIPSeg uses a decoder on top of CLIP’s original image encoder that is trained to perform binary image segmentation.

	Standard	Rephrased	Difference
<b>Caption</b>	0.676	0.668	0.008
<b>Foil</b>	0.707	0.704	0.003
<b>All</b>	0.691	0.686	0.005

**Table 1:** CLIP’s accuracy on the original and rephrased VALSE existence benchmark

	Correct	Ambiguous	Incorrect
	$d > 1$	$1 \geq d > -1$	$d \leq -1$
<b>Caption</b>	72	150	28
<b>Foil</b>	81	145	14

**Table 2:** Number of instances per segment in the VALSE existence dataset

The input to the segmentation task can be either text (which gets encoded using CLIP’s text encoder) or an additional image representing the object to identified in the segmentation. Here, the subject from the image’s caption is used as the text prompt for the segmentation. The subject is identified using an English language tokeniser from the spacy library in Python (Honnibal et al., 2020). Evidently, this analysis can only be performed on the part of the VALSE Existence dataset where the negation is in the foil, since in the other part the subject from the text caption is in fact not present in the image.

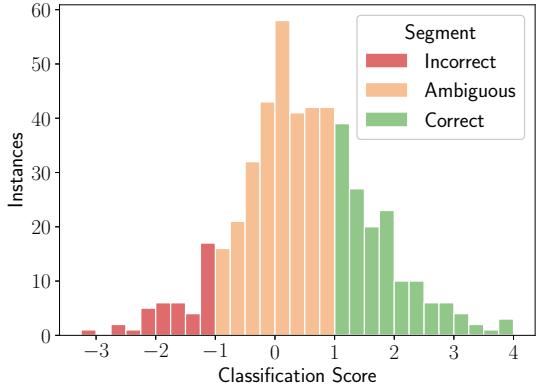
The pixel-wise logits returned by the CLIPSeg decoder are transformed into probabilities using a logistic sigmoid function. The average of these probability values across the image is then used as the relative subject size in the image.

### 3.4 Causal tracing in text encoder

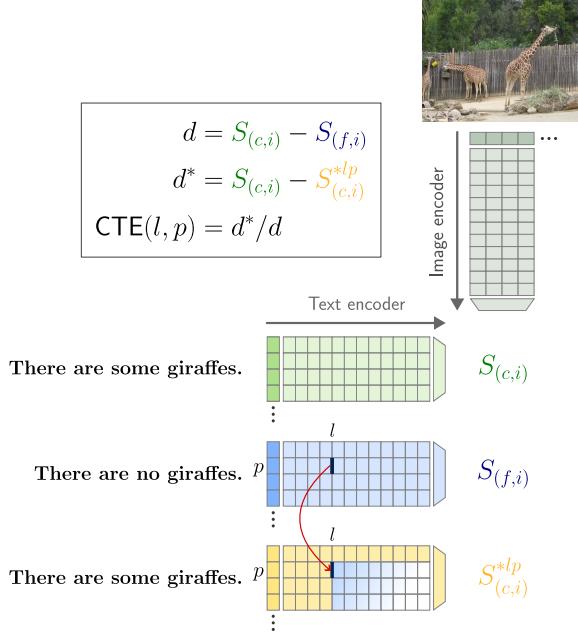
The causal tracing method from Meng et al. (2023) is adapted to the VALSE existence dataset as follows. I outline in detail the methodology for the part of the dataset where the negation is in the foil. I then describe the changes necessary to make the same method work for the remaining part of the dataset where the negation is in the caption. Figure 4 provides a visual summary of the method.

A standard forward pass is carried out with caption, foil, and image, yielding the two similarity scores  $S_{(c,i)}$  and  $S_{(f,i)}$ . This lets us calculate the normal classification score  $d = S_{(c,i)} - S_{(f,i)}$ . Importantly, the activations from the forward pass at each layer and each position in the text encoder are recorded. These activations are used in the subsequent modified forward pass. Here, only the caption (which in the case described does not contain the negator) is used in the forward pass alongside the image. Crucially, during this forward pass, the text encoder’s activation at a given layer and position is replaced by the activation from the foil’s original forward pass at the corresponding layer and position. This process is carried out individually for each combination of layer and position.

Intuitively, this achieves the following. The model processes the non-negated caption, but at a given layer and position it is made to behave as if it was processing the negated foil. If the processing taking place at a certain layer and position is unaffected by the presence of negation, then the replacement of the activation should not lead to any meaningful change in output. If, conversely, a certain layer and position is specialised in processing negation, then substituting



**Figure 3:** Distribution of CLIP’s classification score on VALSE existence. Colour indicates dataset segment.



**Figure 4:** Illustration of the causal tracing methodology adopted for VALSE existence. The activation at a single position and layer from the negated forward pass are inserted into the corresponding layer and position of the non-negated forward pass. This shows what proportion of the original effect can be restored by this layer-position pair. Image and text are taken from VALSE existence (Parcalabescu et al., 2022).

the activation from the negated forward pass into the non-negated one should in fact affect the output in a visible way.

This intuition is quantified in the following way. For a given layer  $l$  and a position  $p$  the modified forward pass produces a similarity score  $S_{(c,i)}^{*lp}$ . This allows us to calculate a modified classification score

$$d^* = S_{(c,i)} - S_{(c,i)}^{*lp}$$

We use this modified classification score to calculate the causal tracing effect of layer  $l$  at position  $p$  as

$$\text{CTE}(l, p) = d^*/d$$

Intuitively, this effect represents the proportion of the original classification score  $d$  that can be “restored” by layer  $l$  at position  $p$ .

Again, if the behaviour of interest (in this case, the correct processing of negation in the text encoder) is highly localised in the text encoder, we would expect only a few layer-position pairs to show a high causal tracing effect. Conversely, if this behaviour is highly distributed across the model, then we would expect many layer-position pairs to have small to moderate effects and none of them to stand out significantly.

To apply this method to cases where the negation is in the caption, one has to swap caption and foil such that, again, the activations from the negated sentence (now the caption) are substituted into the forward pass of the non-negated sentence (now the foil). This means that we obtain a modified classification score, which is used to calculate the causal tracing effect in the same way.

$$d^* = S_{(f,i)}^{*lp} - S_{(f,i)}$$

This method yields a causal tracing effect for each layer and position for each VALSE existence instance. However, since the captions in the dataset are of different lengths, one further step is needed to allow us to average the effect across the dataset. As described in Section 3.2.1, all captions share the same beginning set of tokens, namely [SOT, There, is/are, a/some, subject] where SOT represents the generic start-of-text token. Furthermore all captions and foils end on [., EOT], i.e., a full stop and the generic end-of-text token. However, they differ in the number of tokens in between these two sets. Some captions have no further tokens, others multiple (e.g., “There are airplanes visible in the picture.”). Therefore, the CTE from all positions in between the beginning and end sets of tokens are averaged into one placeholder position called “further subject tokens”. If there are no positions between the beginning and end sets, then a CTE of 0 is recorded at this position. This means that the CTE results from all instances have the same shape, namely  $P \times L$ , where  $P = 8$  and  $L = 13$ . Consequently, we can average these results across the dataset (or a segment thereof). To represent each instance according to its sequence length, the averaged effect at the “further subject tokens” position is weighted by the number of tokens that make up this position in each instance.

Lastly, we want to be able to describe the degree of localisation in particular layers. Localisation is strongest when one position in a layer at the exclusion of all other positions restores the full effect. Conversely, localisation is absent when each position restores the same proportion of the effect. Hence, we can quantify the degree of localisation in a layer  $l$  as the standard deviation of the causal tracing effects at each position in this layer, starting at the negator position.

### 3.5 Negator-selective attention in text encoder

The purpose of this analysis is to identify attention heads in CLIP’s text encoder that selectively pay attention to negators. In the context of the VALSE existence dataset, this means that they specifically pay attention to the “no” token in negated sentences, but not to the “a”, “an”, or “some” tokens at the same position in the corresponding non-negated sentences.

To this end, attention maps are recorded for each attention head in a standard forward pass of the VALSE existence dataset. Since the forward pass consists of both caption and foil, this yields two maps per head. Attention maps indicate how much a position  $p$  attends to each position in the sequence, that is to say, how much each position is contributing to the representation of  $p$  at the subsequent layer. As such, an attention map is an array of size  $P \times P$  where  $P$  is the number of positions in the input sequence. Recall that in Transformers with masked attention (such as the CLIP text encoder) positions can only attend to earlier positions in the sequence, which is why all elements to the right of the diagonal of this array must be 0.

