Identifying Question Ambiguity In Visual Question Answering

by

Anush Kumar Venkatesh

M.S., University of Colorado Boulder, 2024

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Masters of Science

Department of Computer Science

2024

Committee Members:

Dr. Danna Gurari, Chair

Dr. Tom Yeh

Dr. James Martin

Venkatesh, Anush Kumar (Computer Science)

Identifying Question Ambiguity In Visual Question Answering

Thesis directed by Dr. Danna Gurari

The task of visual question answering (VQA) is considered as the key benchmark to evaluate vision language understanding (VLU). However, annotation efforts for VQA tend to produce a variety of answers given for the same visual question. Although previous studies show many possible reasons for the variety of answers, ambiguity plays a major role. In this work, we observe that ambiguity can be further divided into two kinds. First is question ambiguity, when the entity specified in the question cannot be unambiguously mapped to in the image. Second is answer ambiguity, a single question may have multiple valid answers. In our work, we focus on question ambiguity. We enrich the recently published VQA-AnswerTherapy dataset by adding two attributes: first, the entity that users will likely search for based on the question and second, an indication of whether the object can be unambiguously resolved when examining the image. This enrichment will enable us to identify what portion of the dataset has multiple answers due to question ambiguity. Additionally, we evaluate the capability of open-source foundation models to (1) determine whether a visual question is ambiguous (2) identify whether different answers for a visual question point to same visual content and (3) generate all variety of answers possible for a visual question via prompting. This will help evaluate the performance of existing foundation models on the above tasks, especially when dealing with ambiguous visual questions.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

The field of vision language understanding (VLU) includes developing computer systems with the capability to understand an image and reason about it based on text. More generally, the aim is to develop algorithms capable of linking a visual concept and its language representation. Initially, image captioning [15, 13, 23] was the primary task to evaluate VLU. Then, the introduction of a dataset by [1] made the task of visual question answering (VQA) a very relevant problem in the field of VLU as well. VQA entails presenting an image and a corresponding question in order to receive a relevant answer.

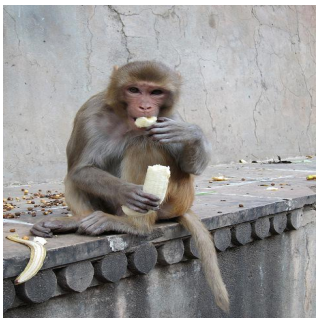
VQA datasets [1, 26, 19, 6] are often curated with crowdsourcing, where users are presented an image along with a question and tasked with providing an answer. A resulting challenge from this approach is that, in practice, crowdsourced answers from different people exhibit a lot of variability. This is exemplified in Figure 1.1.

Previous studies [4, 18, 6] have attempted to understand why different answers are observed. Several reasons include difficult questions, subjectivity, and low quality images, with the most common reason for answer variability being ambiguity. Upon careful observation we notice that not every visual question labeled as ambiguous exhibits a similar kind of ambiguity. We claim that this can be further broken down into (1) question ambiguity and (2) answer ambiguity.

We define question ambiguity as the situation where the entity referenced in the question cannot be unambiguously identified upon viewing its corresponding image. For example, consider the visual question "What is the boy wearing on his head?" The entity referred to here is "the boy".



Question:	What color is the car?	What is the man wearing on his head?	What is this?
Answers:	silver black blue	cap hat beanie	table living room livingroom

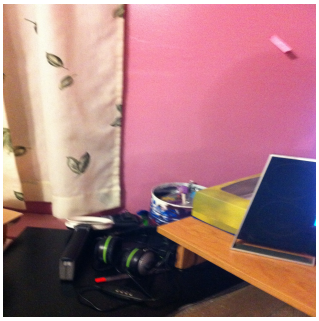


Question:	How many hairs does the monkey have?	What is this?	What does this shirt look like?
Answers:	many thousands lots	blue pen pen ink pen	animal print leopard print cheetah

Figure 1.1: Examples of visual questions containing different answers provided by crowdworkers

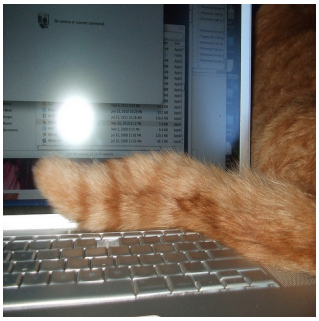
However, if the image contains multiple instances of "the boy", the answer may vary depending on which boy the question is referring to. The only scenario where ambiguity is unlikely is when the image depicts only one instance of "the boy". In all other cases, additional context is needed to resolve the ambiguity. More such examples are provided in Figure 1.2.

We define answer ambiguity as the scenario where the entity referenced in the question can be identified without ambiguity, but there are multiple valid answers. Let's illustrate this with an example: consider the visual question "What is on the table?" The entity referred to here is "the table." Even if the image contains only one instance of "the table", there could be multiple objects


Question:	What kind of donut is the person eating?	What is this?	What kind of jacket is she wearing?
Entity:	donut the person is eating	this	jacket she is wearing

Figure 1.2: Examples of visual questions with question ambiguity



Question:	What is the cats tail laying on?	What dinner is this?	What is the man wearing?
Entity:	cats tail	dinner	man
Answers:	keyboard laptop	healthy choice chicken marinara	shorts life vest

Figure 1.3: Examples of visual questions with answer ambiguity

on it. Providing any one of these objects as an answer would be correct. Therefore, we classify situations where multiple correct answers are possible as instances of answer ambiguity. Further examples are provided in Figure 1.3.

In this work, we aim to understand the prevalence of visual questions exhibiting question ambiguity. To that end, we propose to enrich the existing VQA-AnswerTherapy [6] dataset with two additional attributes: "entity lookup" and "ambiguous question". The "entity lookup" attribute

helps identify what is the entity of interest upon looking at the question. The "ambiguous question" is a binary attribute that helps identify whether the visual question exhibits question ambiguity or not. We believe this exercise of dwelling deeper into a more fine grained analysis of ambiguity will help us develop more robust algorithms to either clarify the ambiguous question before providing an answer to the user or provide answers from all different perspectives.

In addition to creating a new dataset, we aim to evaluate the performance of state-of-the-art, open-source foundation models in identifying whether a visual question is ambiguous or not. Furthermore, we assess the models' ability to map visual concepts represented in text to corresponding locations in the image, which we believe is a crucial skill for machine learning models to discern ambiguity. Lastly, we also probe the foundation models' capability to predict all the diverse sets of answers, similar to the ones provided by the annotators. For these tasks we use different prompting methods on models such as Blip2 [33], InstructBlip [8], and OpenFlamingo [2] and report their performance. This exercise help us in understanding foundation models' capabilities when it encounters an ambiguous visual question versus an unambiguous visual question.

Thus, our main contribution in this work can be summarized as

(1) Enrich the existing VQA-AnswerTherapy [6] with attributes revealing visual questions that exhibit question ambiguity.

(2) Evaluate the capabilities of open source foundation models in identifying if the visual question is ambiguous.

(3) Evaluate the performance of open source foundation models to identify if certain visual concepts point to same or different visual content

(4) Evaluate the performance of open source foundation models in providing all diverse set of answers to mimic human responses.

Chapter 2

Background

2.1 Visual question answering

2.1.1 Visual question answering datasets

The recent wave of excitement around Visual question answering (VQA) algorithms started with the introduction of a dataset by [1], which was later followed by other works (e.g., [26, 19, 6]). The pioneering dataset [1] used images from MSCOCO [38] and employed crowdsourcing to gather questions and answers related to those images. [26] points out limitations in earlier VQA datasets, noting that questions could be answered through a good guess without considering the image. To overcome this, they introduce GQA, a dataset requiring multi-step reasoning abilities to answer visual questions. In [19], the authors emphasize real-world applications for VQA by introducing a dataset comprising images and questions posed by visually impaired individuals. They evaluate the performance of state-of-the-art VQA algorithms at the time and observe a significant decrease in accuracy on this dataset, underscoring the necessity for more robust VQA algorithms.

Straying from the traditional design of VQA datasets, [6] explores the variability in answers from crowdsourced individuals and investigates whether diverse answers correspond to the same or different visual evidence. They introduce a dataset containing images from both VizWiz [19] and VQAv2 [17], revealing that approximately 15.6% of differing answers often point to distinct visual evidence. This prompts new research directions, raising questions about whether computers should emulate human diversity in providing different answers to the same question or take measures to

ensure a single correct answer for a visual question.

Extending prior work, our primary focus is on exploring ambiguity in VQA at a more granular level. Specifically, we build a new dataset that reveals the extent to which the VQA-AnswerTherapy dataset [6] contains ambiguous questions. This dataset can also aid in the development of more robust models capable of handling ambiguity effectively.

2.1.2 Visual question answering algorithms

Visual question answering algorithms have evolved alongside advancements in neural network architectures. Initially, researchers primarily utilized different architectures to extract linguistic and visual features, employing LSTMs and CNNs respectively. These features were later fused into multimodal features using simple techniques such as concatenation. However, with the introduction of transformer-based architectures, we have witnessed a significant shift in approach. Recently, both visual and linguistic features are being extracted using transformer-based architectures. Below, we outline the approaches taken in two phases.

Phase 1: One of the initial baseline algorithms for VQA was presented in [1]. The authors proposed a simple approach: extracting visual features with CNNs and linguistic features with RNNs, then combining them and passing them through an MLP layer to classify the answer to the question. Building upon the baseline proposed in [1], subsequent works [44, 67, 69, 9] aimed at refining the approach to advance the state-of-the-art.

In [44], a similar strategy to the baseline in [1] is employed, but instead of predicting answers through a softmax layer, a generative approach is taken. Visual features from a CNN are appended to each timestep of the question encoding by an LSTM. An LSTM-based decoder is then used to generate answers. This approach offers the advantage of producing answers with multiple words as opposed to single-word answers.

[67] highlight the problem in VQA systems to have a superficial prior, meaning that few answers to the visual questions can be answered without the need to look at the image. To tackle this, the authors propose a novel approach. For each image and its corresponding visual question,

they crowdsource a counterexample, depicting how the image would look if the answer were complementary to the original. This approach compels the VQA system to genuinely comprehend the image before responding to the question. They further frame the questions to elicit yes/no-style answers, and their algorithm involves developing a binary VQA model.

