Coherent Physical Commonsense Reasoning in Foundational Language Models

by

Shane Storks

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in the University of Michigan
2024

Doctoral Committee:

Professor Joyce Chai, Chair
Professor Jason J. Corso
Professor Richard L. Lewis
Associate Professor Lu Wang

Shane Storks

sstorks@umich.edu

ORCID iD: 0000-0002-5826-4426

DEDICATION

This thesis is dedicated to the educators who instilled in me a curiosity and desire to learn, innovate, and spark change.

I give a special honor to Ruth and Mike Cummins, as well as the community of organizers and volunteers behind Macomb Science Olympiad, for teaching me from a young age that I can accomplish anything I set my mind to with focus, determination, and creative thinking. Their years of continued encouragement inspired me to seek a career in science, and gave me the tenacity to pursue and complete this thesis.

／## ACKNOWLEDGEMENTS

I want to thank Prof. Joyce Chai for seeing my potential and giving me the life-changing opportunity to pursue this research in such unprecedented times for artificial intelligence. Prof. Chai has provided invaluable feedback throughout the course of this work, and supported my development as a scientist, educator, and mentor in countless ways.

I would also like to thank my committee members, Profs. Jason Corso, Richard L. Lewis, and Lu Wang, for their thoughtful feedback that shaped this work. I thank my other collaborators and student mentees for their contributions to my research: Dr. Qiaozi Gao, Yichi Zhang, Haoyi Qiu, Brianna Epstein, Wenfei Tang, Xiaoyang Hu, Zheyuan Zhang, Fengyuan Hu, Dr. Sungryull Sohn, Dr. Moontae Lee, Dr. Honglak Lee, Megan Su, Itamar Bar-Yossef, Ruixuan Deng, and Yayuan Li. I thank my other lab-mates in the Situated Language and Embodied Dialog (SLED) lab, Cristian-Paul Bara, Yuwei Bao, Keunwoo Peter Yu, Ziqiao Ma, Jianing Yang, Yinpei Dai, Andy Chung, Josue Torres-Fonseca, Shengyi Qian, and all other former and current SLED members who supported me through hours of helpful discussions and feedback about this work, collaboration on various projects, lifesaving tech and coding support, and their great friendship. I thank my internship mentors and collaborators at Amazon Alexa AI, Govind Thattai, Gokhan Tur, and Aishwarya Reganti, for helping me develop both my research and engineering skills. I thank the many anonymous reviewers who provided constructive feedback on this research.

On a personal note, I would like to thank my incredible support system of family and friends for graciously proofreading my papers, allowing me to rehearse presentations in front of them, giving me numerous pep talks, and being there for me through all the highs and lows of completing this work. I especially thank my husband, Austin Powell, and dog, Popcorn, who I leaned on the most of all. I also thank my colleagues from Queer in AI for offering community and support as I completed this work.

This research was supported by the National Science Foundation (IIS-1617682 and IIS-1949634), the DARPA XAI program through the University of California, Los Angeles (N66001-17-2-4029), LG AI Research, and the DARPA PTG program (HR00112220003). This research was also supported through computational resources and services provided by Advanced Research Computing (ARC), a division of Information and Technology Services (ITS) at the University of Michigan, Ann Arbor.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

LIST OF APPENDICES

ABSTRACT

Recent years in natural language processing (NLP) research have seen a paradigm shift toward foundational language models (LMs), which are self-supervised, transformer-based language models that are trained on large amounts of text data from the web that serve as flexible foundations that can be applied to downstream tasks with a small amount of fine-tuning or in-context learning from demonstrations. While *commonsense reasoning*, i.e., the ability to incorporate implicit background knowledge into natural language understanding (NLU), is a long-standing grand challenge in NLP research with decades of effort spent, these foundational LMs exhibit an apparent human-level proficiency on traditional NLU benchmarks requiring it. However, given limitations of these LMs, including their lack of transparency, tendency to exploit statistical bias in language data, and capability to hallucinate factual information, we argue that traditional benchmarking practices are no longer appropriate to evaluate the commonsense reasoning capabilities of foundational LMs.

In this thesis, we develop a new evaluation paradigm targeting *coherent commonsense reasoning*. While traditional benchmarks boil down NLU into high-level text classification tasks targeting various semantic phenomena, we first propose a notion of *consistency* of LMs' decisions by requiring them to localize these semantic phenomena within long language contexts, serving as evidence for decisions. To enable evaluation of consistency of foundational LM text classifiers, we propose a simple annotation scheme and apply it to two existing benchmarks for NLU and commonsense reasoning. Further, we propose a notion of *verifiability*, which requires LMs to explicitly generate the implicit commonsense background knowledge underlying this evidence, enabling evaluation and comparison to that of humans. We implement this concept in Tiered Reasoning for Intuitive Physics (`TRIP`), a new benchmark for coherent physical commonsense reasoning (PCR) in procedural texts.

Using the collected data, we evaluate and analyze the coherence of foundational LMs under traditional strategies to apply them to downstream tasks. While traditional approaches severely lack coherence in their commonsense reasoning, we develop new fine-tuning and in-context learning strategies inspired by the theory of dual processes in human cognition. Our cognitively motivated approach significantly improves coherence by focusing LMs' attention to the appropriate segments of the language context during each step of reasoning.

Lastly, in the wake of recent foundational vision-and-language models (VLMs) which can be applied to both image and language inputs for broad potential real-world applications, we adapt our notions of consistency and verifiability into visually grounded PCR. After performing an initial study of visual representations underlying these VLMs, we apply them to the challenging PCR task of procedural mistake detection in video frames. We develop automated, reference-free metrics for the relevance and informativeness of VLM-generated explanations in this problem, using them to create a novel, multi-tiered coherence evaluation of accuracy, consistency, and verifiability. We then draw from earlier findings to systematically investigate the impact of various interventions on VLM performance, and show how our evaluation framework can reveal a wealth of insights into the strengths and weaknesses of VLMs, enabling auditing and possible future improvement.

CHAPTER 1

Introduction†

We humans use a variety of knowledge and reasoning to help understand meanings of language. For example, consider these sentences from Marvin Minsky [163]: "Jack needed some money, so he went and shook his piggy bank. He was disappointed when it made no sound." From this, it is not difficult for us to understand that Jack did not find any money, and because of that, Jack was having a negative emotion. What makes us come to this conclusion, which was not explicitly stated in the text, is the knowledge we have about the world and the underlying reasoning process, often called **commonsense reasoning** [53], that allows us to connect pieces of knowledge to reach the new conclusion. For example, we know that a *piggy bank* is a pig-shaped container that holds coins, and that coins are pieces of currency made of metal. Since metal is a hard solid, the coins will make a sound when shaken inside of a container such as a piggy bank; if there is no sound, then there are no coins. It is also likely that we can predict that as piggy banks are typically possessed by children, there is a good chance that Jack is a child. Alternatively, these predictions may be derived from similar events we have experienced as children, and allow us to make similar conclusions by analogy [163]. While this kind of knowledge and reasoning comes so naturally to humans, it is notoriously difficult for machines due to reporting bias [84] and a long tail [53], which make collection and formalization a challenge. Despite significant advances in natural language processing (NLP) in the last several decades, including the recent advent of transformer-based **foundational language models (LMs)** pre-trained on web-scale text data, machines' proficiency in this type of **coherent** natural language understanding (NLU) supported by human-aligned commonsense knowledge and reasoning beyond the text remains under-explored.

In this thesis, we conduct a pioneering investigation into the coherence of commonsense reasoning in NLU achieved by foundational LMs, including methods to evaluate, strengthen,

†Shane Storks, Qiaozi Gao, and Joyce Y. Chai. Recent Advances in Natural Language Inference: A Survey of Benchmarks, Resources, and Approaches. *arXiv preprint arXiv:1904.01172*, 2019.

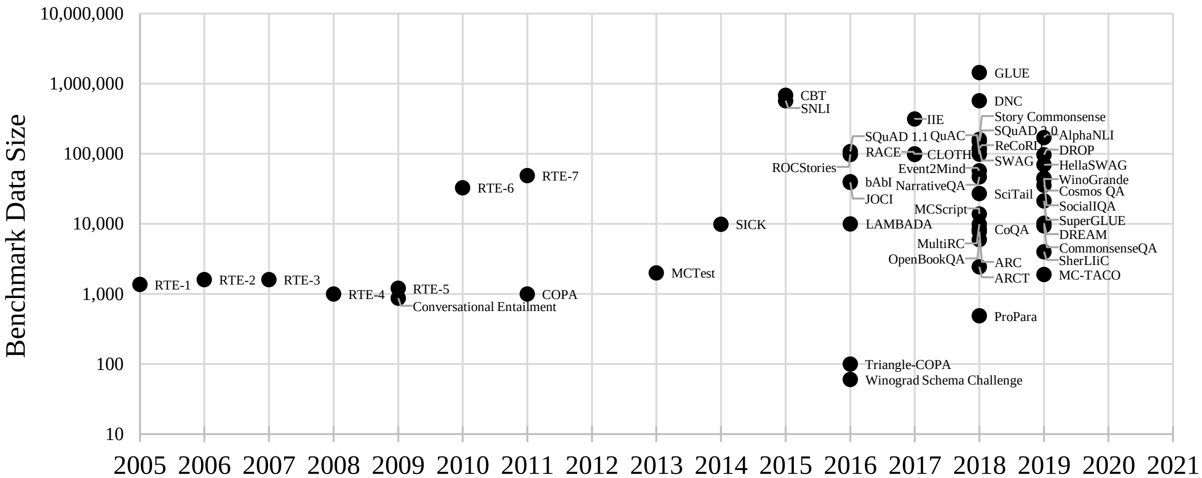


Figure 1.1: Since the early 2000s, there has been an explosion of benchmark tasks geared toward commonsense reasoning. In 2018, we saw the creation of more benchmarks of larger sizes than ever before. Data collected from papers cited in [227].

and apply commonsense reasoning in LMs. Targeting real-world embodied agent applications, we emphasize **physical commonsense reasoning (PCR)** in NLU, where LMs are expected to understand language about concrete objects, actions, and state changes to represent a dynamic physical environment. In this chapter, we first provide an overview of traditional benchmarking practices for commonsense reasoning in NLP systems. We then briefly introduce the advancements in NLP leading to foundational LMs and approaches typically used to apply them to benchmark tasks for commonsense reasoning in NLU, highlighting the key limitations of these approaches that make it difficult to objectively evaluate their performance. Lastly, we introduce the main contributions of this thesis research, and outline the remainder of the thesis which addresses these goals.

1.1 Benchmarking Commonsense Reasoning in NLP Systems

The NLP community has a long history of creating benchmarks to facilitate incremental algorithm development and quantitative evaluation for language processing tasks, e.g., for part-of-speech tagging [153], named entity recognition [86], question answering [97, 244], semantic role labeling [81, 192], and coreference resolution and relation extraction [60]. For earlier benchmarks like these, although it is often the case that some type of commonsense reasoning may be required to reach an oracle performance, they were primarily created to target approaches that apply linguistic context to solve these tasks.

As significant progress has been made by using the earlier benchmarks, recent years have seen a shift in benchmark tasks which are beyond the use of linguistic context, but rather require external knowledge and commonsense reasoning to solve the tasks. These benchmark tasks aim to require a deeper understanding to solve, thus targeting commonsense reasoning. These tasks typically take the form of text classification, where given some language context, a system must assign a label to it or choose from multiple text choices to complete the context or answer a question. Figure 1.1 shows a trend of growth among such benchmarks since 2000. The Recognizing Textual Entailment (RTE) Challenges [43] and Winograd Schema Challenge [128] had been the dominant reasoning tasks for many years, encouraging development of systems through competitions. More recently, there has been an increasing variety of benchmarks with a much larger number of data instances to facilitate training of deep neural networks.

Some examples of these benchmarks that have attracted significant activity in the research community are shown in Figure 1.2.[1] Situations With Adversarial Generations (SWAG) is a straightforward multiple-choice sentence completion task where given the beginning of a sentence, systems must choose the most plausible ending to the sentence based on knowledge about how the world typically works [267]. The General Language Understanding Evaluation (GLUE) benchmark offers a suite of textual entailment tasks focused on different domains and skills, some of which aim to require commonsense reasoning [247]. Textual entailment, originally proposed in the RTE Challenges, requires making a judgement of whether a hypothesis text must be true given some premise text. The particular example shown was adapted from the Winograd Schema Challenge, which poses a challenging problem where a reference (e.g., a pronoun) in a sentence can only be resolved based on implicit commonsense knowledge. Lastly, given two partial observations of a situation, Abductive Reasoning in narrative Texts (ART) [22] requires systems to choose the most plausible hypothesis that connects the observations, a form of abductive reasoning [184].[2] Benchmarks like these serve as useful tools to quickly evaluate commonsense reasoning capabilities in foundational LMs and other NLP systems, and have attracted significant research activity toward commonsense reasoning in NLU.

[1] Benchmarks for commonsense reasoning in natural language understanding are reviewed in greater detail in [227] and [52].

[2] A unique aspect of SWAG and ART is that the datasets were adversarially filtered to be especially challenging for pre-trained language models; see [267, 22] for more details.

(A) SWAG [267]

He pours the raw egg batter into the pan. He...

a. drops the tiny pan onto a plate

b. **lifts the pan and moves it around to shuffle the eggs.**

c. stirs the dough into a kite.

d. swirls the stir under the adhesive.

(B) GLUE [247], Winograd NLI [128]

Premise: The trophy doesn't fit into the brown suitcase because it is too large.

Hypothesis: The trophy is too large.
Label: **entailed**

Hypothesis: The suitcase is too large.
Label: **not entailed**

(C) ART [22]

Observation 1: There was ten feet of snow outside.

Observation 2: In all that time I was unable to check my mail.

Hypothesis:

a. **I couldn't open my door against a drift for 3 days.**

b. It took 10 minutes for the snow plow to come through.

Figure 1.2: Example natural language understanding benchmarks that require commonsense reasoning: Abductive Reasoning in narrative Texts (ART) [22], Situations With Adversarial Generations (SWAG) [267], and the General Language Understanding Evaluation (GLUE) [247]. Correct answers in bold.

1.2 Applying Foundational Language Models to Commonsense NLU Tasks

Nowadays, foundational LMs are the most common approaches to be applied to NLU benchmarks like these. First, we will introduce the key research developments from the last several decades that led to the creation of foundational LMs, as well as how the ways in which they were applied to benchmark tasks changed. We then discuss key limitations of these LMs that arose from this evolution which now make it difficult to objectively judge research progress on commonsense reasoning in NLU, despite their impressive capabilities and apparent superhuman performance on many benchmark tasks.

1.2.1 Evolution of Foundational Language Models

NLP approaches for commonsense reasoning and language understanding have continually evolved over the last several decades. The earliest approaches applied traditional symbolic and statistical methods. Symbolic approaches, derived from classical theories of logic and reasoning [11, 50, 167, 29, 184], parse language into logical forms and apply various operations to make inferences [154, 122, 166, 178, 35, 174, 150, 83, 51]. These approaches were highly accurate, but rigid and difficult to scale. Meanwhile, statistical approaches were used to develop the first instances of **language models** (LMs), which counted co-occurrences of tokens and n-grams to estimate probabilities of sequences in language and classify texts

[216, 148, 221, 33, 21, 208, 112, 36, 207].

The advent of neural networks in statistical language processing led to the ability to learn latent semantic vector representations of tokens [54, 20, 25, 159, 158, 186], which could later be reused as inputs when training task-specific deep neural networks for language tasks [210, 98, 38]. While these neural language models enabled progress on various language tasks (e.g., machine translation), the requirement of large amounts of in-domain training data, difficulties handling long-range dependencies in text, and the challenge of incorporating external background knowledge remained bottlenecks for commonsense reasoning [227, 214]. The invention of attention mechanisms [14], transformers [241], and contextualized token representations [187, 194, 56, 142] were instrumental in further progress, as they enabled LMs to implicitly learn dependencies between every pair of tokens in text, and could be efficiently trained on large corpora of text scraped from the web through self-supervision. Due to their pre-training on open-domain, web-scale data, these **foundational LMs** acquired a breadth of knowledge about the distribution of natural language that could serve as a flexible foundation for various downstream tasks. More than ever before, these LMs were easily adaptable through directly *fine-tuning* the architecture, i.e., training it end-to-end, on a small amount of in-domain data from datasets like the benchmarks discussed previously.

A period of scaling up the training data and complexity of LMs followed, along with various improvements to training paradigms, which led to foundational LMs becoming more ubiquitous and capturing public attention in recent years [195, 34, 236, 2, 76]. This is largely because as these LMs scaled, new capabilities emerged. Most notably, they became capable of generating high-quality, task-specific language without task-specific training. To apply them to downstream tasks, it became viable to *prompt* them directly (zero-shot) or apply *in-context learning* by providing a small number of demonstrations for a task before eliciting a prediction from them [34]. Later work found that it was possible to prompt foundational LMs with a chain-of-thought (CoT) to demonstrate reasoning chains to support inference on downstream tasks in in-context learning, improving performance on complex reasoning tasks [254, 118].

1.2.2 Limitations of Foundational LMs

As shown in Figure 1.3, foundational language models like BERT [56], RoBERTa [142], and T5 [197] marked a period of rapid progress on NLU benchmarks, with state-of-the-art accuracies rising by over 20% (sometimes in a matter of months), and approaching and exceeding human performance. This is striking, as while commonsense reasoning had been thought to be a major bottleneck in automating NLU for decades, these results on NLU benchmarks

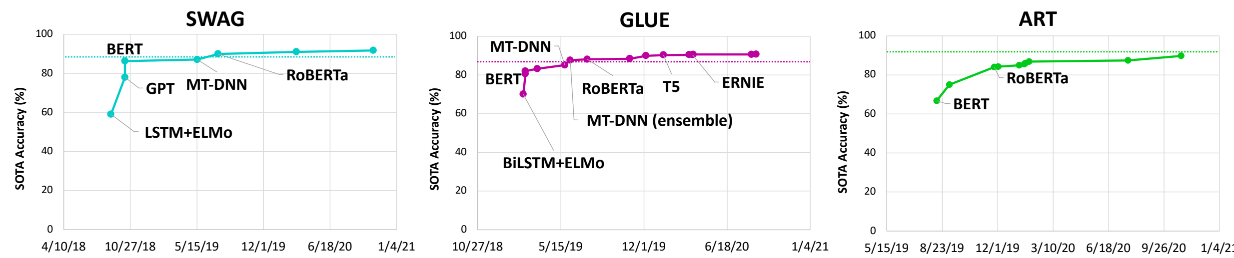


Figure 1.3: Graph of state-of-the-art accuracy by language models (as it approached human accuracy) on three NLU benchmark tasks for evaluating commonsense reasoning: SWAG [267], GLUE [247], and ART [22]. Human performance shown as horizontal dotted lines.

targeting commonsense reasoning suggest the problem may finally be solved. In this thesis, we argue that on the contrary, key limitations of foundational LMs make it difficult to objectively evaluate their commonsense reasoning through traditional benchmarking practices.

Lack of transparency. At their core, deep neural networks are pattern recognition models which learn large numbers of parameters in nonlinear functions to capture variations in high-dimensional language representations. As in many sub-fields of artificial intelligence (AI) where deep learning approaches are heavily applied, transparency and the ability to explain system behaviors are essential for trustworthiness. However, as state-of-the-art LMs have been scaled up, their architectures consist of up to hundreds of billions of learned parameters [34]. Consequently, the underlying reasoning process of these models cannot be easily interpreted, and it is quite opaque why particular conclusions are made by these models. Chapter 2.1 introduces related work toward interpreting decisions made by LMs, which typically requires additional steps like training probing classifiers on top of LMs, systematically manipulating LM inputs or architectures, or curating adversarial data that can target specific phenomena. As there is not yet an agreed-upon or general method to interpret LMs [219], this lack of transparency remains a drawback of LMs as opposed to earlier symbolic and statistical approaches for NLU.

Superficial correlations in language data. This lack of transparency becomes problematic when we consider the presence of superficial statistical biases in datasets used to train and evaluate AI systems, particularly in deep neural networks. Unlike biases in class labels or domains, which are easy to resolve by balancing the data during collection or sampling, such superficial correlations are not straightforward to resolve, and create a number of challenges in practical applications of neural networks. In image classification, the insidiousness of these biases became apparent through the use of adversarial attacks; for example,

[229] found that small changes (virtually undetectable by humans) in images could change the predictions of image classifiers, making it possible for bad actors to manipulate their decisions. This brittleness of neural networks also impacts our ability to benchmark system capabilities, which [202] demonstrated by showing that spurious correlations in training data can cause accurate decisions by image classifiers to be based on invalid evidence from images. For example, when classifying images as wolves or huskies, it was found that the presence of snow in the background of images was the key factor in this decision rather than features of these animals.

This prompted extensive investigation of such biases in language data and brittleness of NLP systems (including recent foundational LMs) through adversarial attacks and training text classifiers on incomplete inputs [4, 215, 109, 90, 191, 104, 209, 79, 177, 143]. While these biases can be linked to meaningful properties of data, e.g., gender bias [209], they are largely driven by more incomprehensible cues like punctuation or otherwise semantically inconsequential tokens or phrases that create correlations between task context (e.g., questions) and answers (e.g., text choices or class labels). For example, [215] found that for the Story Cloze Test [168], a commonsense NLU benchmark requiring systems to choose a plausible ending for a story from two candidates, correct answers were more likely to use exclamation marks (i.e., "!"), despite this being generally unimportant in understanding the plausibility of candidate endings. This incoherent behavior enables fine-tuned foundational LMs to achieve artificially high performance on benchmark tasks, as exploiting shortcuts like these enables them to bypass true reasoning. It is important to note that these biases are dataset-specific, often introduced by patterns in human annotation [215, 79], which do not reflect the broader distribution of natural language and thus undermine the performance of LMs in real-world applications. While Chapter 2.3 introduces approaches from prior work to make fine-tuned LMs resilient to superficial statistical biases, the dominant task-agnostic approach here is adversarial filtering [267], a cyclical approach which requires regenerating and filtering datasets regularly based on whether examples are difficult for fine-tuned state-of-the-art LMs to classify. Furthermore, similar problems with bias have been found even in prompting and in-context learning approaches to apply LMs to downstream tasks [253, 218], which introduce additional challenges specific to text generation settings, e.g., hallucination [108], where LMs generate fluent-sounding language with illusory or false information. The issue of spurious correlations in language data thus appears inescapable for the current paradigm of foundational LMs.

Despite the impressive results and apparent human-like reasoning capabilities of foundational LMs, these limitations compromise traditional benchmark evaluations of their commonsense reasoning and NLU. This problem motivates the creation of deeper evaluations of reasoning which go beyond high-level text classification tasks and require LMs to explain their predictions with coherent, human-aligned reasoning chains. Such auditing tools will be especially important for applying LMs to real-world physical settings which recent work has already begun to explore [101, 5, 220, 87]. In such settings, LMs could be embodied in robots and collaborate with humans in a shared environment, where they would be expected to understand human instructions. This understanding requires PCR and visual grounding to understand the implications of actions on the environment with respect to the current state of the environment. To ensure safety and trust in this setting, it is critical that the depth of LMs' understanding of language and its surroundings matches that of their human partners.

1.3 Contributions

In light of the inability of current benchmarking and modeling practices to account for the incoherent behaviors of foundational LMs, this thesis makes several novel contributions. First, **we develop a multi-faceted evaluation paradigm for coherent commonsense reasoning in NLU**. This paradigm introduces a notion of *consistency* between an LMs' text classification predictions and its beliefs on sub-segments of language contexts, particularly for problems where the context consists of multiple sentences that systems must reason over (incorporating commonsense knowledge as needed) to fully understand. Evaluating consistency in NLU text classification tasks enables a previously unexplored view into the evidence supporting LMs' predictions. Further, this paradigm adopts a notion of *verifiability* of LMs' predictions, where domain-specific commonsense knowledge invoked by the language context is made explicit in LMs and additionally evaluated, enabling greater transparency and ability to compare with human reasoning.

To enable quantitative evaluation of consistency and verifiability in foundational LMs applied to commonsense NLU tasks, **we curate densely annotated benchmark datasets for coherent commonsense reasoning.** To evaluate the consistency of LMs, we propose a simple yet effective technique to annotate discourses of language to characterize the ground truth evidence that supports text classification. We apply this technique to annotate two existing benchmark datasets related to commonsense reasoning [269, 22]. Further, to enable investigating LMs' ability to learn consistent and verifiable reasoning structures, we compile Tiered Reasoning for Intuitive Physics (`TRIP`), a new multi-tiered benchmark dataset of procedural texts annotated with traces of PCR. `TRIP`, like other NLU benchmarks, includes a high-level classification task (physical plausibility), but additionally includes annotated evidence in the form of conflicting sentences creating plausibility conflicts in the texts (targeting consistency), and commonsense physical state changes implied by each sentence (targeting verfiability). These datasets are valuable resources for the community to incrementally strengthen the commonsense reasoning (especially PCR) in LMs while accounting for their spurious behaviors, with a goal of coherent reasoning aligned with that of humans.

Next, **we apply foundational LMs to these benchmark tasks through several empirical experiments**. To understand whether LMs reason consistently in text classification tasks, we fine-tune LMs on our two annotated NLU benchmarks. To investigate whether LMs can learn consistent and verifiable reasoning chains jointly with text classification tasks, we fine-tune LMs on `TRIP` with various loss configurations and perform a detailed analysis. In these experiments, we find that LMs struggle to learn coherent reasoning strategies through traditional fine-tuning methods. Inspired by dual process theories of human cognition, we then propose novel heuristic-to-analytic strategies for fine-tuning and prompting LMs on PCR in `TRIP`. These strategies condition lower-level steps with higher-level steps of the reasoning process, sharply improving the coherence of PCR by focusing LM attention to the most relevant parts of the language context during each step.

To complete this thesis, **we extend the problem of coherent PCR to multimodal settings with recently developed foundational vision-and-language models (VLMs)**. We first explore the strengths and weaknesses of foundational vision-and-language representations in capturing physical concepts, e.g., objects, states, and actions, finding that they capture physical state descriptions better than higher-level action descriptions. Drawing from this, we adapt our concepts of consistency and verifiability to procedural mistake detection (PMD) in egocentric video frames. Specifically, we extend this reasoning-intensive multimodal problem, previously posed as a high-level classification task for whether a procedure was successfully completed, by requiring VLMs to generate low-level explanations for their decisions through visual questions and answers. We develop novel, automated, reference-free evaluation metrics for the relevance and informativeness of VLM-generated explanations in this problem, using them to recreate a multi-tiered coherence evaluation of accuracy, consistency, and verifiability. We then systematically investigate the impact of various interventions on VLM performance: varying the direction of reasoning steps, incorporating our coherence metrics and in-context learning into explanation generation, and applying recently developed approaches to mitigate incoherent visual processing tendencies in VLMs (e.g., hallucination). We show that prioritizing coherence through our proposed metrics can yield large gains in both coherence and accuracy of PMD, while visual hallucination mitigation methods can optimize VLMs' efficiency, confidence, and reliability. Lastly, we show how our evaluation framework can reveal a wealth of insights into strengths and weaknesses of VLMs here, enabling inspection and possible future improvement of systems.

1.3.1 Thesis Outline

The remainder of this thesis is organized as follows:

Chapter 2 summarizes prior work in several areas related to the contributions of this thesis.

Chapter 3 introduces the concept of *consistent commonsense reasoning*, details our process to annotate existing commonsense reasoning benchmarks to evaluate consistency, then evaluates the consistency of foundational LMs fine-tuned on the unannotated benchmark training data.

Chapter 4 introduces the concept of *verifiable commonsense reasoning*, and discusses the curation of the multi-tiered `TRIP` benchmark for consistent and verifiable PCR.

Chapter 5 presents experiments to evaluate the consistency and verifiability of foundational LMs fine-tuned on the sub-tasks of `TRIP`.

Chapter 6 implements, evaluates, and analyzes cognitively motivated PCR strategies for LM fine-tuning and in-context learning.

Chapter 7 performs initial tests of PCR in foundational VLMs, identifying key challenges for applying them to more natural multimodal task settings requiring PCR.

Chapter 8 adapts concepts of coherence to multimodal PCR, resulting in a novel interpretable formulation of PMD in video frames, automated evaluation metrics for evaluating coherence of VLMs' decisions and explanations, and a thorough initial analysis of the impact of various interventions on the accuracy, coherence, efficiency, and reliability of VLMs in PMD.