The attention map is filtered to the column representing the position of the negator in the negated input sentence (or the quantifier in the corresponding non-negated sentence). The values in this column represent the degree to which each subsequent position attends to this negator position. To identify negator-selective attention, we subtract the values from the non-negated sentence from those from the negated sentence (depending on whether caption or foil are negated). Finally, the maximum of the resulting difference values is taken over all source positions and this represents the amount of negator-selected attention of a particular attention head on this particular dataset instance. This procedure can then be repeated over the whole dataset yielding an average negator-selected attention value  $a_{lh}^N$  for each attention head  $h$  in each layer  $l$ .

Instead of taking the maximum value over source positions, negator-specific attention can also be calculated for each source position. In heads with high negator-specific attention, this creates a more fine-grained picture of the negator-specific attention patterns involved.

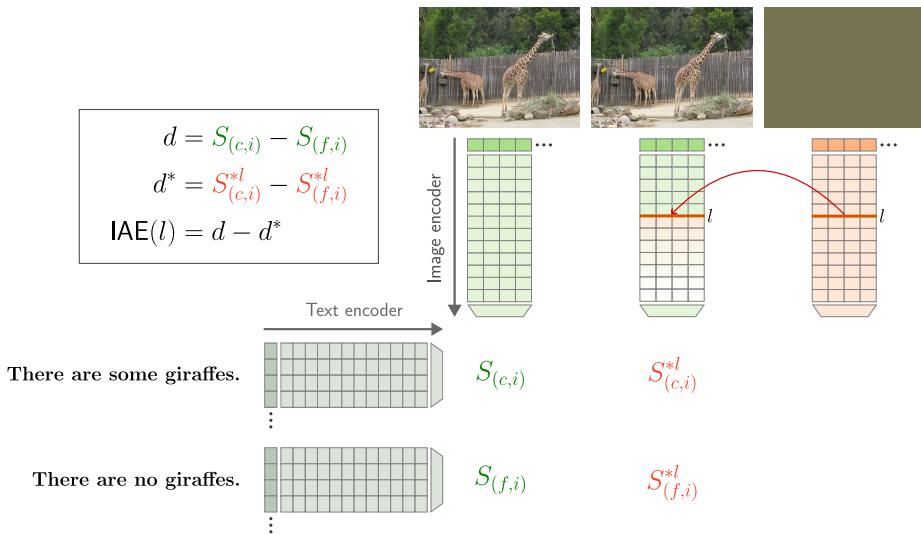
### 3.5.1 Validation on the CANNOT dataset

To test the validity of the results from this analysis, it is further adapted to a subset of the CANNOT dataset (Anschütz et al., 2023). CANNOT was created by taking a set of reference sentences and creating a negated as well as a paraphrased (but meaning-preserving) version of each sentence. The dataset can thus be used to train models to understand and recognise negation.

For the present purposes, the dataset is filtered to 554 negated sentences that contain the word “no” as the determiner of the sentence subject, again using a tokeniser from the spacy library in Python (Honnibal et al., 2020). An example sentence would be “Medical organizations recommend *no* alcohol during pregnancy for this reason”. For each of these sentences, a non-negated counterpart is then generated by replacing the word “no” with “some”. This yields a set of sentence pairs, comparable to the caption-foil pairs from VALSE existence, which thus allows us to apply the same methodology for negator-selective attention.

### 3.6 Image ablation in image encoder

The methodology for image ablation is adapted from Gandelsman et al. (2023) as a way to systematically analyse the role of individual components of the image encoder. The analysis is carried out separately on the MHA and MLP blocks inside the encoder and results are collated to compare the impact of both. Figure 5 summarises the method visually.



**Figure 5:** Illustration of the image ablation methodology adopted for VALSE existence. In the ablated run, the activations of either MHA or MLP at layer  $l$  are replaced by those from a forward pass with an average colour image. Image and text are taken from VALSE existence (Parcalabescu et al., 2022).

For each instance in the dataset, three forward passes are carried out. In the first one, the data remains unmodified, while in the second one the image is replaced by a single-colour image that is obtained by taking the average pixel value of the original image across the three colour channels. The purpose of this is to effectively reproduce the original forward pass with all visual information removed from it. Importantly, in the average-image forward pass the hidden activations of either the MHA or MLP blocks are stored, depending on which component is being analysed.

The final ablated forward pass once again uses the normal, unmodified inputs. However, at a given component in a given layer, the activation is replaced by that from the average-image

forward pass. This is to ablate the component at that layer by making it behave as if it had received no visual information (as was the case in the average-image forward pass). Intuitively, if a component in the image encoder plays a crucial role in the correct classification of VALSE existence then ablating it should significantly reduce the classification score. Conversely, if a component is irrelevant to the task at hand, we should expect no major ablation effect. Similarly to the causal tracing experiment in the text encoder, we can also use these results as an indicator of how localised the processing of VALSE existence data in the CLIP image encoder is.

In both MHA and MLP, the ablation happens at the layer level. Thus, in the ablated forward pass at layer  $l$  we receive an ablated classification score  $d_{\text{MHA}}^{*l}$  and  $d_{\text{MLP}}^{*l}$ , respectively. To quantify the image ablation effect at layer  $l$ , we take the difference between the original classification score and the ablated one:

$$\text{IAE}_{\text{MHA}}(l) = d - d_{\text{MHA}}^{*l}$$

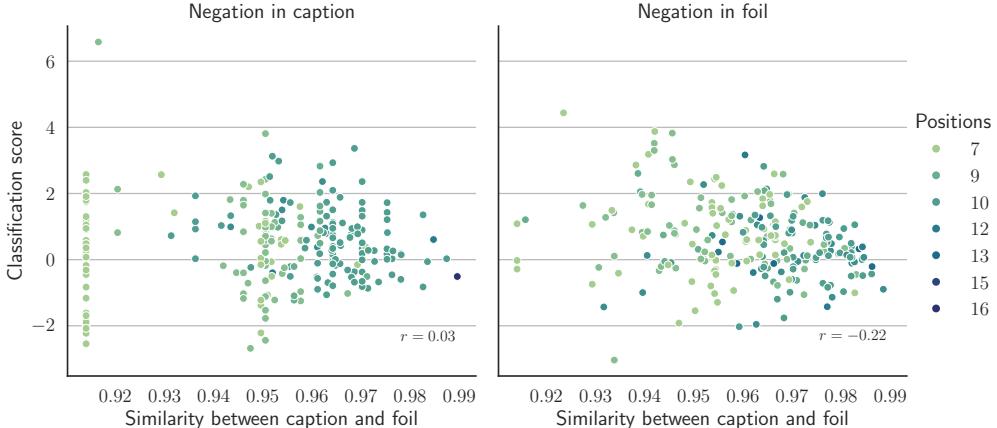
and

$$\text{IAE}_{\text{MLP}}(l) = d - d_{\text{MLP}}^{*l}$$

As such, for each dataset instance, we obtain one image ablation effect per component (MHA and MLP) and layer which can subsequently be averaged across the dataset (or a segment thereof).

## 4 Results

### 4.1 Preliminary analyses

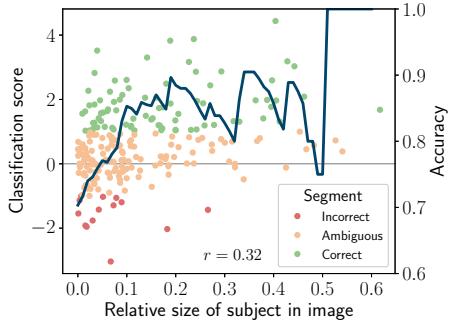


**Figure 6:** Cosine similarity of caption and foil in CLIP’s multimodal embedding space vs. CLIP’s classification score. Colour indicates dataset sequence length (i.e., number of tokens in sequence).

Figure 6 shows the cosine similarity of each instance’s caption and foil in CLIP’s multimodal embedding space against that instance’s classification score, split by whether the negation is in caption or foil. When the negation is in the foil, similarity and score are weakly correlated ( $r = -0.22$ ), whereas no correlation is found when the negation is in the caption ( $r = 0.03$ ). However, note that in the latter case, there appears to be a set of outliers with the same similarity (0.91). These turn out to be 44 instances that all share the caption “There are no people.”. Removing them from this analysis does actually yield a correlation of  $r = -0.20$ , comparable to the one found when the negation is in the foil. These negative, albeit weak,

correlations indicate that CLIP becomes less confident in its classification the more similar caption and foil are.