[69] addresses the challenge of loose global representation in VQA models, leading to less control and explainability. They introduce grounding in VQA to overcome this limitation. Visual question-answer pairs are collected via crowdsourcing, with additional annotations for objects referred to in the questions and answers. Their algorithm involves an encoding stage where a visual feature, encoded using VGG, is combined with token embeddings of the question word by word. At each timestep, the model computes attention for each word over visual features, determining which region of the image to focus on. A simple dot product between answer choices and the encoded multimodal feature is then used to determine the final answer.

[9] raises the bar in the complexity of VQA tasks. They propose that systems should be capable of carrying a conversation between the human to answer all the questions related to the image. They create a dialogue-based dataset through crowdsourcing and experiment with three types of encoders and two types of decoders. The first encoder type involves separate encoding of the image and subsequent questions and answers, deriving a final global feature through fusion. The second type employs recurrent neural networks to jointly embed the image and question-answer pairs, taking advantage of the hierarchical nature of the conversation. The third type develops a memory network where each question-answer pair is encoded into a memory bank for retention before answering the next question. They propose two decoders, one generative with LSTMs and one discriminative. Their results show that using a memory network in combination with a generative decoder produces the best result.

Phase 2: Like other domains, recent works [58, 43, 37, 31, 55, 54] showcase the increasing trend in the usage of transformers for VQA. [58] was among the pioneers in adopting transformers for VQA. Their approach involves extracting object-level features for an image and appending them with position embeddings. Text is converted to its respective word embeddings, and both object

features and word embeddings are passed through unimodal encoders based on the transformer architecture. These unimodal features are then combined and processed through a cross-modal encoder, incorporating a special [CLS] token. The architecture is pre-trained using five different loss functions and subsequently fine-tuned for VQA, achieving superior performance compared to the existing state-of-the-art at that time. Curiously, [43] adopts a similar architecture to [58]. However, instead of utilizing a cross-modal transformer for merging visual and linguistic features, they opt for two transformers with alternating co-attention and self-attention layers. The model undergoes pre-training on multimodal alignment tasks before fine-tuning specifically for VQA.

In [37], the authors highlight that simply concatenating vision and linguistic features and running them through multimodal architectures provide weak supervision for object to text alignment. To solve the issue, they introduce a pre-training strategy where the underlying architecture is similar but the dataset is treated as triplets: linguistic features, linguistic features of words corresponding to objects in the image, and visual features. To pretrain the network, two loss functions are employed; the first involves masked token loss, where the model predicts masked object tokens, and the second uses a contrastive loss. This loss replaces some visual representations with noisy image representations, challenging the model to identify alignment between linguistic and visual features. Upon fine-tuning on the VQA dataset, this approach demonstrated a significant improvement over the previous benchmark.

In [31], the authors demonstrate the powerful representational capabilities of the transformer architecture. They identify that most of the previous approaches often are computation heavy with the unimodal feature extraction processes. Here, they demonstrate that just using a simple embedding layer for unimodal features and passing them onto a cross modal attention transformer in itself can showcase comparable performances to the existing state-of-the-art.

In [55], the authors investigate the benefits of employing a CLIP-based model for VQA. They utilize the CLIP visual encoder to generate image features. For text processing, a T5 language model is employed to create an answer template with a [mask] token. This template is then filled with all potential answers before extracting linguistic features. The answer with the highest CLIP

score is chosen as the prediction.

[54] inspired by [58, 43] try to create a single foundational model that gives good performance with unimodal as well as multimodal tasks. Their model architecture is similar to that chosen in [58] and [43]. However, instead of pretraining the entire model with a single multimodal task, the authors separately pretrain the unimodal encoders and later also train the multimodal encoders. This approach helps the entire model to be more versatile across a range of tasks. They also show a significant improvement in VQA performance.

In our study, we explore various prompting-based methods to evaluate the performance of off-the-shelf open-source foundation models (namely [33], InstructBlip [8], and OpenFlamingo [2]) on the newly proposed dataset. Our focus lies in assessing the capability of these foundation models in addressing visual questions with inherent ambiguity. We aim to determine whether they can provide a diverse set of answers when faced with ambiguity.

2.2 Phrase grounding/Referring expression comprehension

2.2.1 Phrase grounding/Referring expression comprehension Datasets

Many visual grounding datasets exist today. Some early attempts that engaged the community in developing algorithms for visual grounding were datasets called ReferIt [29], Flickr30k Entities [47], gRefCOCO [45], and Visual Genome [32]. Later efforts by [7, 56] aim to further challenge the algorithms requiring them to have both compositional reasoning skills and understanding long-form text.

ReferIt [29] used gamification to gather pairs of object groundings with corresponding referring expressions. In this setup, one user views an image with a highlighted object and is tasked with providing a referring expression. Another user then looks at the same image, reads the given referring expression, and selects the corresponding object. If the selections match, the data point is added to the collection. This approach resulted in the collection of over 100k referring expressions.

Flickr30k Entities [47] expanded on the existing Flickr30k dataset [64], which contains image-

to-sentence descriptions. Annotators were initially tasked with building a coreference chain for a given caption. Subsequently, bounding boxes were provided for the identified coreference chain in the image. The main limitations of [64, 47] were that it contained very few objects per image, making it easy for detection-based algorithms to find the referred object.

[45] identify one of the key drawbacks in ReferIt [29]. They observed that most images in ReferIt [29] contain only one object. This bias could cause machine learning algorithms to simply detect the object in the image without understanding the semantics of the referring expression. To address this issue, they introduced a new dataset called gRefCOCO, which ensures multiple object instances per image.

Subsequently, Visual Genome [32] was introduced to ensure that each image has a dense set of objects with multiple different attributes for each unique object. With the help of crowdsourcing, Visual Genome [32] contains about 100,000 images, with each image having an average of 35 referring expressions, allowing for a dense set of annotations.

The datasets introduced by [29, 47, 32] laid the groundwork for significant improvements in phrase grounding algorithms. However, more recently, a few datasets [7, 56] have emerged to further challenge these algorithms. [7] highlight two main drawbacks of existing datasets. Firstly, there is a shortage of referring expressions that require compositional reasoning. Secondly, there is a lack of distractor images for each referring expression. To address these issues, they propose a new dataset containing an abundance of referring expressions necessitating compositional reasoning skills and provide distractor images for each data point. As a result, algorithms will need to not only resolve the referring expression but also select the correct image from the given set of images.

[56] is one the most recently introduced datasets for visual grounding. The authors highlight that a simple LSTM based encoder without much pre-training results in near state-of-the-art performance. Thus they come up with a new dataset where an image along with its RE is not enough to accurately coming with its corresponding grounding but also needs access to a description of the scene depicted in the image. This leads to the development of new algorithms to also accomodate for the scene descriptions.

In our work, we propose a new dataset that helps us identify the referring expression within the visual question. Unlike prior works, our dataset also marks the referring expression within visual question even with the existence of vague or underspecified words such as “that” or “this”. Our work also considers referrring expressions in the question that could point to multiple regions in the image.

In our research, we introduce a novel dataset that halps us identify the referring expressions from within the visual questions. Unlike previous efforts, our dataset also provides referring expressions to visual questions even in the presence of vague or underspecified terms like ”that” or ”this”. Additionally, our work takes into account referring expressions that may indicate multiple regions within the image. To the best of our knowledge, only [39] provides a referring expression dataset that points to more than one image region.

2.2.2 Phrase grounding/Referring expression comprehension algorithms

Visual grounding algorithms have undergone significant evolution in the past decade that span two key phases. In the first phase, many methods [50, 45, 65, 21, 61, 66, 59] relied on a combination of CNNs and LSTMs. In the second phase, there was a shift towards using transformer-based modules to detect a wider range of objects, meaning the objects to be recognized are not predefined.

Phase 1: One of the earlier works of this phase was [50], which used weak supervision from bounding box proposals generated for an image. The goal is to identify which bounding box aligns well with the given referring expression (REC). These bounding boxes can also be used to regenerate RECs, aiding in determining model loss during training. This method is advantageous as it doesn’t require a large dataset for bounding box proposals.

[45] follows a similar approach, where the LSTM module generates RECs for each bounding box feature extracted from the CNN module. A softmax module is then employed to select the bounding box that closely corresponds to the target REC.

[65] is akin to [45], but the focus here is on generating highly discriminative sentences for

each bounding box proposal in the image. This is achieved by identifying simple differences in each region feature compared to others, encoding it, and appending it to the same region feature. This ensures the LSTM has information on how to discriminate each region in the image.

[21] emphasizes that previous work does not account for identifying how different objects in the REC are connected. The authors break the referring expression into subject, object, and relation pairs, using a language parser and localize them in the image. Later they have separate modules to identify the bounding box proposal matching the relation between different objects in the image. This approach significantly pushed the existing state-of-the-art.

[61] is another example that takes advantage of the structure in language. The main idea is to produce attention masks for sub phrases in the referring expression that are compositional in nature. Simple noun phrases in language can be composed into bigger phrases, and the authors aim for the attention masks of the bigger phrase to be a composition of small masks. Using structure also helps the authors train the neural network with an additional structural loss.

[66] maintains a general framework for visual grounding similar to the ones mentioned earlier but with a focus on improving the quality and reducing the number of object proposals to simplify the downstream network. The approach involves training a network based on Faster-RCNN on the Visual Genome dataset to expand the target classes for object proposals. The resulting region proposals, combined with language embeddings, go through an MLP layer to generate softmax scores for all object proposals. The top K proposals are then sent downstream for further predictions.

In [59], the authors intend to represent regions in an image as a graph, where nodes correspond to nouns/objects. Only the top K objects near each other are connected by an edge. They employ a language-guided attention module to identify the best-matching nodes in the subgraph to the given referring expression (REC) and determine the target region.

Phase 2: The success of the aforementioned approaches led to an expansion of the REC task to detect more open-ended objects and incorporate usage of transformer based modules [51, 62, 10, 28, 35, 68, 34, 57].

In [51], the authors suggest moving away from traditional object proposals in favor of anchor

boxes. This involves extracting image features using feature pyramid networks (FPNs), obtaining language features with an LSTM, and combining them with all the anchor box proposals to generate multimodal features. These features are then input into a classification-based module for predictions. The authors also introduce the use of focal loss to address the challenge of a large volume of negative samples with anchor boxes.

[62] focuses on solving the REC task in a single stage, marking the beginning of the adoption of transformers. The authors use FPNs to extract image features, combine them with BERT-based [11] language features, and pass them to a YOLO-inspired [49] module to predict anchor boxes and their corresponding confidence scores.