CHAPTER 2

Related Work

This thesis makes contributions to several areas of ongoing research:

1. **Interpreting and evaluating reasoning in language models** (Chapters 3-6)
2. **Benchmarking physical commonsense reasoning in AI systems** (Chapter 4)
3. **Improving reasoning in language models** (Chapters 5-6)
4. **Applying dual process theory in AI** (Chapter 6)
5. **Grounded language understanding in foundational multimodal models** (Chapters 7-8)
6. **Procedural mistake detection for task guidance** (Chapter 8)

In this chapter, we highlight relevant work from each area, and contextualize the contributions of this thesis within each area.

2.1 Interpreting and Evaluating Reasoning in Language Models

To address the black-box nature of neural NLP models, including foundational language models (LMs), and questions about the spuriousness of their decisions, a broad space of past work has proposed methods to interpret and evaluate their reasoning processes in various tasks and domains. [18, 219] provide detailed reviews of work in this space, while we highlight some of the most relevant work below.

In addition to the adversarial attacks discussed in Chapter 1.2.2 which exposed incoherent behaviors in LMs, several prior works have behaviorally tested NLP models like LMs to evaluate their reasoning capabilities. Some work has studied coherence of generated language through the proxy task of text ordering [123, 144]. Other work has collected and generated specialized data to stress test the coherence and consistency of models' reasoning processes in various settings [203, 57], some targeting specific aspects like natural language inference [155], question answering [107], causal inference [111], and memorization of training data [92]. Related to these efforts, in Chapter 3 of this thesis, we propose an easily-accessed, versatile evaluation of reasoning consistency in text classifiers which can be enabled from only a small amount of additional annotations on spans of a discourse. Using this evaluation method, we show that foundational LMs fine-tuned on commonsense reasoning tasks often make decisions based on evidence inconsistent with that used by humans. Further, in Chapters 4 and 5, we develop Tiered Reasoning for Intuitive Physics (`TRIP`), a novel specialized benchmark to evaluate the consistency and verifiability of intuitive physical commonsense reasoning (PCR), an under-explored challenge in natural language understanding (NLU).

Meanwhile, another line of work seeks to understand model behaviors through their internals. To interpret the semantic vector representations used by LMs to make decisions and the knowledge they actually capture in various task settings, a common approach is to train lightweight (often linear) probing layers on them [3, 64, 234, 96, 106, 233, 40]. An alternative approach is to elicit and analyze attention-based explanations for model decisions and generated language over other possible decisions and generations [105, 262]. Parallel work has applied causal methods to judge the importance of aspects of model inputs in reasoning, locate knowledge in model parameters, and identify roles of model components [132, 157, 77, 225, 226, 99, 248]. This line of work, recently referred to as mechanistic interpretability, has developed numerous statistical methods to analyze the flow of information in transformer-based models during training and inference [170, 243, 78, 48, 172, 165, 235, 95]. The majority of this thesis focuses on developing behavioral evaluations for the coherence of commonsense reasoning, especially PCR, in foundational LMs, as there remains significant room for improvement in this area even through such evaluations. However, in Chapter 6, we contribute to this line of work by proposing metrics for the faithfulness, precision, and recall of model attention over a discourse, using it to demonstrate the advantage of a cognitively inspired LM prompting approach proposed there.

2.2 Benchmarking Physical Commonsense Reasoning in AI Systems

Some past works have proposed NLP benchmark datasets around the domain of PCR, which offer various classification tasks that have been used to evaluate the capabilities of LMs.

In this area, benchmarks take various forms. Some benchmarks target the low-level prediction of changing physical states implied by procedural text [46, 232, 268]. Other benchmarks use higher-level tasks like question answering and textual entailment to measure LMs' understanding of specific aspects of PCR, such as physical properties of actions and objects [71, 24, 12], temporal reasoning [278, 279], and spatial reasoning [164]. Visual [113, 15] and multimodal [102, 49, 9, 217] benchmarks also investigate systems' commonsense understanding of the physical world through perception and interaction. Different from existing benchmarks, this thesis proposes `TRIP` (Chapter 4), a first-of-its kind benchmark which combines aspects of previous text-based benchmarks for a multi-tiered evaluation targeting coherent PCR. This multi-tiered task creates a reasoning chain from low-level physical states to higher-level decisions of locating plausibility conflicts and the end task of choosing a text which is more physically plausible. Through this benchmark, we find that classification tasks are not enough for LMs to learn to reason coherently, and thus it becomes essential to elicit multi-tiered reasoning chains like these to support decision-making tasks. This may be especially important for machine reasoning in real-world physical settings, where humans and agents powered by LMs may share an environment.

2.3 Improving Reasoning in Language Models

Another broad body of work has developed strategies to strengthen the reasoning capabilities in NLP models. These efforts can be categorized into two areas: supervised methods to train and fine-tune LMs to reason, and unsupervised methods to prompt LMs in a way that facilitates reasoning.

While progress has been made by fine-tuning LMs on various specialized reasoning tasks to augment their capabilities in natural language inference [255, 240], mathematical reasoning [80], temporal reasoning [279], and other forms of commonsense reasoning [198, 213], such effort is largely data-driven and thus vulnerable to superficial statistical bias. Some approaches have been proposed to remove biases from language by filtering out data too easily discriminated by state-of-the-art fine-tuned text classifiers [267, 176], and to improve robustness and consistency of systems through specialized architectures, learning objectives, and data augmentation [19, 39, 135, 161, 13]. Meanwhile, other work has attempted to compile large amounts of semi-structured commonsense knowledge [212, 169] and inject this knowledge into pre-trained LMs [31, 275] in order to enable knowledge-supported language understanding and on-the-fly explanation. Related to these efforts, in Chapter 5, we proposed a multi-tiered strategy to fine-tune LMs to jointly make decisions and explain them in coherent PCR, finding that removing a high-level decision-making objective from training improved coherence of reasoning. Following this, some work (discussed in Chapter 6) similarly attempted to build in explicit coherent reasoning structures [149, 204, 130], but these approaches reason from the bottom up over various representations of the world state, or jointly optimize all reasoning steps without dependency. In Chapter 6 of this thesis, we instead found that the coherence of commonsense reasoning in fine-tuned LMs could be majorly improved through a top-down heuristic-analytic reasoning strategy inspired by dual process theories of human cognition.

With the introduction of GPT-3 [34], in-context learning became a common way to apply LMs to new tasks without in-domain gradient-based training, where one prompts the LM with helpful task-specific knowledge or even full demonstrations at inference time before finally asking it to solve a task. A number of works found applications of in-context learning in PLMs for complex reasoning tasks [47, 230, 223]. Among these works, significant improvements came from inserting or generating free-text reasoning chains in prompts to support task predictions [254, 118]. These findings sparked the exploration of many different in-context learning and sequential prompting approaches to strengthen reasoning in PLMs and tackle various tasks [116, 179, 266, 258, 261, 145, 251]. These methods usually rely on an assumption that by decomposing a high-level task into many low-level sub-tasks, the model can solve the low-level sub-tasks easily, which helps it achieve better performance on the high-level task. However, in complex cases like commonsense reasoning, even lower-level sub-tasks are hard to solve due to the requirement of retrieving and incorporating knowledge beyond the text. As such, in Chapter 6, this thesis proposes a heuristic-analytic in-context learning approach which instead uses higher-level decisions to refine the generation of low-level commonsense knowledge from LMs.

2.4 Applying Dual Process Theory in AI

Dual process theories of cognitive psychology have recently attracted interest in various areas of AI. [10] apply them to augment reinforcement learning algorithms with deliberative planning of policies through tree search. [72] combine them for more efficient navigation with AI agents that evolve from slow to fast decision-making while navigating. Similarly inspired by dual process theories, [100] apply logical reasoning over representation learning for more accurate commonsense knowledge base completion. [125] use dual-process inspired associative selection combined with evidence generation to perform question answering on scientific articles. [28] propose additional research questions and directions around the application of dual process theories of human cognition in AI. Complementary to these past works, we apply dual process theories of human cognition in coherent PCR with foundational LMs in

Chapter 6, both through fine-tuning and in-context learning.

2.5 Grounded Language Understanding in Foundational Multimodal Models

A large amount of recent work has been devoted to grounding foundational LMs for NLU and reasoning in the visual modality, especially relevant to tasks involving physical reasoning about real-world environments. The most relevant work to this thesis has occurred in three general areas: developing multimodal representations suitable for PCR, analyzing and evaluating NLU and physical commonsense in foundational vision-and-language models (VLMs), and formulating strategies to improve the visual reasoning capabilities of foundational VLMs.

First, prior work has attempted to build and strengthen multimodal representations specifically for physical reasoning tasks, such as understanding object states and state changes [42, 138]. A significant line of work has attempted to integrate a notion of visual imagination into physical commonsense NLU through physical simulators [17, 58, 141, 120, 119] and intuitive simulation with language-conditioned image and video generation [242, 151, 147, 260, 115, 131, 263]. Complementary to these efforts, which largely target synthetic settings, in Chapter 7, we probe foundational multimodal representations to evaluate their suitability for physical state prediction in natural images. Notably, we take advantage of foundational LMs and text-to-image diffusion models to implement textual and visual intuitive physical simulation strategies, finding that these representations capture declarative textual physical state descriptions better than descriptions of actions or images of objects in particular states.

Vision-and-language pre-training paradigms have rapidly evolved in recent years and seen much success in multimodal tasks [231, 136, 272, 117, 238, 134, 6, 2, 133, 44, 185, 140, 139, 76, 271, 1], enabling the development of VLMs which can generate language about input images. Several works have analyzed the grounded NLU capabilities of VLMs [103, 264], including relevant studies on their capturing of object properties [91, 270] and physical actions [94, 265, 110, 175]. Other works have attempted to benchmark their capabilities in multimodal applications requiring physical reasoning, such as robotic planning and manipulation [101, 5, 220, 87, 173], as well as procedural mistake detection (PMD), where a VLM must determine whether a human has performed a procedure correctly based on a text description of the procedure and an image or video of the current state of the environment [62, 16]. In Chapter 8, we extend previous work to analyze the capability of VLMs to generate explanations of their decisions in PMD, a previously unexplored aspect of the problem.

These explanations are formulated as a series of questions and answers that infer the success conditions from the procedural text and check for them in an image, thus testing VLMs' grounded NLU capabilities. To evaluate explanations, we propose automated metrics based on pre-trained NLI models to judge the coherence of generated explanations for decisions. We then apply these metrics, along with in-context learning from human-generated questions, to improve question generation and thus improve the accuracy and coherence of VLM decisions and explanations.

Related to the above, several works have specifically studied the visual perception capabilities of foundational VLMs. VLMs are vulnerable to visual hallucination and illusion [45, 137, 274, 88], making it difficult to apply them to reasoning tasks requiring visual perception. A recent line of work has attempted to address this by proposing strategies to manipulate VLMs' input images and output logits in such a way that removes illusory and irrelevant information [245, 127, 8]. An alternative approach is to train question-aware visual encoders to ensure visual representations capture only the most prompt-specific information [73]. Other works have attempted to address this through VLMs' textual inputs by utilizing foundational LM-generated questions and feedback to guide VLMs through complex reasoning and question answering problems [224, 37, 280]. As mentioned above, in Chapter 8, we follow these works to frame PMD as a self-dialog of questions and answers generated by VLMs. We then examine the impact of some hallucination mitigation strategies [245, 127, 8] on VLMs' performance in PMD, finding that they can improve the efficiency of explanation and reliability of decisions at a possible cost of coherence.

2.6 Procedural Mistake Detection for Task Guidance

Lastly, prior work has developed resources and approaches for PMD in the broader problem space of task guidance. Early systems for task guidance provided the user with pre-defined, task-specific, context-agnostic information without the capability to track the state of the environment, provide user-specific feedback, or generalize to new tasks [181, 180, 250, 126, 201]. *Interactive task guidance* based on visual perception, language communication, and mixed reality is a recently emerging area of AI research that aims to address these limitations, largely stemming from a DARPA program on the topic [55].

To enable research on this problem as well as the sub-problem of procedural mistake detection (PMD), prior work has produced annotated datasets [62, 16, 249, 183] and built development platforms [27]. Related to these resources, in Chapter 8, we recast the Ego4D dataset [85] as a PMD problem, providing a large-scale data source for PMD in simple everyday procedures. This dataset lowers some barriers to understanding VLMs' capabilities in PMD by focusing on narrated actions rather than recipes (which can include difficult-to-perceive aspects like temperatures, times, and small measurement quantities), and excluding any dialog or social interactions. We refer to this simplified dataset as Ego4D for PMD (Ego4D-PMD). In addition to Ego4D-PMD, we propose automated, reference-free coherence metrics for natural language explanations in PMD.

These prior works have also begun to develop approaches to address this problem. [249, 183] fine-tuned specialized neural models for PMD and other sub-problems of interactive task guidance based on features from video frames, eye tracking, and hand tracking. Meanwhile, [62] explored the role of foundational VLMs for binary PMD, finding that task- and domain-specific fine-tuning was required to achieve a viable level of performance. [16] extended this inquiry to apply foundational LMs and VLMs to the entire problem of interactive task guidance, including environmental state tracking, user modeling and interaction, and PMD. This work found that a major bottleneck of PMD is that visual information extracted through VLMs tends to be noisy and overly vague or high-level. All of these early approaches struggled to achieve a viable level of performance in PMD, possibly because they largely approached PMD as a classification problem without language or explicit reasoning. As mentioned above, in Chapter 8, we formulate PMD as an explicit reasoning problem through a self-dialog of questions and answers generated by a foundational VLM, enabling evaluation of the coherence of PCR in foundational VLMs for PMD. We further apply various interventions in both question generation and answering, achieving significant gains in accuracy, coherence, efficiency, and reliability of VLMs for PMD. Notably, this reformulation of the problem enables previously impossible visualization and understanding of common bottlenecks in PMD, such as gaps in commonsense knowledge and erroneous visual perception (e.g., object hallucination).

CHAPTER 3

Consistent Commonsense Reasoning†

As discussed in Chapter 1, while foundational language models (LMs) have approached or exceeded human performance on many existing natural language understanding (NLU) benchmarks requiring commonsense reasoning, the coherence of these state-of-the-art models and their alignment to human reasoning is not well understood. This is perhaps because benchmarks geared toward NLU only cover the tip of the iceberg, typically focusing on a high-level end task rather than diving deeper into the kind of coherent, robust understanding based on commonsense knowledge and reasoning that humans are capable of. Specifically, NLU in machines is often boiled down to text classification, where a classifier (typically built with a foundational LM in recent work) is tasked with recognizing whether a text contains a particular semantic class, e.g., textual entailment [43, 32], commonsense implausibility [205, 168, 22, 24], or combinations of several phenomena meant to serve as comprehensive diagnostics [190, 247, 246]. Without regard to the underlying evidence from the language context that is used to reach a conclusion, systems are rewarded for correct predictions on the task without "showing their work."

To make meaningful improvement on machine NLU, it is important to have more informative performance measures. To address this issue, the key contribution of this chapter is to introduce a novel model- and task-agnostic evaluation framework that allows a quick assessment of text classifiers' ability in terms of the coherence of their predictions. Specifically, we apply text classifiers to all sub-spans of sentences in a discourse, enabling us to isolate the specific contextual evidence used to make prediction on the full discourse. We then compare their predictions with that of humans to determine whether the evidence used to make decisions is *consistent* with humans. We demonstrate our framework in two different NLU benchmark tasks, highlighting its versatility. This evaluation framework, although

†Shane Storks and Joyce Chai. Beyond the Tip of the Iceberg: Assessing Coherence of Text Classifiers. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3169–3177, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

simple in ideas and implementation, is effective as a quick measure to provide insight into the coherence of machines' predictions. Finally, we use this framework to determine whether foundational LMs fine-tuned on classification-based NLU tasks use consistent contextual evidence to arrive at conclusions.

3.1 Defining Consistency in Text Classification

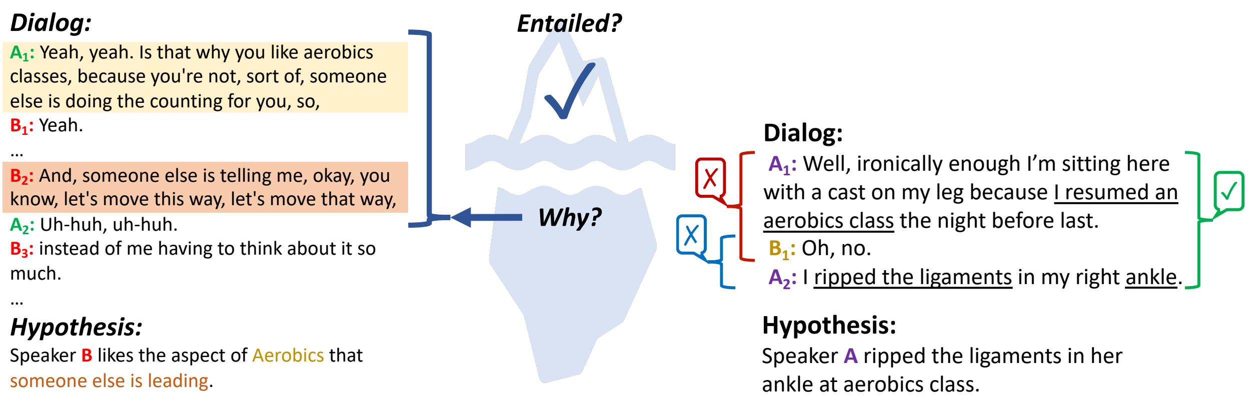
For any text classification task requiring reasoning over a discourse, a coherent classifier should use the same evidence as humans do in reaching a conclusion. For any positive example, we expect that there are specific regions of the text which contain the semantic class of interest and thus directly contribute to the positive label. Conversely, for any negative example, there should be no such regions of the text. First, we will propose novel consistency metrics to capture whether foundational LM text classifiers can give consistent and human-aligned predictions on these regions to support the end task conclusion.

Depending on specific tasks, this measure can have different implementations while maintaining the same high-level goal. In the following sections, we will use two example benchmark datasets, Conversational Entailment (**CE**) [269] and Abductive Reasoning in narrative Text (**ART**) [22], to illustrate how this metric can be applied. We intentionally chose these two distinctive benchmark datasets for our investigation which follow typical commonsense NLU task formulations. CE is formulated as a textual entailment task, while ART is a multiple-choice text plausibility classification task. CE is small-scale, created over ten years ago before the era of deep learning, while ART is a large-scale (∼171k examples) dataset created more recently. Through these two different datasets, we aim to demonstrate the versatility of this framework.

3.1.1 Consistency in Textual Entailment

CE poses a textual entailment task where context is given as several turns of a natural language dialog, and we must determine whether the dialog entails a hypothesis sentence. All required information is explicitly given in the dialog. In each positive example, only some dialog turns directly contribute to the entailment, while others are irrelevant to the hypothesis. For example, as shown in Figure 3.1a, turns A_1 and B_2 together entail the hypothesis, while others are not necessary for entailment.

As shown in Figure 3.1b for CE, we can label individual spans of a discourse that entails a hypothesis with whether or not consecutive sub-spans of the discourse also entail the hypothesis. Here, while the entire dialog from A_1 through A_2 entails the hypothesis, the



(a) Entailment and corresponding evidence. (b) Sub-span annotation to capture evidence.

Figure 3.1: In Conversational Entailment (CE) [269], language model text classifiers only predict whether a hypothesis is entailed by a dialog, while ignoring the underlying evidence in the discourse toward this conclusion. To enable a systematic evaluation of the consistency of models' predictions with the underlying evidence contributing to them, we label each sub-span of dialog with whether it entails the hypothesis (✓ for yes, ✗ for no).

spans from A_1 through B_1 and B_1 through A_2 do not, as they omit details required by the hypothesis. Given an example of length N,[1] we can decompose it into $N + \binom{N}{2}$ possible consecutive sub-spans[2] to label with human judgements.

For a correctly classified example, we can then perform inference on all sub-spans. If the system additionally classifies all of them correctly, we consider the prediction to be consistent. We then calculate **consistency** on the task as the percentage of examples coherently classified. Extremely simple to compute, this provides valuable insight beyond the surface of end task accuracy, measuring how well the classifier's perceived evidence toward the conclusion aligns with that of humans. Alternatively, the average sub-span accuracy may be considered as a more lenient measure.

3.1.2 Consistency in Plausibility Classification

ART, meanwhile, is a multiple-choice text classification benchmark for commonsense plausibility recognition. The task is to determine which of two candidate sentences most plausibly fits between two given context sentences when considering commonsense constraints on the world. This translates naturally into a choice between two three-sentence stories (differing

[1]Length can be defined in units of dialog turns, sentences, paragraphs, or other appropriate units of the text. Text should be decomposed such that individual sub-spans are not malformed or fragmented, so token- and character-level sub-spans will typically be inappropriate for this evaluation.

[2]There are $\binom{N}{2}$ combinations of starting and ending points for multi-sentence sub-spans, plus N individual sentences.

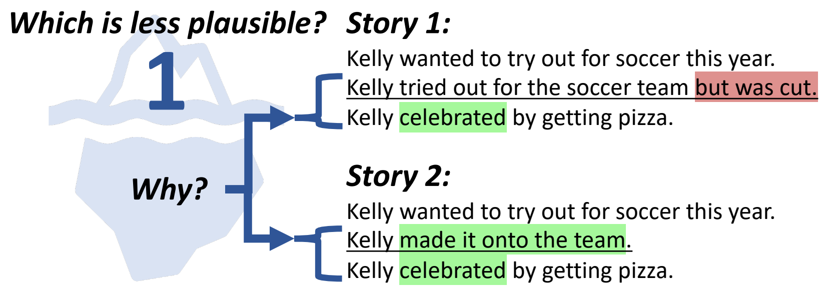


Figure 3.2: In Abductive Reasoning in narrative Texts [22], systems only compare two texts by their commonsense plausibility, ignoring which parts of the stories support this conclusion.

only by the second sentence), one of which has some implausibility (the positive choice). For example, as shown in Figure 3.2, Story 1 is implausible because while the second sentence describes a negative event, the third sentence indicates celebration. Meanwhile, in Story 2, the agent is celebrating a positive event.

Multiple-choice tasks. To account for multiple-choice tasks like ART, where we identify one of two texts to be semantically implausible, we must adjust this setup. We still consider sub-spans of the context, breaking down each pair of texts into $N + \binom{N}{2}$ pairs of sub-spans. Intuitively, the model's choice on each pair should again align with that of humans. However, there is a possibility that none of the texts contain the positive class. In such cases, the classifier should not make a confident prediction, and instead believe the texts are equally likely. Confidence should be defined based on the classifier's internal model of the probability distribution over all possible class labels, i.e., text choices (typically calculated by applying softmax over the activations of several neural network branches). This is conceptually visualized in Figure 3.3, where a classifier should only become confident that Story B is implausible once both the second and third sentence are present, as *the trash* is less likely to end up on *the floor* with a *hole in the top* of the bag.

Generally, let $T_{a:b}$ represent the consecutive sub-sequence of text T from unit a through b, e.g., sentences a through b of text T. Consider a set $S_{1:N}$ of M texts of length N such that $S = \{T^1_{1:N}, T^2_{1:N}, \cdots, T^M_{1:N}\}$, and a classifier f such that $f(S_{1:M}) \in [1, M]$.[3] When classifying a set $S_{a:b}$, let $f(S_{a:b}) = c^*$ be considered a *confident* prediction if $\max_{c \in [1,c^*) \vee (c^*,M]} (p(c^*) - p(c)) \geq \rho$, where $p(c)$ refers to probability of class c under the classifier's output distribution, and ρ is a confidence threshold. Where there is no positive text within $S_{a:b}$, then the desired outcome (ground truth) is for $f(S_{a:b})$ to be a non-confident prediction. This should be reflected in the calculation of consistency.

[3]While text choices may be different lengths, this can be trivially resolved by padding.

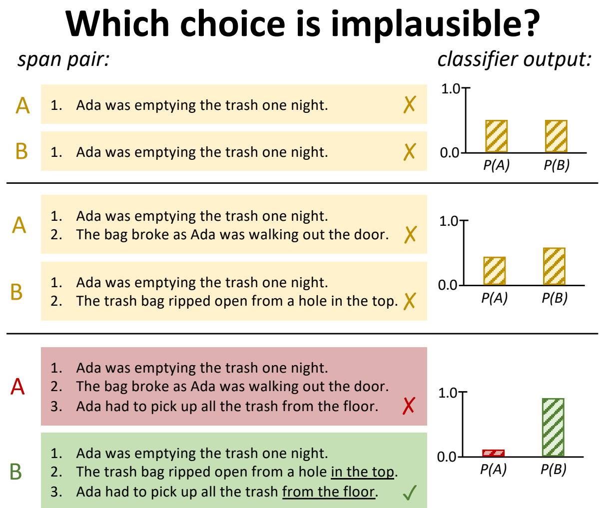


Figure 3.3: In ART, a multiple-choice text classification problem, we can label sub-spans with the least plausible choice, although in some cases, both choices are plausible. To address this, we consider the classifier's posterior probability for each choice; it is ideal if the classifier has low confidence in such instances.

3.2 Annotating Text Classification Benchmarks for Consistency Evaluation

To enable the type of evaluation described in Chapter 3.1 for our benchmarks, additional annotation is required. CE contains 50 unique dialog sources from the Switchboard corpus [82]. We randomly selected 10 testing sources to form the test set and left all remaining sources for training and validation, creating an 80%/20% split for training and validation (703 examples) versus testing (178 examples). We annotated the positive examples in the test set with the range of dialog turns entailing each hypothesis, allowing us to generate ground truth labels for the consistency measurement. Examples were labeled by two separate annotators and cross-verified with a near-perfect Cohen's κ [41] of 0.91, then a third annotator resolved any disagreements.

To transfer ART to our framework, we annotated 200 random examples from the public validation set (1532 examples) with the evidence for implausibility. There are 3 possible cases in implausible story choices: 1) the second sentence conflicts with the first and/or third sentence, 2) the second sentence is malformed or nonsense, presumably due to annotation error or adversarial filtering [267], and 3) the first and third sentence conflict with each other by default, and the second sentence does not resolve this. These cases are labeled by two annotators then merged with a fair Cohen's κ of 0.30 (perhaps lower due to subjectivity of commonsense-based problems), and a third annotator again resolving disagreements. 11

examples were discarded as two annotators agreed that both story choices were entirely plausible, presumably due to annotation error in ART.

3.3 Evaluating the Consistency of Fine-Tuned LMs

To evaluate the consistency of reasoning in state-of-the-art text classifiers, we fine-tune LMs on CE and ART, then use the annotations collected in Chapter 3.2 to evaluate their consistency, i.e., the proportion of end task predictions that are supported by valid evidence from the language context. Specifically, we choose three transformer-based language models from recent years: BERT [56], RoBERTa [142], and DeBERTa [93].[4] On CE, we additionally apply transfer learning from MultiNLI [256], a large-scale textual entailment dataset with some dialog-based problems. We measure both the *accuracy*, i.e., the proportion of instances where the end task prediction is correct, and *consistency* of models on respective evaluation sets. We refine the proposed metric for consistency into two forms: strict and lenient. Given a set of evaluation instances, *strict consistency* refers to the proportion of instances where the end task prediction is not only correct, but also coherent as described in Chapter 3.1. While strict consistency only rewards systems for examples where all sub-span predictions are correct, *lenient consistency* averages the sub-span accuracy over all examples for a less rigid reward. We include this alternate form of consistency to accommodate some disagreement with our annotations (which can be subjective based on measured inter-annotator agreement) without severe penalty.

Training details. Following common practice, systems are trained with cross-entropy loss toward the end task of text classification, maximizing accuracy on the validation set for model selection. On CE, we used 8-fold cross-validation split by dialog sources, then re-trained the model with the highest average validation accuracy on all folds. Pre-trained model parameters and implementations come from Hugging Face `transformers` [257],[5] each trained with the AdamW optimizer [146]. We performed a grid search over a wide range of learning rates and a maximum of 10 epochs. Training batch sizes are fixed based on available GPU memory. Selected hyperparameters can be found in Appendix A.