Figure 6 also encodes sequence length with longer sequences (darker colour) tending to exhibit greater caption-foil similarity. This is to be expected since caption and foil differ in exactly one position. If the total number of positions increases, then the relative size of this difference decreases, leading to greater similarity. These results suggest that CLIP’s failure to correctly classify some FALSE Existence instances might be partly due to instances with longer captions and foils that are more similar in their representation and therefore more difficult to tell apart. However, filtering the dataset to instances with shorter sequences does not meaningfully improve CLIP’s accuracy, suggesting that sequence length plays a minor role at best.



**Figure 7:** Relative size of subject in the image vs. CLIP’s classification score. All instances where the subject from the caption is shown in the image. Colour indicates dataset segment. The blue line shows classification accuracy when imposing a minimum subject size threshold.

	Positions	Size <sup>†</sup>	Similarity	Score
<b>Positions</b>	1.000	0.030	0.574	-0.041
<b>Size<sup>†</sup></b>	0.030	1.000	-0.052	0.321
<b>Similarity</b>	0.574	-0.052	1.000	-0.048
<b>Score</b>	-0.041	0.321	-0.048	1.000

<sup>†</sup> Only calculated when negation is in the foil.

**Table 3:** Correlation results from preliminary analyses. Positions: Sequence length (number of tokens). Size: relative size of subject in image. Similarity: cosine similarity of caption and foil in CLIP’s multimodal embedding space. Score: CLIP’s classification score. Note that correlations between similarity/positions and score are affected by outliers (see Figure 6 (left))

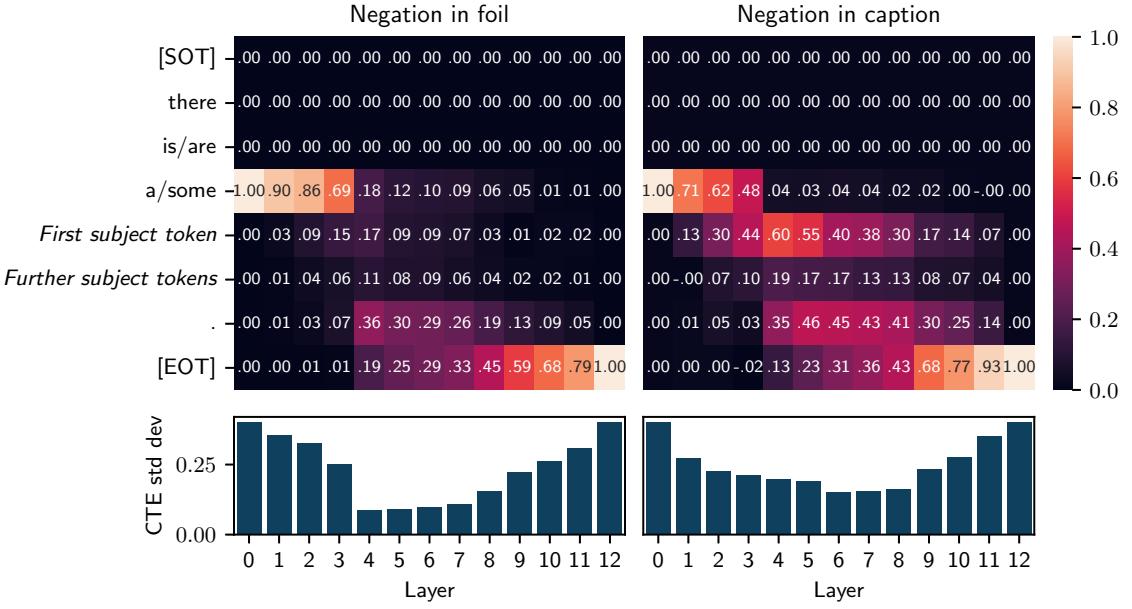
Figure 7 shows the relative size of the caption’s subject in the image, as estimated by the CLIPSeg model (see Section 3.3), compared to this instance’s classification score. The correlation of  $r = 0.32$  shows that images with larger subjects tend to be classified more accurately. In fact, when imposing a subject size threshold of 0.1 (i.e., which however filters out 43% of instances), CLIP already achieves an accuracy of 0.85. The accuracy as a function of the subject size threshold is shown by the line in Figure 7. Note, however, that the validity of these results decreases with higher thresholds, as the remaining sample size gets very small. Nonetheless, these results suggest that CLIP would exhibit better existence classification results on a dataset with larger subjects shown in the images.

Table 3 shows correlations of all variables included in this preliminary analysis.

## 4.2 Causal tracing in text encoder

The left heatmap in Figure 8 shows the causal tracing effect per layer and position for the correct segment of the data where the negation is in the foil. The effects of the first three positions are all 0. This is expected since CLIP’s text encoder uses masked attention and therefore these positions cannot be affected by tokens later in the sequence. Thus, none of the positions prior to the negator can be affected by its presence or absence, which is the only effect measured here.

Furthermore, the effect at the negator position in layer 0 (i.e., the embedding layer) is 1.



**Figure 8:** Causal tracing results of the correct segment, split by whether negation is in foil or caption. The heatmaps show the CTE of each layer-position pair in the text encoder. The bar charts show the standard deviation of all CTE in the corresponding layer as an overall measure of localisation. Layer 0 denotes the embedding layer.

Recall that we are patching activations from the negated sentence into the forward pass of the non-negated sentence and that these two sentences only differ at the position of the negator. Thus, performing causal tracing at the negator position in layer 0 is equivalent to replacing the non-negated sentence with the negated one. We therefore restore the full difference between the two sentences and obtain an effect of 1.

For similar reasons, we observe an effect of 1 at the last position in the final layer. Recall that CLIP uses the output at the last position in the final layer as the input into the projection that returns the text’s multimodal embedding. Thus, replacing the activation from the non-negated forward pass at the last position in the final layer with that from the negated forward pass is tantamount to replacing the final result (i.e., the similarity score  $S$ ) of the non-negated sentence with that of the negated sentence. Hence, we restore the full difference and obtain a CTE of 1.

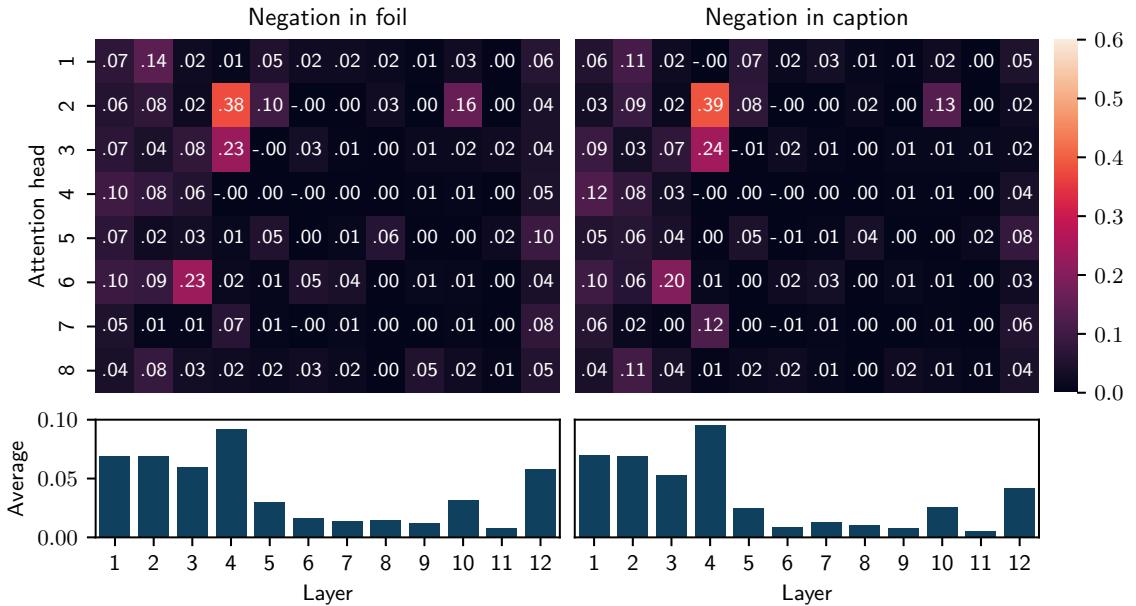
We are thus interested in the effect of components that lie in between the negator position in layer 0 and the last position in the final layer, as these possibly mediate the model behaviour we are interested in, namely the correct processing of negation in the text input. Figure 8 shows that the effect is limited to only a subset of positions and layers and seems to suggest a path through the model. In particular, in layers 0-3 the effect is practically limited to the negator position, suggesting that in these early layers the negation information is processed mainly at its original position. The effect at the negator position then drops sharply at layer 4 and further decreases until the final layer. This indicates that the negator position only plays a pivotal role in the early layers and that the processing is in fact “shifted” to the second-to-last and last position at layer 4. In the central layers 4-7 these two positions seem to play an equally important role, judging by their respective CTE, and from layer 8 onwards, the effect is concentrated in the last layer.