From [10], a complete shift to transformer-based approaches is evident. Initially, the authors feed ResNet features of the image into a vision transformer, extract text features from BERT, and fuse them with image features using a cross-attention transformer block. These multimodal features then go into a regression module for directly predicting bounding box coordinates.

Unlike previous approaches aiming to predict a single bounding box for the entire REC, [28], inspired by [5], introduces a framework to pretrain a system for identifying all bounding boxes for each subphrase in a given text. This enables the generation of high-quality multimodal embeddings with a richer understanding of correspondences between text and an image. The authors propose to train a transformer based module that uses cross attention on multimodal features (image and text) to produce a fixed number of bounding box predictions and its class by corresponding class distribution in the input text. This approach eliminates the need for a fixed set of target classes, making the module adaptable for various downstream tasks. In this case, the authors report achieving state-of-the-art results in the REC task. The downside of this approach is that input queries to the transformer are fixed which may result in the module always trying to look for the same object.

The authors from [35] circumvent this problem. They propose to use a transformer module to generate input query embedding which encodes the information that needs to be looked for in the image based on the input text. These query embeddings are now fed into the cross attention

transformer module as input to look at the multimodal features and predict the required bounding boxes as well as its corresponding segmentation.

In [68], similar to [66], the focus is on enhancing the object detection aspect of the REC component to boost overall system performance. The authors aggregate four large object detection datasets and conduct extensive pretraining on a vision language transformer. The model is trained using masked token loss for both image and text, along with a three-way contrastive loss. When fine-tuned on various downstream tasks, the resulting model improves the state-of-the-art.

Later works like [34, 57] draw inspiration from the success of CLIP [48]. [34] highlight that CLIP [48] based training, although powerful, learns to match the entire caption to an image. However, the authors argue that it's also crucial to learn the connections between subphrases and image regions. They suggest using BERT to extract word-level features and the dyHead module to extract sub-region-level features from the images. Both encoders are trained in a CLIP-style manner, aiming to match sub-region-level features in the image to their corresponding text matches. The authors from [57] also take inspiration from CLIP and attempt to improve the zero shot performance of REC. They suggest extracting sub phrases from the text using a language parser and utilizing CLIP to obtain probabilities for each subphrase with object proposals. However, the authors note that CLIP struggles with resolving spatial relations. To address this, they introduce a spatial resolver module to refine and provide the final predictions.

In our study, our focus is not on directly predicting the visual content indicated by the referring entity. Instead, we aim to determine whether existing open-source state-of-the-art foundation models, such as Blip2 [33], InstructBlip [8], and OpenFlamingo [2], can discern whether a referring expression could point to more than one entity.

2.3 Referring expression segmentation

Referring expression segmentation (RES) is closely related to referring expression comprehension (REC). The task entails identifying regions in an image described by text. Unlike REC, which focuses on outputting a bounding box, RES is expected to provide a segmentation mask in-

stead. The identified portion in the image corresponding to the referring expression (RE) is called the foreground mask, while the non-matching parts are termed as the background mask. The key distinction of solutions for REC compared to RES lies in the output. In REC, modules were either classifiers to select region proposals or regressors predicting bounding box coordinates.

2.3.1 Referring expression segmentation Datasets

Certain datasets such as Flickr30k Entities [47], gRefCOCO [45], and Visual Genome [32] came with segmentation masks for REs, whereas for a few datasets, such as ReferIt [29], the segmentation masks were derived from [14]. This was possible because the images for ReferIt [29] was built on top of [14]. Other datasets that provided segmentation masks for REs were UNC and UNC+ [65]. The UNC dataset is based on MSCOCO dataset using a two-player game similar to ReferIt [29]. UNC+ dataset is similar to the UNC dataset. However, there is a restriction of this dataset that no words in the referring expressions indicate location. Namely, the expression of the objects only describe the appearance information. Lastly we want to mention [39] as it emphasizes a unique challenge to RES. Until now most of the datasets proposed for RES have referring expressions pointing to a single object. In [39], however, the authors underscore the importance of also considering referring expressions that point to multiple objects. Hence they propose a new dataset that consists of referring expressions which point to both a single object as well multiple objects.

2.3.2 Referring expression segmentation Algorithms

Similar to VQA and REC, the development of algorithms for RES can be divided into two key phases. Initial works [22, 40, 36, 46, 52, 63, 24, 25] relied on generating visual and linguistic features using CNNs and LSTMs and later fusing them either via simple concatenation based methods or attention mechanism before upsampling the features to produce the required result. Later works such as [12, 30, 41] demonstrate how transformer-based architectures can be adapted to RES. The two phases are discussed below.

Similar to the development of algorithms for VQA and REC, the evolution of algorithms for RES can be divided into two key phases. In the initial phase [22, 40, 36, 46, 52, 63, 24, 25], researchers relied on generating visual and linguistic features using CNNs and Long Short-Term Memory networks LSTMs. These features were then fused using simple concatenation-based methods or attention mechanisms before upsampling to produce the required result. Subsequent works, such as those by [12, 30, 41], demonstrate how transformer-based architectures can be adapted to RES. Below, we discuss these two phases in detail.

Phase 1: A common approach for the model output involves using a 1 x 1 convolution layer to upsample features to match the dimensions of the given image. [22] follows a similar path. They propose using a CNN initialized with VGG [53] to extract image features and append additional channels encoding spatial level information. Text features are extracted from an LSTM and reshaped to match the image feature dimensions. The combination of these features is then passed to a classifier module, which upsamples to produce background and foreground masks.

In [40], the authors draw inspiration from how humans approach the task. They argue that humans, when resolving referring expressions in an image, don't do it in one go after reading the entire expression; instead, they do it incrementally. In line with this idea, the authors, similar to previous methods, employ a CNN to initially extract image features and include spatial-level encodings. These features are appended at every time step of LSTM processing the word level feature of the input text. The combined features are then fed into a multi-modal LSTM. Consequently, the multi-modal LSTM incrementally processes each word and image feature to generate the final output.

In [36], the authors emphasize that prior methods rely on a single output module to upsample multi-modal features for producing a segmentation map, often neglecting the full utilization of multi-scale features in the image. To address this limitation, they first process the image using a CNN and merge it with spatial-level encoding, along with text-level features derived from an LSTM. Subsequently, they incrementally upsample the features through a module. With each increment, the features are upsampled to the dimensions of earlier shallow CNN features. This

ensures that these features can be combined before moving to the next step. This approach observed an improvement in the IOU score from the previous methods.

In [46], following a similar approach to [40], the authors aim to enhance the multi-modal representation incrementally. Like previous methods, they use a CNN to extract image-level features. However, instead of an LSTM, they employ a simple RNN to process word-level features from the given referring expression (RE). These features are then fed into a synthesis module, tasked with providing scores for the referring expression in the visual space incrementally, word by word, ultimately generating a low-resolution map. Finally, the low-resolution map is upsampled to produce the desired output.

Later works [52, 63, 24, 25] use the attention mechanisms in different ways to solve RES. In [52], similar to previous approaches, CNNs and LSTMs are used to extract image and text features, respectively. The features at all timesteps of the LSTM are preserved. Subsequently, attention is employed to identify which feature map produced by the CNN corresponds to particular phrases in the given text. The attention weights can then be used to obtain a weighted average of all the visual features. Finally, a combination of all features is passed to the classifier and upsampled to generate the output.

In [63], the authors aim to leverage multi-scale features and capture long-range dependencies in both the image and text. Image features are extracted through a CNN initialized with ResNet [20], retaining feature maps from the last three convolutional layers. Individual words from the RE are encoded into their respective word embeddings and concatenated with the image feature maps at all levels. These multimodal feature maps, at all scales, are then input to a cross-modal self-attention module to identify the most relevant features. Finally, a gated fusion module aggregates features from all scales to produce the output. The authors argue that the incorporation of the cross-modal self-attention module aids in capturing long-range dependencies, thereby leading to a substantial improvement in the existing state-of-the-art.

[24] also like [63] try to encode multi-modal features via cross attention and capture long range dependencies. Their methodology is slightly different. Unlike the utilization of word level

embeddings in [63], the authors use an LSTM to incorporate sentence level features. Interestingly, their reported scores on ReferIt [29] are extremely similar, however [24] shows slightly better performance on UNC and UNC+ [65] datasets.

In [25], addressing Referring expression segmentation (RES) similarly to [66] for Referring expression comprehension (REC), visual features and text features are initially extracted through CNN and LSTM, respectively. These features are fused using bilinear fusion. Subsequently, the multi-modal features are input into a cross-attention module aiming to highlight relevant features and suppress others. Finally, a multimodal graph is formed as an adjacency matrix, representing nodes with region features. A separate module identifies relations between different nodes, determining the relation that best matches the given text to produce the desired output.

[27] underline that many prior works directly address the segmentation problem without prioritizing the intermediate step of localizing the region of interest first. To address this, they propose a two-step algorithm for RES. The first module focuses on localizing required regions of interest. Here extracted visual features and text features from CNN and GRU respectively are first fused with an MLP layer. A unique relevance filtering module then encodes a kernel based on the input text, which is passed to a transformer-based decoder to predict a rough heat map for localization. The second module utilizes these features to generate a more accurate segmentation mask.

Phase 2: Recent works [12, 30, 41] highlight the growing influence of transformers in Referring expression segmentation (RES).

[12] draws inspiration from attention mechanisms with the main concept that a REs should focus on relevant parts in the image. However, instead of relying on a single query (for a referring expression), they introduce a query generation module responsible for producing multiple query embeddings based on visual and text features. The idea is to have these multiple embeddings concentrate on different parts of the text. These query embeddings are then passed to a transformer-based decoder that attends to visual features, generating a segmentation mask for each query embedding. A separate module aggregates these masks to produce the final output.

[30] is the first to introduce an algorithm solving RES solely using transformers. They utilize a vision transformer to process visual features and a language transformer for textual features. These modalities' features are combined using a fusion transformer. The output of the fusion is encoded along with a randomly initialized seed embedding through a linguistic seed encoder to create classifier maps. A separate decoder generates the target segmentation map by computing the inner product between the multimodal feature and the classifier maps. The authors further demonstrate a significant improvement compared to the previous state-of-the-art. [41] takes a unique approach to Referring expression segmentation (RES). Instead of directly predicting the segmentation mask, the authors aim to predict the coordinates of a bounding polygon around the required foreground region. This approach mirrors how humans annotate a segmentation task. The process involves encoding visual and linguistic features separately using transformers. The unimodal features are then fused into multimodal features using a different transformer. These multimodal features are passed to a regression-based decoder to predict the polygon coordinates for the specified regions of interest. Surprisingly, this simple approach helps in achieving a new state-of-the-art across many RES datasets.