Discussion of results. Results on the test set of CE and public validation set of ART are listed in Table 3.1. All results show a statistically significant drop in performance from classification accuracy to strict consistency under a McNemar test [156] with $p <$ 1e-5,

[4] We use the "large" configuration of all models, which have 24 hidden layers and 16 attention heads.
[5] https://huggingface.co/transformers/

CE, *test*:

Model	**Accuracy** (%)	**Strict Consistency** (Δ; %)	**Lenient Consistency** (Δ; %)
majority	57.8	–	–
BERT	55.8	28.5 (-27.3)	35.7 (-20.1)
RoBERTa	70.9	39.0 (-31.9)	47.5 (-23.4)
↪ + MNLI	78.5	50.6 (-27.9)	58.2 (-20.3)
DeBERTa	67.4	37.2 (-30.2)	45.2 (-22.2)

ART, *validation*:

Model	**Accuracy** (%)	**Strict Consistency** (Δ; %)	ρ	**Lenient Consistency** (Δ; %)	ρ
majority	55.0 (50.1)	–	–	–	
BERT	66.7 (66.7)	42.3 (-24.4)	0.15	43.7 (-23.0)	0.85
RoBERTa	87.8 (84.2)	55.0 (-32.8)	0.1	59.3 (-28.5)	0.05
DeBERTa	88.4 (85.7)	59.8 (-28.6)	0.85	61.8 (-26.6)	0.95

Table 3.1: Accuracy, strict consistency, and lenient consistency on CE and ART for state-of-the-art text classifiers. Δ is the total performance drop from the classification accuracy to each consistency measure, and each ρ is the confidence threshold achieving the highest consistency. For ART, accuracy on the full validation set is given in parentheses.

some dropping below majority-class accuracy. While lenient consistency is slightly higher for both tasks, we still see large drops from accuracy. This demonstrates that while our text classifiers can achieve high classification accuracy on CE and ART, they do not deeply understand the tasks. Much of their performance is supported by incoherent intermediate predictions. Although pre-training on MultiNLI improves the end task accuracy on CE, it still suffers from comparably significant drops to the consistency measures. On ART, while all models see significant performance drops, DeBERTa, the state-of-the-art system for the task, achieves the best accuracy and consistency measures, as well as the highest chosen ρ values, which generally indicates more confident predictions. Even though it only marginally outperforms RoBERTa in accuracy, we see larger improvements in consistency measures and the chosen ρ, suggesting DeBERTa is more robust.

3.4 Summary of Findings

In this chapter, we proposed a simple and versatile method to evaluate the consistency of text classifiers, particularly targeting the problem where end task prediction depends on a discourse rather than a single sentence. By annotating a small amount of data in a benchmark, this method supports a quick assessment on whether machines' end task performance is supported by coherent intermediate evidence from the language context, in line with the evidence humans might use.

We used this method to evaluate the consistency of various foundational LMs fine-tuned on CE [269] and ART [22]. Our results showed that on these reasoning-intensive text classification tasks for NLU, LMs' high performance is attributed to inconsistent evidence from the language context, and is thus artificially achieved. Future work in commonsense reasoning and NLU driven by benchmarks should consider similar examinations beyond the end task accuracy, whether this be through our proposed consistency measures or other appropriate means. As we showed, such effort is quite straightforward and effective, and could facilitate progress toward more powerful classifiers that can support human-aligned reasoning.

CHAPTER 4

Verifiable Physical Commonsense Reasoning in Natural Language Understanding†

The previous chapter proposed a notion of *consistent* reasoning to identify and objectively evaluate the evidence from the language context used by foundational language models (LMs) to make predictions on natural language understanding (NLU) tasks. While consistency covers a surface-level component of coherent reasoning, it also remains unclear whether machines can perform *verifiable* reasoning based on background commonsense knowledge as humans do. As discussed in Chapter 1, this background knowledge can be difficult to collect, as it is rarely mentioned explicitly in language, it can vary widely between people and cultures, and its distribution has a long tail.

Physical commonsense reasoning (PCR), also referred to as naïve physics [53] or intuitive physics [121], has recently gained attention in the NLP community [74, 70, 46, 30, 71, 24]. From a young age, humans possess commonsense knowledge and reasoning skills about a wide variety of physical phenomena, such as movement, rigidity, and balance [26]. This problem is consequently thought to be especially challenging for machines because physical commonsense is considered obvious to most humans, and majorly suffers from reporting bias [70]. As most foundational LMs are trained only on written communications, it remains unclear whether they can learn this type of reasoning [23]. On the other hand, due to its concreteness, physical commonsense is relatively subjective compared to other types of commonsense knowledge, making it an ideal testbed for analyzing the coherence of commonsense reasoning.

In this chapter, we introduce Tiered Reasoning for Intuitive Physics (`TRIP`), a benchmark for coherent PCR. `TRIP` poses a high-level end task for story plausibility classification,

†Shane Storks, Qiaozi Gao, Yichi Zhang, and Joyce Chai. Tiered Reasoning for Intuitive Physics: Toward Verifiable Commonsense Language Understanding. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4902–4918, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

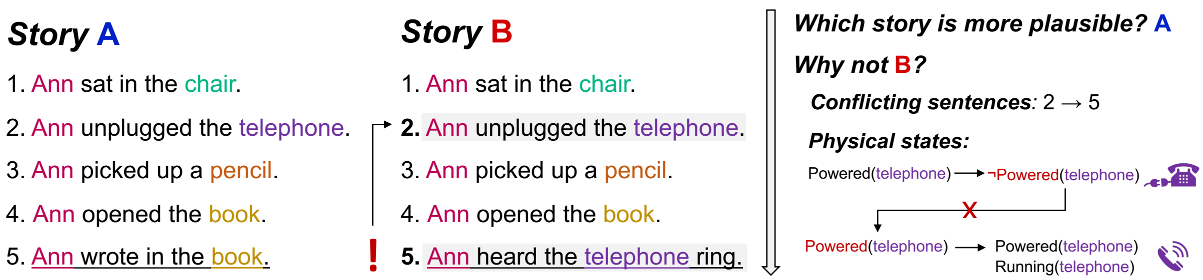


Figure 4.1: Story pair from TRIP, along with the tiers of annotation available to represent the reasoning process.

a common proxy task for commonsense reasoning problems [205, 168, 213, 24]. Notably, however, it includes dense annotations for each story capturing multiple tiers of reasoning beyond the end task. From these annotations, we propose a tiered evaluation, where given a pair of highly similar stories (differing only by one sentence which makes one of the stories implausible), systems must jointly identify (1) the plausible story, (2) a pair of conflicting sentences in the implausible story, and (3) the underlying physical states in those sentences causing the conflict. The goal of TRIP is to enable a systematic evaluation of machine coherence toward the end task prediction of plausibility. In particular, we evaluate whether a high-level plausibility prediction can be *verified* based on lower-level understanding, for example, physical state changes that would support the prediction.

4.1 Tiered Reasoning for Intuitive Physics (TRIP)

The Tiered Reasoning for Intuitive Physics (TRIP) is a benchmark for PCR that provides traces of reasoning for an end task of plausibility prediction. The dataset consists of human-authored stories, such as those in Figure 4.1, describing sequences of concrete physical actions. Given two stories composed of individually plausible sentences and only differing by one sentence (i.e., Sentence 5), the proposed task is to determine which story is more plausible. To understand stories like these and make such a prediction, one must have knowledge of verb causality[1] and precondition[2], and rules of intuitive physics.[3]

Plausible stories were crowd-sourced from Amazon Mechanical Turk.[4] To convert each story into several implausible stories, we hired separate workers to each write a new sentence to replace a sentence in the original story, such that the new story after replacement is

[1]For example, *cutting* an object causes it to be in pieces, and *melting* an object causes it to be in liquid form.

[2]For example, to *cut* an object, it must be in solid form, but to *stir* an object, it must be in liquid form.

[3]For example, the constraint that an object inside of a container moves when its container moves.

[4]https://www.mturk.com/

no longer realistic in the physical world. To ensure quality, these workers flagged stories which were incoherent or did not describe realistic actions. We eliminated those stories and performed a manual round of validation to remove any remaining bad stories and correct typos.

4.1.1 Controlled Data Curation

TRIP was carefully curated and restricted to support probing of reasoning abilities possessed by text classifiers. Compared to current benchmark trends, this dataset has the following unique properties.

4.1.1.1 Objectivity in Physical Commonsense

As commonsense knowledge differs between humans based on region, culture, and other factors [51], plausible reasoning tasks can become ambiguous and subjective, for example, in open-domain commonsense reasoning problems [273, 22]. To address this issue, we directed story authors to write sentences involving concrete actions, which can be unambiguously visualized in the physical world, while avoiding mental actions such as to *think* or *like*. We limit stories to typical household happenings by directing annotators to write stories in one of six possible "rooms" seen in everyday life.

To further reduce subjectivity and block other confounding factors that may result from complex use of language, we encourage crowd workers to write sentences in a simple declarative form, typically starting with the agent of the story, followed by a verb, a direct object, and an optional indirect object. The simplicity of language use would additionally allow us to focus less on linguistic processing and semantic phenomena, and more on investigating machines' reasoning ability.

4.1.1.2 Plausibility in Longer Context

Many benchmarks for plausible reasoning only (or most frequently) provide one sentence of context, with similarly short choices to complete the context [205, 267, 23]. In TRIP, we imposed several restrictions to require reasoning over multiple sentences with associated physical state changes. First, we required annotators to write stories at least five sentences long. Further, when collecting new sentences to convert plausible stories into implausible stories, we required that the new sentence should be plausible in isolation, and only become implausible when considering the world state implied by other sentences in the story. This constraint encourages stories to be rich in interesting action dynamics rather than nonsense sentences such as "Mary fried eggs on the printer" or "Tom ate the spoon," which may be

Measure	Train	Val.	Test	All
# plausible stories	370	152	153	675
# implausible stories	799	322	351	1472
avg. # sentences	5.1	5.0	5.1	5.1
avg. sentence length	8.3	8.0	8.5	8.3
# story authors	97	57	62	134
avg. # stories/author	3.8	2.7	2.5	5.0
avg. # conflicting sentence pairs	1.2	1.2	1.2	1.2
# physical state labels	18.8k	8.74k	9.09k	36.6k

Table 4.1: Statistics of the `TRIP` dataset. Implausible stories in each partition are generated from and paired with the plausible stories in the same partition.

easier to recognize through distributional biases. As this new sentence can conflict with any other sentence(s) in the story, solving the task requires reasoning over the entire context.

4.1.1.3 Multi-Tiered Annotation

To enable a systematic investigation of a system's reasoning process, we manually provided three levels of annotation. As shown in Figure 4.1, the first level is the *end task label* to indicate which of the two story choices are more plausible. By design, most implausible story choices have exactly one pair of *conflicting sentences*, e.g., Sentences 2 and 5 in the example. The second level of annotation identifies these sentences in each story. The third level justifies the implausibility with labels for the underlying *physical states*, giving a detailed account of the physical changes associated with each sentence. In our example, unplugging *the phone* in Sentence 2 causes it to lose power, while Sentence 5 requires that the phone is powered in order to *ring*.

Table 4.1 lists various statistics to summarize the resulting dataset. While this dataset is small by today's standards, our goal is depth, not breadth. Rather than training models on a surplus of data to simply achieve high accuracy on the end task, we aim to use our deep, multi-tiered annotations to probe the capability of NLP models to perform coherent reasoning toward the end task. Next, we provide additional details for annotating conflicting sentences and physical states.

Conflicting sentence annotation. For each implausible story, an annotator identified one or more pairs of conflicting sentences. On a random set of 100 implausible stories from the training data, a second annotator labeled these pairs of sentences, reaching a near-perfect Cohen's κ [41] of 0.929, supporting the objectivity of these labels.

Label	Human Location	Object Location	Other Attributes
0	irrelevant	irrelevant	irrelevant
1	disappeared	disappeared	*false* → *false*
2	moved	picked up	*true* → *true*
3	–	put down	*true* → *false*
4	–	put on	*false* → *true*
5	–	removed	___ → *no*
6	–	put in container	___ → *true*
7	–	taken out of container	*false* → ___
8	–	moved	*true* → ___

Table 4.2: Label space and meanings for human location, object location, and other attributes. Each label represents a specific physical change (or lack of change).

Physical state annotation. In order to generate the rich physical state annotations in `TRIP`, we defined a space of 20 physical attributes (5 for humans, 15 for objects)[5] which capture most conflicts found in the stories. This was collected in part from related attribute spaces proposed in [74] and [30], and chosen based on a random set of implausible training stories, specifically the nature of their conflicts and physical changes objects underwent during the stories. For each entity in each sentence in the dataset, we annotate the implied values of these attributes before (precondition) and after (effect) the events of the sentence take place. This step of the annotation was a substantial effort.

More specifically, physical states were annotated with values from the attribute-specific label spaces in Table 4.2, each of which represented directions of physical state change (e.g., attribute became true or attribute became false). In the training data, we manually labeled each entity in the sentence with these attributes and values. For predicting precondition and effect in non-location attributes as done in this work, it is straightforward to collapse this space into *true*, *false*, or *unknown* for each. For human location labels, we use the full label space for predicting both precondition and effects for simplicity. Meanwhile, for object location labels, we simplify the problem by mapping them to smaller precondition and effect label spaces. While this does not significantly affect verifiability, this should be expanded in

[5] For humans, we track *location*, *hygiene*, and whether a human is *conscious*, *dressed*, or *wet*. For objects, we consider *location* and whether or not an object *exists*, is *clean*, connected to *power*, *functional*, *in pieces*, *wet*, *open*, *hot*, *solid*, *occupied* (i.e., containing another object), *running* (i.e., turned on), *movable*, *mixed*, or *edible*.

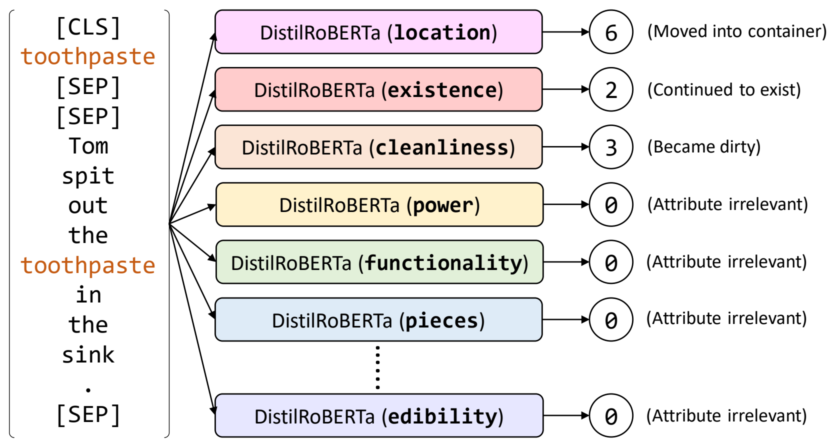


Figure 4.2: Structure of the physical state classifier used for semi-automatic annotation, consisting of 20 parallel instances of DistilRoBERTa [211]. Each instance outputs an integer representing a particular kind of change (or lack of change) in the corresponding attribute.

a full solution for better interpretability. For more detailed explanations, future work may consider tracking spans of text describing entity locations along the lines of [7].

To expand our manual physical state annotations to the validation and testing data, we used a semi-automatic approach to label physical states. First, we used the existing annotations to fine-tune classifiers to predict values for each attribute given a sentence-noun pair. To achieve this, each story was broken down into all possible sentence-entity pairs, using spaCy[6] to identify noun phrases.[7] As shown in Figure 4.2, these sentence-noun pairs were passed into the physical state classifier,[8] implemented with 20 parallel instances of DistilRoBERTa$_{\text{BASE}}$ (82 million parameters) [142, 211] with classification heads, one for each physical attribute. Using this collection of classifiers, we generated candidate physical state annotations for the remaining data, then manually revised them. For a representative subset of 157 sentences from 31 stories in the training data, a second annotator repeated this process, finding a substantial Cohen's κ of 0.7917.

4.1.2 Proposed Tasks

From the `TRIP` dataset, we propose several tiered tasks as shown in Figure 4.1. Together, these tasks form a human-interpretable reasoning process supported by a chain of evidence.

[6] https://spacy.io/

[7] While relevant entities in each sentence are available to models at inference time for convenient evaluation, these can be fairly reliably re-extracted using spaCy.

[8] Followed [89] for formatting the input in order to generate entity-centric embeddings.

Physical state classification. From our physical state annotations, we propose two tasks for each sentence-entity pair in each story choice: precondition and effect state classification. For example, consider the entity *potato* in the sentence "John cut the cooked potato in half." First, we should predict that the potato was solid in order to be *cut*, i.e., the precondition label for the `solidity` attribute is *true*. Second, we should predict that the potato was in pieces as a result of being *cut*, i.e., the effect label for the `in pieces` attribute is *true*.

Conflict detection. Next, we define the task of conflict detection as identifying a pair of sentences in the form $S_i \rightarrow S_j$. S_j is a *breakpoint*, i.e., the point where the story first becomes implausible given the context so far, while S_i serves as *evidence* that explains the breakpoint, usually causing a conflicting world state. For example, in Figure 4.1, Sentence 5 is a breakpoint, while Sentence 2 is the evidence that explains why the story becomes implausible after Sentence 5. Note that it is possible that a story may have multiple pairs of conflicting sentences beyond the breakpoint and evidence pair. However, across the dataset, the average number of conflicting sentence pairs is only 1.2, so one conflicting sentence pair is a sufficient and simpler explanation for the conflict (albeit not exhaustive).

Story classification. Lastly, the end task is to determine which of two stories is the plausible one. This should be determined based on any conflicts detected within the two stories.

4.1.3 Benchmark Goals

It is important to note that while one can treat these tasks separately, the goal of this benchmark is to solve them jointly to form a coherent reasoning chain: physical state classification explains conflict detection, which further explains story classification. Unlike most existing benchmarks in this area, which assess language understanding ability through some high-level end tasks, the goal of our benchmark is to enable development of systems for interpretable, consistent, and verifiable reasoning toward language understanding.

It is also worth noting that although data bias is an issue for high-level benchmark tasks where systems are not required to justify their predictions, we are not directly targeting this issue. Recent work has attempted to remove biases from benchmark data and thus prevent exploitation of them in performing high-level tasks [267, 176]. In contrast, our framing of language understanding as being built from the ground up (i.e., from low-level to high-level tasks) provides systems with the proper supporting evidence toward high-level tasks, and thus can potentially mitigate some of the problems around data bias.

4.2 Defining Verifiability in `TRIP`

To enable a better understanding of machines' ability in coherent reasoning toward end task performance, we apply the several multi-tiered evaluation metrics. This includes the introduction of a new metric for *verifiability*, which captures commonsense background knowledge underlying the understanding of surface-level language context.

Accuracy. The traditional metric of end task accuracy, i.e., the proportion of testing examples where plausible stories are correctly identified.

Consistency. The proportion of testing examples where not only the plausible story is correctly identified, but also the conflicting sentence pair for the implausible story is correctly identified. This is to demonstrate the consistency with identified conflicts when reasoning about plausibility.

Verifiability. The proportion of testing examples where not only the plausible story and the conflicting sentence pair for the implausible story are correctly identified, but also underlying physical states (i.e., preconditions and effects) that contribute to the conflict are correctly identified.[9] This is to demonstrate that the detected conflict can be verified by a correct understanding of the underlying implausible change of physical states.

It is worth noting that this notion of verifiability, although different, is motivated by the notion of *verification* in software engineering [189]. This term refers to determining whether a given software solution satisfies its architectural and design requirements, and is built from the correct sub-components. Along this line, our notion of verifiability can be seen as a method to evaluate whether a language understanding system's reasoning process is built up from the correct components.

Each successive metric dives deeper into the coherence of reasoning that supports the end task prediction. Consequently, if accuracy is a, consistency is b, and verifiability is c, then $a \geq b \geq c$. A system that reliably produces a coherent chain of reasoning is demonstrated by $a \approx b \approx c$.

In Chapter 5, we will evaluate the accuracy, consistency, and verifiability of various foundational LMs fine-tuned on the tiered sub-tasks of `TRIP`.

[9] At least one nontrivial, i.e., non-default, positive-class physical state label must be predicted in the preconditions of the breakpoint sentence and effects of the evidence sentence, and all such predictions must be correct.

CHAPTER 5

Coherence of Physical Commonsense Reasoning in Fine-Tuned Language Models†

Chapter 3 showed that fine-tuning foundational language models (LMs) on natural language understanding (NLU) classification tasks elicits artificially high accuracy through incoherent reasoning supported by inconsistent evidence. An alternative method to tackle NLU tasks requiring complex reasoning is to fine-tune LMs as multi-tiered reasoners.

To explore this possibility, in this chapter, we design a baseline architecture and training paradigm for the Tiered Reasoning for Intuitive Physics (`TRIP`) benchmark proposed in Chapter 4. We present experiments in fine-tuning this architecture powered by foundational LMs for coherent reasoning, comparing various configurations of loss functions for the multi-tiered sub-tasks of `TRIP`. Through this, we seek to unearth insights into how overly high-level learning objectives may interact with lower-level reasoning-based learning objectives. We conduct a detailed analysis to reveal several insights on the reasoning behaviors of fine-tuned LMs, creating opportunities for future work.

5.0.1 A Tiered Baseline for `TRIP`

Figure 5.1 displays a high-level view of our proposed baseline system to solve `TRIP`. It uses a foundational LM backbone to individually embed each sentence-entity pair in each story, classifies physical precondition and effect states, then identifies conflicting sentences from these. Given a pair of stories, it aggregates conflict predictions for each story to decide which is more plausible.

---

†Shane Storks, Qiaozi Gao, Yichi Zhang, and Joyce Chai. Tiered Reasoning for Intuitive Physics: Toward Verifiable Commonsense Language Understanding. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4902–4918, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

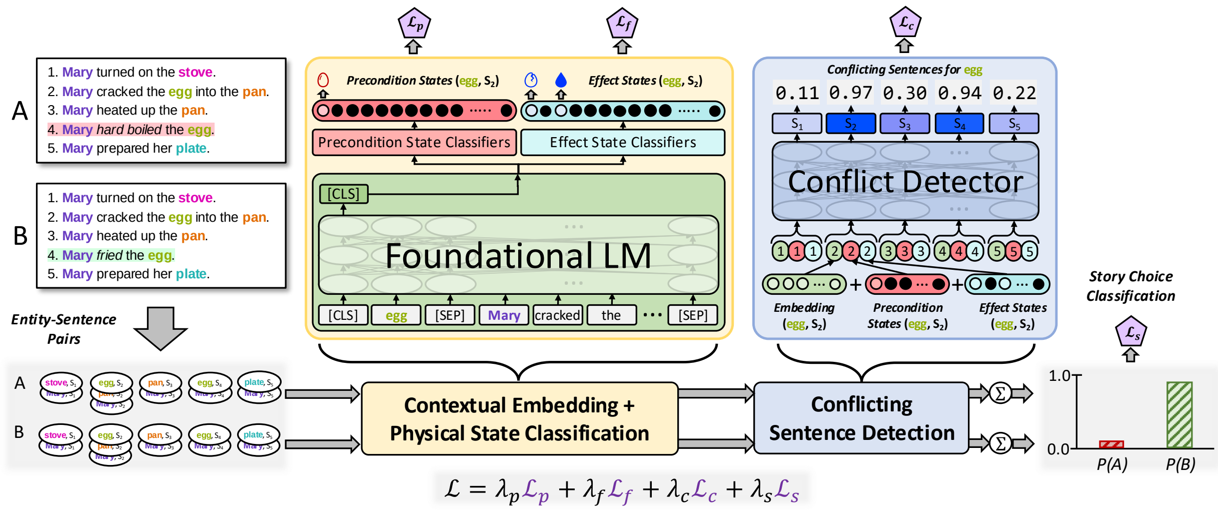


Figure 5.1: Proposed tiered reasoning system with loss functions $\mathcal{L}_p$ for precondition state classification, $\mathcal{L}_f$ for effect state classification, $\mathcal{L}_c$ for conflicting sentence detection, and $\mathcal{L}_s$ for story choice classification. The model is trained end-to-end by optimizing the joint loss $\mathcal{L}$, a weighted sum of these loss functions.

5.0.1.1 Module Implementations

Each module of this baseline is implemented through a neural network architecture. Here, we describe some details of the implementations.

Contextual Embedding. The Contextual Embedding module is implemented with a transformer-based foundational language model. Generally, this module takes as input a sentence and the name of an entity from a story, following an entity-first input formulation [89], and outputs a dense, contextualized numerical representation. While there are some model-specific variations in special tokens, given an entity e and a sentence $t_1, t_2, \cdots, t_n$, we structure the input sequence as " [CLS] e [SEP] t_1 t_2 $\cdots$ t_n [SEP]," where [CLS] is a special token meant for input to classification layers, and [SEP] is a special separator token for multi-text inputs.

Precondition and Effect Classifiers. The Precondition and Effect Classifiers are implemented as typical feedforward classification heads for contextual embeddings, with one precondition classifier and one effect classifier for each of the 20 physical attribute tracked in the dataset. Specifically, each classifier is made up of two feedforward layers, each preceded by a dropout layer (using model specific defaults for dropout probability), with tanh activation in between them. The first layer performs a linear transformation on an input contextual embedding, while the second layer projects the hidden state to the size of the label space for the corresponding attribute. Argmax is applied to the output for classification. Altogether, the predictions from these classifiers label physical states of each entity in each sentence of the story.

Conflict Detector. For each entity and its predicted physical states over all sentences in a story, the Conflict Detector predicts whether there is some conflict in the entity's physical states, specifically flagging a pair of conflicting sentences through multi-label classification. We use another transformer for this module (6 additional layers with 8 attention heads), but model the high-level sequence of sentences in a story rather than the low-level sequence of tokens in a sentence. For each sentence-entity pair, we input the contextual embedding, as well as the classification logits behind all physical state predictions. We project both representations through linear layers to the same size, then concatenate them to form an entity dynamics representation.[1] This representation for each sentence is input to the transformer, and the resulting hidden states are concatenated. Lastly, we use a feedforward layer followed by sigmoid activation to transform the hidden state to a belief probability of each sentence conflicting with another sentence in the story.

Story choice prediction. Given any detected conflicts, we lastly select which of the two given stories is plausible. As each Conflict Detector output represents a belief that the physical states of an entity in a particular sentence conflict with that of another sentence, we can simply sum the negative outputs for each story and apply softmax to determine which story is least likely to have a conflict.

5.0.1.2 Model Training

We train the architecture's parameters through gradient descent on the overall loss $\mathcal{L}$:

$$\mathcal{L} = \lambda_p \mathcal{L}_p + \lambda_f \mathcal{L}_f + \lambda_c \mathcal{L}_c + \lambda_s \mathcal{L}_s$$

$\mathcal{L}$ sums individual cross-entropy loss functions $\mathcal{L}_p$ for precondition classification, $\mathcal{L}_f$ for effect classification, $\mathcal{L}_c$ for conflict detection, and $\mathcal{L}_s$ for story choice classification, each balanced by respective weights λ_p, λ_f, λ_c, λ_s summing to 1. In preliminary experiments, we found the best balance between state classification and the other tasks with the following assignment of weights: $\lambda_p = \lambda_f = \frac{0.4}{|A|}$, $\lambda_c = \lambda_s = 0.1$, where $|A|$ is the number of attributes tracked, i.e., 20. When omitting different loss functions, we rebalance the weights by ensuring $\lambda_c + \lambda_s = 0.2$, or $\lambda_c = \lambda_s$ where state classification losses are omitted.

[1] Appendix A.2.1 presents an ablation study for each component of the entity dynamics representation input to the Conflict Detector.

5.0.2 TRIP Experiments

Using TRIP, we evaluate several variations of the proposed reasoning system powered by selected pre-trained language models: BERT [56], RoBERTa [142], and DeBERTa [93].[2] These models offer a range in design choices such as model complexity and size of pre-training data. We begin with an evaluation from the perspective of the end task, then take a detailed look at the lower-level tasks.

5.0.2.1 Tiered Baseline Results

Recall that we consider four loss functions for training the tiered system: $\mathcal{L}_p$ for precondition classification, $\mathcal{L}_f$ for effect classification, $\mathcal{L}_c$ for conflicting sentence detection, and $\mathcal{L}_s$ for story choice classification. To investigate how each loss affects model performance, we train instances using several combinations of them. The results of this study on the validation set are listed in Table 5.1.