For a given layer, we can quantify the degree of localisation using the standard deviation of the CTE at each position (starting at the negator position), since the CTE at earlier positions

must be 0). This is shown in the bar charts in Figure 8. In line with the interpretation above, localisation is high in the early layers 0-3, then drops sharply in layer 4, remains low in the middle layers, and goes up again in the late layers 9-12.

The right part of Figure 8 shows the results from the same experiment on the correct segment of the data where the negation is in the *caption*. The general pattern of these results is comparable to the ones described above, where the negation is in the *foil*. Again, the negator position plays an important role in the early layers which then sharply decreases in layer 4. However, unlike in the prior case, the first subject position also already has a visible effect in the early layers, leading to reduced localisation. The effect of the first subject position becomes most pronounced in the middle layers which constitutes the most substantial difference between the two sets of results and in fact leads to greater localisation in the middle layers. Finally, in the late layers 9-12 the effect is once again concentrated in the last position, a pattern similar to the previous results. Possible reasons for the increased role of the first subject position are discussed in Section 5.1.

### 4.3 Negation-selective attention in text encoder



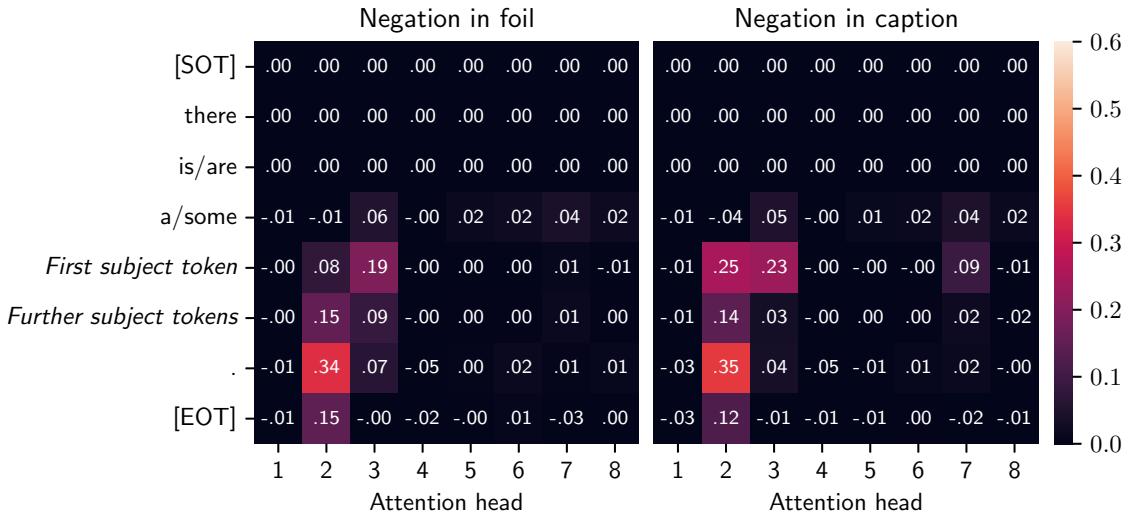
**Figure 9:** Negator-selective attention across all dataset segments, split by whether negation is in foil or caption. The heatmaps indicate the degree of negator-selective attention for each attention head and layer. The bar charts show the average of each layer as an overall measure of negator-selective attention.

Figure 9 shows the negator-selective attention of each attention head of each layer in CLIP’s text encoder, divided by whether the negation is in foil or caption. The patterns in both parts of the dataset are practically identical, which is to be expected since this analysis only considers processes in the text encoder and is not affected by its alignment with the vision encoder. As a general observation, only a small subset of heads display any negator-selective attention (8% of heads with  $a_{lh}^N > 0.1$ ) and the majority of them are found in the early layers. The most negator-selective attention head is found in layer 4.

Note that these results are reported across all dataset segments (incorrect, ambiguous, correct), since the patterns do not meaningfully differ between them. This suggests that negator-selective attention cannot explain the difference in CLIP’s classification performance on different instances of FALSE existence, since the same patterns are found in correctly and incorrectly classified

cases. In fact, none of the attention heads that show negator-selective attention of at least 0.1 show a correlation between negator-selective attention and classification score (all  $|r| < 0.2$ ).

Layer 4, where the most negator-specific attention is found, is the same layer, where the causal tracing results from Section 4.2 suggested that negation information is “moved” from its original position to later positions. To provide further evidence for this narrative, we can analyse the source of this negator-specific attention, that is to say, which specific positions pay particular attention to the negator position in the identified heads of interest. This is shown in Figure 10. In line with the hypothesis of negation information being moved to later position, the source of negator-specific attention in Head 2 is the second-to-last position. Furthermore, when the negation is in the caption, we find that additional negator-specific attention comes from the first subject position, which aligns with the greater role this position plays in this part of the dataset, as already suggested by the causal tracing results in Section 4.2.



**Figure 10:** Source of negator-selective attention in layer 4 across all dataset segments, split by whether negation is in foil or caption. The heatmaps show the degree of negator-selective attention from each sequence position (y-axis) in each attention head (x-axis).

#### 4.3.1 Validation on the CANNOT dataset

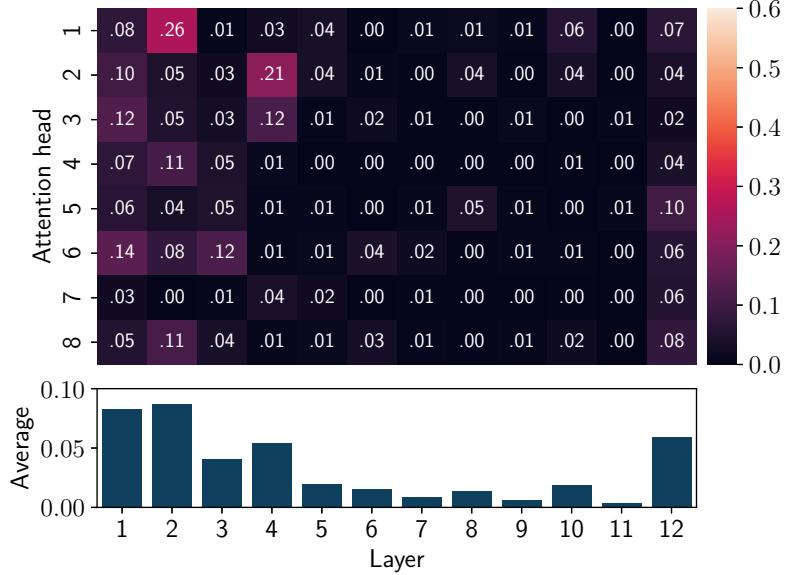
Figure 11 shows negator-selective attention on the analysed subset of the CANNOT dataset. Just like on the VALSE dataset, most negator-selective attention is found in the early layers 1-4. Head 2 in layer 4 once again shows particularly high negator-selective attention, albeit not the highest, which here is found in head 1 in layer 2. In summary, this provides converging evidence for the negator-selective attention results found in VALSE existence.