Lastly, [39] introduced a new dataset for Referring expression segmentation (RES) that contains expressions mapping to multiple objects in an image and proposed a baseline model to tackle the task. The authors utilized a SWIN [42] transformer to extract image features and a BERT-based [11] encoder to extract language features. These features are then fed into a specialized module that partitions the image features into patches and performs cross-attention between the patches and the language features. Subsequently, each individual patch is upsampled to generate minimaps, which are then combined to derive the final output. The core concept behind this approach is to initially predict for smaller patches before aggregating them for the entire image.

Although our dataset do not provide segmentation masks for the referring expressions in the question, the dataset can be easily extended to include segmentation masks easily via crowdsourcing in the future. Just like mentioned earlier in REC, the segmentation masks can also include annotations for vague or under-specified expressions such as "this" or "that" which can help improve

the existing algorithms for RES.

Chapter 3

Enriched VQA-AnswerTherapy dataset

In this chapter, we describe our creation of a new dataset that flags for which visual question is ambiguous and what content in the image the question is asking about.

3.1 Dataset creation

3.1.1 Dataset source

We extend the existing VQA-AnswerTherapy [6] dataset, which consists of visual questions sampled from the VQAV2 [17] and VizWiz [19] datasets. It offers a diverse and well-rounded collection of images from various sources. VQAV2 images are sourced from the MSCOCO dataset, while VizWiz comprises real-world images taken by visually impaired individuals. Notably, VQA-AnswerTherapy stands out as the only existing VQA dataset that includes multiple user-provided answers to an image along with corresponding answer groundings for each of the multiple responses.

Selecting VQA-AnswerTherapy as a starting point is advantageous for several reasons. The dataset has undergone meticulous filtering by the authors to reduce noise, including the removal of incorrect examples and exclusion of questions containing multiple sub-questions, thereby eliminating complex scenarios. It contains multiple possible answers for visual questions which also allows us to assess question ambiguity, as discussed in later sections. Furthermore, it allows us to explore the correlation between answers with different groundings with question ambiguity. It's worth noting that VQA-AnswerTherapy publicly provides both train and validation splits. For our task, we opt to utilize all examples from both splits.

3.1.2 Task design

We developed a user interface designed to present a single example from the entire dataset. It displays the image along with its corresponding question and all unique answers provided for that visual question.

Using this interface, the user's primary task involves addressing two key questions. Firstly, the user must identify and record the entity described in the question, with a focus on noun phrases. For instance, in the question "What color is this shirt?", the entity to identify is "this shirt". Later, we outline the rules for selecting the correct entity for all questions. This field will be labeled as "entity lookup" for future reference. Secondly, the user provides a binary label indicating whether the extracted entity can be unambiguously located solely by looking at the image. A "yes" indicates that the answer is unambiguously resolvable, while a "no" suggests otherwise. This field will be labeled as "question ambiguity" for future reference. Once the user is satisfied with the answers and clicks "submit", the interface proceeds to the next example in the dataset.

We now elaborate on the annotation rules.

(1) **Questions with a single target entity**: If the question contains only one noun phrase, the user is instructed to extract that specific noun phrase in the entity lookup field. This is exemplified in Figure 3.1

(2) **Questions with multiple target entities**: When the question includes multiple noun phrases, the user is instructed to extract all. Additionally, elements of the question connecting these noun phrases should also be recorded in the entity lookup field. An example for this is provided in Figure 3.2

(3) **Questions containing demonstrative pronouns**

   (a) In cases where the entity is not explicitly stated in the question and is often phrased using demonstrative pronouns such as "this" or "that," the user is instructed to record the demonstrative pronoun as the entity lookup for which they need to search.


	(a)	(b)
Question:	What is the food on?	What color is this shirt?
Answers:	table, dishes, plate	tan, beige, off white
Entity Lookup:	the food	this shirt
Question Ambiguity:	Yes	No

Figure 3.1: Example annotation for visual questions with single target entity

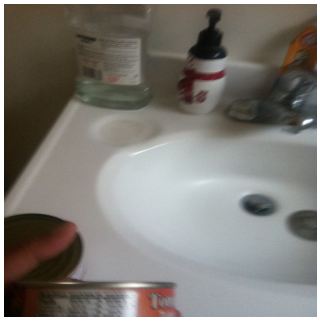
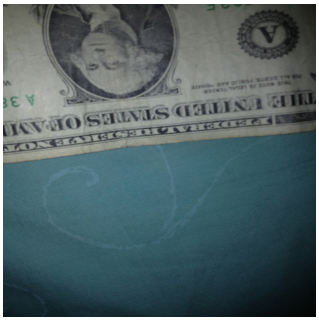

	(a)	(b)
Question:	What is the object in this image?	What is behind the woman on phone?
Answers:	carpet, TV, rug	mountains, mountain
Entity Lookup:	object in this image	the woman on phone
Question Ambiguity:	Yes	No

Figure 3.2: Example annotation for visual questions with multiple target entity

(b) Examples include questions like “What is this?” or “What is that?” where the demonstrative pronoun indicates the salient object in the image.

(c) More examples can be found in Figure 3.3

(4) **Questions where articles such as (“the”, “an”, “a”) are used along with entity**

Question:	What is this?	What is this?
Answers:	can, sink	money, dollar bill
Entity Lookup:	this	this
Question Ambiguity:	Yes	No
	(a)	(b)

Figure 3.3: Example annotation for visual questions containing demonstrative pronouns


Question:	where is the tissue	What is the boy holding?
Answers:	next to toilet, beside toilet	glove, mitt
Entity Lookup:	the tissue	the boy
Question Ambiguity:	Yes	No
	(a)	(b)

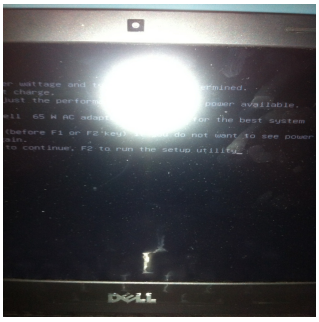
Figure 3.4: Example annotation for visual questions containing articles combined with entity

(a) If the entity is combined with an article in the question, the user is directed to record the entity along with it's article.

(b) For instance, in the question "What is that on the TV?", the user is instructed to record "the TV" as the "entity lookup".

(c) More examples can be found in Figure 3.4

(5) **Questions Referring to the Entire Image**

Question:	What, what do you see?	What do you see?
Answers:	floor, kitchen	computer screen, dell computer screen
Entity Lookup:	entire image	entire image
Question Ambiguity:	Yes	No
	(a)	(b)

Figure 3.5: Example annotation for visual questions containing referring to entire image

Question:	Is that a beer or a coke?	Is this a regular coke or a diet coke?
Answers:	coke, diet coke	diet coke, diet
Entity Lookup:	that	coke
Question Ambiguity:	No	No
	(a)	(b)

Figure 3.6: Example annotation for visual questions containing multiple options

(a) In cases where the question refers to the entire image, such as "What do you see?" or "Where am I?" the user is instructed to mark the entity lookup as "entire image."

(b) More examples can be found in Figure 3.5

(6) **Questions with options provided**

(a) If the question asks the user to recognize an entity from provided options and options are not related, such as "Is this some kind of bone or a horn?", then the user is directed

to record the demonstrative pronoun as the entity. In this case, the entity should be recorded as "this".

(b) If the question asks the user to recognize an entity from provided options and options are not related, such as "Is this regular coke or diet coke?", then the user is directed to record the noun phrase as the entity lookup. In the example provided, the focus is on differentiating between varieties of coke, so the entity lookup is marked as "coke."

(c) More examples can be found in Figure 3.6

For all the above scenarios, except when recording a demonstrative pronoun for entity lookup, the user should label the "question ambiguity" as "yes" if the noun phrase or the entity can be unambiguously resolved by looking at the image and "no" otherwise. Only when we use a demonstrative pronoun in the entity lookup field, we ask the user to look at the answers provided by the annotators to record "question ambiguity". If all the answers point to the same object, the user is instructed to record "question ambiguity" as "yes" and "no" otherwise.

Chapter 4

Dataset analysis

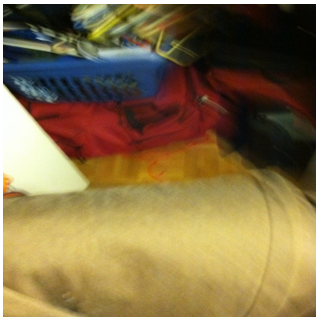
We found that approximately 3.4% of all of visual questions were flagged as ambiguous questions during our annotation efforts. We provide further analysis to better understand our new dataset.

	Overall	VizWiz	VQAv2
Ambiguous Questions	152	59	93
Unambiguous Questions	4288	2494	1794

Table 4.1: Number of ambiguous vs unambiguous visual questions in VQA-AnswerTherapy dataset

4.1 Prevalence of ambiguous questions

We summarize the exact number of ambiguous and unambiguous visual questions are presented in Table 4.1. We also summarize the number of ambiguous questions coming from VQAv2 [17] dataset versus those from the VizWiz dataset [19], since VQA-AnswerTherapy [6] is a dataset that combines images from VQAv2 and VizWiz. Out the 152 datapoints that showcase question ambiguity, roughly 38.8% originate from VizWiz, whereas 61.2% of the come from VQAv2 dataset. This means that a majority of the ambiguous questions arise from VQAv2 dataset rather than the VizWiz dataset. To understand why this might be the case, we perform some qualitative analysis. Figure 4.1 represents a sample taken from the examples belonging to VizWiz dataset that do not exhibit question ambiguity. Often visually impaired individuals mostly zoomed in on a single object for their pictures, thereby reducing the likelihood of question ambiguity. On the other hand,



Question: What color are these pants?
Answers: tan, khaki

What is this?
extra spearmint gum, gum

What is this?
blue pen, pen, ink pen

Question: What is that?
Answers: sprite can, sprite

What is this?
telephone, phone

What is this?
remote control, remote

Figure 4.1: Examples from VizWiz dataset of visual questions lacking question ambiguity

images from VQAv2 do not necessarily focus on a single object. Nevertheless, even when ambiguous questions arise, the expectation is that VQA models should demonstrate robustness by either prompting users to resolve the ambiguity or providing answers from various perspectives.