The role of end task supervision. In the first section of Table 5.1, we train the system jointly on all four loss functions. Here, we see low verifiability and consistency for all three LMs, while the end task accuracy is relatively high, reaching 78.3% when using BERT. When we omit the story classification loss in the second section, however, we see sharp gains in verifiability and consistency for all models, with RoBERTa jumping from 0.9% verifiability and 6.8% consistency to 10.6% and 22.4%, respectively. This comes at a slight cost of end task accuracy for BERT and RoBERTa.

This suggests that while fine-tuning systems based on a high-level classification loss targeting the end task can improve the end task accuracy, this drastically reduces the interpretability of the underlying reasoning process. One potential explanation for this is that this loss drives the system to exploit spurious statistical cues in order to further increase the end task accuracy. This gives us motivation to move away from using over-simplified end tasks to train and evaluate language understanding. In fact, if we fine-tune RoBERTa's contextual embedding directly on the end task of TRIP without intermediate classification layers, we can achieve up to 97% accuracy, but have no insight toward verifiability or consistency of the system. This raises questions about the validity of such a result.

Natural emergence of intermediate predictions. In the third and fourth sections of Table 5.1, we respectively omit conflict detection loss and state classification losses to explore whether conflicting sentences or physical states would emerge naturally in the reasoning

[2] We use the "large" configurations of BERT (355M parameters) and RoBERTa (355M parameters), and the "base" configuration of DeBERTa (140M parameters).

All Losses

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
random	47.8	11.3	0.0
BERT	**78.3**	2.8	0.0
RoBERTa	75.2	6.8	0.9
DeBERTa	74.8	2.2	0.0

Omit Story Choice Loss

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
BERT	73.9	**28.0**	9.0
RoBERTa	73.6	22.4	**10.6**
DeBERTa	75.8	24.8	7.5

Omit Conflict Detection Loss

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
BERT	50.9	0.0	0.0
RoBERTa	49.7	0.0	0.0
DeBERTa	52.2	0.0	0.0

Omit State Classification Losses

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
BERT	75.2	17.4	0.0
RoBERTa	71.4	2.5	0.0
DeBERTa	72.4	9.6	0.0

Table 5.1: End and tiered task metrics for tiered classifiers on the validation set of `TRIP` trained on varied combinations of loss functions. Random baseline (averaged over 10 runs) makes tiered predictions at random.

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
random	49.5	10.7	0.0
BERT	70.9	21.9	8.3
RoBERTa	**72.9**	19.1	**9.1**
DeBERTa	**72.9**	**22.2**	6.6

Table 5.2: Metrics for the best tiered systems on the test set of `TRIP`. Compared to random baseline.

process. When omitting conflict detection loss, all metrics degrade to near or below random performance. Clearly, conflict detection is not implicitly learned from the downstream story classification loss, and since the story choice classification directly depends on the conflict detection output, the end task accuracy drops as well.

Meanwhile, when omitting physical state classification loss, verifiability unsurprisingly drops to zero, but high accuracy on the end task can still be achieved by all models (up to 75.2%). Notably, this suggests that reasonable supporting evidence is not required in order to achieve high accuracy on the end task. This casts further doubt that existing state-of-the-art results on other commonsense language understanding benchmarks possess any kind of coherent reasoning beyond end classification tasks which over-simplify the problem.

In Table 5.2, we present the testing results for the best loss function configuration of the system, i.e., omitting story choice classification loss. Compared to the validation set results in Table 5.1, we see slight drops in consistency and verifiability, further demonstrating the difficulty of this problem.

5.0.3 Results Analysis

Given the poor performance along our proposed metrics, we next consider the connections between the tiered tasks, and what goes wrong in unverifiable end task instances. We focus our analysis on the systems achieving the highest verifiability on the validation set in Chapter 5.0.2.1.

Failure mode distribution. Figure 5.2 provides a detailed breakdown of the combinations of failure modes on the validation set. Of the 73.6% of validation instances that are classified correctly on the end task, almost half of these (31.4% overall) are entirely unverified, with incorrect physical states and conflicts predicted by the system. Similarly, of the 26.4% of instances with *incorrect* end task predictions, about half (13% overall) have incorrect physical state and conflict predictions. Meanwhile, a combined 31.1% of instances

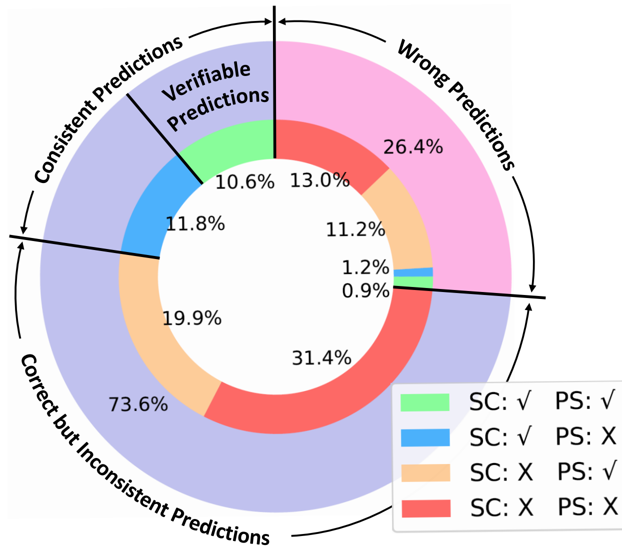


Figure 5.2: Distribution of RoBERTa successes and failures on `TRIP`. SC (sentence conflict) and PS (physical state) denote whether the predicted conflicting sentences or physical states are correct (✓) or not (×).

Model	Prec. F1 (%)	Eff. F1 (%)	Confl. F1 (%)
BERT	**54.9**	57.2	66.3
RoBERTa	51.2	51.2	**69.6**
DeBERTa	52.8	**57.3**	63.6

Table 5.3: Macro-F1 scores of best tiered systems on aggregate precondition, effect, and conflicting sentence classification. Scores averaged over all attributes for physical state classification.

correctly predict physical states in the conflicting sentences of the implausible story, but fail to detect a conflict in those sentences (19.9% are correct at the end task, while 11.2% are not). These instances, represented by orange wedges in the graph, are a significant disconnect in the reasoning process.

Low-level task performance. To further address this disconnect, we examined system performance from the perspective of physical state classification and conflict detection. First, Table 5.3 lists the validation metrics for our best baselines on the tasks of precondition and effect classification (by sentence-entity pair), as well as conflicting sentence detection (by end task instance). Across the board, we find reasonable performance on all tasks.[3]

The best performing baseline from Table 5.1 is trained using loss functions for both phys-

[3]Appendix A.2.2 presents additional results for physical state classification performance by attribute.

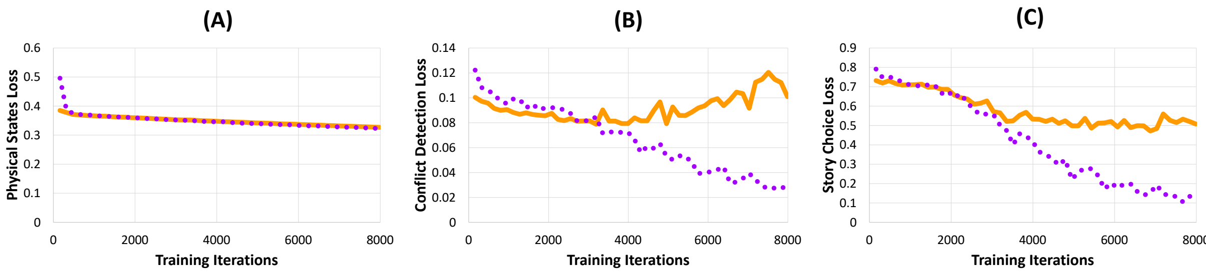


Figure 5.3: Training (purple, dotted) and validation (orange, solid) losses for best tiered RoBERTa system trained on TRIP for 10 epochs. Uses the best configuration of the loss functions (as found in Chapter 5.0.2.1) for (A) physical state classification, (B) conflict detection, and (C) story choice classification. Validation loss recorded 4 times per epoch, with training loss averaged over the trained batches since the previous recording.

ical state classification and conflict detection. Given this configuration, we further examined how each task is learned. Figure 5.3 shows training curves for the loss functions of physical state classification (averaged for precondition and effect), conflicting sentence detection, and story choice classification. Notably, though story choice classification is not used as a training objective, this end task is learned fairly well (albeit overfitting), with training and validation losses generally decreasing through training. This shows that learning to reason from the lower-level tasks is successful to some degree. However, the lower-level tasks appear challenging to learn. For physical state classification, losses decrease steadily, but slowly. For conflict detection, the losses also decrease slowly, and the model begins overfitting the training data, perhaps indicating a need for more training data at this challenging step. Future work may consider automatic data augmentation techniques to resolve this.

Connecting states to conflicts. To dig deeper into the connection between physical states and plausibility conflicts, we next examined correct physical state predictions by attribute in Figure 5.4. In the graph, we indicate the percentage of predictions supporting a successfully detected conflict, which may be interpreted as a *utility* measure of each attribute toward conflict detection. We find that some attributes, like whether an electrical object is `running`, rarely contribute to successful conflict detections (only 26.1%) despite having reasonably high F1 score (0.69). Other attributes, like `wet`, are more likely to appear in successful conflict detections when predicted correctly, even though their overall classification performance is lower. This provides strong insights for targeted improvement, for example, to better take advantage of lower-level predictions toward high-level tasks.

Sample system outputs. Figure 5.5 presents sample outputs from the tiered RoBERTa system. In Example (a), the prediction is entirely verifiable. The system correctly chooses

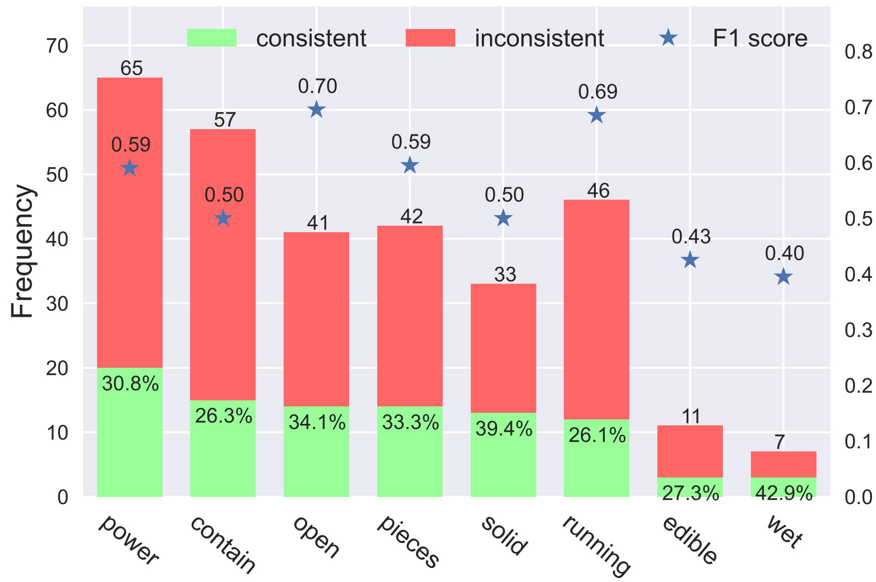
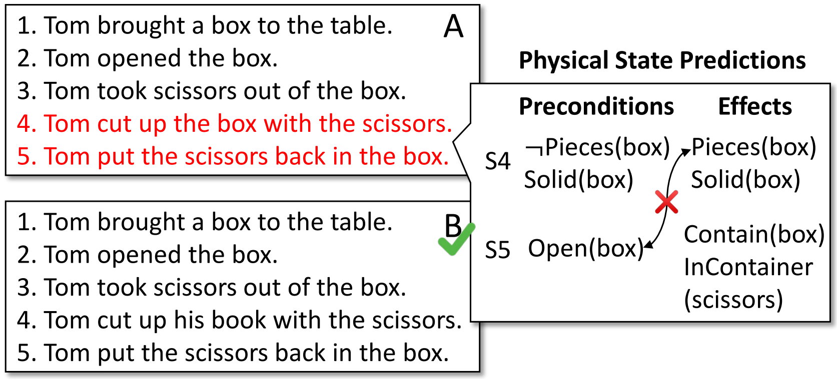


Figure 5.4: Contribution of correct RoBERTa-predicted physical states to consistency evaluation for selected attributes. The macro-F1 score of precondition and effect predictions is shown by blue stars. Among all correctly predicted states (for both effects and preconditions), the bar regions indicate whether these states appear in successfully detected conflicting sentences.

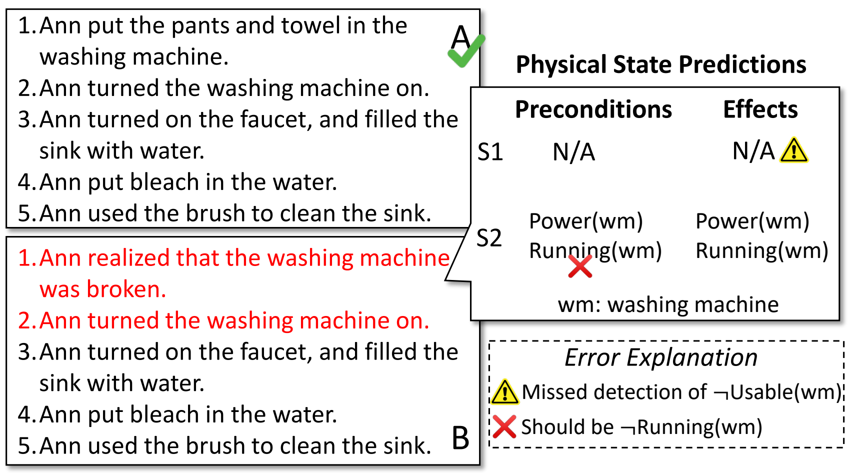
the plausible story, identifies Sentences 4 and 5 as the conflicting sentences in the implausible story, and even predicts that the *box* is `in pieces` after Sentence 4, and thus cannot become `open` in Sentence 5. In Example (b), the prediction is consistent but unverifiable, as the system identifies a conflict between Sentences 1 and 2, but cannot support the conflict with correct underlying physical states in either sentence. Although some relevant attributes are identified for the breakpoint sentence, e.g., `power` and `running`, they are not quite right. Meanwhile, no states are predicted for the evidence sentence.

5.0.4 Summary of Findings

In this section, we applied LMs to `TRIP`, our proposed tiered benchmark dataset for physical commonsense reasoning posing a new challenge of jointly solving low-level to high-level tasks to form a coherent reasoning process. We experimented with several variations of multi-tiered LM fine-tuning paradigms to solve the tasks. Our results show that in many cases, *supervising foundational LMs based on high-level classification tasks in order to learn commonsense NLU leads to inconsistent and unverifiable reasoning*, and inability to capture intermediate evidence toward the end task. Instead, we should train systems to jointly incorporate multiple types of lower-level evidence to solve reasoning tasks coherently. Our detailed analysis of these results offers strong intuition for future progress toward this goal.



(a) A verifiable prediction.



(b) A consistent but unverifiable prediction.

Figure 5.5: Sample outputs from the baseline system. The detected conflicting sentences are in red, and physical state predictions are shown on the right.

As such, TRIP and our baselines provide an important first step toward verifiable, human-aligned commonsense language understanding, and a direction for development of artificially intelligent systems in this area.

TRIP uses physical commonsense reasoning as an example, but we expect that a similar approach can apply to many aspects of NLU. In the context of spurious behaviors of LMs, our results reveal a new challenge to build machines that can reason logically and coherently, similar to what we expect from human reasoning. As these machines ultimately will work with humans, such alignment in reasoning is critical, as it will improve accountability and transparency in human-machine enterprise.

CHAPTER 6

Cognitively Motivated Strategies for Coherent Physical Commonsense Reasoning[†]

In the previous chapter, we showed that applying traditional fine-tuning approaches to adapt foundational language models (LMs) to physical commonsense reasoning (PCR) leads to incoherent behaviors, particularly inconsistent and unverifiable decisions. This possibility of incoherence in reasoning with LMs makes them difficult to rely on in practice, and creates a demand for more reliable, logical, and transparent reasoning strategies compatible with differentiable architectures like pre-trained LMs [152, 124].

Meanwhile, in theories of cognitive psychology, drawing conclusions in reasoning problems and coherently rationalizing them have long been thought to come from dual processes of human cognition [252, 65, 66, 68, 239, 69]: fast, associative *heuristic* thinking based on experience, and slower, deliberative *analytic* thinking, which requires more working memory. Specifically, prior work theorizes that heuristic processes enable us to extract the most relevant information from the context and provide quick intuition for decisions, which can then inform analytic processes that operate on this information to perform inference and rationalize when needed [66, 67, 114].

In this chapter, inspired by the synergy between these dual processes in humans, we propose analogous heuristic-analytic reasoning (HAR) strategies for LMs, which bootstrap lower-level (analytic) rationalization from higher-level (heuristic) decision-making. Targeting coherent PCR, e.g., in the `TRIP` benchmark introduced in Chapter 4, we implement HAR for LM fine-tuning (Chapter 6.2) and in-context learning (Chapter 6.3).[1] In the in-context

[†]Zheyuan Zhang, Shane Storks, Fengyuan Hu, Sungryull Sohn, Moontae Lee, Honglak Lee, and Joyce Chai. From Heuristic to Analytic: Cognitively Motivated Strategies for Coherent Physical Commonsense Reasoning. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7354–7379, Singapore, 2023. Association for Computational Linguistics.

[1]Additional performance analysis of both approaches on types of `TRIP` plausibility conflicts provided in Appendix B.5.

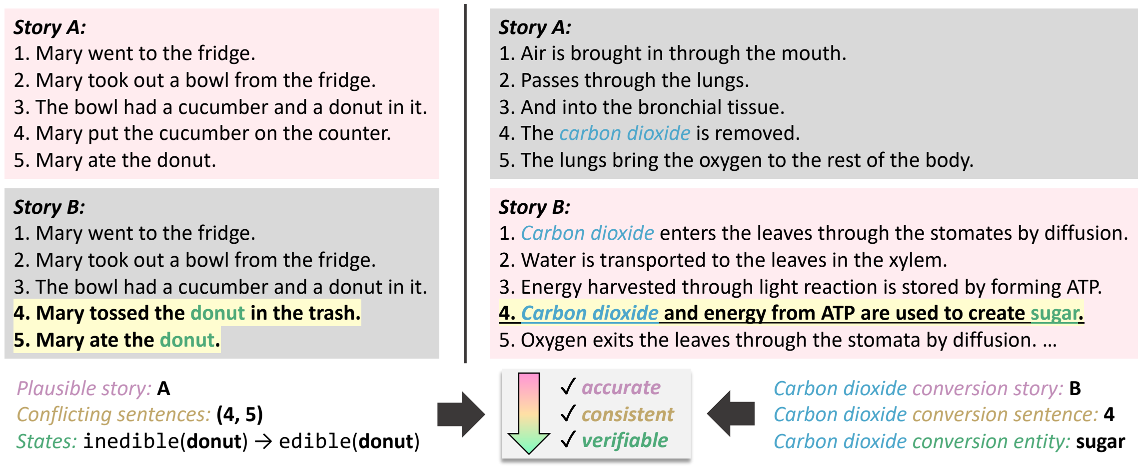


Figure 6.1: TRIP (left; 228) compared to reframed Tiered-ProPara (right) task for coherent physical commonsense reasoning. Each task requires multiple levels of reasoning from surface-level story and sentence selection and commonsense physical state prediction. While *accuracy* only evaluates the ability to perform the highest-level task, *consistency* and *verifiability* are used to evaluate lower-level steps and judge the coherence of reasoning.

learning setting, given the role of these dual processes in humans to filter out irrelevant information, we explore how various reasoning strategies influence LMs' self-attention weights on the language context.

6.1 Reframing ProPara for Coherent Physical Commonsense Reasoning

To supplement the results on TRIP, we recast ProPara, a previously existing dataset of texts about scientific processes annotated with the dynamic existence and location of entities throughout the processes [46]. While ProPara originally focused on the low-level task of predicting the states of entities before and after each sentence of passages, we propose Tiered-ProPara, a novel reframing of the task which requires multiple levels of reasoning. As shown in Figure 6.1, in this version of the task, a system is presented with two passages from ProPara with shared entities, and asked in which story is a particular type of entity, e.g., *carbon dioxide*, converted into another entity. In addition to this, the system must identify the sentence in that story in which the conversion occurs, and what the entity is converted into. Similarly to TRIP, we can use these two lower-level tasks to evaluate consistency and verifiability of system predictions on the end task of choosing a story. More details on how we generate this data are given in Appendix B.1.

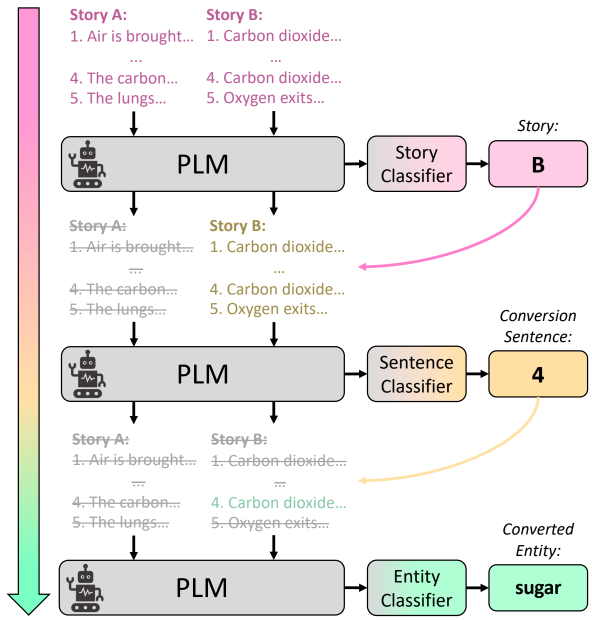


Figure 6.2: Heuristic-analytic reasoning for fine-tuning LMs, where the language context is iteratively refined using classification predictions during training and inference. In Tiered-ProPara, after the LM is used to classify which story contains a conversion, the other story is deleted from the model inputs. After classifying which sentence describes the conversion, other sentences are deleted. Lastly, the resulting entity after the conversion is identified.

6.2 Heuristic-Analytic Reasoning for LM Fine-Tuning

Fine-tuning LMs is one popular approach for adapting them to downstream tasks, applied in recent work toward coherent commonsense reasoning [149, 204], and suitable for applications with compute or privacy restrictions. As shown in Figure 6.2, we can build explicit heuristic-analytic reasoning structure into LM fine-tuning for our target tasks by deleting parts of the language context that are no longer relevant as the model makes predictions for each step of reasoning, both during training and inference. While our approach provides just one example, this reasoning trick can apply to LM fine-tuning for any multi-step reasoning problem where the most relevant language context to support reasoning changes with each step.

6.2.1 Fine-Tuning Experiments

Next, we introduce our experiments with HAR in fine-tuning LMs.

Implementation. To implement HAR in fine-tuning, we use the recent state-of-the-art Coalescing Global and Local Information (CGLI) model for `TRIP` and ProPara [149] as a backbone for HAR in fine-tuning. We apply two tricks to better focus the model on relevant

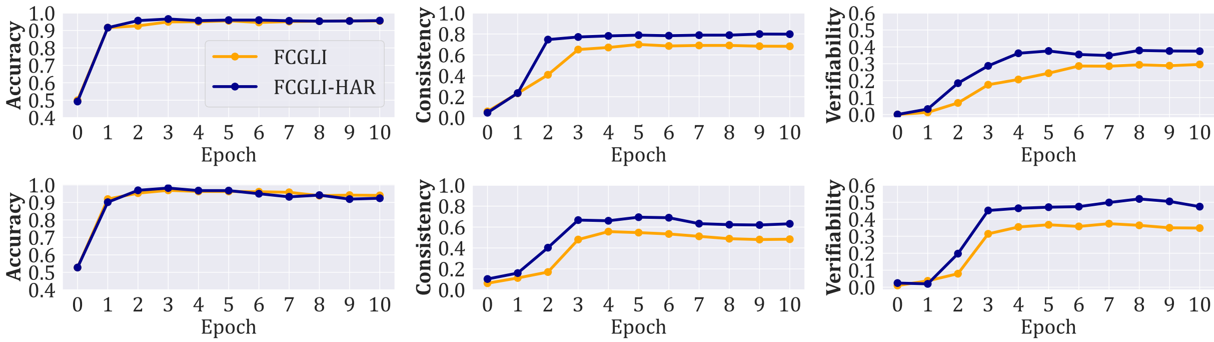


Figure 6.3: Validation metrics for unstructured FCGLI baseline and FCGLI with heuristic-analytic reasoning (FCGLI-HAR) through epochs of training on `TRIP` (top) and Tiered-ProPara (bottom).

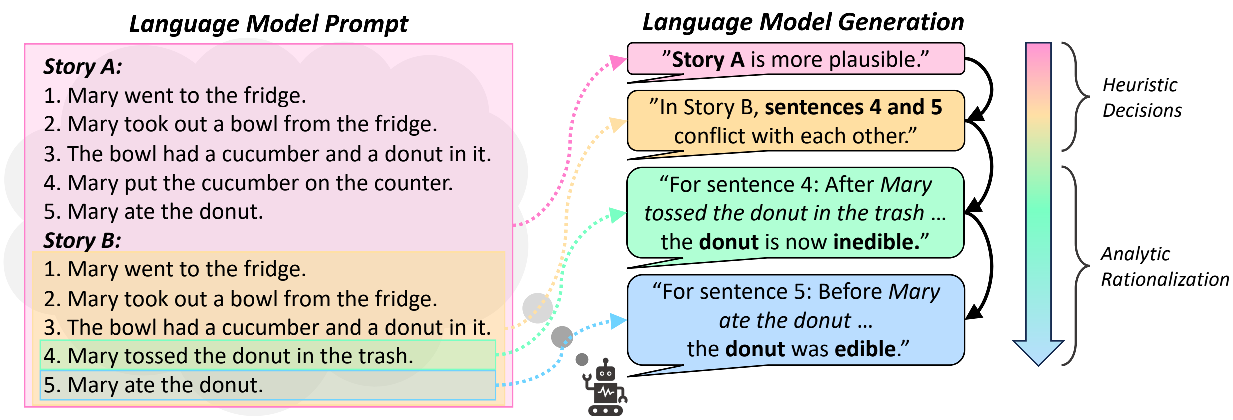


Figure 6.4: Heuristic-analytic reasoning (HAR) for in-context learning with pre-trained language models (LMs). HAR uses chain-of-thought prompting to bootstrap low-level analytic rationalization (e.g., physical state prediction) from high-level heuristic decision-making (e.g., implausible story and conflicting sentence selection), focusing the LM's attention to the most relevant context in each reasoning step.

information. First, while the original CGLI model takes one text as input, Focused CGLI (FCGLI) takes two texts, enabling the model to consider both together. Additionally, we filtered down training annotations in `TRIP` for low-level state prediction tasks to only include the most relevant physical states, i.e., those causing conflicts.[2] This allows the model to focus on learning the most important physical states from the high-dimensional hypothesis space. Development and testing data remain unchanged for fairness. FCGLI with heuristic-analytic reasoning (FCGLI-HAR) applies the above iterative deletion strategy to the input context of FCGLI.

[2]More fine-tuning details provided in Appendix B.2.

TRIP

Approach	Accuracy	***Consistency***	***Verifiability***
RoBERTa	72.9	19.1	9.1
CGLI	94.1	**77.3**	28.0
Breakpoint	80.6	53.8	32.4
FCGLI	93.7	66.2	33.8
FCGLI-HAR	94.3	75.4	**41.1**

Tiered-ProPara

Approach	Accuracy	***Consistency***	***Verifiability***
FCGLI	94.5	56.7	36.2
FCGLI-HAR	95.1	**83.6**	**57.4**

Table 6.1: TRIP and Tiered-ProPara results for baselines (introduced in Chapter 6.2.1), and heuristic-analytic reasoning with fine-tuned LMs (FCGLI-HAR).

Baselines. To measure the advantage of informing low-level reasoning steps with higher-level steps through sequential heuristic and analytic processes in FCGLI-HAR, we use FCGLI, where all reasoning steps are performed jointly without dependency between them, as a baseline. To contextualize TRIP results with past work, we also include RoBERTa [142] results from Chapter 5, CGLI [149], and Breakpoint Transformer [204].