### 4.4 Image ablation in image encoder

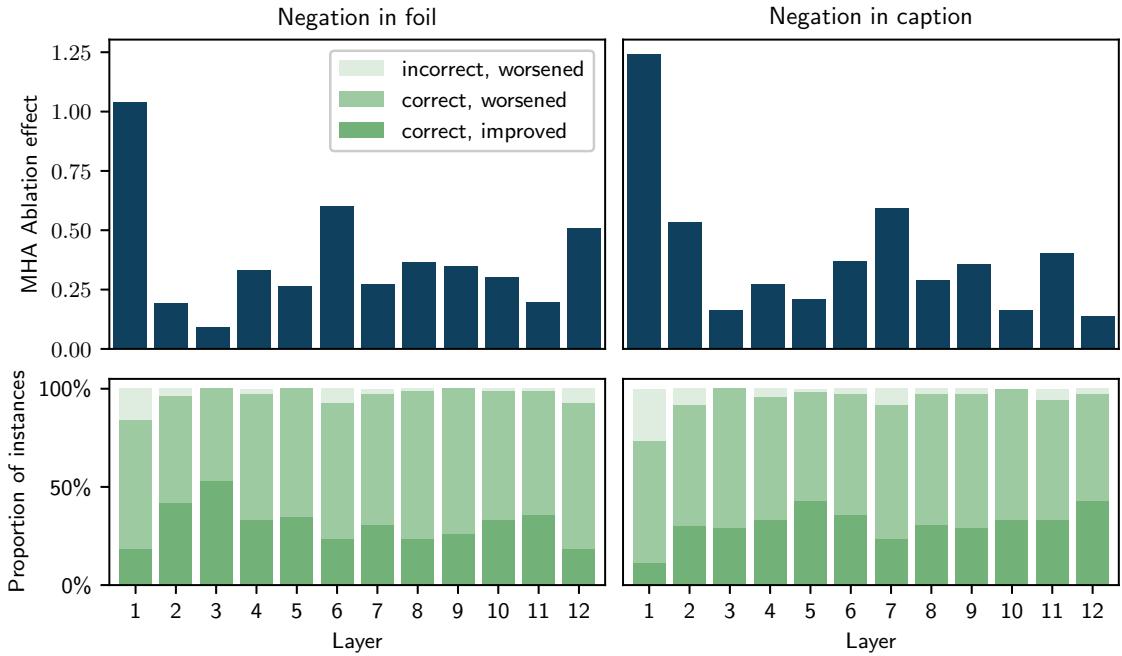
#### 4.4.1 Multi-head attention (MHA)

Figure 12 (top, left) shows the effect of ablating the MHA in each layer of the image encoder in the correct part of the dataset where the negation is in the *foil*. The strongest effect is found in layer 1 ( $\text{IAE}_{\text{MHA}}(1) = 1.04$ ). The remaining layers show mixed results with layers 6 and 12 slightly standing out from the rest.

Figure 12 (bottom, left) aids the interpretation of these results. It classifies each dataset instance based on whether the ablation worsened or improved CLIP’s classification score and whether



**Figure 11:** Negator-selective attention on the CANNOT dataset, to validate the results from Figure 9. The heatmaps indicate the degree of negator-selective attention for each attention head and layer. The bar charts show the average of each layer as an overall measure of negator-selective attention.



**Figure 12:** MHA image ablation effects of the correct segment, split by whether negation is in foil or caption. The upper bar charts indicate the ablation effects in each layer. The lower stacked bar charts indicate for each layer how many instances obtained an improved/worse and correct/incorrect classification as a result of ablation.

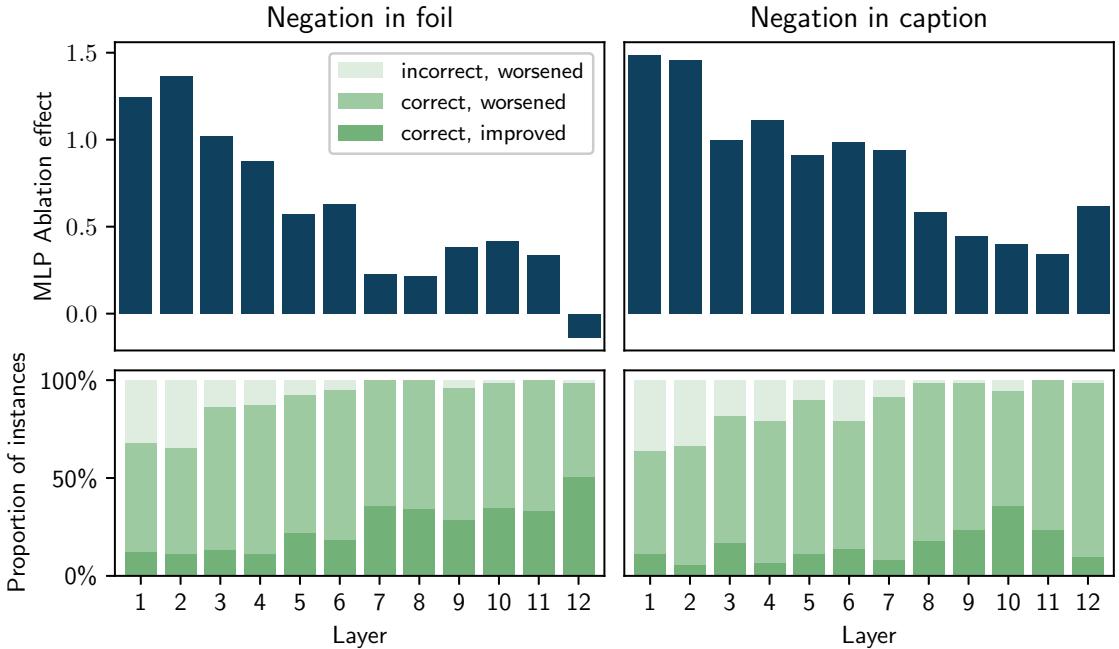
the ablated classification is correct or incorrect. Conceptually, ablation removes information from the model and should therefore worsen performance and the ablation effect is designed to quantify this deterioration in performance. However, in many layers we find that classification of a considerable proportion of instances in fact improves as a result of ablation, thus precluding

a straight-forward interpretation of the measured ablation effects. Even in layer 1 which seems to show the clearest effect, 18% of instances in fact show improved classification as a result of ablation.

The right part of Figure 12 shows similar results for the correct part of the dataset where the negation is in the *caption*. Again, the strongest effect is in layer 1 ( $\text{IAE}_{\text{MHA}}(1) = 1.24$ ), where this time only 11% of instances show improved performance. By contrast, none of the other layers show a clear ablation effect, with improved instances ranging from 24% to 43%.

In summary, these results provide tentative evidence for the role of the MHA in the first layer of the image encoder. However, given that ablation did in fact *improve* the classification of some proportion of instances, a straight-forward interpretation remains elusive. Possible reasons for these unintuitive results are discussed in Section 5.2.

#### 4.4.2 Multi-layer perceptron (MLP)



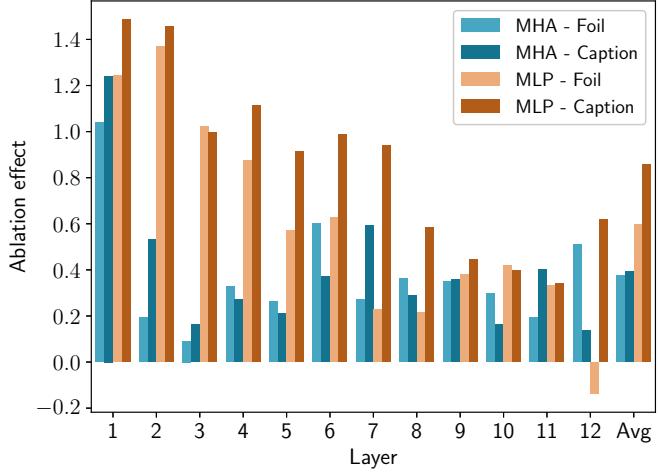
**Figure 13:** MLP image ablation effects of the correct segment, split by whether negation is in foil or caption. The upper bar charts indicate the ablation effects in each layer. The lower stacked bar charts indicate for each layer how many instances obtained an improved/worse and correct/incorrect classification as a result of ablation.

Figure 13 (top left) shows the image ablation results on the MLP for the correct part of the dataset where the negation is in the foil. Unlike the MHA results from Section 4.4.1, these results suggest a general trend with early layers being more important to the classification task than later layers (highest effect  $\text{IAE}_{\text{MLP}}(2) = 1.37$ ). While there remains a non-negligible proportion of improved instances even in the early layers 1-4 (11–13%), the pattern here allows for a less ambiguous interpretation than in Section 4.4.1.

The right part of Figure 13, where the negation is in the caption, again shows the highest effect in the early layers ( $\text{IAE}_{\text{MLP}}(1) = 1.49$ ). It also shows higher absolute effects, compared to instances where the negation is in the foil. These differences are highlighted in Section 4.4.3.

#### 4.4.3 Comparison of the role of MHA and MLP

Figure 14 shows a comparison of all image ablation effects discussed above. Most notably, it confirms that the MLP, shows an almost twice as high effect than the MHA (avg.  $\text{IAE}_{\text{MLP}} = 0.73$ , avg.  $\text{IAE}_{\text{MHA}} = 0.38$ ). The effect of the MLP is greater in every layer except for layer 12, suggesting it plays a more important role in the VALSE Existence task. Furthermore, the effect of the MLP is greater in cases where the negation is in the caption (avg. caption  $\text{IAE}_{\text{MLP}} = 0.86$  vs. avg. foil  $\text{IAE}_{\text{MLP}} = 0.60$ ), whereas in the case of the MHA the negation position does not seem to play a role. Reasons for these effects will be discussed in Section 5.2.