Finally, we explore the connection between question ambiguity and its impact on generating answers that point to different visual content in the VQA-AnswerTherapy dataset. We discovered that 49.3% of visual questions with ambiguity showcase a tendency to produce answers that point to different visual content. This underscores that ambiguity is one of the key factors in contributing to diverse answers. Therefore, in this work we highlight the importance of needing to build robust models capable of handling ambiguous questions. We summarize all these numbers in Table 4.2.

	Number of examples
Ambiguous questions with diff answer grounding	75
Ambiguous questions with same answer grounding	77
Unambiguous questions with diff answer grounding	573
Unambiguous questions with same answer grounding	3715

Table 4.2: Number of ambiguous and unambiguous visual questions that have answers pointing to same or different visual content



Question:	What kind of donut is the person eating?	Where is the tissue?	What color is the car?
Answers:	chocolate, frosted	next to toilet, beside toilet	silver, black, blue
Entity Lookup:	donut the person is eating	the tissue	the car

Figure 4.2: Examples of ambiguous visual questions from VQAv2 dataset



Question:	Is this?	This, what's this?	what is that?
Answers:	desk, office	slippers, rug	living room, chair
Entity Lookup:	this	this	that

Figure 4.3: Examples of ambiguous visual questions from VizWiz dataset

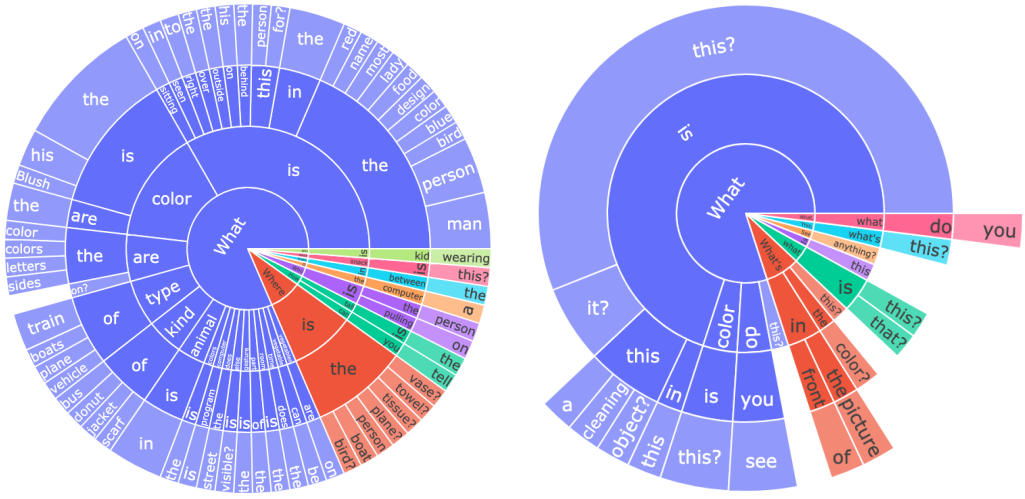


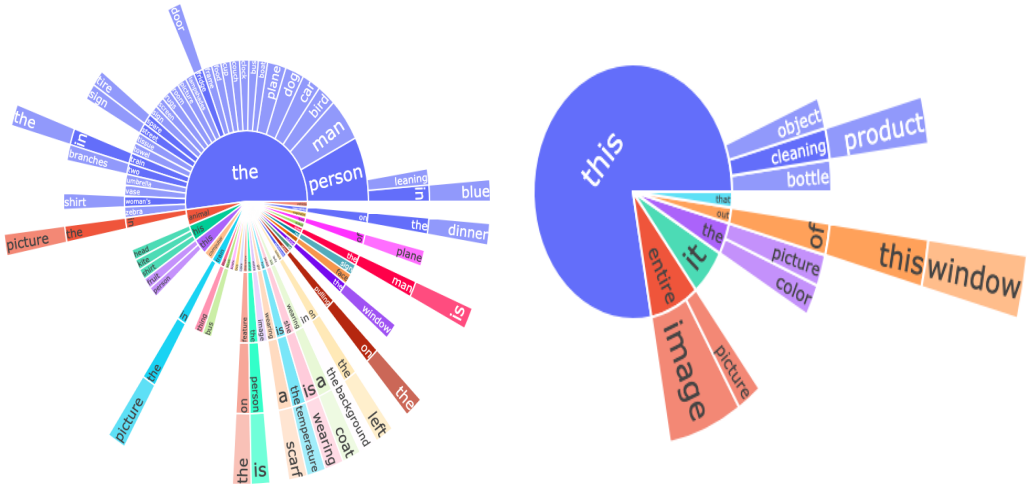
Figure 4.4: Sunburst diagram depicting the distribution of questions by their first four words of a sample from (a) VQAv2 and (b) VizWiz. All these questions showcase question ambiguity. The words starts towards the center and radiates outwards.

4.2 Analyzing question types which are ambiguous

In this section we try to identify if there are any recurring patterns in the words used in ambiguous questions. Figure 4.4 presents a sunburst diagram of visual questions flagged as ambiguous coming from VQAv2 and VizWiz dataset. Figure 4.5 presents a sunburst diagram of the "entity lookup" field from the ambiguous visual questions from VQAv2 and VizWiz dataset.

Upon observing the sunburst diagram for questions and entity descriptions from VQAv2, we observe a diverse range of questions. Although many questions start with "What," no clear pattern emerges. Conversely, in the case of VizWiz data points, the majority of ambiguous questions follow the pattern of "What is this." It's noteworthy that, in general, VizWiz predominantly features questions stating "What is this?" Below, we provide qualitative examples of cases marked as ambiguous, all containing the question "What is this?". We notice that some of these examples lack a clearly identifiable prominent object, leading to ambiguity.

We further present the word clouds representing questions and entity descriptions from both VQAv2 and VizWiz in Figures 4.6 and 4.7. The patterns align consistently with the previously



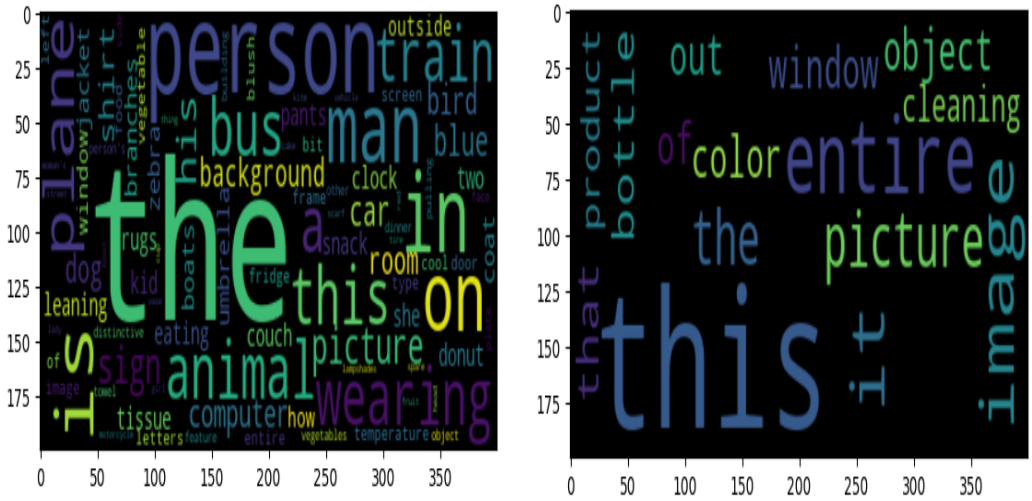
(a) (b)

Figure 4.5: Sunburst diagram depicting the distribution of entities extracted from visual questions from (a) VQAv2 and (b) VizWiz. All entities extracted were from visual questions showcasing question ambiguity. The words starts towards the center and radiates outwards.

(a) (b)

Figure 4.6: Word cloud diagram depicting the frequency of unique words in the visual questions from (a) VQAv2 and (b) VizWiz. All these words were taken from questions that showcase question ambiguity.

displayed sunburst diagram. We observe the number of unique words is significantly higher in the visual questions originating from VQAv2 when compared to VizWiz. Additionally, the visual questions coming from VizWiz seem to be dominated by the presence of demonstrative pronouns,



(a) (b)

Figure 4.7: Word cloud diagram depicting the frequency of entities in the visual questions from (a) VQAv2 and (b) VizWiz. All these entities extracted were part of visual questions that showcase question ambiguity.

whereas the visual questions from VQAv2 seem to have a more number of unique entities.

4.3 Comparison of entity descriptions between other datasets

The entity descriptions extracted from our dataset can be treated as phrases that point to some some regions in the image. Thus, we can compare the entity descriptions of our dataset to the phrases present by other phrase grounding datasets. In this section, we compare the distribution of parts-of-speech (POS) tags and description lengths of the object descriptions in our dataset and compare it to OmniLabel, RefCOCO/g, and Flickr30k.

Figure 4.8 shows the number of unique nouns, adjectives, and verbs in the entity look ups of our dataset versus that of OmniLabel, RefCOCO/g, and Flickr30k. We observe that our dataset has the least number of unique words. This could be in part because a majority of the questions in our dataset are phrased as “What is this?” leading to a lesser variety of unique words. This also could stem from the much smaller size of our dataset in absolute terms.