Results. Results are listed in Table 6.1.[3] On TRIP, FCGLI-HAR exceeds or achieves comparable performance to baselines on all three metrics, pushing verifiability up to 41.1% and setting a new state-of-the-art result on the most difficult sub-task of TRIP. On Tiered-ProPara, FCGLI-HAR exceeds the FCGLI baseline on all metrics, with consistency and verifiability reaching a respective 83.6% and 57.4%, thus beginning to close the gap between accuracy and these coherence metrics. This shows that *HAR is indeed a promising strategy to improve coherence and dependability of reasoning in LMs.*

6.2.2 Learning Curves for FCGLI-HAR

We plotted the validation metrics through fine-tuning FCGLI and FCGLI-HAR in Figure 6.3. Unsurprisingly, we see that consistency and verifiability increase slower than accuracy through training, suggesting that these lower-level objectives are indeed most difficult to learn. However, FCGLI-HAR converged 1-2 training epochs faster than FCGLI on both datasets, suggesting that HAR may enable more efficient learning of coherent reasoning.

[3]Statistical significance testing of fine-tuning performance improvements in Appendix B.3.

6.3 Heuristic-Analytic Reasoning for LM In-Context Learning

While HAR can benefit domain-specific applications when integrated into LM fine-tuning, it requires expensive training on in-domain data that may sacrifice generalizability. To alleviate the limitations of fine-tuning, we also integrated HAR into in-context learning, taking advantage of emergent abilities of large LMs to perform coherent reasoning through free-form language generation.

Prompting techniques like chain-of-thought (CoT; 254) can be useful in in-context learning to demonstrate valid reasoning strategies to reach conclusions. However, CoT has traditionally been used to improve performance on complex high-level tasks by breaking them down into simpler low-level steps and reasoning from the bottom up. Meanwhile, the most complex sub-task and bottleneck in our problem is low-level physical state prediction [228], which is impossible to further break down, as descriptions of actions directly invoke a world state based on physical commonsense rules of how the world works. Since it would thus be difficult to use traditional CoT to generate a useful explanation to support physical state prediction, we apply HAR through a reverse CoT method where tiered tasks are demonstrated and predicted in a top-down sequence from high-level decisions to low-level rationalization.[4]

In `TRIP`, as shown in Figure 6.4, LMs are conditioned to first predict plausibility, a relatively easy heuristic process. To further refine the relevant context, the LM then predicts conflicting sentences, another heuristic judgement that need not be directly based on low-level commonsense knowledge, but may still benefit from being conditioned on the higher-level implausible story prediction. Lastly, the LM rationalizes these decisions with low-level physical states, an analytic process requiring the integration of external background knowledge about actions and objects. Instead of breaking down physical state prediction further (the typical purpose of CoT), we hypothesize that conditioning this sub-task with higher-level heuristic processes in HAR helps focus the model on the correct context and reason more coherently. We similarly apply HAR in Tiered-ProPara by prompting the LM to first select a story and sentence in which a particular entity is converted to another, and finally rationalizing these decisions with the name of the resulting entity after conversion, which requires commonsense understanding of how entities change as a result of various actions and processes.

[4]LMs are first conditioned with 4 consistent demonstrations of the ICL-HAR strategy from the training set. More details in Appendix B.6.

6.3.1 In-Context Learning Experiments

Next, we introduce our experiments on in-context learning with HAR (ICL-HAR).

Implementation. We apply ICL-HAR with InstructGPT[5] [34, 182] and LLaMA-65B[6] [236] using greedy decoding. Since LLaMA is limited to a context length of 2048 tokens, while prompts for `TRIP` include over 3000 tokens to familiarize the model with the physical state classification label space,[7] we apply it to a filtered version of `TRIP` which only includes instances where annotated states involve only the top-6 most frequent physical precondition-effect pairs.

Unstructured baseline. We propose a more traditional, unstructured in-context learning (ICL-U) baseline to help measure the advantage of HAR strategies applied in in-context learning. Instead of prompting LMs to predict all 3 reasoning steps in sequence, we extract each step of the task through separate but comparable prompts.[8] The LM is provided both input stories for each task, and given 4 task-specific demonstrations comparable to ICL-HAR. We then combine the 3 extracted predictions on each testing example into one reasoning chain to calculate evaluation metrics on them.

Traditional chain-of-thought baseline. Despite the anticipated limitations discussed above, we augment the ICL-U baseline with traditional CoT for comparison, creating an additional in-context learning with CoT (ICL-CoT) baseline. Specifically, we prompt InstructGPT[9] with "let's think step by step about $\langle$*sub-task*$\rangle$" [118] to generate a free-text explanation for each separate reasoning step. We then append these explanations to their respective prompts before eliciting predictions from the models. Unlike HAR, which enforces a top-down chain-of-thought, this traditional application of CoT allows the LM to attempt to break down each step before making a prediction for it.

Results. As shown in Table 6.2, we observe sharp performance improvements[10] from ICL-HAR over the baselines, particularly in the coherence metrics of consistency and verifiability, where our proposed strategy primarily comes into play. Meanwhile, compared to both

[5] Specifically, we use the `text-davinci-002` version through an Azure OpenAI deployment.

[6] Specifically, we use a HuggingFace [257] compatible version available at https://huggingface.co/decapoda-research/llama-65b-hf at the time of writing.

[7] See Appendix B.6.1 for more information.

[8] Example prompts provided in Appendix B.6.2.

[9] Based on preliminary experiments with both LMs, we expected InstructGPT to generate more reasonable explanations on the in-context demonstration examples, so we used its explanations in prompting both InstructGPT and LLaMA.

[10] Statistical significance testing of in-context learning performance improvements in Appendix B.3.

baselines, HAR improves InstructGPT consistency on `TRIP` from 40.7% up to 47.9%, and verifiability from a maximum of 10.8% up to 23.9%, over a 100% improvement on the latter. LLaMA sees similar improvements, especially in verifiability. On Tiered-ProPara, InstructGPT's consistency improves from 19.2% to 31.5%, and verifiability improves from 7.5% to 20.7%, nearly a 200% improvement on the latter. LLaMA again sees similar improvements. These results demonstrate that compared to common approaches for in-context learning with LMs, *human-inspired heuristic-analytic reasoning can significantly improve coherence and reduce hallucination.*

As expected, *traditional chain-of-thought in the ICL-CoT baseline brought only marginal improvements in most cases*, especially for verifiability, as physical state prediction (the bottleneck in coherent physical commonsense) cannot be further decomposed. Instead, we saw that the free-form explanations generated by InstructGPT for physical state prediction typically repeated specific sentences and actions from the story, which introduced no new information. ICL-HAR reveals a possible new use case for chain-of-thought-style prompting: refining LM attention to the most relevant language context. We investigate this further in Chapter 6.3.2.

Interestingly, *HAR not only brought vast improvements on verifiability metrics, but also some improvements in consistency.* In other words, the mid-level sentence selection tasks benefited slightly from being conditioned on higher-level story selection tasks, despite being considered heuristic processes that simply refine the context. This may suggest that rather than belonging to separate dual processes, these consecutive steps of reasoning may fall along a spectrum from heuristic to analytic processes in a recursive manner in LMs.

We also observed that LLaMA's accuracy decreased with ICL-HAR. As discussed in Chapter 4, it is not ideal for accuracy to far exceed consistency and verifiability, each of which require end-task predictions to be accurate as a prerequisite, as this indicates incoherent reasoning with insufficient support for the end-task prediction. Therefore, *drops in accuracy with HAR are not necessarily problematic.* Nonetheless, we believe it occurs due to the smaller complexity of LLaMA (65B), making it more sensitive to long prompts and generations. A quick fix could be to sequentially prompt the LM multiple times with shorter prompts for each reasoning step, where the prompt for each step is informed by the previous step's prediction. We explore this approach in Appendix B.4, and indeed find the accuracy does not drop with HAR.

InstructGPT

Approach	Acc.	`TRIP` *Cons.*	*Ver.*	Acc.	**Tiered-ProPara** *Cons.*	*Ver.*
ICL-U	70.9	40.7	7.1	54.9	17.4	5.2
ICL-CoT	75.0	40.7	10.8	50.7	19.2	7.5
ICL-HAR	72.6	**47.9**	**23.9**	54.9	**31.5**	**20.7**

LLaMA

Approach	Acc.	`TRIP` *Cons.*	*Ver.*	Acc.	**Tiered-ProPara** *Cons.*	*Ver.*
ICL-U	70.4	42.3	14.8	51.2	3.8	1.4
ICL-CoT	74.6	42.3	19.7	57.3	9.4	4.2
ICL-HAR	55.6	**44.4**	**35.2**	41.8	**17.8**	**13.1**

Table 6.2: Accuracy, consistency, and verifiability percentages for in-context learning with heuristic-analytic reasoning (ICL-HAR) in LMs, compared to an unstructured in-context learning (ICL-U) baseline that tackles reasoning steps through separate focused prompts.

6.3.2 Faithful Attention in ICL-HAR

To explore possible reasons for why HAR strengthens LMs' reasoning, we last compare and examine models' attention weights when generating language in our in-context learning experiments, where the model has access to the entire input language context throughout inference. Earlier in this section, we hypothesized that ICL-HAR enables the model to focus in on key parts of the context to make decisions and rationalize them more faithfully, similar to its role in human reasoning. For example, as shown in Figure 6.4, after a LM identifies the implausible story in `TRIP`, it should attend more to that story when identifying conflicting sentences. After identifying these sentences, it should attend more to those sentences when generating physical commonsense evidence.

In order to validate this hypothesis, we aggregate and normalize transformer self-attention weights for each story or sentence within the input prompt,[11] then use the ground truth reasoning chains for `TRIP` and Tiered-ProPara to evaluate the faithfulness of them.[12] To our knowledge, prior work has not studied attention weights in this way for in-context learning with LMs, so we hope to yield new insights on the nature of reasoning with them. We next introduce the evaluation criteria used for attention weights, then the results of the analysis.

[11] More details in Appendix B.7.1.

[12] It is important to note that the research community has adopted a broad space of methods and used various signals to interpret the predictions of language models. While this is just one avenue for such interpretabilit, [276] provides a detailed review of other approaches that could be applied in future work in this vein.

6.3.2.1 Attention Evaluation Criteria

We propose two kinds of measures to capture the faithfulness of attention and its relationship with coherence in `TRIP` and Tiered-ProPara: attentional ratios and attentional precision and recall.

Attentional ratios. To evaluate the faithfulness of LMs' attention, we can compare the attention weights for the correct segments of language context (i.e., stories or sentences) versus others through an *attentional ratio.* In both `TRIP` and ProPara, the model must first identify one of two stories containing some physical phenomenon before identifying which sentence(s) in that story contain it (*sentence selection step*). In `TRIP`, in cases where it correctly identifies the implausible story, we can calculate the attentional ratio for sentence selection by taking the ratio of the mean attention weight of the implausible story (i.e., where the LM must attend to identify conflicting sentences) to that of the plausible story. Similarly, in Tiered-ProPara, when the model correctly identifies which story contains an entity conversion, we calculate the ratio of the mean attention weight for the story containing an entity conversion to that of the story that does not.

When the model correctly identifies which sentence(s) contain a phenomenon (i.e., a plausibility conflict or entity conversion), the model must lastly generate physical commonsense knowledge for those sentences to rationalize its decisions (*physical state prediction step*). In `TRIP`, we calculate the attentional ratio for physical state prediction by taking the ratio of the mean attention weight for conflicting sentences (i.e., sentences from which physical states must be predicted) to that of all other sentences. In Tiered-ProPara, we similarly calculate the ratio between the mean attention weights for the conversion sentence and other sentences.

Together, these ratios can provide a sense of how strongly the LM is attending to the relevant language context to produce each level of the reasoning chain. We expect that higher ratios indicate more faithful rationalizations from the model.

Attentional precision and recall. Beyond the faithfulness of attention, we would like to understand how faithful attention relates to coherent reasoning, i.e., the LM's predicted sentence(s) and physical state(s) to rationalize which story it chose. For each of these reasoning steps, there are four possible combinations of faithfulness of model attention and correctness of its predictions:[13]

1. Attends to the *correct* context, and generates a *correct* prediction (**true positive**)

[13] Attention faithfulness classified by a threshold. More details and supporting examples listed in Appendix B.7.2.

Sentence Selection Step

Approach	TRIP Ratio	TRIP Prec.	TRIP Rec.	Tiered-ProPara Ratio	Tiered-ProPara Prec.	Tiered-ProPara Rec.
ICL-U	0.96	42.6	39.6	0.90	14.8	30.6
ICL-HAR	**1.07**	**75.2**	**48.7**	**1.80**	**51.1**	**58.2**

Physical State Prediction Step

Approach	TRIP Ratio	TRIP Prec.	TRIP Rec.	Tiered-ProPara Ratio	Tiered-ProPara Prec.	Tiered-ProPara Rec.
ICL-U	1.23	43.0	35.4	1.21	14.6	25.9
ICL-HAR	**1.95**	**79.8**	**98.2**	**2.20**	**72.1**	**83.3**

Table 6.3: Attentional ratio, average precision (%), and average recall (%) for LLaMA baseline and HAR strategy, during different physical commonsense reasoning steps. Precision and recall averaged across several attention thresholds, as outlined in Appendix B.7.2.

2. Attends to the *correct* context, but generates an *incorrect* prediction (**false positive**)
3. Attends to the *incorrect* context, and generates an *incorrect* prediction (**true negative**)
4. Attends to the *incorrect* context, but generates an *correct* prediction (**false negative**)

We can calculate the precision and recall of attention to measure how the correctness of attended language context correlates with correctness of these model predictions (and thus coherence of reasoning). Given a set of evaluation examples, we define *attentional precision* as the number of true positives divided by all positives, representing how often the LM is correct given faithful attention. We define *attentional recall* as the number of true positives divided by the sum of true positives and false negatives, representing how often the LM attends faithfully given a correct prediction. Together, these metrics can provide an impression of the connection between faithful attention and coherent reasoning under different prompting strategies.

6.3.2.2 Attention Analysis Results

To understand why HAR improves coherence so significantly, we compare LM self-attention distributions as a reasoning chain is generated in the in-context learning setting. As this analysis requires access to LM internals, we can only use open-source models like LLaMA here. Following findings from prior work that the middle layers of transformer-based language representations contain the most transferable semantic information [188, 234], we extract self-attention weights from the center-most 20 layers of the transformer backbone in LLaMA.

As shown in Table 6.3, ICL-HAR sharply exceeds the unstructured ICL-U baseline across

Story A: `51.1%`
1. The book was on the table.
2. Sarah took the book to the copier.
3. Sarah copied a few pages and used the stapler to keep them together.
4. Sarah could not find her notebook.
5. Sarah put the notebook next to the keyboard.

Story B: `48.9%`
1. The book was on the table.
2. Sarah took the book to the copier.
3. Sarah copied a few pages and used the stapler to keep them together.
4. Sarah put the pages inside her notebook.
5. Sarah put the notebook next to the keyboard.

Story A: `61.7%`
1. The book was on the table.
2. Sarah took the book to the copier.
3. Sarah copied a few pages and used the stapler to keep them together.
4. Sarah could not find her notebook.
5. Sarah put the notebook next to the keyboard.

Story B: `38.3%`
1. The book was on the table.
2. Sarah took the book to the copier.
3. Sarah copied a few pages and used the stapler to keep them together.
4. Sarah put the pages inside her notebook.
5. Sarah put the notebook next to the keyboard.

Figure 6.5: Story-wise attention visualization on `TRIP` in sentence-of-conversion detection, ICL-U (top) vs. ICL-HAR (bottom).

Story A: `41.0%`
1. Coal is heated in the boiler.
2. The water tank over the boiler is heated.
3. Creates steam.
4. The steam is funneled to the piston.
5. Piston uses the steam as energy.
6. The piston causes the crankshaft to move.

Story B: `59.0%`
1. Plates on the Earth's crust move slowly past each other.
2. As the plates move, they exert a great force.
3. When the force is large enough, the crust breaks.
4. The stress is released as energy.
5. The energy moves through the Earth in the form of waves.
6. We feel the earthquake.

Story A: `16.3%`
1. Coal is heated in the boiler.
2. The water tank over the boiler is heated.
3. Creates steam.
4. The steam is funneled to the piston.
5. Piston uses the steam as energy.
6. The piston causes the crankshaft to move.

Story B: `83.7%`
1. Plates on the Earth's crust move slowly past each other.
2. As the plates move, they exert a great force.
3. When the force is large enough, the crust breaks.
4. The stress is released as energy.
5. The energy moves through the Earth in the form of waves.
6. We feel the earthquake.

Figure 6.6: Attention visualization on Tiered-ProPara in selecting which sentence *energy* is converted in, baseline ICL-U (top) vs. ICL-HAR (bottom). Attention averaged across stories and reflected by the intensity of color.

Story A:
1. Tom found he is out of ice cream. 9.0%
2. Tom peeled a hard boiled egg. 5.5%
3. Tom sliced the egg with a knife. 4.6%
4. Tom washed the knife in the sink. 4.4%
5. Tom ate ice cream for dessert. 8.6%
Story B:
1. Tom poured a glass of milk. 10.4%
2. Tom peeled a hard boiled egg. 25.4%
3. Tom sliced the egg with a knife. 3.3%
4. Tom washed the knife in the sink. 16.2%
5. Tom ate ice cream for dessert. 12.5%

Story A:
1. Tom found he is out of ice cream. 21.3%
2. Tom peeled a hard boiled egg. 7.1%
3. Tom sliced the egg with a knife. 5.3%
4. Tom washed the knife in the sink. 4.4%
5. Tom ate ice cream for dessert. 15.4%
Story B:
1. Tom poured a glass of milk. 7.2%
2. Tom peeled a hard boiled egg. 8.2%
3. Tom sliced the egg with a knife. 2.4%
4. Tom washed the knife in the sink. 20.8%
5. Tom ate ice cream for dessert. 7.9%

Figure 6.7: Sentence-wise attention visualization on `TRIP` in state change prediction for baseline ICL-U (left) vs. ICL-HAR (right). Attention averaged across sentences.

Story A:
1. Plants have roots. 10.8%
2. The roots grow out. 3.9%
3. Roots have fibers that are attached to them. 8.6%
4. They attract water. 6.7%
5. They suck up water. 5.1%
6. They absorb the water. 12.1%
Story B:
1. The air is cold. 6.0%
2. Water is in the air. 4.4%
3. The water forms tiny ice crystals. 9.9%
4. The ice crystals collide with each other. 4.3%
5. The ice crystals stick to each other. 3.0%
6. The ice crystals get bigger as more of them stick together. 3.4%
7. The ice crystals get too heavy to be in the air. 3.8%
8. The ice crystals become snowflakes. 9.4%
9. The snow flakes fall to the ground as snow. 8.6%
What happened to water?

Story A:
1. Plants have roots. 2.3%
2. The roots grow out. 1.5%
3. Roots have fibers that are attached to them. 3.4%
4. They attract water. 2.8%
5. They suck up water. 3.2%
6. They absorb the water. 11.4%
Story B:
1. The air is cold. 4.4%
2. Water is in the air. 7.4%
3. The water forms tiny ice crystals. 26.5%
4. The ice crystals collide with each other. 9.9%
5. The ice crystals stick to each other. 4.9%
6. The ice crystals get bigger as more of them stick together. 6.5%
7. The ice crystals get too heavy to be in the air. 4.0%
8. The ice crystals become snowflakes. 6.6%
9. The snow flakes fall to the ground as snow. 5.4%
What happened to water?

Figure 6.8: Sentence-wise attention visualization on Tiered-ProPara in entity conversion prediction, ICL-U (top) vs. ICL-HAR (bottom).

all attentional metrics, both when selecting sentences (i.e., conflicting sentences or sentences with conversions) and predicting states of entities. While observed ratios show that *ICL-HAR has more faithful attention to relevant parts of the context*, the high values of attentional precision (up to 80%) and recall (up to 98%) show that *faithful attention and coherent reasoning go hand-in-hand*; faithful attention in ICL-HAR is likely to bring coherent reasoning, and vice-versa. This demonstrates that HAR enables more trustworthy reasoning in LMs.

Lastly, we present example visualizations of self-attention patterns [259] for ICL-HAR compared to the ICL-U baseline in Figures 6.5, 6.6, 6.7, and 6.8. For example, in Figure 6.6, we see that in the sentence selection step in Tiered-ProPara, LLaMA had higher average attention on the story containing a conversion of *energy* under ICL-HAR than the baseline. Similarly, in Figure 6.7, we see that under ICL-HAR, LLaMA paid more attention to the conflicting sentences in the physical state prediction step in `TRIP`, whereas the baseline had high attention on irrelevant sentences. This shows how HAR can help focus LMs on the most relevant language context at each step of reasoning.

6.4 Summary of Findings

In this chapter, we took inspiration from the synergy between heuristic and analytic processes in human reasoning to explore how high-level decision-making tasks in commonsense reasoning can condition and drive lower-level rationalization tasks supporting them in LMs. We proposed two general strategies to integrate heuristic-analytic reasoning (i.e., HAR) into LM fine-tuning and in-context learning, and found that HAR sharply improved reasoning coherence, outperforming competitive baselines on two benchmark tasks. In fine-tuning, we saw that HAR enabled LMs to learn to reason not only more coherently, but also faster. Meanwhile, in in-context learning, we found that improvements were enabled by more faithful attention to the language context within each step of reasoning, shedding light on the nature of incoherence in language generation. While this human-inspired approach shows promising strides toward more trustworthy reasoning from LMs, future work should continue to dive deeper into cognitively motivated strategies to further strengthen coherent reasoning in AI systems and improve human-machine alignment.

CHAPTER 7

Physical Perception and Causality in Foundational Multimodal Representations†

The previous chapters of this thesis have made strides in evaluating and strengthening the coherence of physical commonsense reasoning (PCR) in natural language understanding (NLU) for foundational language models (LMs). As discussed in Chapter 1, such effort is especially important for trustworthiness between humans and intelligent agents powered by foundational LMs in real-world, physical settings. However, the work thus far has yet to move beyond text-based settings. In the real world, commonsense NLU may also depend on a number of other modalities, especially visual perception (e.g., through images or videos). To bring foundational LMs into tasks in this space, a parallel line of work has developed foundational vision-and-language models (VLMs) [231, 136, 272, 117, 238, 134, 6, 2, 133, 44, 185, 140, 76], which extend the language modeling paradigm to incorporate images and videos into LM inputs, enabling LMs to generate language to describe or reason over them.

While early works attempted to jointly learn VLMs by pre-training them end-to-end on large-scale datasets for vision-and-language tasks such as image and video captioning, more recently this is typically achieved through learning a projection from a frozen pre-trained visual representation into the input space of a frozen pre-trained foundational LM, implemented by a neural network architecture (as shown in Figure 7.1. Foundational VLMs acquire broad capabilities in visual understanding from their pre-training, accessible by simply prompting the VLM. It is common for recent state-of-the-art VLMs to use Contrastive Language-Image Pre-training (CLIP) for visual representation [193, 133]. CLIP is trained through contrastive learning over paired images and texts to learn dual vision and language vector representations, such that the representations for related images and texts should be highly similar (and vice versa). This was an early example of zero-shot visual understanding

†The experiments described in this chapter are from an intermediate unpublished study led by Shane Storks. We acknowledge Wenfei Tang, Sungryull Sohn, Moontae Lee, Honglak Lee, and Joyce Chai for helpful discussions and contributions to this work.

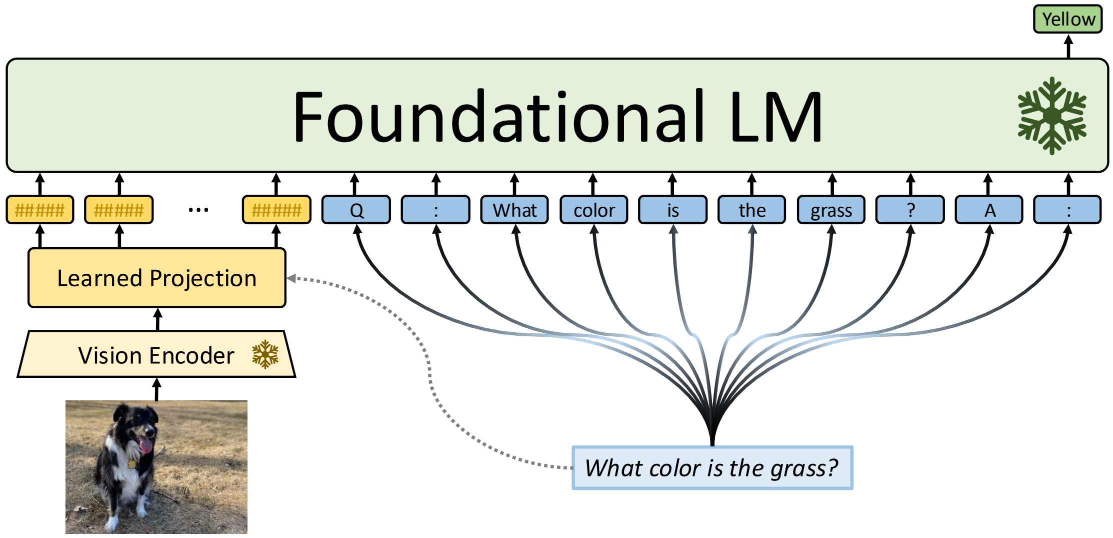


Figure 7.1: Foundational vision-and-language models (VLMs) learn a projection from the outputs of a frozen foundational vision encoder to the inputs of a frozen foundational language model (LM), optionally informed by text inputs.

achieved through large-scale pre-training, and thus serves as a strong frozen visual encoder to integrate into VLMs.

Given that today's state-of-the-art foundational VLMs are built on top of LMs, we can expect similarly incoherent tendencies in them. For example, past work has discovered new forms of hallucination specific to VLMs, such as the fabrication of objects when generating image captions [45, 137]. As such, when applying them to physical settings, it is essential that we ensure information generated by them is consistent with both the language and visual context, and can be verified through observable states of the environment. A key component of this is to understand how VLMs represent physical concepts like objects, actions, and physical states. This is particularly difficult in visual perception, as objects may appear vastly different depending on what physical state they are in, and physical states can also appear vastly different depending on the object they are applied to. For example, an *orange* will look different depending on whether it is *peeled* or *sliced*, while a *peeled orange* looks quite different from a *peeled banana*.

In this chapter, we perform an initial investigation of the viability of multimodal language-and-vision representations for PCR. Specifically, we apply CLIP to action-effect prediction (AEP) [75], a task where an AI system is given a physical action in the form of a verb-noun pair, e.g., "peel orange," and must rank a large pool of candidate images for how well they portray the effect of this action. To capture the challenges of visually representing actions,

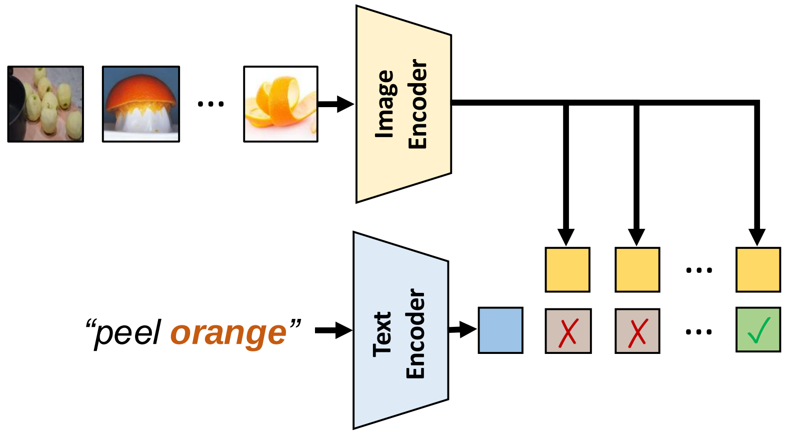


Figure 7.2: To apply dual-encoder multimodal representations like CLIP [193] to action-effect prediction (AEP), candidate images of an action are ranked based on the similarity of their representations to the representation of a verb-noun pair describing the action.

each action in the dataset is unique, and may have multiple correct images.[1] By applying CLIP to this task, we seek to understand how its visual representation captures objects and their physical states, and how well this representation can be mapped to language descriptions of actions. In comparison with zero-shot CLIP, we also attempt to augment CLIP with intuitive physical causality by applying generative LMs and text-to-image diffusion models to simulate effects of actions both textually and visually.