**Figure 14:** Comparison of image ablation results of the correct segment. Results are split by MHA and MLP and by whether negation is in foil or caption. The rightmost set of bars indicate the average effects across all layers.

## 5 Discussion

### 5.1 Localisation patterns in text encoder

The causal tracing results from Section 4.2 suggest that the processing of negation in CLIP’s text encoder is relatively localised in the early (1-3) and late (8-12) layers. This means that negation is largely represented at singular positions, namely at the negator position in the early layers and the final position in the late layers. Since the CTE at other positions in these layers is relatively low, we can infer that the model does not represent much negation-related information at these positions.

In layer 4, the amount of information pertaining to negation represented at the negator position drops sharply, as indicated by the drop in CTE, and this coincides with the finding of negator-selective attention heads in layer 4. The locations of these attention heads overlap with those found on the CANNOT dataset, which provides initial evidence that the CLIP text encoder uses some attention heads for specific syntactic functions. This would align with results from other language models that demonstrated syntactic functions in various attention heads (Clark et al., 2019; Vig and Belinkov, 2019). Further experiments on more diverse datasets containing negation could add to the robustness of these findings, while expanding the present analyses to other pieces of the VALSE benchmark might reveal attention heads responsible for processing other linguistic phenomena.

Two results stand out with respect to the middle layers. First, localisation is generally lower

than in the early and late layers, with no single position restoring more than 60% of the original effect. Second, the first subject token position appears to play a unique role in cases where the negation is in the caption.

As for the low overall localisation, this implies that CLIP represents negation at more than one position in these layers and that no single position on its own contains enough information to restore the majority of the effect. In fact, in cases where the negation is in the foil, the effects of all positions in these layers sum up to less than 1, suggesting that “the whole is greater than the sum of its parts” and that the model relies on *combining* the representations at each position in order to make correct judgements about negations. The present methodology only analyses simple effects of individual positions in individual layers but follow-up experiments could investigate the interactive effect of pairs of positions by simultaneously patching the activations at multiple positions in a given layer. Alternatively, a series of probing experiments could further test this hypothesis of distributed negation information. Here, probes could be trained on the activations of individual layer-position pairs to predict the presence or absence of negation in the input. If the conclusions drawn here with respect to localisation are correct, then we would expect probes trained on positions in early and late layers with high CTE to be more accurate than those trained on positions in the middle layers. If it is true that no individual position in the middle layers sufficiently represents negation, then probes should in fact be unable to recognise negation in these layers.

As for the unique role of the first subject token position in cases where the caption is negated, it might be instructive to consider the asymmetrical nature of the tasks carried out by the model, when the negation is the foil or caption, respectively. When the negation is in the foil, then the label’s subject is shown in the image and, intuitively, once it is detected, a decision can be made and no further processing is necessary. Conversely, when the negation is in the caption, the entire image needs to be scanned to ensure that the label’s caption is in fact absent from all parts of the image. This difference does not allow for a straight-forward explanation of the asymmetry found in the causal tracing results of the text encoder but it could be part of the reason why the first subject token position appears to play a role up until deeper layers of the network, when the negation is in the caption. The effects of the subject in deeper layers could imply that the subject information is in fact more deeply processed and thus more strongly represented in the final text encoder’s output which, in turn, could be conducive to the model’s task of “searching” for the subject in the image’s representation. However, it must be noted that these explanations are speculative and must not be accepted without further experiments. Such experiments could, for instance, analyse patterns of attention paid to the subject positions or investigate whether ablating or obscuring subject information in the input affects cases with negated caption more than those with negated foils.

Further experiments could also refine these results by distinguishing the roles of MHA and MLP. The finding of negator-selective attention heads have highlighted the importance of MHA in certain layers and positions but the present results do not indicate to what extent the MHA rely on subsequent processing in the MLP or whether, conversely, ablating the MLP would not harm model’s performance.

As a final general comment on these results, it remains difficult to evaluate the overall levels of localisation found here without comparison to other tasks and potentially other models. While the degree of localisation found here does appear substantial it may or may not differ significantly from other model behaviours.

## 5.2 Localisation patterns in image encoder

The analyses carried out on the image encoder have produced less clear results than those on the text encoder. At least two questions require further discussion. First, why is the effect of the MLP higher than that of the MHA? Second, why does ablation often seem to improve classification performance?

The results suggest that the MLP in the image encoder is more important than the MHA, although it must be noted that this effect is not overly large with the average MLP effect being 1.8 times higher than the average MHA effect. Given that the vast majority of Transformer interpretability literature has focused on text transformers, not much is known about the distinct roles of MHA and MLP in vision transformers. The present findings do seem to contradict those of Gandelsman et al. (2023) who reported that the MLP can be ablated without affecting model performance and that only the late MHA modules seem to play a critical role. However, a direct comparison between these results does not seem justified given the difference in dataset size, test task, and ablation method.

Elhage et al. (2021), focusing on text transformers, describe the role of MHA as moving information from and to different positions. Compared to typical transformer interpretability studies in the text domain where the input often consists of single sentences, the number of input positions in the vision transformer tends to be higher (in the present study, images are split into  $7 \times 7$  patches with one classification token prepended, yielding 50 positions). As such, we might hypothesise that the representation at each individual position contains less information than is the case with typical text input which is distributed over fewer positions. Also note that the sequence of positions in the image encoder is obtained by flattening 2-dimensional information into one dimension. Thus, here the role of attention heads must be partly to restore this 2-dimensional information, which arguably is a more complex operation than the purely 1-dimension task of MHA in text encoders.

Taken together, this could imply that the magnitude of the MHA’s task in the image encoder is such that it requires to be completed over multiple layers. In other words, the operations at each individual layer are insufficient to produce an accurate representation of the whole image. Recall that the present method only ablates one layer at a time and cannot make statements about the combined role of multiple layers. As such, it could be that the full role of the MHA only becomes evident when ablating full sets of layers. To validate these assumptions about the role of the MHA in the image encoder, it would be useful to directly analyse its attention patterns in future experiments. Such work could, for example, test for the existence of horizontally and vertically oriented attention heads and analyse the “attention distance” of different heads. This concept was proposed by Dosovitskiy et al. (2021) to indicate whether heads attend only to positions representing their immediate surrounding area or in fact to positions representing more distant parts of the image. While they found that the original ViT model did in fact show high attention distance already in early layers, these results do not necessarily have to generalise to CLIP’s image encoder, given its different training paradigm and dataset.

Alternatively, it might be the case that given the higher number of positions and the lower amount of information represented at each position, “merely” shifting information between positions does not suffice to perform well at the task at hand. In this case, further processing of the information at each position is required and this task might be carried out by the MLPs which in fact operate on the output of the MHA in each layer. Hence, they play a larger role in this task. However, this is a more general claim about the operations inside of vision transformers and would lead us to expect other studies to confirm the role of the MLP and this is precisely not what Gandelsman et al. (2023) reported. In general, it must be noted that

either of these interpretations are highly speculative and require further interpretability studies to gain support.

While the present analysis has suggested that the MLP plays a more important role than the MHA, the diffuse results of ablation *improving* the classification of some instances perhaps call into question the validity of the method itself. The largest methodological shortcoming likely lies in the use of inputs (single-coloured images) which the model never learned to process correctly since they were not part of the training data. This is related to a criticism by Chan et al. (2022) who note that both mean and zero ablation take models off distribution in a way that can have unpredictable effects.

In light of this, it makes sense to expect an ablated component in the present study to behave erratically which does not necessarily need to translate into consistently worse classification performance, but rather sometimes randomly improves classification. In fact, it would be conceivable that ablation primarily distorts the image’s multimodal embedding in such a way as to reduce the image’s similarity with both caption *and* foil. If this was the case, we would expect significantly lower image-caption and image-foil similarities in ablated runs compared to the original scores. However, a brief follow-up analysis refuted this hypothesis, showing that overall ablation did not significantly reduce the image-caption or image-foil similarity.

To resolve these methodological shortcomings, a possible follow-up experiment could use an edited version of the original image where only the subject is removed, instead of a flat single-colour image. Note that merely masking the image subject would not be sufficient since it potentially still allows the model to recognise the subject based on its contour. In addition, this would once again pose the problem of using “unrealistic” inputs that would put the model off distribution in possibly erratic ways. Instead, the image would need to be edited in such a way that the contour is filled in with realistic background imagery.