Figure 4.9 displays a histogram of lengths for the entity look ups of our dataset versus that of OmniLabel, RefCOCO/g, and Flickr30k. To generate this graph, we sample 10000 datapoints from

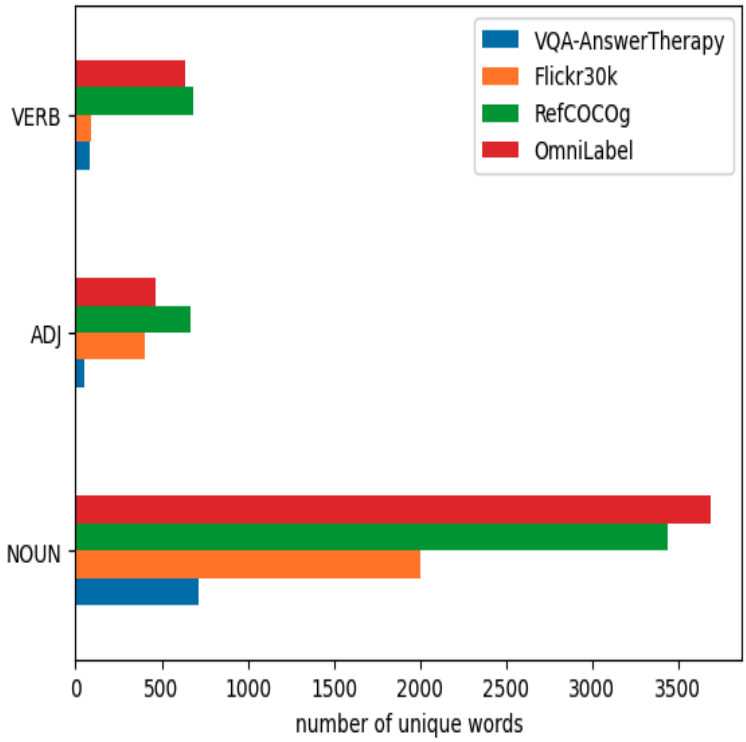


Figure 4.8: Percentage of distribution of unique nouns, adjectives and verbs for our dataset compared to other datasets .

each dataset of OmniLabel, RefCOCO/g, and Flickr30k. However, since VQA-AnswerTherapy has 4440 datapoints, we retain all of them. We observe that the entity descriptions in our dataset is shorter than observed for other datasets.

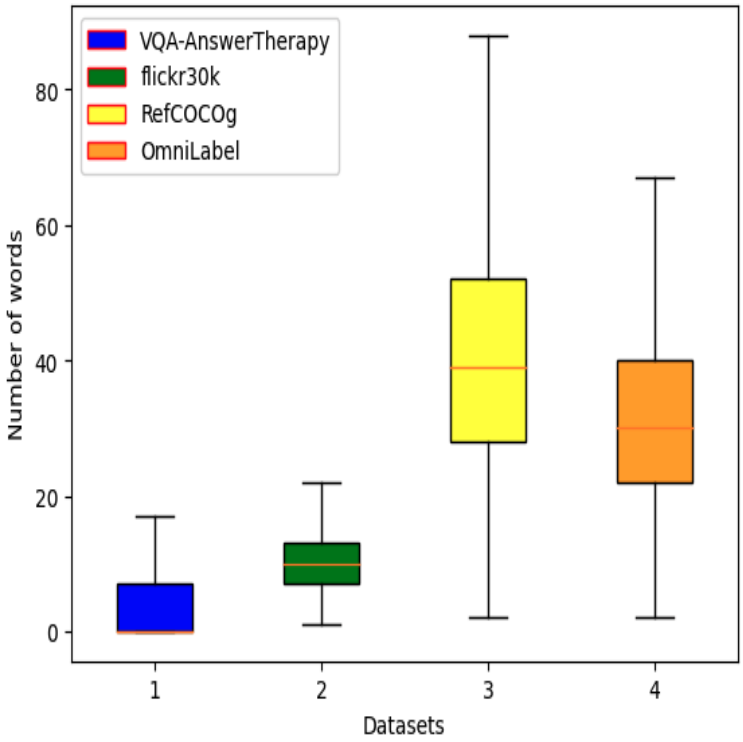


Figure 4.9: Boxplots showing entity description lengths (in number of words) for VQA-AnswerTherapy and other datasets.

Chapter 5

Algorithmic evaluation

5.1 Task definition

In this section, our objective is to assess the performance of existing open-source foundational vision language models on our new dataset. Our primary focus is to determine whether these models can recognize ambiguity in visual questions. We hypothesize that foundation models will excel in this task only if they can discern whether visual concepts refer to the same location in the image or not. The VQA-AnswerTherapy dataset, aside from providing multiple answers provided by annotators to a visual question, also includes a field called "binary_label". This field takes the value "same" when the answers point to the same visual content and "different" otherwise. We can utilize this attribute to probe whether models have the capability to map visual concepts to it's corresponding location in the image. Lastly, we also evaluate whether these models are capable of generating all diverse sets of answers similar to humans. To that end, we propose three novel tasks.

(1) **Ambiguity identification** - Given an image, question, and all answers (curated by crowdsourcing), can the model identify if the question was ambiguous.

(2) **Visual identification** - Given an image, question, and all answers (curated by crowdsourcing), can the model identify if all the answers point to the same visual content or not.

(3) **Diverse answer generation** - Given an image and question about it, can the model predict all the diverse set of answers received from different humans.

We focus on evaluating three foundation models: Blip2 [33], InstructBlip [8], and OpenFlamingo [2]. These models have demonstrated strong performance on numerous downstream tasks related to vision language understanding, including VQA. Each is open-source to enable researchers to utilize them for specific downstream tasks, such as VQA, via prompting. For our evaluation, we rely on four prompting methods.

Prompting method 1 (Standard) We use a standard template for each of the tasks, without providing much additional context. In other words, the foundation model only receives the image as context to complete the required task mentioned in the prompt. The templates used are

(1) **Ambiguity identification** - "Question: {question} Answer: {answer} Question: Is the question ambiguous?"

(2) **Visual identification** - "Question: {question} Answer: {answer} Question: Do all the given answers for the question point to the same visual content in the image?"

(3) **Diverse answer generation** - "Indicate every possible answer to the given question. Question: {question} Answer:"

Prompting method 2 (Captions) In this method, we provide the foundation model with both the image and its corresponding caption as context for each task. Initially, we prompt the model to generate a caption for the given image, which is then used as additional context for our tasks. The templates utilized are as follows:

(1) **Ambiguity identification** - "Context: {caption} Question: {question} Answer: {answer} Question: Is the question ambiguous?"

(2) **Visual identification** - "Context: {caption} Question: {question} Answer: {answer} Question: Do all the given answers for the question point to the same visual content in the image?"

(3) **Diverse answer generation** - "Context: {caption} Indicate every possible answer to the given question. Question: {question} Answer:"

Prompting method 3 (Zero shot Chain of Thought) - [60] demonstrate that chain of thought prompting enhances the performance of foundation models on various downstream tasks. The core concept involves prompting the model to solve the task step by step instead of providing the answer directly. Inspired by this approach, we have devised the following prompt templates for our task.

(1) **Ambiguity identification** - "Please answer the following question by reasoning step by step. Question: {question} Answer: {answer} Question: Is the question ambiguous?"

(2) **Visual identification** - "Please answer the following question by reasoning step by step. Question: {question} Answer: {answer} Question: Do all the given answers for the question point to the same visual content in the image?"

(3) **Diverse answer generation** - "Please indicate every possible answer to the visual question by reasoning step by step. Question: {question} Answer:"

Prompting method 4 (Few Shot) - Few shot prompting involves providing a certain number of examples, along with their expected outputs, to the model as additional context before it can execute the task at hand. We experiment with prompting method 1 and 2 with few shot prompting. The prompt templates remain consistent for each respective task. We report results when the number of examples provided for additional context are 4 and 8 respectively.

It should be noted that we apply prompting methods 1, 2, and 3 to the foundation models Blip2 [33] and InstructBlip [8], while we use prompting method 4 for OpenFlamingo [2].

5.2 Evaluation methodology

We conduct our experiments on our enriched VQA-AnswerTherapy dataset. This dataset is ideal for our tasks for three reasons. First, it provides annotations for identifying which visual question is ambiguous. Second, it provide a field labeled "binary_label" that enables us to determine whether all the answers provided by the annotators to a visual question point to the same visual

content in the image or not. Lastly, it also provides all the multiple set of answers provided to us by the annotators. These characteristics helps us easily evaluate the output of our models.

5.2.1 Ambiguity identification

For our ambiguity identification task, we expect the foundation models to answer "yes" whenever it encounters an ambiguous visual question and answer "no" whenever the model encounters an unambiguous visual question. For this task, we use accuracy as the evaluation metric. We use the field "question ambiguity" from VQA-AnswerTherapy dataset as the ground truth. The distribution of ambiguous versus unambiguous visual questions were presented in Table 4.1.

5.2.2 Visual identification

For our visual identification task, we expect the foundation models to answer "yes" whenever the different answers provided by the annotators point to the same location in the image and answer "no" otherwise. For this task, we use accuracy as the evaluation metric. We use the field "binary_label" from VQA-AnswerTherapy dataset as the ground truth. Table 5.1 provides us a distribution of number of examples that mark the "binary_label" as "same" versus that of "different" across the entire dataset.

Binary Label	**Number of examples**
Same	3535
Different	636
Total	4171

Table 5.1: Distribution same versus different in binary_label field of VQA-AnswerTherapy dataset

5.2.3 Diverse answer generation

This situation gets tricky since the model might provide sequences with fewer or more words than those in the ground truth, making direct comparison challenging. To address this, we suggest

calculating the percentage of answers from the ground truth that the model includes in its prediction.For instance, if the model captures all the answers for a given ambiguous visual question, it gets a score of 1. If it includes half of the answers from the ground truth, it gets a score of 0.5, and so on. Finally we calculate the mean over all scores across the dataset to get the final score. For instance, if the model captures all the answers for a given ambiguous visual question, it gets a score of 1. If it includes half of the answers from the ground truth, it gets a score of 0.5, and so on. Finally we calculate the mean over all scores across the dataset to get the final score. Table 5.2 provides additional examples.

	Ground truth			
Visual Question	**Answer 1**	**Answer 2**	**Predicted**	**Score**
What type of vehicle is here?	Tow truck	Bus	Bus and tow truck	1.0
			Bus	0.5
What kind of car is this?	Compact	Hatchback	Compact and hatchback	1.0
			Compact	0.5

Table 5.2: Sample accuracy calculation for diverse answer generation task

5.3 Results

5.3.1 Ambiguity identification

We summarize the results of ambiguity identification test in Tables 5.3, 5.4, and 5.5 respectively.