7.1 Zero-Shot Action-Effect Ranking

Applying CLIP to AEP, a cross-modal language-to-image task, is straightforward. As shown in Figure 7.2, given an action described in language l and a set of candidate matching effect images V_e, we apply the CLIP language and visual encoders E_L and E_V to generate vector representations of the same dimension. Each candidate effect image $v_e \in V_e$ is ranked in descending order by cosine similarity $\cos(E_L(l), E_V(v_e))$.

For a more CLIP-friendly prompt, we form l from the verb-noun pair (l_v, l_n) given in the AEP data. We use the template "A photo of a $\langle l_v \rangle$ $\langle l_n \rangle$," where we use `mlconjug3`[2] to convert l_v to its past participle and spaCy[3] to adjust prompts for grammaticality as needed. For example, given the verb-noun pair "slice orange," we generate the prompt for CLIP as "A

[1]To reproduce the train-test split in the original paper, we follow [75] in randomly sampling 10% of each verb-noun pair's positive and negative images for training, 30% for validation, and 60% for testing to make up the pools of candidate effect images in each split.

[2]https://mlconjug3.readthedocs.io/

[3]https://spacy.io/

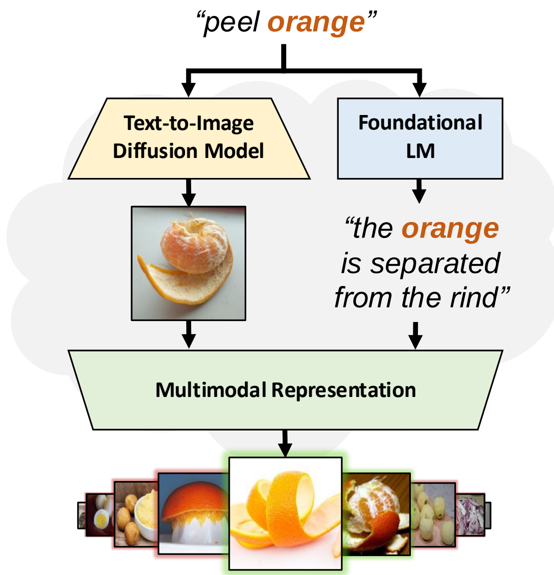


Figure 7.3: Generative foundation models as simulation engines to simulate the effects of physical actions. In action-effect prediction [75], we can simulate the effect of *peel orange* by prompting foundational text-to-image diffusion and language models (LMs) for additional multimodal context before using CLIP [193], a foundational dual-encoder vision-and-language representation, to rank effect images.

photo of a sliced orange." After ranking, we evaluate CLIP's zero-shot performance through label ranking average precision (LRAP), which is higher when correct images are ranked higher, and top-n accuracy metrics Acc@n for $n \in \{1, 5, 20\}$, which measure the proportion of verb-noun pairs for which the top n images are correct.

7.2 Zero-Shot Action-Effect Ranking with Intuitive Action Simulation

CLIP represents the language and images associated with a physical situation the same regardless of whether they are applied to physical perception or reasoning. Inspired by work toward intuitive physics-engine approaches to physical understanding [121], we hypothesize that when used as a backbone for PCR, they could benefit from being augmented with explicit reasoning through an imagination-inspired simulation engine, as shown in Figure 7.3. Specifically, we can condition generative models on task inputs, and use them to generate additional helpful context to support prediction of effect states. Notably, this creates an intuitive separation between reasoning, which happens within the generative models for simulation, and perception, which happens in the CLIP backbone to infer the effect state.

The availability of generative foundation models for images [200, 199] and language [34] enables us to perform both textual and visual simulation, and thus understand where each is most beneficial.

7.2.1 Intuitive Textual Simulation

The recent scaling of large pre-trained language models [34] has enabled the ability to prompt them for zero-shot reasoning and planning with astounding results [118, 101]. Inspired by this success, we explore their capability to support PCR by augmenting the CLIP backbone with textual simulation.

In AEP, we achieve this by generating language descriptions for the effect state of an action. Given a verb-noun pair (l_v, l_n), each simulation is generated by prompting a PLM with the text fragment "Question: When someone $\langle l_v \rangle$ an $\langle l_n \rangle$, what happens to the $\langle l_n \rangle$? Answer: The $\langle l_n \rangle$ is now," where `mlconjug3` is again used to convert l_v to third-person singular form. From the GPT-3 generated text T, we form the imagined effect l_e^i by the more CLIP-friendly prompt "A photo of a $\langle l_v \rangle$ $\langle l_n \rangle$. The $\langle l_n \rangle$ is $\langle T \rangle$," where l_v is again converted to an adjective form.

For example, given the verb-noun pair "slice orange," we prompt the PLM as described, and receive an output T to describe the state of the *orange*, e.g., "in multiple pieces." From this, we create a CLIP prompt "A photo of a sliced orange. The orange is in multiple pieces." Compared to the simulation-free CLIP prompt "A photo of a sliced orange," we expect this simulation-augmented prompt to be more informative and better guide the model to choose the correct image. Upon generating k textual simulations $l_e^i \in L_e^i$, we now perform inference by averaging their similarity to each candidate effect image $v_e \in V_e$ when represented with language and visual encoders E_L and E_V:

$$\sum_{l_e^i \in L_e^i} \cos\left(E_L(l_e^i), E_V(v_e)\right)$$

7.2.2 Intuitive Visual Simulation

The development of foundational text-to-image diffusion models [200, 199] has produced systems capable of generating realistic images conditioned on text prompts, which have been shown to be somewhat helpful in simulating situations and world states to improve language understanding [147, 260, 141] and planning [115]. To apply them to physical state prediction, we augment the CLIP backbone with visual simulation powered by these models.

In AEP, we achieve this by generating simulated images for the effect state of an action using such a diffusion model. Given a verb-noun pair (l_a^v, l_a^n), each simulation is generated by prompting the diffusion model with the same prompt used to interface directly with CLIP, i.e., for "slice orange," the prompt would be a "A photo of a sliced orange." We expect that generated images may look quite similar to the ground truth images for the action's effect, and thus serve as a useful signal for this task. Upon generating k visual simulations $v_e \in V_e^i$, we can again perform inference through an ensembled similarity search similar to the method described in Chapter 7.2.1.

7.2.3 Multimodal Simulation

We may also consider using both textual and visual simulation jointly to augment the pipeline. We consider two strategies for combining these signals: *parallel* and *sequential*. Given sets L^i of textual simulations and V^i of visual simulations, we can combine them in parallel by simply incorporating all of them as possible signals to support the final prediction by CLIP. Or, we can do a sequential two-stage simulation by first generating L^i using a foundational LM then using the k generated textual simulations as prompts for a text-to-image diffusion model to then generate visual simulations. In the sequential approach, only visual simulations are considered, but they're conditioned on textual simulations.

7.2.4 Experimental Results

The results on the AEP testing set are shown in Table 7.1. We use an instance of CLIP using vision transformer (ViT) [61] as a visual encoder.[4] Textual simulation is implemented with the InstructGPT variant of GPT-3 [182], while visual simulation is implemented with DALL-E 2 [199].

Zero-shot performance. Without simulation, CLIP achieves impressive zero-shot results on AEP, significantly exceeding all metrics from supervised convolutional neural network (CNN) baselines from [75], one of which was trained on a large dataset of images retrieved from the web. This demonstrates the viability of foundational multimodal representations, and thus VLMs, for capturing actions in images.

Impact of textual simulation. Augmenting CLIP with just $k = 1$ textual simulation shows a sharp performance improvement to 50.7% top-1 accuracy, with similar increases in other metrics. This shows that using a foundational LM to generate language simulating action effects, which we have extensively studied in previous chapters, may better reveal

[4]Pre-trained CLIP weights can be found at https://huggingface.co/laion/CLIP-ViT-L-14-laion2B-s32B-b82K.

Model	LRAP	Acc@1	Acc@5	Acc@20
CNN	18.2	32.9	62.9	80.7
CNN + Web Images	29.0	41.4	75.0	92.1
CLIP	40.6	45.0	84.3	94.3
CLIP + Textual ($k = 1$)	43.3	**50.7**	85.0	**98.6**
CLIP + Textual ($k = 5$)	**43.5**	50.0	**85.7**	97.9
CLIP + Visual ($k = 1$)	23.4	32.9	65.0	86.4
CLIP + Visual ($k = 5$)	24.6	35.7	66.4	87.9
CLIP + Textual + Visual (Parallel; $k = 1$)	31.9	42.9	72.1	92.1
CLIP + Textual + Visual (Parallel; $k = 5$)	34.0	45.7	72.1	94.3
CLIP + Textual + Visual (Sequential; $k = 1$)	25.0	34.3	72.1	88.6
CLIP + Textual + Visual (Sequential; $k = 5$)	27.0	38.6	68.6	90.7

Table 7.1: Metrics for CLIP on action-effect prediction (AEP) [75]. In the first section, zero-shot CLIP is compared to supervised baselines from [75]. In the remaining sections various configurations of action simulation are applied with $k = 1$ and 5 simulated effect texts and/or images respectively. In multimodal ("Both") imagination, generative model outputs are combined in parallel or sequentially as described in Chapter 7.2.3.

physical state information from CLIP's visual representations as opposed to only using descriptions of actions themselves (i.e., through verb-noun pairs). Increasing k to 5 brings shows minimal further improvements.

Impact of visual simulation. Surprisingly, visual simulation surprisingly causes performance degradation. Increasing the number of visual simulations k leads to a slight recovery in performance, but the performance of CLIP is still better without using them at all. This may suggest that visual representations vary significantly for images of the same physical concepts. Since language provides powerful abstractions for physical concepts, language-based representations for physical concepts may be more consistently mapped to a variety of images portraying the same physical concepts. This may further explain the performance degradation in the multimodal simulation settings that integrate both textual and visual simulated effects. In Chapter 7.2.4.1, we qualitatively analyze sample simulated effect images and texts to better understand this disparity.

7.2.4.1 Sample Simulations

Based on the results so far, it seems that textual simulations is helpful for PCR, but visual simulation is not. As such, we provide some sample simulations for each modality to

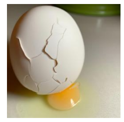
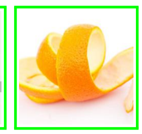
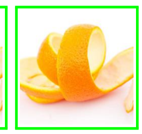
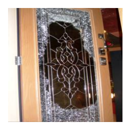
Verb-Noun Pair	Simulated Effect Text	Reference Effect Text	Simulated Effect Image	Reference Effect Image	Top-Ranked Effect Images (Textual, Visual, Both)
chop carrot	*the carrot is in smaller pieces*	*the carrot is cut into small pieces*			  
crack egg	*the egg is broken*	*the egg is broken open*			  
ignite wood	*the wood is on fire*	*the wood is on fire*			  
peel orange	*the orange is peeled*	*the orange has no skin*			  
smash door	*the door is broken*	*the door is broken in*			  

Table 7.2: Intuitive simulation outputs for selected verb-noun pairs, compared to selected reference human-annotated effect texts and images provided in the AEP dataset. The top-ranked effect image by the pipeline is also shown when the inference includes only textual simulations, visual simulations, or both (highlighted in green for correct choices and red for incorrect choices).

qualitatively examine their relevance and coherence to their prompts. Sample simulations compared to reference human-annotated counterparts are shown for selected verb-noun pairs in Table 7.2.

First, we observe that both the textual and visual simulation models generate quite realistic and relevant outputs for each verb-noun pair, with generated effect texts nearly or exactly matching reference effect texts, and images appearing to represent verb-noun pairs mostly well. Nonetheless, they contribute to mixed success in the final ranking, depending on whether imagined texts, images, or both are used in the ranking. For example, we see that while both types of simulation lead to correct predictions on *chop carrot*, *ignite wood*, and *peel orange*, visual simulation causes incorrect predictions for *crack egg* and *smash door*. This may suggest that CLIP may be better at fine-grained semantic matching of text to images than images to images. Interestingly, the incorrect predictions are still nearly correct; for *crack egg*, an image of a boiled egg is selected, and for *smash door*, an image of a smashed window is selected. even when the pipeline fails, it still produces a somewhat reasonable incorrect answer.

7.3 Summary of Findings

In this chapter, we introduced foundational vision-and-language models (VLMs), and investigated the capacity of CLIP [193], a typical language-dependent visual representation used in VLMs, to be used for action-effect prediction. First, we found that CLIP exhibits an impressive capability to represent actions outperforming supervised baselines from prior work. Notably, its performance is enhanced if we simulate the effects of actions with a foundational LM, then incorporate the generated physical state descriptions into action representations. This may suggest that VLMs' visual representations are better equipped to handle more direct descriptions of physical states of objects than the high-level actions that have been applied to objects. Furthermore, we found that using text-to-image diffusion models to simulate images of action effect states hurt CLIP's performance. This suggests that VLMs' visual representations are not well-equipped to represent physical concepts abstractly and generally, and thus language may be advantageous because of its ability to describe and abstract physical concepts with symbols.

CHAPTER 8

Coherent Physical Commonsense Reasoning for Procedural Mistake Detection in Video Frames†

The problem of automated, interactive task guidance has recently attracted attention in the AI research community [16, 249, 183, 27]. A successful intelligent agent for this problem can observe a human user through video (usually egocentric) and interact with them through language and visual cues to guide them through completing a task. One key component of such an agent is **procedural mistake detection** (PMD): the ability to detect when the user performs an action that deviates from a procedural text, e.g., a recipe or instruction manual. This ability requires physical commonsense reasoning (PCR) to anticipate the success conditions for actions from this procedural text, extract relevant physical state information from the visual scene, and reconcile these sources of information to determine whether a mistake has occurred.

Mistake detection has proven to be a challenging problem within task guidance. One thread of work here has attempted to fine-tune primarily vision-based classifiers without incorporating language [249, 183], while another has attempted to apply foundational language models (LMs) and vision-and-language models (VLMs) to this problem [62, 16], but both types of approaches have failed to achieve a viable level of accuracy in detecting mistakes. In the latter effort, [16] finds that while the web-scale multimodal pre-training of these models enables flexibility and generalization to a wide variety of procedures, they often produce noisy, vague, or otherwise insufficient information to facilitate reasoning about the success of procedure execution in visual scenes. This capability to extract key task-relevant visual information may be crucial for improving mistake detection accuracy in foundation models,

†The work described in this chapter is from an ongoing project led by Shane Storks. We acknowledge Yayuan Li, Itamar Bar-Yossef, Zheyuan Zhang, Fengyuan Hu, Ruixuan Deng, Megan Su, Jason J. Corso, and Joyce Chai for helpful discussions and contributions to this project.

as well as providing system interpretability for user trust. However, prior work has largely overlooked the interpretability of mistake detection, instead targeting binary and categorical classification tasks in their system design and quantitative evaluations.

In this chapter, inspired by the previously presented work toward coherent commonsense reasoning, we adapt our notions of consistency and verifiability to the problem of *interpretable procedural mistake detection* (interpretable PMD) in egocentric video frames with foundational VLMs.[1] To enable this inquiry, we transform an existing egocentric video dataset for experimentation with foundational VLMs on this problem. We then propose and validate novel reference-free automated metrics for the coherence of PCR in this problem based on a fine-tuned natural language inference (NLI) model, which we adapt into a multi-tiered evaluation paradigm for both accuracy and coherence in mistake detection. In our experiments, we formulate the problem as a multi-tiered reasoning process consisting of iterative visual question generation (VQG), visual question answering (VQA), and a determination of success based on visual information and/or information collected from VQA. Since foundational VLMs are still in early stages and highly vulnerable to illusion and hallucination [45, 137, 274, 88], their reasoning capabilities are not yet well-understood, and there are thus many under-explored questions about how best to apply them to multimodal reasoning tasks like mistake detection. Several of these questions guide us in laying the groundwork for studying coherent PCR in interpretable PMD with foundational VLMs:

1. *How does the direction of multi-tiered reasoning (top-down or bottom-up) impact the accuracy and coherence of PMD?*

2. *In addition to VLM likelihood, can reference-free coherence metrics provide a useful signal for more accurate or coherent PMD?*

3. *Can foundational LMs' in-context learning capability be leveraged for more accurate or coherent PMD?*

4. *How do available methods to mitigate visual illusion and hallucination* [245, 127, 8] *impact accuracy and coherence of PMD?*

To address these questions, we perform experiments evaluating impact of various interventions in question generation and answering on the performance of foundational VLMs in interpretable PMD. Notably, we show how our multi-tiered metrics for this problem enable previously impossible rich insights into VLMs' reasoning from various perspectives, enabling

[1] Here, we are particularly targeting lightweight, open-source VLMs that are feasible and affordable to run in online settings like interactive task guidance.

a previously impossible level of interpretability in mistake detection that could support targeted improvement in downstream engineering of task guidance systems.

8.1 Redefining Mistake Detection for Interpretability and Coherence

Given a text description of a procedure and a video frame of the procedure being performed by a user, the task of PMD is to judge whether or not the procedure was performed correctly. Since prior work has made limited progress on this problem by primarily focusing on the visual modality and binary or categorical classification, we additionally wish to draw from the previous chapters of this thesis to elicit interpretable and coherent explanations for mistakes in natural language.

In this section, we first formally define the problem of interpretable PMD in an approachable manner for recent open-source VLMs. We then introduce a naturalistic benchmark dataset we curated for evaluating interpretable PMD in open-source VLMs.

8.1.1 Interpretable Procedural Mistake Detection

For interpretable PMD, we not only want a decision from the system on whether or not a mistake has occurred in executing a procedure, but also an explanation for why. Since current state-of-the-art approaches for mistake detection do not have a viable level of accuracy, especially those based on foundational VLMs, explanation is crucial to ensure the human user can understand the system's reasoning process and act on, disregard, or even correct its outputs accordingly. Further, given findings that VLMs perform better on complex visual reasoning problems after breaking them down into lower-level steps [224, 280], it is possible that eliciting such explanations can improve the accuracy of VLMs.

Formally, interpretable PMD provides the following inputs:

- A short *procedural text* P
- A single *video frame* F which may or may not show the successful completion of the procedure described in P

Given these inputs, as shown in Figure 8.1, a system must return the following outputs:

- A binary decision y for whether the procedure has been successfully completed ($y = 0$ indicates success, and $y = 1$ indicates a detected mistake)

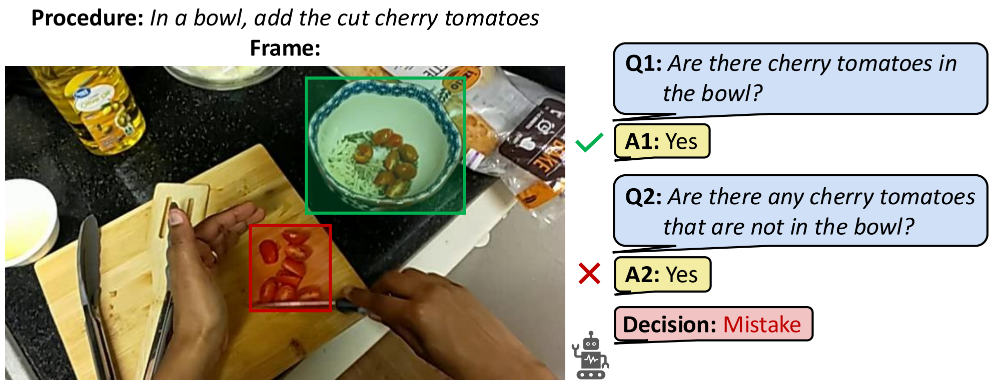


Figure 8.1: In *interpretable procedural mistake detection (PMD)*, foundational VLMs must not only to judge whether a video frame shows a successful or mistake state for a given procedure (e.g., "In a bowl, add the cut cherry tomatoes"), but also generate visual questions and answers to reveal key physical states of the environment that justify the decision. Procedure and frame from CaptainCook4D [183].

- A sequence of generated questions $\mathcal{Q}$ and yes-no answers $\mathcal{A}$ which together provide a sufficient explanation for the decision y

Later in this chapter, we will introduce specific approaches to evaluate and generate these explanations. As this process requires generating several pieces of information, it is expected that this would be applied once at the end of procedure execution to verify the state of the environment, e.g., when the user asks a task guidance system to advance to the next step. Based on the results of our study, one could explore streamlined, lighter-weight approaches to apply VLMs to a stream of video frames in a live online setting. However, as the capability to reason over sequences of frames is still limited in state-of-the-art VLMs, we leave this for future work. In the meantime, formulating the problem in this way enables our initial experiments to begin building a meaningful understanding of the behaviors of VLMs in PMD. As future VLMs become better at capturing physical states in videos, the formulation of this problem can be revisited.

8.1.2 Constructing a Dataset for Procedural Mistake Detection

Various benchmark datasets have been created for mistake detection from egocentric video, each of which includes video and procedural text along with various other modalities [62, 16, 249, 183]. Videos are annotated with detailed information about mistakes. While these datasets are useful resources for research in task guidance, most of them include dialog interaction between a user and instructor agent which often causes mistakes to be corrected before or while they happen. Furthermore, they include mistakes around temperature, tim-

Figure 8.2: Selected examples from our reformulated Ego4D [85] for Procedural Mistake Detection (Ego4D-PMD). For each matching pair of a video frame and procedural text (in this case, "Fold the cloth with your hands"), we generate a success example, and generate various types of mistake examples by sampling alternate video frames: *incomplete* execution of the procedure, execution with the *wrong verb* (e.g., wringing a cloth instead of folding), execution with the *wrong noun* (e.g., folding a paper instead of a cloth), and execution with both the *wrong verb and noun* (e.g., opening a notepad instead of folding a cloth). Images slightly cropped for space.

ing, small measurements, and other physical properties of the environment that are difficult for open-source VLMs, which are mostly optimized for representing single images, to perceive. While some of these difficulties could be overcome through a two-way dialog between the agent and a user, this makes it difficult to isolate mistakes occurring in the videos and dive deep into the PCR behind detecting them. As such, we follow [62] in recasting Ego4D [85], a narrated procedural video dataset, into an offline mistake detection format. While the recasted dataset targets the coherence of PCR in mistake detection, it could still be used to inspect the capabilities of a larger dialog system for task guidance.

8.1.2.1 Ego4D for Mistake Detection

Ego4D is a large-scale egocentric video dataset for everyday activities with dense annotations for various aspects of the videos [85]. Ego4D's hand and object interactions data subset includes videos of physical actions being performed with various objects. Each video is annotated with narrations describing fine-grained procedures being performed, timestamps for when it begins and ends, and category labels for the verb and noun characterizing the procedure. This makes an ideal testbed for evaluating foundational VLMs' physical commonsense understanding of real-world actions, but the data is not formulated for mistake detection. We thus apply several preprocessing steps to the data to create a new Ego4D for Procedural Mistake Detection (Ego4D-PMD) benchmark that includes successful cases and a variety of mistake types for each annotated procedure, outlined below. Example data from Ego4D-PMD is shown in Figure 8.2.

Generating success examples. As discussed above, Ego4D's hand object interaction data is annotated in units of egocentric video clips of individual actions being performed by humans. We can form an example of a successful execution of the procedure by pairing each video clip with its annotated natural language narration of the procedure. Since most VLMs are not optimized to reason over multiple frames and videos, and those that are are still in very early stages, we sample exactly one frame from each video clip. Specifically, as each clip is carefully annotated with a postcondition time for the action, i.e., the time that the action has been completed, we simply sample the video frame at this annotated time and pair it with the text narration.

Generating mistake examples for incomplete procedures. One natural type of mistake a user could make is not finishing a procedure. In addition to postcondition times, each video clip is annotated with a precondition time. Following a similar approach in [62], we can generate a mistake example by sampling a frame at the precondition time and pairing it with the video clip's narration text. We expect that by doing this, the sampled frame will show the procedure at an incomplete state, and contain most of the same objects as the success example for the same clip. This poses a difficult challenge of identifying the key physical properties of the scene that would indicate completion.

Generating mistake examples for mismatched verbs and nouns. Mistakes also happen when a user applies the wrong type of action to an object, causing an unexpected state, as well as when a user uses the wrong object or ingredient in a procedure. Following this intuition, we generate additional easier mistake examples from each clip by matching each clip with other clips that have a mismatched verb, noun, or both. While each clip is annotated with verb and noun categories, these categories are coarse-grained, making it impossible to guarantee that two clips with the same verb or noun label actually involve the same verb or noun, thus preventing sampling clips that share the same verb or noun. Instead, we apply the `AllenNLP`[2] semantic role labeler to each narration text to identify the key participants in each procedure. For each clip, we then attempt to sample the postcondition frames from three mismatched clips: one with a mismatched verb (but matching nouns), one with a mismatched noun (but matching verb), and one with a mismatched verb and noun. We then pair these frames with the source clip's narration text, creating mistake examples with varying levels of overlap with the source clip.[3] While it is not always possible to find every such alternative clip for each clip in Ego4D, we can usually find at least one of them.

[2] https://allenai.org/allennlp

[3] We acknowledge Yayuan Li for significant contributions to curating this data.

Transforming narrations into instructions. The narration texts annotated in Ego4D are declarative statements about the actions being performed in each clip. This is not an accurate depiction of typical interactive task guidance and PMD settings, which usually revolve around instructional texts like recipes or guidebooks. As such, we convert each narration, e.g., "Someone washes the lettuce," into imperative form, e.g., "Wash the lettuce," using `spaCy`.[4] Further, some narrations describe procedures that are not suited for comparing physical state changes in text and images, such as social interactions, interactions with animals, interactions with electronic devices, and movements that are impossible to precisely characterize from the narration text (e.g., in "Move plate"). We use the verb and noun category annotations on each clip to filter out such cases.

Ensuring data quality. We perform several additional steps to ensure high-quality mistake detection task instances. First, we remove clips where the precondition and postcondition frames are overly similar (i.e., at least 0.95 cosine similarity). We remove clips that are too dark (i.e., where the mean of all normalized RGB values is less than 0.2). When sampling frames from source clips, we sample several candidates within a small range around the precondition or postcondition timestamp, then select the least blurry candidate by the variance of the images' Laplacian. Some videos in Ego4D show the same action being performed over and over (e.g., "Roll a ball of dough"), which can make it difficult to determine whether the state of the environment shown in a clip is the result of the current procedure or a prior one (given only a single frame). While future work applying video-optimized VLMs for interpretable PMD in long-horizon tasks will need to address this challenge, this adds an unnecessary complexity to an already challenging task for current VLMs. As such, we remove any clips such that the same procedure in the clip has already been performed previously in the video. Lastly, we remove a handful of videos in Ego4D that we notice to be corrupted or significantly distorted.

The statistics of the full Ego4D-PMD dataset are presented in Tables 8.1 and 8.2. As listed there, to avoid spending unnecessary compute on the experiments in this chapter, we randomly sample a subset of 500 validation examples and 2,000 testing examples (evenly split between success and mistake cases) for the forthcoming experiments. The validation data is used in these experiments unless otherwise specified.

[4] https://spacy.io/

Example Type	Train	Validation	(Sample)	Test	(Sample)	Total
Success	56,643	13,058	250	18,057	1000	87,758
Mistake	106,600	25,423	250	34,182	1000	166,205
Mistake *(Incomplete)*	20,261	4,908	51	6,545	194	31,714
Mistake *(Wrong Verb)*	12,171	2,694	31	3,747	108	18,612
Mistake *(Wrong Noun)*	37,065	8,914	87	11,843	344	57,822
Mistake *(Wrong Verb & Noun)*	37,103	8,907	81	12,047	354	58,057

Table 8.1: Distribution of example types in each partition of our proposed Ego4D [85] for Procedural Mistake Detection (Ego4D-PMD) dataset.

Action Label Type	Train	Validation	(Sample)	Test	(Sample)	All
Verbs	83	77	55	78	71	83
Nouns	440	365	151	390	257	487
Verb-Noun Pairs	3,976	2,185	326	2,658	833	5,363

Table 8.2: Distribution of unique verb, noun, and verb-noun pair categories in each partition of our proposed Ego4D [85] for Procedural Mistake Detection (Ego4D-PMD) dataset. Verb and noun categories are annotated for each narration in the Ego4D dataset.

8.2 Evaluating Coherence of Procedural Mistake Detection

In the previous chapters of this thesis, we collected detailed annotations to support evaluating coherence in PCR tasks. In this setting, we instead opt to propose automated coherence metrics for generated questions and answers. We do this for two reasons. First, in mistake detection, there may be multiple valid ways to detect a mistake through asking and answering visual questions, each of which could involve asking different questions and different numbers of questions.[5] Second, in a real-world setting like PMD for task guidance, we argue that automated metrics are better suited for understanding and improving a deployed system than an offline benchmark.