### 5.3 Causes of CLIP’s moderate classification performance

Section 4.1 highlights some variables that appear to play a role in CLIP’s classification performance on the VALSE Existence dataset. In particular, the label’s length and the subject’s size in the image show non-negligible correlations with respect to the classification score. This suggests that CLIP is better at the VALSE Existence task when labels are shorter and therefore produce less similar multimodal embeddings and when the subject in the image is sufficiently large.

Further experiments could aim to identify the role of other input variables, for example image features like contrast and brightness, and quantify the combined predictive power of these variables with respect to CLIP’s classification score. Arguably, the more variance in classification score can be explained on the basis of such variables, the less CLIP’s benchmark score can be interpreted as an indicator of its linguistic understanding, thus calling into question the validity of the VALSE benchmark. However it must be noted, that none of the correlations found in the present study are particularly high and thus further analyses are needed to support this conclusion.

A last point worthy of discussion pertains to the suitability of localisation methods for analysing model behaviour that is shown with only moderate reliability. Note that the methods used in the present study had originally been proposed and applied to language model capabilities that are shown reliably across a large corpus of data, e.g., indirect object identification (Wang et al., 2022), simple factual knowledge (Meng et al., 2023), or docstring completion (Heimersheim and Janiak, 2023). By contrast, the capability of interest in the present study, namely understanding

negation in a multimodal context, is not shown reliably (CLIP’s accuracy is only 66.9%) and only across a relatively small dataset ( $n = 490$ ).

Methods like causal tracing and mean ablation can be a powerful method for discovering mechanisms underlying consistently shown behaviour across a homogenous dataset. However, they do not intuitively lend themselves to *comparing* situations in which model behaviour is shown with those in which it is not shown. That is because they focus on the degree to which an effect that represents a particular model behaviour can be restored or ablated, but as such the methodology breaks down when said effect isn’t present in the first place.

Taken together, this points to two methodological shortcomings of the present study. First, whilst illuminating the role of various components in the CLIP model in processing negation, it cannot provide strong insights into why this processing yields correct classifications only in a fraction of cases. Second, since correct classification only occurs in a subset of instances of VALSE, which only featured a moderately sized dataset in the first place, the results described here are based on only a small number of samples and require a larger and potentially more diverse dataset to obtain greater validity.

## 6 Conclusion

This thesis analysed the components involved in CLIP’s performance on the VALSE existence benchmark. It discovered localised information processing in the text encoder, facilitated by negator-selective attention heads. Analyses on the image encoder yielded more diffuse results that might be caused by methodological shortcomings and don’t permit definitive interpretations. Furthermore, dataset variables (chiefly sequence length and subject size in the image) were identified that appear to explain part of CLIP’s only moderate performance on VALSE existence. Importantly, these results can be seen as covariate effects, suggesting that performance on VALSE existence cannot solely be interpreted as an indicator of the model’s linguistic understanding. Future research should aim to validate these findings on larger, more diverse datasets, as well as to arrive at more rigorous and robust descriptions of the mechanisms this study has discovered. In particular, focus should be placed on studying other linguistic phenomena as a way to benchmark the degree of localisation found here. Such studies could also broaden our understanding of attention heads that serve particular syntactic functions. In addition, it remains to be seen how far the present findings generalise to other VL models, especially those with distinct architectures from the one studied here.

## References

- A. Akula, S. Gella, Y. Al-Onaizan, S.-C. Zhu, and S. Reddy. Words Aren't Enough, Their Order Matters: On the Robustness of Grounding Visual Referring Expressions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6555–6565, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.586. URL <https://www.aclweb.org/anthology/2020.acl-main.586>.
- J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, and K. Simonyan. Flamingo: A Visual Language Model for Few-Shot Learning, Nov. 2022. URL <http://arxiv.org/abs/2204.14198>.
- M. Anschütz, D. M. Lozano, and G. Groh. This is not correct! Negation-aware Evaluation of Language Generation Systems, July 2023. URL <http://arxiv.org/abs/2307.13989>.
- J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer Normalization, July 2016. URL <http://arxiv.org/abs/1607.06450>.
- T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020. URL <https://papers.nips.cc/paper/2020/hash/1457c0d6bfc4967418bfb8ac142f64a-Abstract.html>.
- E. Bugliarello, R. Cotterell, N. Okazaki, and D. Elliott. Multimodal Pretraining Unmasked: A Meta-Analysis and a Unified Framework of Vision-and-Language BERTs, May 2021. URL <http://arxiv.org/abs/2011.15124>.
- E. Bugliarello, L. Sartran, A. Agrawal, L. A. Hendricks, and A. Nematzadeh. Measuring Progress in Fine-grained Vision-and-Language Understanding, May 2023. URL <http://arxiv.org/abs/2305.07558>.
- L. Chan, A. Garriga-Alonso, N. Goldwosky-Dill, R. Greenblatt, J. Nitishinskaya, A. Radhakrishnan, B. Shlegeris, and N. Thomas. Causal scrubbing, a method for rigorously testing interpretability hypotheses. *AI Alignment Forum*, 2022.
- K. Clark, U. Khandelwal, O. Levy, and C. D. Manning. What Does BERT Look at? An Analysis of BERT's Attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy, Aug. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4828. URL <https://aclanthology.org/W19-4828>.
- R. Csordás, S. van Steenkiste, and J. Schmidhuber. Are Neural Nets Modular? Inspecting Functional Modularity Through Differentiable Weight Masks, Mar. 2021. URL <http://arxiv.org/abs/2010.02066>.
- A. Das and P. Rad. Opportunities and Challenges in Explainable Artificial Intelligence (XAI): A Survey, June 2020. URL <http://arxiv.org/abs/2006.11371>.
- J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional

- Transformers for Language Understanding, May 2019. URL <http://arxiv.org/abs/1810.04805>.
- A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, June 2021. URL <http://arxiv.org/abs/2010.11929>.
- N. Elhage, N. Nanda, C. Olsson, T. Henighan, N. Joseph, B. Mann, A. Askell, Y. Bai, A. Chen, T. Conerly, N. DasSarma, D. Drain, D. Ganguli, Z. Hatfield-Dodds, D. Hernandez, A. Jones, J. Kernion, L. Lovitt, K. Ndousse, D. Amodei, T. Brown, J. Clark, J. Kaplan, S. McCandlish, and C. Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL <https://transformer-circuits.pub/2021/framework/index.html>.
- Y. Gandelsman, A. A. Efros, and J. Steinhardt. Interpreting CLIP's Image Representation via Text-Based Decomposition, Oct. 2023. URL <http://arxiv.org/abs/2310.05916>.
- A. Ghorbani and J. Zou. Neuron Shapley: Discovering the Responsible Neurons, Nov. 2020. URL <http://arxiv.org/abs/2002.09815>.
- G. Goh, N. C. †, C. V. †, S. Carter, M. Petrov, L. Schubert, A. Radford, and C. Olah. Multimodal Neurons in Artificial Neural Networks. *Distill*, 6(3):e30, Mar. 2021. ISSN 2476-0757. doi: 10.23915/distill.00030. URL <https://distill.pub/2021/multimodal-neurons>.
- Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering, May 2017. URL <http://arxiv.org/abs/1612.00837>.
- P. Hase, M. Bansal, B. Kim, and A. Ghandeharioun. Does Localization Inform Editing? Surprising Differences in Causality-Based Localization vs. Knowledge Editing in Language Models, Jan. 2023. URL <http://arxiv.org/abs/2301.04213>.
- K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition, Dec. 2015. URL <http://arxiv.org/abs/1512.03385>.
- S. Heimersheim and J. Janiak. A circuit for Python docstrings in a 4-layer attention-only transformer. *Alignment Forum*, Feb. 2023. URL <https://www.alignmentforum.org/posts/u6KXXmKFbXfWzoAXn/a-circuit-for-python-docstrings-in-a-4-layer-attention-only>.
- L. A. Hendricks and A. Nematzadeh. Probing Image-Language Transformers for Verb Understanding, June 2021. URL <http://arxiv.org/abs/2106.09141>.
- J. Hoelscher-Obermaier, J. Persson, E. Kran, I. Konstas, and F. Berez. Detecting Edit Failures In Large Language Models: An Improved Specificity Benchmark, June 2023. URL <http://arxiv.org/abs/2305.17553>.
- M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd. spaCy: Industrial-strength natural language processing in python. 2020. doi: 10.5281/zenodo.1212303.
- A. Jabri, A. Joulin, and L. van der Maaten. Revisiting Visual Question Answering Baselines, Nov. 2016. URL <http://arxiv.org/abs/1606.08390>.
- C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. V. Le, Y. Sung, Z. Li, and T. Duerig. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision, June 2021. URL <http://arxiv.org/abs/2102.05918>.