Are models capable of recognizing whether visual questions are ambiguous or not? Upon reviewing the results from Tables 5.3, 5.4, and 5.5, we observe that the overall performance of InstructBlip is significantly higher compared to that of Blip2 and OpenFlamingo. However, upon closer examination of the results, considering the model's performance for ambiguous and unambiguous questions individually, we notice that both Blip2 and InstructBlip tend to excel only in one category of visual questions. Blip2 shows good performance for ambiguous visual questions, whereas InstructBlip performs well with unambiguous visual questions. This indicates

Accuracy	Prompt method		
	Standard	**Captions**	**Chain of thought**
Overall	0.085	0.086	0.285
Ambiguous Questions	0.873	0.940	0.733
Unambiguous Questions	0.058	0.056	0.269

Table 5.3: Accuracy for Blip2 on ambiguity identification task

Accuracy	Prompt method		
	Standard	**Captions**	**Chain of thought**
Overall	0.836	0.856	0.818
Ambiguous Questions	0.166	0.173	0.146
Unambiguous Questions	0.860	0.880	0.841

Table 5.4: Accuracy for InstructBlip on ambiguity identification task

	Accuracy	Prompt method	
		Standard	**Captions**
4 shot	Overall	0.694	0.710
	Ambiguous Questions	0.246	0.253
	Unambiguous Questions	0.710	0.713
8 shot	Overall	0.372	0.385
	Ambiguous Questions	0.986	0.990
	Unambiguous Questions	0.350	0.350

Table 5.5: Accuracy for OpenFlamingo on ambiguity identification task

that Blip2 has a tendency to answer "yes" to most prompts in this task, while InstructBlip tends to answer "no". OpenFlamingo, on the other hand, seems to perform reasonably better than Blip2 and InstructBlip in this task. However, it also exhibits a tendency to answer "yes" when the number of examples provided as context is 4, and a tendency to answer "no" when the number of examples provided as context is 8. Due to these inconsistencies, we cannot conclusively determine whether these foundation models are adept at identifying ambiguous visual questions.

Which prompting method performs best on the ambiguity identification test? The results from Table 5.3 suggest that the performance of Blip2 slightly improves when captions are introduced as additional context, as well as when zero-shot chain of thought prompting is used.

Accuracy score for Binary Label	Prompt method		
	Standard	**Captions**	**Chain of thought**
Overall	0.835	0.833	0.853
Same	0.979	0.976	0.997
Different	0.001	0.003	0.012

(a) Overall versus same versus different

Accuracy score for Question type	Prompt method		
	Standard	**Captions**	**Chain of thought**
Ambiguous Questions	0.506	0.506	0.506
Unambiguous Questions	0.847	0.845	0.865

(b) Ambiguous versus unambiguous visual questions

Table 5.6: Accuracy for Blip2 on visual identification task

However, in the case of InstructBlip, as seen in Table 5.4, performance only improves with captions. This trend is also evident in the case of OpenFlamingo, as shown in Table 5.5.

5.3.2 Visual identification

We present the results of our visual identification test in Tables 5.6, 5.7, and 5.8 respectively. In this test, we categorize the results into two parts. The first part evaluates the overall accuracy of the models on this task, as well as the model's performance individually for all data points categorized based on their "binary_label" marked as either "same" or "different". The second part evaluates the model's performance for all data points categorized as ambiguous or unambiguous. This helps us understand the model's behaviour when it encounters an ambiguous visual question versus an unambiguous visual question.

Are models capable of recognizing whether different answers point to the same visual content or not? Upon reviewing the results from part (a) Tables 5.6, 5.7, and 5.8 we recognize that it is not the case. We find a standard pattern that the Blip2 and InstructBlip usually answers "yes" and OpenFlamingo answers "no" irrespective of the questions asked. This suggests to us that models may not be able to correlate answers back to the image, meaning models lack the

Accuracy score for Binary Label	Prompt method: Standard	Captions	Chain of thought
Overall	0.839	0.826	0.839
Same	0.981	0.963	0.981
Different	0.009	0.031	0.009

(a) Overall versus same versus different

Accuracy score for Question type	Prompt method: Standard	Captions	Chain of thought
Ambiguous Questions	0.506	0.506	0.493
Unambiguous Questions	0.851	0.838	0.851

(b) Ambiguous versus unambiguous visual questions

Table 5.7: Accuracy results for InstructBlip on visual identification task

	Accuracy score for Binary Label	Prompt method: Standard	Captions
4 shot	Overall	0.385	0.265
	Same	0.347	0.345
	Different	0.606	0.605
8 shot	Overall	0.153	0.243
	Same	0.009	0.003
	Different	0.992	1.0

(a) Overall versus same versus different

	Accuracy score for Question type	Prompt method: Standard	Captions
4 shot	Ambiguous Questions	0.493	0.512
	Unambiguous Questions	0.381	0.356
8 shot	Ambiguous Questions	0.493	0.502
	Unambiguous Questions	0.141	0.145

(b) Ambiguous versus unambiguous visual questions

Table 5.8: Accuracy results for OpenFlamingo on visual identification task

capability to ground answers in images. However, we do concede that a deeper exploration of various prompt templates are necessary to solidify the results. Additionally, focusing on the results from

part (b) of Tables 5.6, 5.7, and 5.8, we observe that Blip2 and InstructBlip demonstrate relatively higher performance when the visual question is unambiguous. This highlights the importance of developing more robust VQA algorithms capable of handling different types of ambiguity. It is worth mentioning that since the majority of visual questions from VQA-AnswerTherapy are unambiguous and considering the models' bias towards answering "yes" to most prompts, higher scores are observed for unambiguous visual questions. Conversely, due to OpenFlamingo's tendency to respond "no" to most prompts, we observe higher scores for ambiguous questions.

Which prompting method performs best on the visual identification test? Comparing the results from Tables 5.6, 5.7, we find that introducing image captions as additional context yields a slight improvement in the model's capability of recognizing that some answers do not point to the same visual content. On the contrary, this cannot be claimed in the case of OpenFlamingo (Table 5.8). As mentioned earlier OpenFlamingo inherently has the bias to answer "no" to all the questions asked. Adding image captions does not seem to help. Furthermore, the difference in performance between a standard prompting template and chain of thought prompting is negligible, indicating that chain of thought prompting does not seem to provide significant improvement. A similar observation was made by [3] when it comes to VQA. Furthermore, it is noteworthy that OpenFlamingo exhibits lower performance for unambiguous questions compared to Blip2 and InstructBlip. We suspect that its bias towards answering "no" to most visual questions contributes to this behavior. Considering that the number of visual questions with identical answer groundings (Table 4.2) is significantly higher in the dataset, the performance of OpenFlamingo for unambiguous questions is adversely affected.

5.3.3 Diverse answer generation

We summarize the results of Diverse answer generation task in Tables 5.9, 5.10, and 5.11 respectively. Similar to the visual identification task, we categorize the results into the same two parts for analysis.

Accuracy score for Binary Label	Prompt method		
	Standard	**Captions**	**Chain of thought**
Overall	0.573	0.591	0.584
Same	0.583	0.600	0.595
Different	0.513	0.540	0.521

(a) Overall versus same versus different

Accuracy score for Question type	Prompt method		
	Standard	**Captions**	**Chain of thought**
Ambiguous Questions	0.573	0.573	0.576
Unambiguous Questions	0.573	0.592	0.584

(b) Ambiguous versus unambiguous visual questions

Table 5.9: Accuracy results for Blip2 on diverse answer generation

Accuracy score for Binary Label	Prompt method		
	Standard	**Captions**	**Chain of thought**
Overall	0.570	0.549	0.562
Same	0.589	0.569	0.580
Different	0.459	0.429	0.456

(a) Overall versus same versus different

Accuracy score for Question type	Prompt method		
	Standard	**Captions**	**Chain of thought**
Ambiguous Questions	0.581	0.552	0.572
Unambiguous Questions	0.570	0.548	0.561

(b) Ambiguous versus unambiguous visual questions

Table 5.10: Accuracy results for InstructBlip on diverse answer generation

Are models able to generate diverse set of answers for the same question? All the results in Tables 5.9, 5.10, and 5.11 center around an average score of 0.5, which means that on average these models were able to get only half of the diverse set of answers, which means that there is still a significant room for improvement. Observing the examples from VQA-AnswerTherapy, we find that most visual question have two different ground truth answers, with very few examples

	Accuracy score for Binary Label	Prompt method	
		Standard	**Captions**
4 shot	Overall	0.585	0.546
	Same	0.619	0.579
	Different	0.465	0.443
8 shot	Overall	0.577	0.537
	Same	0.624	0.579
	Different	0.430	0.406

(a) Overall versus Same versus Different

	Accuracy score for Question type	Prompt method	
		Standard	**Captions**
4 shot	Ambiguous Questions	0.543	0.512
	Unambiguous Questions	0.543	0.502
8 shot	Ambiguous Questions	0.534	0.540
	Unambiguous Questions	0.537	0.515

(b) Ambiguous versus Unambiguous visual questions

Table 5.11: Accuracy results for OpenFlamingo on diverse answer generation

containing more than two ground truth answers. This may further explain the accuracy number being close to 50%.

Which prompting method performs best at diverse answer generation test? Results in Tables 5.9, 5.10, and 5.11 suggest that no particular prompting method necessarily improves the model's capability to produce diverse answers. Additionally, no pattern emerges even when we try to compare the performance of the models for ambiguous and unambiguous visual questions. This result is important as it helps us to determine the future direction that we need to take in order to get the models to produce a variety of answers. The language models within Blip2, InstructBlip, and OpenFlamingo seem to be pre-trained to only produce results that are most probable. In the future, we aim to either try different sampling techniques or instruction tuning these foundation models to produce varied outputs.

Does few shot prompting help? Few shot prompting doesn't seem to help in either of our tasks. Additionally contrary to belief, increasing the number of few shot exemplars hurt the

model performance more. This is evident comparing the numbers from Table 5.8 and 5.11

Chapter 6

Conclusion

Our research aimed to investigate the role of ambiguity, particularly question ambiguity, in leading to provide multiple correct answers for a given visual question. To achieve this, we conducted an annotation task to enrich the existing VQA-AnswerTherapy dataset with two additional fields: “entity lookup” and “question ambiguity”. The “entity lookup” field identifies the entity that users need to locate in the image based on the visual question. The “question ambiguity” field, a binary attribute, determines whether the question is ambiguous. Our annotation efforts revealed that approximately 3.4% of all visual questions in the VQA-AnswerTherapy dataset exhibit question ambiguity. Moreover, the majority of visual questions (roughly 61.2%) demonstrating question ambiguity originate from the VQAv2 dataset, while the remainder originate from the VizWiz dataset. Our observations indicate that images from VizWiz typically feature only one zoomed-in object per image, resulting in a less likely scenario for question ambiguity. Conversely, images from VQAv2 often contain multiple instances of an object per image, increasing the likelihood of question ambiguity. Additionally, we found that visual questions showing question ambiguity from VizWiz are often vague, with entity lookups consisting of under-specified words such as ”this” and ”that,” while those from VQAv2 are more diverse.