In this section, we describe our application of a fine-tuned NLI model to calculate such metrics, proposing two coherence metrics for interpretable procedural mistake detection: *relevance*, an evaluation metric for generated questions, and *informativeness*, an evaluation metric for predicted answers to those questions. Lastly, we describe how these metrics are incorporated into new definitions of consistency and verifiability for this problem.

[5]For example, in trying to determine the success of the procedure "In a bowl, add the cut cherry tomatoes" from [183], we could reasonably ask one question "Are all the cherry tomatoes in the bowl?" or two questions "Are there cherry tomatoes in the bowl?" and "Are there any cherry tomatoes outside of the bowl?"

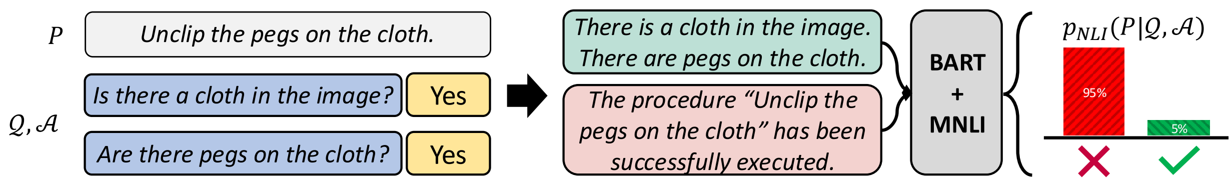


Figure 8.3: Usage of BART [129] fine-tuned on MultiNLI [256] to judge procedural success.

8.2.1 Using NLI Models to Judge Success

NLI, also referred to as textual entailment, is the task of determining whether a hypothesis text must be true given a premise text, and has long been studied in the NLP community [43]. Once thought to be a grand challenge for commonsense reasoning, many human-annotated resources have been compiled for this task, and thus significant progress has occurred [227]. Recent work has successfully leveraged this progress by applying LMs fine-tuned for NLI to improve the competence, confidence, and coherence of LMs for tasks like conversational dialog [255, 63], summarization [206] and visual question answering [224].

As shown in Figure 8.3, LMs fine-tuned for this task can similarly prove useful for measuring the coherence of machine-generated explanations for mistake detection. Formally, given:

- Procedural text P
- A sequence of n binary questions $\mathcal{Q} = \{Q_1, Q_2, \ldots, Q_n\}$ and their answers $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$ (each "No" or "Yes")
- Fine-tuned NLI model f_{NLI} that transforms a premise and hypothesis string into a probability distribution of entailment versus contradiction
- Rephrasing transformation t_{QA} that converts a question Q and answer A into a declarative statement
- Procedure success prompt template t_P

We calculate the NLI model's probability distribution of a success versus a mistake as follows:

$$p_{NLI}(P|\mathcal{Q},\mathcal{A}) = f_{NLI}\big(t_P(P)|f_{QA}(Q_0, A_0), f_{QA}(Q_1, A_1), \ldots, f_{QA}(Q_n, A_n)\big) \quad (8.1)$$

We implement f_{NLI} with BART [129] fine-tuned on the large-scale MultiNLI dataset [256].[6] To implement t_{QA}, we will follow [224] and prompt a foundational LM[7] with 10 demonstrations of transforming a question and answer, e.g., "Is there a bowl on the table?" and "Yes," into a declarative statement, e.g., "There is a bowl on the table." For the procedural text P, we choose a success prompt template t_P "The procedure $\langle P \rangle$ has been successfully executed."

8.2.2 Relevance

A coherent mistake detection decision should be supported by relevant questions about the state of the environment.[8] We propose to measure the *relevance* of a question Q' to the success of a procedure P, given previous questions $\mathcal{Q} = \{Q_1, \ldots, Q_{i-1}\}$ and answers $\mathcal{A} = \{A_1, \ldots, A_{i-1}\}$, as follows:

$$\texttt{Rel}(Q'|P, \mathcal{Q}, \mathcal{A}) = \left| p_{NLI}(P|Q' \cup \mathcal{Q}, \text{“No”} \cup \mathcal{A}) - p_{NLI}(P|Q' \cup \mathcal{Q}, \text{“Yes”} \cup \mathcal{A}) \right| \tag{8.2}$$

This definition for relevance quantifies how much impact the answer to the proposed question Q' can have on the success probability (as estimated by the NLI model). If this probability is estimated to be a similar value for "Yes" and "No" answers, this suggests that Q' would not reveal much pertinent information (beyond what was already gathered in $\mathcal{Q}$ and $\mathcal{A}$) about whether or not the procedure P was successfully executed by the user, and thus the relevance will be low. If the success probabilities vary widely depending on the answer, this suggests that Q' can reveal important new information to help decide whether P was successfully executed by the user.

Comparison with human judgements. To judge whether this relevance metric correlates with human judgements of relevance, we recruited 5 annotators (all English speakers with with conferred or in-progress undergraduate degrees). We presented each annotator with 10 randomly selected VLM-generated questions Q', along with previous questions and answers $\mathcal{Q}$ and $\mathcal{A}$.[9] Annotators were instructed to rate the relevance (i.e., given the previ-

[6] Model weights can be downloaded at https://huggingface.co/facebook/bart-large-mnli.

[7] To conserve GPU memory, we will later choose to use the evaluated VLM's LM backbone to facilitate rephrasing. Prompt details in Appendix C.3.

[8] For example, given a procedure "In a bowl, add the cut cherry tomatoes," the question "Are there tomatoes in the bowl?" is relevant to the success of the procedure, while the questions "Is the bowl blue?" and "Is the person wearing a white shirt?" are less relevant.

[9] Questions to annotate were sampled from the outputs of the best and worst VLM approaches presented later in Table 8.4. More details in Appendix C.1.

ous questions and answers, how helpful could an answer to this question be in determining whether the task was successfully completed) on a scale from 1-5 (least to most relevant). Between the resulting 50 annotations and corresponding automated metrics, we found a moderate Spearman correlation [222] of $\rho = 0.55$ ($p = 0.000029$). This suggests that this automated measure of relevance is indeed correlated with human judgements of relevance.

8.2.3 Informativeness

Beyond relevant questions, a coherent mistake detection decision should also be supported by informativeness answers to those questions. Since a highly relevant question does not guarantee a highly informative answer,[10] and errors made by the VLM in answering questions could unintentionally introduce conflicting information, it is necessary to evaluate the quality of predicted answers in justifying the mistake detection decision. To achieve this, we propose to measure the *informativeness* of a predicted answer A' for a question Q' to the success of a procedure P, given previous questions $\mathcal{Q} = \{Q_1, \ldots, Q_{i-1}\}$ and answers $\mathcal{A} = \{A_1, \ldots, A_{i-1}\}$, as follows:

$$\texttt{Inf}(A'|Q', P, \mathcal{Q}, \mathcal{A}) = 1 - H\big(p_{NLI}(P|Q' \cup \mathcal{Q}, A' \cup \mathcal{A})\big) \tag{8.3}$$

H is the binary entropy of the success probability returned by p_{NLI} from the NLI model, calculated by $H(p) = -p\log_2 p - (1-p)\log_2(1-p), p \in [0,1]$.

This definition for informativeness quantifies how much information the answer to this question gives us toward determining the success of the procedure. As such, if the success probability given this answer A' to Q' is confident, this indicates that A' (along with previous questions and answers $\mathcal{Q}$ and $\mathcal{A}$) are sufficient to determine whether or not the procedure was successfully completed, and thus informativeness will be high. On the other hand, a success probability closer to a uniform distribution suggests that the information gathered thus far still has not made the user's success clear, and thus will yield low informativeness.

Comparison with human judgements. To judge whether this informativeness metric correlates with human judgements of relevance, we recruited 5 annotators (all English speakers with conferred or in-progress undergraduate degrees). We presented each annotator with 10 randomly selected VLM-generated questions Q' and answers A', along with previous

[10]For example, in the procedure "In a bowl, add the cut cherry tomatoes," "Are there tomatoes in the bowl" is a relevant question, but a "Yes" answer to that question does not give us enough information to confirm that the procedure is 100% complete (there could be more *tomatoes* still left outside the bowl). On the other hand, a "No" answer allows us to confidently judge that the procedure is incomplete.

questions and answers $\mathcal{Q}$ and $\mathcal{A}$.[11] Annotators were instructed to rate the relevance (i.e., based on all the information we have, how sure is the annotator about whether the procedure was successfully completed) on a scale from 1-5 (least to most informative). Between the resulting 50 annotations and corresponding automated metrics, we found a weaker Spearman correlation [222] of $\rho = 0.32$ ($p = 0.024$). Interestingly, if we multiply the automated informativeness metrics by the relevance for Q', $\mathcal{Q}$, and $\mathcal{A}$, we find a stronger Spearman correlation of $\rho = 0.50$ ($p = 0.00022$). This suggests that while informativeness does have a relationship with human judgements, when multiplying it by relevance this relationship is stronger and more significant. This might be because the concepts of relevance and informativeness are themselves related. Intuitively, in most cases, a relevant question should be informative, and an irrelevant question should be uninformative. Meanwhile, fine-tuned NLI models could theoretically score an answer to an irrelevant question as informative, and vice versa. When proposing multi-tiered coherence metrics in Chapter 8.2.4, we incorporate the inductive bias reflected in these human judgements by multiplying informativeness by relevance. Later, in Chapter 8.7, we visualize the full distribution of relevance and informativeness on our evaluation data to better understand this issue.

8.2.4 Accuracy, Consistency, and Verifiability

Using these metrics, we next propose a multi-tiered evaluation of accuracy, consistency, and verifiability in line with the previous work in this thesis. Formally, this evaluation requires the following information:

- Task data instance $D = \{P, F, y\}$ consisting of a procedural text P, a video frame F, and a binary ground truth label y^* for whether a mistake has occurred in the execution of P (1 if mistake, 0 if success)
- Sequence of n generated questions $\mathcal{Q} = \{Q_1, Q_2, \ldots, Q_n\}$, their predicted answers $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$ based on F, and likelihoods for those answers $\mathcal{L} = \{l_1, l_2, \ldots, l_n\}$ (derived from the softmax of evaluated VLM's likelihoods for "Yes" and "No")
- Sequence of m ($m \leq n$) high-confidence answer indices $\mathcal{C}$, such that for all $i \in \mathcal{C}$, $l_i \geq s$, where s is a minimum answer sureness threshold
- Predicted binary mistake detection decision y (1 if mistake, 0 if success)

[11] Questions to annotate were sampled from the outputs of the best and worst VLM approaches presented later in Table 8.4. More details in Appendix C.1.

For each instance D, *accuracy* is defined as the traditional binary mistake detection accuracy:

$$\texttt{Acc}(D, y) = \begin{cases} 1 & y = y^* \\ 0 & \text{else} \end{cases} \tag{8.4}$$

For any sequence of items $\mathcal{S} = \{S_1, \ldots, S_{|\mathcal{S}|}\}$ and sequence of indices $\mathcal{I}$, let $\mathcal{S}^{\mathcal{I}}$ be the subsequence of elements S_i for all $i \in \mathcal{I}$. Now, relevance measures whether a generated question can yield new information consistent with a decision about the success of the procedure at hand. As such, we define *consistency* by the mean marginal relevance for each question in $\mathcal{Q}$:

$$\texttt{Con}(D, E, y) = \begin{cases} \frac{1}{n} \sum_{i=0}^{n} \texttt{Rel}(Q_i | P, \mathcal{Q}^{\mathcal{C}} \cap \mathcal{Q}^{\{1,\ldots,i\}}, \mathcal{A}^{\mathcal{C}} \cap \mathcal{A}^{\{1,\ldots,i\}}) & y = y^* \\ 0 & \text{else} \end{cases} \tag{8.5}$$

In other words, for accurate mistake detection predictions, consistency is the average relevance for each question with respect to all previous confidence-filtered questions and answers. As shown, to ensure consistency is strictly less than or equal to accuracy and that we do not reward VLMs for generating questions that do not help them arrive at the correct decisions, it is set to zero for instances with incorrect mistake detection decisions.

Informativeness judges whether the model's predicted answers provide enough information about the state of the environment to fully verify a decision about the success of a procedure. To additionally measure whether the decision the answers point to match the ground truth decision for the task instance, *verifiability* is defined by the maximum reference-adjusted informativeness of confidence-filtered answers:

$$\texttt{Ver}(D, E, y) =$$
$$\begin{cases} \texttt{Con}(D, E, y) \max\big(\max_{i \in \mathcal{C}} \texttt{Inf}^*(A_i | Q_i, P, \mathcal{Q}^{\mathcal{C}} \cap \mathcal{Q}^{\{1,\ldots,i\}}, \mathcal{A}^{\mathcal{C}} \cap \mathcal{A}^{\{1,\ldots,i\}}, y^*), 0\big) & y = y^* \\ 0 & \text{else} \end{cases} \tag{8.6}$$

Relevance is included as a coefficient for two reasons: first, to ensure verifiability is strictly less than or equal to consistency, and second, because we found earlier that informativeness has a stronger correlation with human judgements when multiplied by relevance. Verifiability is again set to zero for instances with incorrect mistake detection decisions. The reference-adjusted informativeness $\texttt{Inf}^*(A_i | Q_i, P, \mathcal{Q}^{\mathcal{C}} \cap \mathcal{Q}^{\{1,\ldots,i\}}, \mathcal{A}^{\mathcal{C}} \cap \mathcal{A}^{\{1,\ldots,i\}}, y^*)$ is equiva-

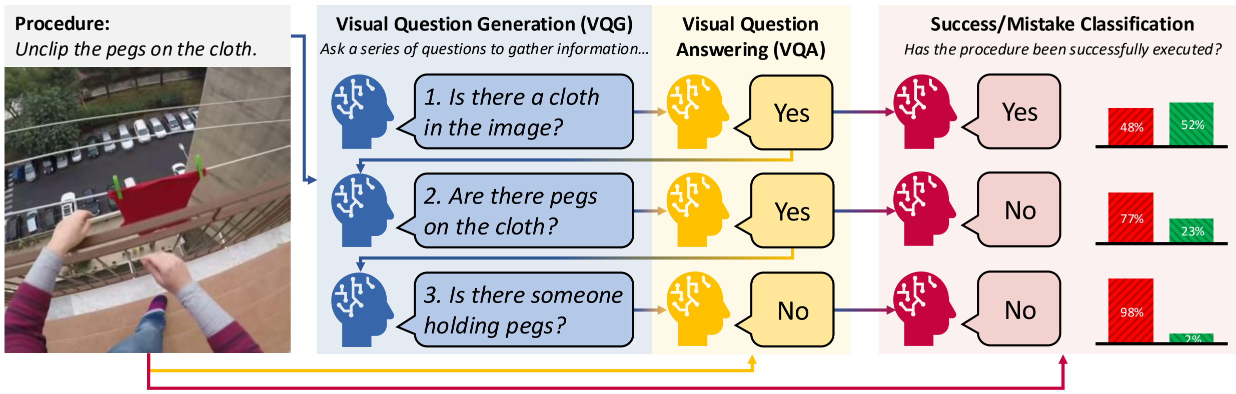


Figure 8.4: Overview of self-reflective explanatory dialog between VLM and itself to facilitate interpretable procedural mistake detection. Here, reasoning occurs in a bottom-up direction (i.e., mistake detection decision occurs after explanation), but as discussed in Section 8.4, this can also be formulated in a top-down manner with the mistake detection occurring before and conditioning explanation.

lent to $\texttt{Inf}(A_i|Q_i, P, \mathcal{Q}^{\mathcal{C}} \cap \mathcal{Q}^{\{1,\ldots,i\}}, \mathcal{A}^{\mathcal{C}} \cap \mathcal{A}^{\{1,\ldots,i\}}, y^*)$ if and only if the most likely outcome in $p_{NLI}(P|\mathcal{Q}^{\mathcal{C}}, \mathcal{A}^{\mathcal{C}})$ is equivalent to y^*, i.e., the NLI model's probability of success for answers and questions through A_i and Q_i agrees with the ground truth label for the success of the procedure. Otherwise, it is negated. This allows us to incorporate whether the information gathered in generated questions and answers is actually leading to the correct answer. In other words, for accurate mistake detection predictions, verifiability is the consistency-weighted maximum informativeness of any confidence-filtered answer found to be indicative of the correct mistake detection decision, with respect to previous confidence-filtered questions and answers. It is worth noting that this notion of verifiability is slightly different than the one defined in Chapter 4, since we do not have ground truth labels for answers to questions generated by VLMs. As such, rather than directly capturing the correctness of physical state information extracted from the context, verifiability evaluates the potential usefulness of the information extracted from the context.

Accuracy, consistency, and verifiability are averaged over an entire PMD dataset and system predictions to produce global evaluation metrics for interpretable PMD systems.

8.3 Applying VLMs to Interpretable Procedural Mistake Detection

As shown in Figure 8.4, in applying VLMs to interpretable PMD, we formulate PCR as a self-reflective dialog between the VLM and itself.[12] To facilitate this dialog, our experiments apply InstructBLIP[13] [44] and LLaVA 1.5-7B[14] [140]. Both of these VLMs are based on Vicuna-7B [277], an instruction-tuned version of LLaMA 2-7B [237], but apply different architectures, training datasets, and training strategies to integrate vision into the model. Neither VLM was trained on Ego4D data. To conserve GPU memory, VLM weights are 4-bit quantized at inference time. To provide sufficient information for interpretable PMD, the dialog must consist of several steps (in no particular order): proposing questions to ask about the procedure at hand (visual question generation or VQG), answering them (visual question answering or VQA), and making a final decision about whether or not the procedure has been successfully completed in the given video frame (SuccessVQA [62]). This structure goes beyond past approaches for PMD with foundational VLMs, which at worst only elicited the final decision [62], and at best used procedure-agnostic prompts to caption images before making a final decision, nonetheless ignoring this information in quantitative evaluations [16].

8.3.1 Visual Question Generation

To facilitate VQG, we prompt the VLM to generate a series of questions given the text description of the procedure. Each question is generated based on the full dialog history with previous questions and their answers, enabling deductive reasoning about the status of the procedure. To ensure the VLM generates yes-no questions, we constrain generation during decoding to enforce that each generated text begins with a word that can signal a yes-no question,[15] does not include the word *or*, and ends with a question mark. To encourage logical questions while ensuring variety, we apply greedy beam search decoding with $k = 8$ beams, returning the top 4 candidate questions.[16] Out of these candidates, we remove any that are exactly the same as previously generated questions, then select the most likely candidate based on the LM's log-likelihoods. Question generation is not conditioned on the video frame,

[12] Prompt templates provided in Appendix C.2.

[13] Model weights can be downloaded at https://huggingface.co/Salesforce/instructblip-vicuna-7b.

[14] Model weights can be downloaded at https://huggingface.co/liuhaotian/llava-v1.5-7b.

[15] Specifically, questions must begin with *is*, *does*, or *has*, along with all plural and past tense forms of these verbs.

[16] Due to generation constraints, it is often the case that the VLM does not successfully generate all 8 candidates.

as we found significant performance degradation when VLMs were conditioned on the video frame while generating questions, often leading to completely nonsensical questions, e.g., "Is is is is is is?"

8.3.2 Visual Question Answering

To facilitate VQA, each generated question is immediately answered by the VLM, conditioned on the video frame. To produce the answer, we extract the resulting logits from the forward pass of the VLM for both the *Yes* and *No* tokens, then apply a softmax over them to form a probability distribution for the binary answer. If the probability of the most likely answer exceeds the minimum answer sureness threshold introduced earlier in Chapter 8.2.4, we append it to the dialog history; otherwise, we append the string "Unsure." Also discussed there, these unsure cases are excluded from the previous questions and answers in calculating relevance and informativeness in coherence evaluations, and excluded from the example-level calculation of verifiability. In all forthcoming experiments, we set the sureness threshold to $s = 60\%$. It is important to note that we exclude the dialog history from the context during VQA, as we again observed significant performance degradation when VLMs answered visual questions in the context of a longer dialog. This was especially prominent when several similar questions were generated and answered in a dialog, which often caused the VLM to creep from being initially unsure about the answer to being confidently wrong.

8.3.3 Success or Mistake Determination

Given the video frame and procedural text, we also need to elicit a prediction from the VLM about whether the procedure has been successfully executed in the frame, and we follow [62] in referring to this sub-task as SuccessVQA. To facilitate SuccessVQA, we simply append a question to the dialog history directly asking this. Both the video frame and any questions and answers generated before this step are included. Similarly to VQA, the logits of the *Yes* and *No* tokens are used to produce a probability distribution over a success or mistake decision. The final decision is then determined by a mistake confidence threshold τ, which is varied in the forthcoming experiments.

8.4 Impact of Reasoning Direction

In Chapter 6, we took inspiration from human cognitive psychology to investigate how the direction of the steps of PCR impacted accuracy and coherence in foundational LMs. There, we found that a top-down reasoning strategy using quick decisions about the end classification task helped condition foundational LMs to better predict physical states of the environment, a much more complicated task. In this section, we will similarly investigate how the direction of reasoning impacts performance in this problem.

In the top-down approach, we first prompt the VLM to judge whether the procedure was successfully executed before generating questions and answers to justify the decision. Meanwhile, in the bottom-up approach, we do the converse and first prompt the VLM to generate questions and answers, then ultimately use that information to decide whether the procedure was successfully executed. For both approaches, we generate $n = 10$ questions and answers, and measure the percentage accuracy, consistency, and verifiability.[17] We select the best result across 99 thresholds $\tau \in \{0.01, 0.02, 0.03, \ldots, 0.97, 0.98, 0.99\}$ based on the validation set accuracy.

In the bottom-up approach, the SuccessVQA step is conditioned on generated questions and answers. As such, to prevent over-generating noisy information, which degrades the accuracy and thus the consistency and verifiability, we implement an early stopping mechanism for this approach. This mechanism inserts a SuccessVQA check after each generated question and answer, and uses the resulting success likelihood to determine whether to stop generating questions. Generation stops early (i.e., before $n = 10$ questions have been generated) if one of the following conditions are met:

- The likelihood of success *stabilizes*, changing by less than 10% for two consecutive iterations of VQG, VQA, and SuccessVQA
- The likelihood of success becomes *highly confident*, subceeding 5% or exceeding 95%

Such early stopping is not possible in the top-down approach, since the probability of success is only calculated once before generating questions and answers. To ensure a relatively fair comparison, we set the number of iterations n for all task instances in the top-down approach as close as possible to the mean number of iterations taken by the bottom-up approach for each VLM.

8.4.1 Multi-Tiered Coherence Evaluation Results

We compare the performance of top-down and bottom-up approaches in Figure 8.5 and Table 8.3, observing mixed results. Across most thresholds, the top-down approach achieves lower false positive and false negative rates. At the selected threshold, the top-down approach achieves the highest accuracy, while the bottom-up approach achieves higher verifiability. On

[17] Verifiability relies on informativeness, which is a measure of information in bits that varies between 0 and 1, so percentage verifiability simply refers to the proportion of maximum information.

VLM	Direction	τ	**Accuracy** (%)	**Consistency** (%)	**Verifiability** (%)
InstructBLIP	Top-Down ($n = 4$)	0.49	59.2	9.5	2.8
InstructBLIP	Bottom-Up	0.40	**63.4**	**10.0**	**4.0**
LLaVA	Top-Down ($n = 3$)	0.58	**65.0**	**21.8**	6.2
LLaVA	Bottom-Up	0.81	60.6	18.1	**10.0**

Table 8.3: VLM multi-tiered coherence evaluation results on Ego4D-PMD validation set for top-down and bottom-up reasoning approaches. Each result is reported for the maximum-accuracy mistake confidence threshold τ, and the top-down result is reported for a uniform number of iterations n for each task instance based on the mean number of iterations taken by the bottom-up approach.

the one hand, this suggests that forcing the VLM to first generate an explanation through iterative VQG and VQA in the bottom-up approach hinders it in making an accurate mistake detection decision. On the other hand, this approach delivers more verifiable explanations, suggesting that first making a mistake detection decision also hinders later VQG and VQA steps.

Qualitatively, the lower verifiability in the top-down approach seems to happen because conditioning VLMs with a high-level question about success influences them to generate overly cumbersome and high-level questions afterward. For example, the following questions are generated by LLaVA using the top-down approach:

- “Is the person working on the procedure ’Pick up the plastic bowls in the cabinet’?”
- “Is the person wearing a lab coat while working on the procedure ’Put liquid in the paper’?”
- “Is the person working on the procedure ’Drop a glass on the countertop’ still in the process of dropping the glass?”

Overall, for this reason, and because the accuracy of the bottom-up approach can possibly benefit from improvements in generated questions and answers, we choose to use the bottom-up approach in forthcoming experiments.

8.5 Encouraging Coherence in Question Selection

While the previous results selected candidate questions generated through beam search by their likelihood, an alternative approach could be to rerank the candidates using the reference-free coherence metrics introduced in Chapters 8.2.2 and 8.2.3. This could possibly

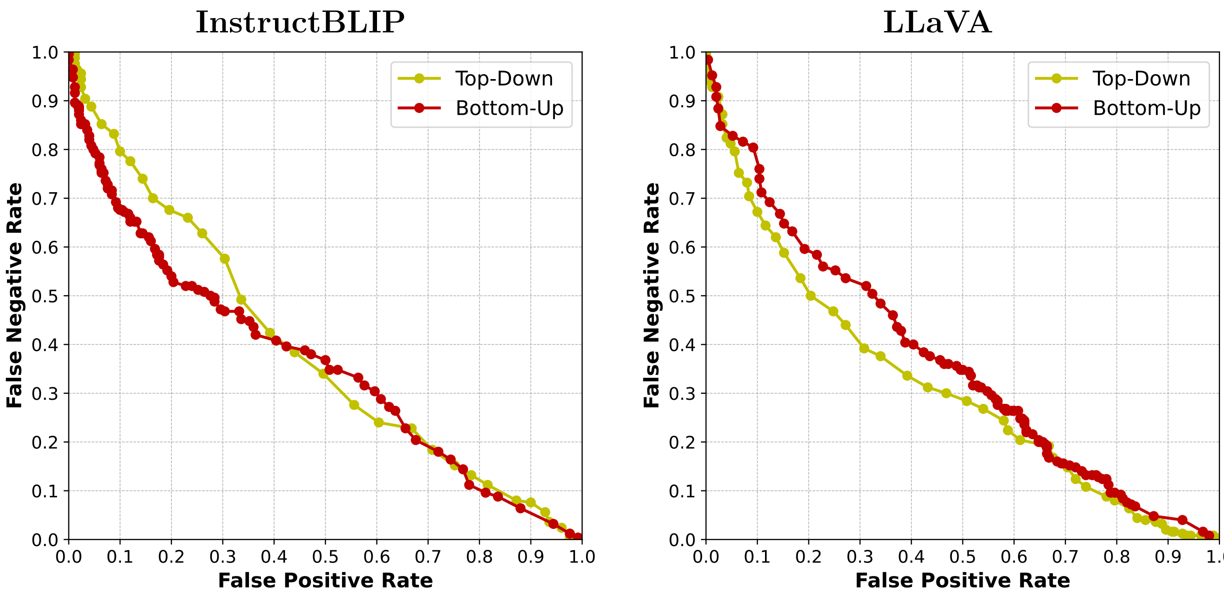


Figure 8.5: Mistake detection error tradeoff (DET) curves for VLMs applied to the Ego4D-PMD validation set with top-down and bottom-up reasoning approaches. Here, a "positive" refers to a mistake case, while a "negative" refers to a successful case.

encourage the selection of questions that are more likely to bring in new, salient, and helpful information.

Furthermore, while our current setting is entirely zero-shot, in Chapter 6, we showed how the in-context learning capability of foundational LMs enables them to acquire coherent PCR from just a few in-context demonstrations. As such, we will explore whether supplementing existing candidate questions from each iteration with additional questions generated through a similar in-context learning approach can support more accurate or coherent results from the VLM's explanatory dialog.

For the remainder of this section, we will introduce two approaches we use to augment the candidate question pool for more coherent candidates: *coherence-based reranking* and *candidate generation through in-context learning*. We will then present the multi-tiered evaluation results for these approaches, and analyze how the impact of these two approaches extends beyond accuracy and coherence into efficiency of generating explanations.