- O. Kovaleva, A. Romanov, A. Rogers, and A. Rumshisky. Revealing the Dark Secrets of BERT. In K. Inui, J. Jiang, V. Ng, and X. Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4365–4374, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1445. URL <https://aclanthology.org/D19-1445>.
- M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL <https://aclanthology.org/2020.acl-main.703>.
- J. Li, R. R. Selvaraju, A. D. Gotmare, S. Joty, C. Xiong, and S. Hoi. Align before Fuse: Vision and Language Representation Learning with Momentum Distillation, Oct. 2021. URL <http://arxiv.org/abs/2107.07651>.
- J. Li, D. Li, C. Xiong, and S. Hoi. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation, Feb. 2022. URL <http://arxiv.org/abs/2201.12086>.
- J. Li, D. Li, S. Savarese, and S. Hoi. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models, June 2023. URL <http://arxiv.org/abs/2301.12597>.
- L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang. VisualBERT: A Simple and Performant Baseline for Vision and Language, Aug. 2019. URL <http://arxiv.org/abs/1908.03557>.
- X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, Y. Choi, and J. Gao. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks, July 2020. URL <http://arxiv.org/abs/2004.06165>.
- F. Liu, G. Emerson, and N. Collier. Visual Spatial Reasoning, Mar. 2023. URL <http://arxiv.org/abs/2205.00363>.
- J. Lu, D. Batra, D. Parikh, and S. Lee. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks, Aug. 2019. URL <http://arxiv.org/abs/1908.02265>.
- T. Lüddecke and A. S. Ecker. Image Segmentation Using Text and Image Prompts, Mar. 2022. URL <http://arxiv.org/abs/2112.10003>.
- S. M. Lundberg and S.-I. Lee. A Unified Approach to Interpreting Model Predictions. In *Neural Information Processing Systems (NIPS)*, volume 31, Long Beach, CA, USA, 2017.
- Z. Ma, J. Hong, M. O. Gul, M. Gandhi, I. Gao, and R. Krishna. CREPE: Can Vision-Language Foundation Models Reason Compositionally?, May 2023. URL <http://arxiv.org/abs/2212.07796>.
- K. Meng, D. Bau, A. Andonian, and Y. Belinkov. Locating and Editing Factual Associations in GPT, Jan. 2023. URL <http://arxiv.org/abs/2202.05262>.
- R. Mokady, A. Hertz, and A. H. Bermano. ClipCap: CLIP Prefix for Image Captioning, Nov. 2021. URL <http://arxiv.org/abs/2111.09734>.

- J. Mu, S. Bhat, and P. Viswanath. All-but-the-Top: Simple and Effective Postprocessing for Word Representations, Mar. 2018. URL <http://arxiv.org/abs/1702.01417>.
- N. Nanda, L. Chan, T. Lieberum, J. Smith, and J. Steinhardt. Progress measures for grokking via mechanistic interpretability, Oct. 2023. URL <http://arxiv.org/abs/2301.05217>.
- L. Parcalabescu and A. Frank. MM-SHAP: A Performance-agnostic Metric for Measuring Multimodal Contributions in Vision and Language Models & Tasks, May 2023. URL <http://arxiv.org/abs/2212.08158>.
- L. Parcalabescu, M. Cafagna, L. Muradjan, A. Frank, I. Calixto, and A. Gatt. VALSE: A Task-Independent Benchmark for Vision and Language Models Centered on Linguistic Phenomena. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8253–8280, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.567. URL <https://aclanthology.org/2022.acl-long.567>.
- J. Pearl. Direct and indirect effects. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, UAI'01, pages 411–420, San Francisco, CA, USA, Aug. 2001. Morgan Kaufmann Publishers Inc. ISBN 978-1-55860-800-9.
- A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language Models are Unsupervised Multitask Learners. *OpenAI*, 2019.
- A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning Transferable Visual Models From Natural Language Supervision, Feb. 2021. URL <http://arxiv.org/abs/2103.00020>.
- C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):140:5485–140:5551, Jan. 2020. ISSN 1532-4435.
- A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents, Apr. 2022. URL <http://arxiv.org/abs/2204.06125>.
- T. Räuker, A. Ho, S. Casper, and D. Hadfield-Menell. Toward Transparent AI: A Survey on Interpreting the Inner Structures of Deep Neural Networks, Sept. 2022. URL <http://arxiv.org/abs/2207.13243>.
- M. T. Ribeiro, S. Singh, and C. Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier, Aug. 2016. URL <http://arxiv.org/abs/1602.04938>.
- R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-Resolution Image Synthesis with Latent Diffusion Models, Apr. 2022. URL <http://arxiv.org/abs/2112.10752>.
- O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec. 2015. ISSN 1573-1405. doi: 10.1007/s11263-015-0816-y. URL <https://doi.org/10.1007/s11263-015-0816-y>.
- L. S. Shapley. Notes on the N-Person Game — II: The Value of an N-Person Game. Technical report, RAND Corporation, Aug. 1951. URL [https://www.rand.org/pubs/research\\_memoranda/RM0670.html](https://www.rand.org/pubs/research_memoranda/RM0670.html).

- R. Shekhar, S. Pezzelle, Y. Klimovich, A. Herbelot, M. Nabi, E. Sangineto, and R. Bernardi. FOIL it! Find One mismatch between Image and Language caption. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 255–265, Vancouver, Canada, 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1024. URL <http://aclweb.org/anthology/P17-1024>.
- A. Singh, R. Hu, V. Goswami, G. Couairon, W. Galuba, M. Rohrbach, and D. Kiela. FLAVA: A Foundational Language And Vision Alignment Model, Mar. 2022. URL <http://arxiv.org/abs/2112.04482>.
- W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai. VL-BERT: Pre-training of Generic Visual-Linguistic Representations, Feb. 2020. URL <http://arxiv.org/abs/1908.08530>.
- A. Suhr, S. Zhou, A. Zhang, I. Zhang, H. Bai, and Y. Artzi. A Corpus for Reasoning about Natural Language Grounded in Photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1644. URL <https://www.aclweb.org/anthology/P19-1644>.
- H. Tan and M. Bansal. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1514. URL <https://aclanthology.org/D19-1514>.
- T. Thrush, R. Jiang, M. Bartolo, A. Singh, A. Williams, D. Kiela, and C. Ross. Winoground: Probing Vision and Language Models for Visio-Linguistic Compositionality, Apr. 2022. URL <http://arxiv.org/abs/2204.03162>.
- T. H. Truong, T. Baldwin, K. Verspoor, and T. Cohn. Language models are not naysayers: An analysis of language models on negation benchmarks, June 2023. URL <http://arxiv.org/abs/2306.08189>.
- A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need, Dec. 2017. URL <http://arxiv.org/abs/1706.03762>.
- J. Vig and Y. Belinkov. Analyzing the Structure of Attention in a Transformer Language Model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 63–76, Florence, Italy, Aug. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4808. URL <https://aclanthology.org/W19-4808>.
- K. Wang, A. Variengien, A. Conmy, B. Shlegeris, and J. Steinhardt. Interpretability in the Wild: A Circuit for Indirect Object Identification in GPT-2 small, Nov. 2022. URL <http://arxiv.org/abs/2211.00593>.
- R. Wolfe and A. Caliskan. Contrastive Visual Semantic Pretraining Magnifies the Semantics of Natural Language Representations. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3050–3061, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.217. URL <https://aclanthology.org/2022.acl-long.217>.
- M. Yuksekgonul, F. Bianchi, P. Kalluri, D. Jurafsky, and J. Zou. When and why vision-language models behave like bags-of-words, and what to do about it?, Mar. 2023. URL <http://arxiv.org/abs/2210.01936>.

- Y. Zeng, X. Zhang, and H. Li. Multi-Grained Vision Language Pre-Training: Aligning Texts with Visual Concepts. In *Proceedings of the 39th International Conference on Machine Learning*, pages 25994–26009. PMLR, June 2022. URL <https://proceedings.mlr.press/v162/zeng22c.html>.
- P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao. VinVL: Revisiting Visual Representations in Vision-Language Models, Mar. 2021. URL <http://arxiv.org/abs/2101.00529>.
- B. Zhou, Y. Sun, D. Bau, and A. Torralba. Revisiting the Importance of Individual Units in CNNs via Ablation, June 2018. URL <http://arxiv.org/abs/1806.02891>.
- Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei. Visual7W: Grounded Question Answering in Images, Apr. 2016. URL <http://arxiv.org/abs/1511.03416>.