Furthermore, we conducted an algorithmic analysis to evaluate whether existing state-of-the-art, open-source foundation models can accomplish three tasks: (1) determing if the provided visual question exhibits question ambiguity (2) determining if the answers provided by annotators refer to the same or different visual context, and (3) predicting diverse sets of answers solely based on the

visual question, similar to those provided by annotators. We employed four prompting methods for each task and compared the overall performance of these models with respect to visual questions that exhibit question ambiguity versus those that do not. Our findings indicate that none of the models, Blip2, InstructBlip, or OpenFlamingo, excel in the aforementioned tasks. We show that for the first and second tasks, models have a tendency to either answer "yes" to all the prompts or "no". We show that this bias hurts the performance of the models and thus using these foundation models naively without finetuning may not be the desired approach for downstream applications. Furthermore, we observe that none of the models are good at performing the third task. All of them seem to produce just one set of the ground truth answers. Thus they are unable to mimic the diverse set of answers that humans can provide. This observation holds true across all prompting methods, emphasizing the need to develop more robust models.

Chapter 7

Future work

In our work, we enrich VQA-AnswerTherapy dataset to identify questions that showcase question ambiguity. However, as discussed earlier there is also a factor of answer ambiguity. In the future, we propose to continue to enrich the dataset with additional fields to help us recognize the kind of questions that exhibit answer ambiguity. The combination of annotations for question ambiguity and answer ambiguity will help us truly understand the role of ambiguity in general to elicit diverse set of answers at a finer level. Furthermore, we would also be able to test the capability of different foundation models when posed with questions showcasing different kinds of ambiguity.

Additionally, in our work we test the outputs of the models with a greedy decoding strategy. However, there exists many other popular decoding techniques such as beam search decoding, top-p and top-k sampling etc. In the future, we aim to evaluate the foundation models with different decoding strategies in order to make our evaluation exhaustive.

Bibliography

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In Proceedings of the IEEE international conference on computer vision, pages 2425–2433, 2015.

[2] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. Openflamingo: An open-source framework for training large autoregressive vision-language models. arXiv preprint arXiv:2308.01390, 2023.

[3] Rabiul Awal, Le Zhang, and Aishwarya Agrawal. Investigating prompting techniques for zero- and few-shot visual question answering. arXiv preprint arXiv:2306.09996, 2023.

[4] Nilavra Bhattacharya, Qing Li, and Danna Gurari. Why does a visual question have different answers? In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 4271–4280, 2019.

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In European conference on computer vision, pages 213–229. Springer, 2020.

[6] Chongyan Chen, Samreen Anjum, and Danna Gurari. Vqa therapy: Exploring answer differences by visually grounding answers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 15315–15325, 2023.

[7] Zhenfang Chen, Peng Wang, Lin Ma, Kwan-Yee K Wong, and Qi Wu. Cops-ref: A new dataset and task on compositional referring expression comprehension. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10086–10095, 2020.

[8] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. Advances in Neural Information Processing Systems, 36, 2024.

[9] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 326–335, 2017.

[10] Jiajun Deng, Zhengyuan Yang, Tianlang Chen, Wengang Zhou, and Houqiang Li. Transvg: End-to-end visual grounding with transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1769–1779, 2021.

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

[12] Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. Vision-language transformer and query generation for referring segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 16321–16330, 2021.

[13] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2625–2634, 2015.

[14] Hugo Jair Escalante, Carlos A Hernández, Jesus A Gonzalez, Aurelio López-López, Manuel Montes, Eduardo F Morales, L Enrique Sucar, Luis Villasenor, and Michael Grubinger. The segmented and annotated iapr tc-12 benchmark. Computer vision and image understanding, 114(4):419–428, 2010.

[15] Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. From captions to visual concepts and back. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1473–1482, 2015.

[16] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. Neural computation, 12(10):2451–2471, 2000.

[17] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 6904–6913, 2017.

[18] Danna Gurari and Kristen Grauman. Crowdverge: Predicting if people will agree on the answer to a visual question. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, pages 3511–3522, 2017.

[19] Danna Gurari, Qing Li, Chi Lin, Yinan Zhao, Anhong Guo, Abigale Stangl, and Jeffrey P Bigham. Vizwiz-priv: A dataset for recognizing the presence and purpose of private visual information in images taken by blind people. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 939–948, 2019.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[21] Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. Modeling relationships in referential expressions with compositional modular networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1115–1124, 2017.

[22] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, pages 108–124. Springer, 2016.

[23] Yushi Hu, Hang Hua, Zhengyuan Yang, Weijia Shi, Noah A Smith, and Jiebo Luo. Promptcap: Prompt-guided image captioning for vqa with gpt-3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 2963–2975, 2023.

[24] Zhiwei Hu, Guang Feng, Jiayu Sun, Lihe Zhang, and Huchuan Lu. Bi-directional relationship inferring network for referring image segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 4424–4433, 2020.

[25] Shaofei Huang, Tianrui Hui, Si Liu, Guanbin Li, Yunchao Wei, Jizhong Han, Luoqi Liu, and Bo Li. Referring image segmentation via cross-modal progressive comprehension. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10488–10497, 2020.

[26] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 6700–6709, 2019.

[27] Ya Jing, Tao Kong, Wei Wang, Liang Wang, Lei Li, and Tieniu Tan. Locate then segment: A strong pipeline for referring image segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 9858–9867, 2021.

[28] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1780–1790, 2021.

[29] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 787–798, 2014.

[30] Namyup Kim, Dongwon Kim, Cuiling Lan, Wenjun Zeng, and Suha Kwak. Restr: Convolution-free referring image segmentation using transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 18145–18154, 2022.

[31] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In International Conference on Machine Learning, pages 5583–5594. PMLR, 2021.

[32] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. International journal of computer vision, 123:32–73, 2017.

[33] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In International conference on machine learning, pages 19730–19742. PMLR, 2023.

[34] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. Grounded language-image pre-training. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 10965–10975, June 2022.

[35] Muchen Li and Leonid Sigal. Referring transformer: A one-step approach to multi-task visual grounding. Advances in neural information processing systems, 34:19652–19664, 2021.

[36] Ruiyu Li, Kaican Li, Yi-Chun Kuo, Michelle Shu, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. Referring image segmentation via recurrent refinement networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5745–5753, 2018.

[37] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16, pages 121–137. Springer, 2020.

[38] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, pages 740–755. Springer, 2014.

[39] Chang Liu, Henghui Ding, and Xudong Jiang. GRES: Generalized referring expression segmentation. In CVPR, 2023.

[40] Chenxi Liu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan Yuille. Recurrent multimodal interaction for referring image segmentation. In Proceedings of the IEEE international conference on computer vision, pages 1271–1280, 2017.

[41] Jiang Liu, Hui Ding, Zhaowei Cai, Yuting Zhang, Ravi Kumar Satzoda, Vijay Mahadevan, and R Manmatha. Polyformer: Referring image segmentation as sequential polygon generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 18653–18663, 2023.

[42] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF international conference on computer vision, pages 10012–10022, 2021.

[43] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. Advances in neural information processing systems, 32, 2019.

[44] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. Ask your neurons: A neural-based approach to answering questions about images. In Proceedings of the IEEE international conference on computer vision, pages 1–9, 2015.

[45] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana-Maria Camburu, Alan Loddon Yuille, and Kevin P. Murphy. Generation and comprehension of unambiguous object descriptions. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 11–20, 2015.

[46] Edgar Margffoy-Tuay, Juan C Pérez, Emilio Botero, and Pablo Arbeláez. Dynamic multimodal instance segmentation guided by natural language queries. In Proceedings of the European Conference on Computer Vision (ECCV), pages 630–645, 2018.

[47] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In Proceedings of the IEEE international conference on computer vision, pages 2641–2649, 2015.

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

[49] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 779–788, 2016.

[50] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In European Conference on Computer Vision, 2015.

[51] Arka Sadhu, Kan Chen, and Ram Nevatia. Zero-shot grounding of objects from natural language queries. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 4694–4703, 2019.

[52] Hengcan Shi, Hongliang Li, Fanman Meng, and Qingbo Wu. Key-word-aware network for referring expression image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), pages 38–54, 2018.

[53] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[54] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 15638–15650, 2022.

[55] Haoyu Song, Li Dong, Weinan Zhang, Ting Liu, and Furu Wei. Clip models are few-shot learners: Empirical studies on vqa and visual entailment. In Annual Meeting of the Association for Computational Linguistics, 2022.

[56] Yibing Song, Ruifei Zhang, Zhihong Chen, Xiang Wan, and Guanbin Li. Advancing visual grounding with scene knowledge: Benchmark and method. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 15039–15049, 2023.

[57] Sanjay Subramanian, William Merrill, Trevor Darrell, Matt Gardner, Sameer Singh, and Anna Rohrbach. Reclip: A strong zero-shot baseline for referring expression comprehension. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 5198–5215, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[58] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. arXiv preprint arXiv:1908.07490, 2019.

[59] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1960–1968, 2019.

[60] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems, 35:24824–24837, 2022.

[61] Fanyi Xiao, Leonid Sigal, and Yong Jae Lee. Weakly-supervised visual grounding of phrases with linguistic structures. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5945–5954, 2017.

[62] Zhengyuan Yang, Boqing Gong, Liwei Wang, Wenbing Huang, Dong Yu, and Jiebo Luo. A fast and accurate one-stage approach to visual grounding. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 4683–4693, 2019.

[63] Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. Cross-modal self-attention network for referring image segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10502–10511, 2019.

[64] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. Transactions of the Association for Computational Linguistics, 2:67–78, 2014.

[65] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14, pages 69–85. Springer, 2016.

[66] Zhou Yu, Jun Yu, Chenchao Xiang, Zhou Zhao, Qi Tian, and Dacheng Tao. Rethinking diversified and discriminative proposal generation for visual grounding. arXiv preprint arXiv:1805.03508, 2018.

[67] Peng Zhang, Yash Goyal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Yin and yang: Balancing and answering binary visual questions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5014–5022, 2016.

[68] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 5579–5588, 2021.

[69] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4995–5004, 2016.