8.5.1 Coherence-Based Candidate Question Selection

For the bottom-up reasoning approach, we implement a coherence-based candidate question reranking approach as follows. Given a set of question candidates $\hat{\mathcal{Q}}$ for procedural text P along with previous confidence-filtered questions $\mathcal{Q}$ and answers $\mathcal{A}$, we can select the best question Q^* by maximizing the product of relevance and potential informativeness:

$$Q^* = \arg\max_{Q \in \hat{\mathcal{Q}}} \texttt{Rel}(Q|P, \mathcal{Q}, \mathcal{A}) \max_{A \in \{\text{"Yes"}, \text{"No"}\}} \texttt{Inf}(A|Q, P, \mathcal{Q}, \mathcal{A}) \tag{8.7}$$

This ranking prioritizes well-rounded questions which could yield both the most impactful information for the final determination of success, and the most confidence in the final determination of mistake or success. Q^* is then concatenated to the dialog history and answered by the VLM, in line with the previously presented approaches.

8.5.2 Improving Candidate Questions with In-Context Learning

Applying in-context learning in interpretable PMD is not straightforward, as each bottom-up explanatory dialog includes an image, a variable number of iterations of VQG and VQA, and a final mistake detection decision. All these complexities make interpretable PMD an impractical task to guide VLMs through with in-context learning. Instead, we propose to apply in-context learning to improve the text-based VQG step by providing examples of human-written questions.

As shown in Figure 8.6, we achieve this by manually annotating 20 procedures from the Ego4D training data with 3 reasonable questions one could ask about a given procedure to judge its success.[18] On average, these human-written example questions achieve 53.9% relevance and 83.2% maximum informativeness (i.e., for either a *yes* or *no* answer). We prompt the VLM (without any input image) with these example procedures and questions, the current procedure at hand, and the previous 2 questions proposed by the VLM (as available) to incorporate information the VLM already collected. We then generate 4 additional candidate questions, again using beam search with $k = 8$ beams. To minimize the impact of ordering, in-context examples are randomly shuffled in every prompt.

8.5.3 Multi-Tiered Coherence Evaluation Results

The evaluation results from both above approaches are presented in Figure 8.7 and Table 8.4.

Coherence-based question reranking. Unsurprisingly, we find that introducing coherence-based reranking of candidates sharply improves the consistency and verifiability of resulting explanations. Interestingly, though, this also sharply improves the accuracy of the approach across most mistake confidence thresholds to as high as 63.6%. This shows that when more relevant and informative information is extracted from the scene, VLMs can

[18] Procedures and questions listed in Appendix C.4.

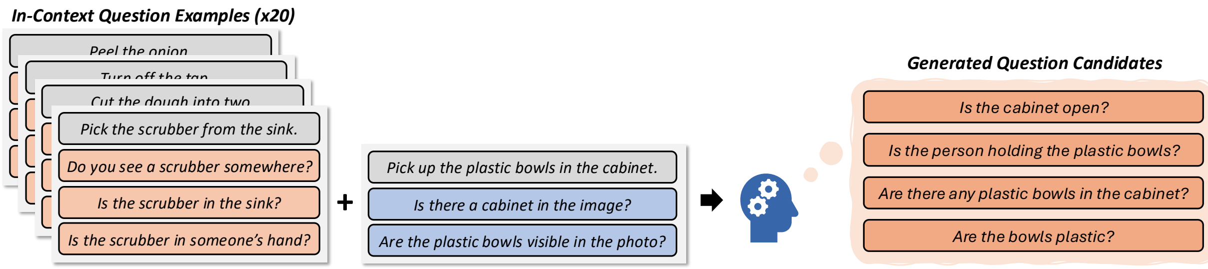


Figure 8.6: To bolster visual question generation (VQG), we apply in-context learning from 20 sets of 3 human-written questions for procedures. Ideally, this enables the generation of more coherent questions.

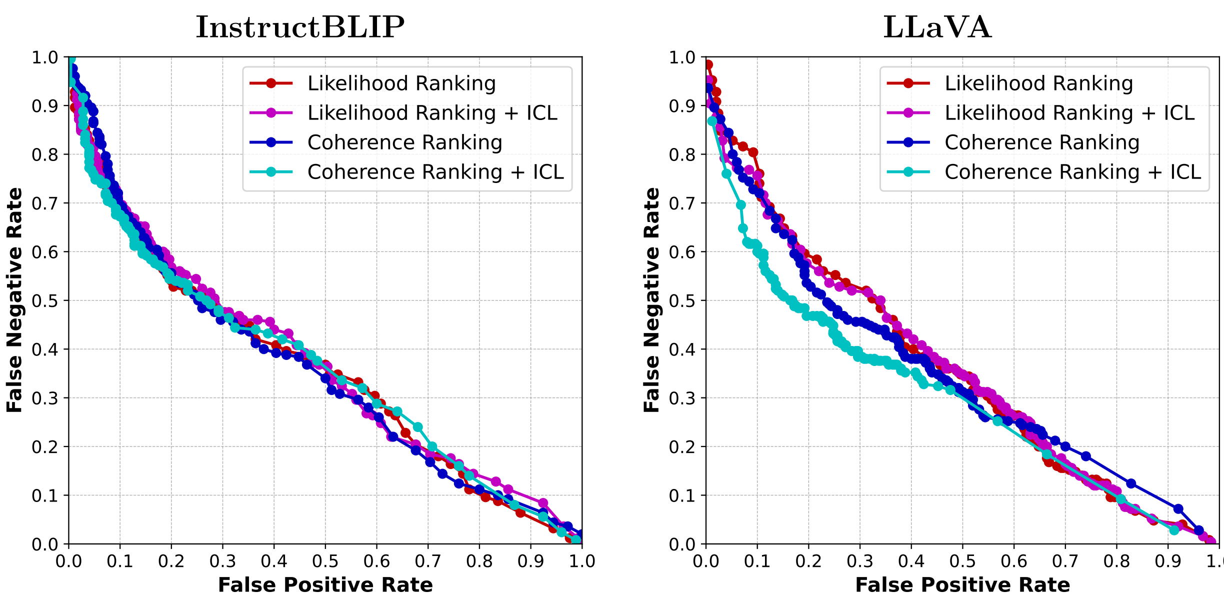


Figure 8.7: Mistake detection error tradeoff (DET) curves for VLMs applied to the Ego4D-PMD validation set with likelihood- and coherence-based candidate question selection approaches, with optional supplementary candidates generated through in-context learning (ICL).

VLM	Ranking	ICL	τ	**Accuracy** (%)	**Consistency** (%)	**Verifiability** (%)
InstructBLIP	Likelihood	✗	0.40	**63.4**	10.0	4.0
InstructBLIP	Likelihood	✓	0.40	61.6	10.7	4.4
InstructBLIP	Coherence	✗	0.30	62.8	15.6	7.7
InstructBLIP	Coherence	✓	0.32	63.0	**21.3**	**11.3**
LLaVA	Likelihood	✗	0.81	60.6	18.1	10.0
LLaVA	Likelihood	✓	0.83	61.4	19.1	10.7
LLaVA	Coherence	✗	0.69	63.6	33.5	22.1
LLaVA	Coherence	✓	0.72	**67.0**	**46.1**	**37.5**

Table 8.4: VLM multi-tiered coherence evaluation results on Ego4D-PMD validation set for likelihood- and coherence-based candidate question selection approaches, with optional supplementary candidates generated through in-context learning (ICL). Each result is reported for the maximum-accuracy mistake confidence threshold τ.

make better final mistake detection decisions. Furthermore, questions that a VLM judges to be most likely are not naturally the most coherent, in line with earlier findings with foundational LMs.

In-context learning for candidate question generation. Supplementing candidate questions with those generated through in-context learning from human-written examples also exhibits a sharp improvement in accuracy, consistency, and verifiability to their best respective values of 67.0%, 46.1%, and 37.5%. Notably, accuracy now exceeds the level of top-down reasoning, showing that conditioning mistake detection decisions with coherent questions and answers (rather than the converse, as done in the top-down approach) can promote more accurate mistake detection.

While incorporating these additional candidates is quite beneficial when using coherence-based ranking, likelihood-based ranking sees no significant changes in accuracy or coherence. This suggests that coherence-based ranking is better suited to identify potentially helpful questions even if those questions come from another context. This is unsurprising, though, as the beam search that generates the initial set of candidate questions from the dialog context also relies on the VLM likelihood. While it is possible that the beam search misses high-likelihood candidate questions, it is unlikely that the VLM would score questions from a completely different context in a similar range to those coming from those initial candidates.

To shed more light on where selected candidate questions come from in each approach, we visualize the distribution of question sources in Figure 8.8. As expected, candidates generated with in-context learning are only rarely selected in the likelihood-based ranking, amounting to about 24.2% of VQG iterations for InstructBLIP, and 9.4% of VQG iterations for LLaVA. On the other hand, they are selected more frequently in the coherence-based

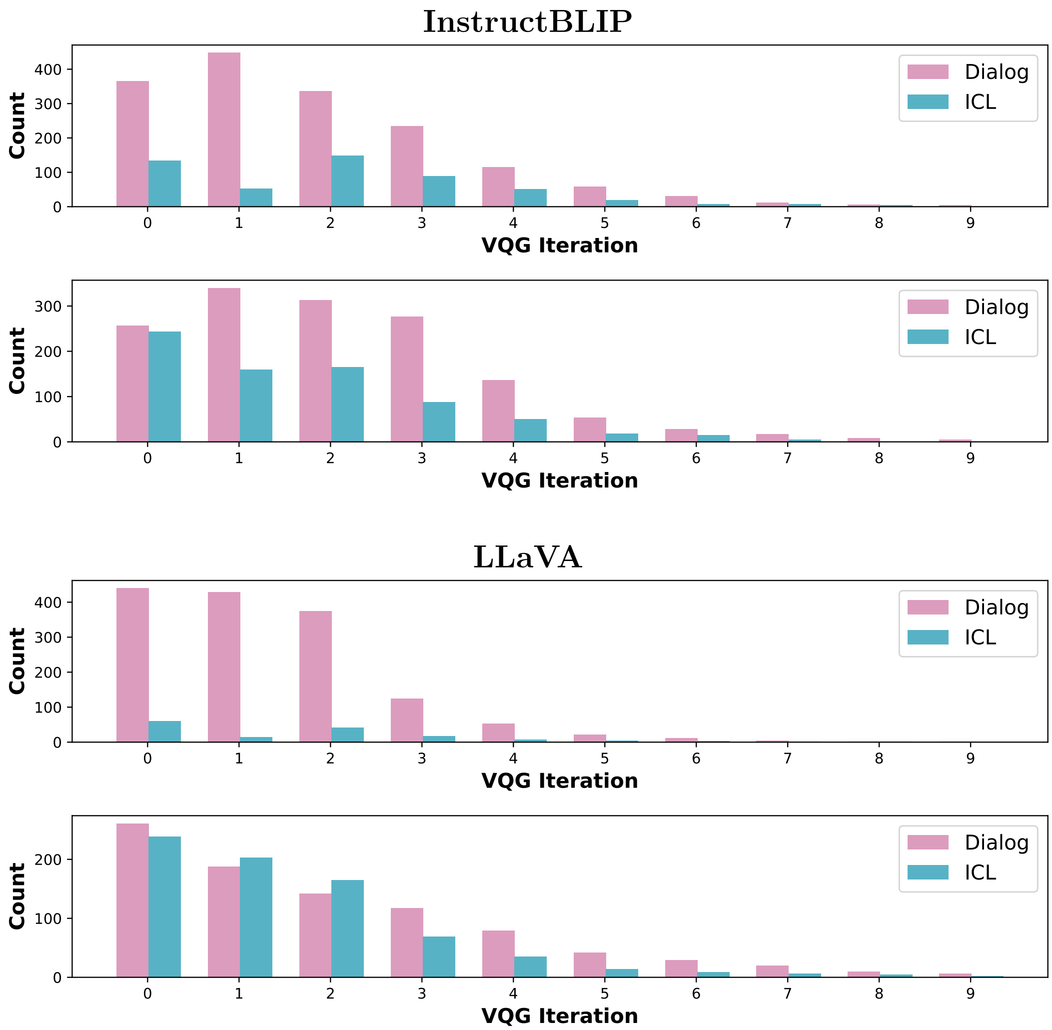


Figure 8.8: Histograms of VLMs' selected question sources, either explanatory dialog context or in-context learning (ICL) examples, by visual question generation (VQG) iteration for likelihood-based question selection (top) and coherence-based question selection (bottom).

ranking, amounting to about 34.1% of VQG iterations for InstructBLIP, and 45.5% of VQG iterations for LLaVA. Interestingly, in-context learning candidates are more dominant in earlier iterations, while candidates generated based on the dialog context are relatively more common in later iterations. This may suggest that after selecting a few questions from in-context learning in earlier iterations, the VLM is able to utilize them to generate better questions from the dialog context in later iterations. Alternatively, this could suggest that candidates from in-context learning have limited variety, and thus are less likely to be selected in later turns to avoid redundant questions or information.

8.5.4 Explanation Efficiency and Confidence Analysis

In addition to improving the accuracy and coherence of PMD in VLMs, we may wonder how these approaches impact the efficiency of the explanatory dialog and confidence in the VLM's predictions. Such factors can be important in the practical utility and trustworthiness of systems for interpretable PMD. We measure them in several ways:

1. *Number of iterations* of VQG and VQA that occur before the dialog terminates
2. *Information gain* from the dialog toward the likelihood of success in SuccessVQA steps (in bits)
3. Average *VQA confidence*, i.e., maximum likelihood for answers to generated questions
4. *Expected calibration error (ECE)* [171] of the likelihood of success, calculated over 10 bins of likelihood ranges
5. *Area under the risk-coverage (AURC) curve* [59] for the likelihood of success[19]

Together, these measurements provide an impression of how efficient, confident, and reliable each approach is in interpetable PMD. The results of this analysis are shown in Table 8.5. From the results, we see that while all combinations of approaches take similar numbers of iterations on average, the information gain from approaches using coherence-based question selection and candidates generated from in-context learning is significantly higher. While the earlier evaluation already showed that questions generated from these approaches are theoretically more relevant and informative, this demonstrates that in practice, these improvements to VQG enable VLMs to use approximately the same number of questions to gather more useful information to determine whether the procedure has been successfully completed.

Further, while the average VQA likelihood and ECE are similar for all approaches, the AURC is significantly lower for the approach using coherence-based question selection and candidates generated from in-context learning. This suggests that we can restrict VLM predictions under this approach to specific risk tolerances to maximize reliabilty of decisions that the VLM makes while maintaining a relatively higher coverage of task instances that the VLM can make predictions for.

[19] AURC is calculated in a selective prediction setting, where the VLM abstains from making a mistake detection decision if the maximum likelihood from the SuccessVQA step does not meet a threshold. For a given threshold, we calculate the *risk* as the error rate for those predictions the VLM made, and *coverage* as the proportion of examples in the dataset that the VLM made a prediction for. AURC is then calculated from a comprehensive set of thresholds $\{0.0, 0.01, 0.02, \ldots, 0.98, 0.99, 1.0\}$.

VLM	Ranking	ICL	# Iter. ↓	Inf. Gain ↑	VQA Conf. ↑ (%)	ECE ↓ (%)	AURC ↓ (%)
InstructBLIP	Likelihood	✗	**4.13**	0.348	75.9	**18.0**	**35.4**
InstructBLIP	Likelihood	✓	4.25	0.366	**76.2**	19.3	36.9
InstructBLIP	Coherence	✗	4.17	0.359	75.9	18.9	38.6
InstructBLIP	Coherence	✓	4.36	**0.460**	75.3	22.4	36.0
LLaVA	Likelihood	✗	3.27	0.383	77.5	**23.5**	33.4
LLaVA	Likelihood	✓	**3.22**	0.403	**77.6**	24.8	31.6
LLaVA	Coherence	✗	3.37	0.496	77.4	23.8	32.9
LLaVA	Coherence	✓	3.28	**0.650**	76.6	24.4	**25.3**

Table 8.5: VLM explanatory dialog efficiency and decision confidence evaluation on Ego4D-PMD validation set for likelihood- and coherence-based candidate question selection approaches, with optional supplementary candidates generated through in-context learning (ICL).

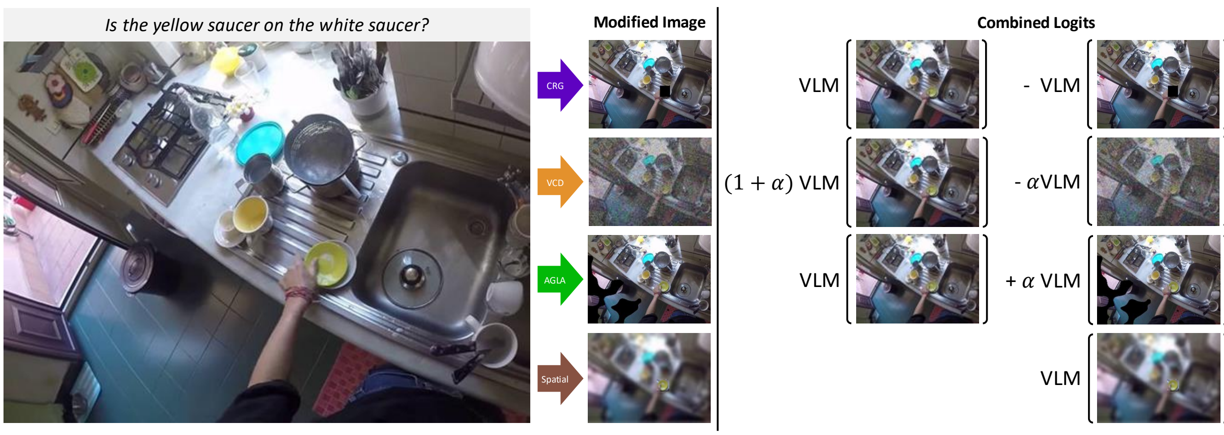


Figure 8.9: Comparison of evaluated visual hallucination mitigation strategies: Contrastive Region Guidance (CRG) [245], Visual Contrastive Decoding (VCD) [127], Assembly of Global and Local Attention (AGLA) [8], and our proposed question-aware spatial filter.

8.6 Impact of Visual Hallucination Mitigation Strategies

Given the pervasive problems of visual hallucination, illusion, and other incoherent visual processing behaviors in VLMs, recent work has proposed various training-free methods to mitigate these problems [245, 127, 8]. As shown in Figure 8.9, these methods typically modify images before inputting them to the VLM at inference time, then may combine the resulting logits with those from prompting the VLM with the unmodified image. In this section, we first compare these methods, including a method we design for targeting spatial information in questions, then investigate how such methods impact the accuracy and coherence of interpretable PMD.

Contrastive Region Guidance. Contrastive Region Guidance (CRG) [245] identifies objects mentioned in visual questions, then modifies the input image by using foundational open-vocabulary object detection models to mask these objects out. The VLM then answers the input question twice, once with the original image and once with the modified image. To form the final logits from which a yes-no prediction can be extracted, the logits for the modified image are subtracted from those of the original image. When replicating this approach, if no target objects are identified, the logits for the original image are directly returned. Ideally, CRG prevents distracting background information from factoring into the VLM's prediction.

Visual Contrastive Decoding. Visual Contrastive Decoding (VCD) [127] modifies the input image by applying diffusion noise. The VLM answers the input question for both the original and modified image, yielding original image logits L_o and modified image logits L_m. These logits are combined using an equation $(1+\alpha)L_o - \alpha L_m$, where we assign $\alpha = 1.0$ following the original paper. From these combined logits, we then extract a yes-no prediction. Ideally, VCD prevents illusory or spurious information in images from factoring into the VLM's prediction.

Assembly of Global and Local Attention. Assembly of Global and Local Attention (AGLA) [8] modifies the input image by applying a lighter-weight image-text matching module across patches of the image to compare each patch to the input question. Regions of the image judged to be irrelevant to the question are masked out. The VLM answers the input question for both the original and modified image, yielding original image logits L_o and modified image logits L_m. These logits are combined using an equation $L_o + \alpha L_m$, where we assign $\alpha = 2.0$ following the original paper. Unlike the original paper, we remove a step that zeroed out all but the highest-probability tokens in the original frame logits, as this often impacted the logits for "Yes" and "No," which are required in our evaluation. Ideally, AGLA enables the VLM to attend to only the most question-relevant information in images.

Question-aware spatial filter. We also attempt to implement such a hallucination mitigation approach in this thesis. We are motivated by the lack of prompt-specific visual attention in most VLMs. This stands in contrast to humans, who are capable of leveraging language guidance to adapt and control their attention to the most relevant visual information for the task at hand [160]. This capability seems particularly important for this task, as the questions generated by VLMs often target only specific sub-regions of an image, which are usually indicated by asking about the state of a specific object (e.g., in "Are the tomatoes sliced?"), or including spatial relations in questions (e.g., in "Are the tomatoes in the bowl?").

To investigate whether we can use these cues from generated questions to better focus VLMs on the most relevant information, we implement a rule-based question-aware spatial filter. This filter works by first identifying noun phrases and/or spatial prepositional phrases in the question using `spaCy`,[20] accounting for negation where appropriate (e.g., in "Are there any tomatoes that are not in the bowl?"). We use this information to then determine which regions of the image should be attended to, which are expressed in a logical form, e.g., ¬`bowl` for "Are there any tomatoes that are not in the bowl?". We then apply OWL-ViT [162] to the input image to extract bounding boxes for the identified object(s) in this logical form, and apply a Gaussian blur (kernel size 55) to the appropriate regions of an image. The VLM then answers the input question based on this modified image instead of the original image, and the resulting logits are used to extract a yes-no answer. Similarly to AGLA, this spatial approach ideally enables VLMs to focus only on the most relevant regions of an image to answer questions.

8.6.1 Multi-Tiered Coherence Evaluation Results

We augment our best approach thus far, bottom-up reasoning with coherence-based question ranking supplemented with candidates generated through in-context learning, with each of the above visual hallucination mitigation strategies. The evaluation results are presented in Figure 8.10 and Table 8.6, and interestingly show that these approaches have mostly small impacts on the accuracy and coherence of PMD in VLMs. We observe that CRG yields some of the lowest accuracy, consistency, and verifiability for both VLMs, causing performance to be slightly worse than not using any hallucination mitigation strategy. VCD improves the accuracy of InstructBLIP slightly, and yields the highest consistency and verifiability for LLaVA. AGLA yields some of the highest consistency and verifiability for InstructBLIP, and remarkably low mistake confidence thresholds for both VLMs. Lastly, our spatial filtering strategy does not appear significantly more or less beneficial than any of the comparison approaches.

8.6.2 Explanation Efficiency and Confidence Analysis

While these hallucination mitigation strategies have minimal impact on accuracy and coherence, we next test whether there is any advantage in the efficiency of explanation or the confidence of predictions. We measure these aspects using the same values as we did earlier

[20] https://spacy.io/

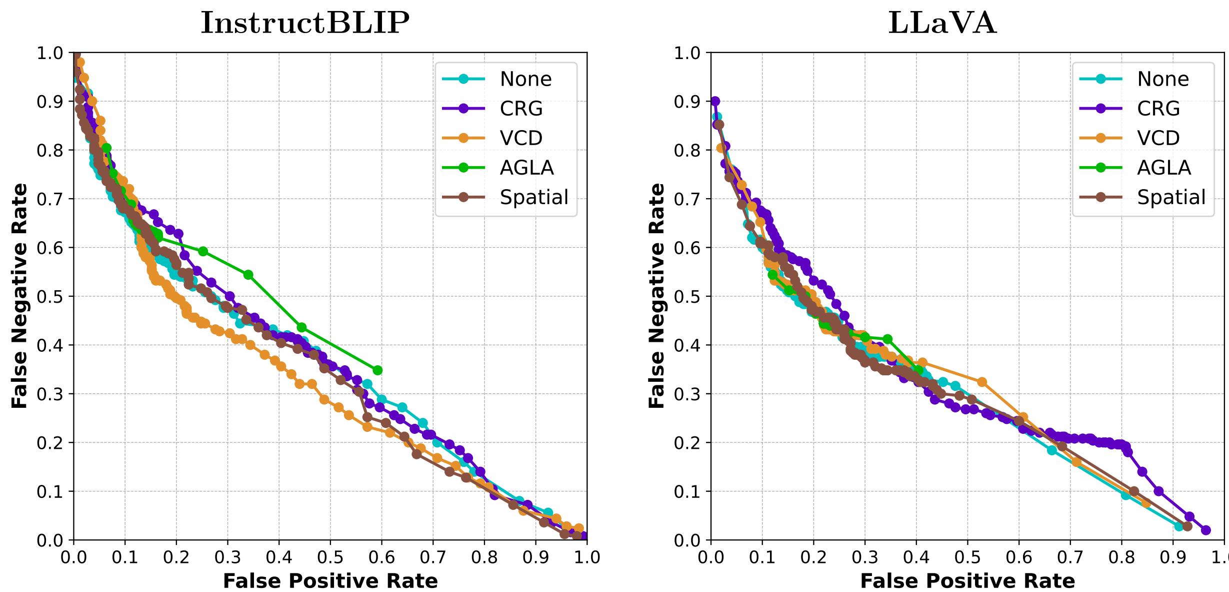


Figure 8.10: Mistake detection error tradeoff (DET) curves for VLMs augmented with various training-free visual hallucination mitigation methods introduced in Chapter 8.6, compared to using no such method ("None").

VLM	**Strategy**	τ	**Accuracy** (%)	**Consistency** (%)	**Verifiability** (%)
InstructBLIP	None	0.32	63.0	21.3	11.3
InstructBLIP	CRG	0.50	60.4	21.5	10.1
InstructBLIP	VCD	0.34	**65.8**	22.2	**11.7**
InstructBLIP	AGLA	0.16	61.6	**25.0**	11.6
InstructBLIP	Spatial	0.26	62.6	21.1	11.1
LLaVA	None	0.72	67.0	46.1	37.5
LLaVA	CRG	0.51	64.8	43.1	29.3
LLaVA	VCD	0.41	67.2	**46.5**	**36.1**
LLaVA	AGLA	0.37	66.8	45.6	33.6
LLaVA	Spatial	0.44	67.0	45.4	35.1

Table 8.6: VLM multi-tiered coherence evaluation results on Ego4D-PMD validation set for VLMs augmented with various training-free visual hallucination mitigation strategies introduced in Chapter 8.6, compared to using no such method ("None"). Each result is reported for the maximum-accuracy mistake confidence threshold τ.

VLM	Strategy	# Iter. ↓	Inf. Gain ↑	VQA Conf. ↑ (%)	ECE ↓ (%)	AURC ↓ (%)
InstructBLIP	None	4.36	0.460	75.3	22.4	36.0
InstructBLIP	CRG	4.38	0.332	72.0	**16.6**	36.3
InstructBLIP	VCD	4.21	0.442	80.3	19.2	33.2
InstructBLIP	AGLA	**2.49**	**0.880**	**89.6**	36.5	**29.7**
InstructBLIP	Spatial	4.31	0.452	75.4	23.4	33.5
LLaVA	None	3.28	0.650	76.6	24.4	25.3
LLaVA	CRG	3.54	0.364	73.4	**15.3**	29.6
LLaVA	VCD	2.94	0.714	83.3	26.3	26.1
LLaVA	AGLA	**1.64**	**0.943**	**90.3**	32.6	**23.4**
LLaVA	Spatial	3.34	0.639	76.4	23.6	25.6

Table 8.7: VLM explanatory dialog efficiency and decision confidence evaluation on Ego4D-PMD validation set for VLMs augmented with various training-free visual hallucination mitigation strategies introduced in Chapter 8.6, compared to using no such method ("None").

in Chapter 8.5.4, and the results are shown in Table 8.7. Interestingly, we find AGLA sharply outperforms other strategies in number of iterations, information gain, VQA confidence, and AURC for both VLMs. This suggests that AGLA, while achieving similar accuracy and coherence to earlier approaches, drives the VLM to arrive at a decision faster and with more confidence. We observe that CRG yields the lowest ECE, suggesting that its SuccessVQA likelihood is the best calibrated compared to other strategies. This property may be important for the reliability of decisions in higher-risk settings, or to enable the system to ask the human user for help when it is unsure.

Lastly, we find that our proposed spatial filtering strategy does not compete with other strategies under this efficiency and confidence evaluation. It seems that such a simple rule-based approach simply does not consistently provide an advantage in applying VLMs to interpretable PMD. Future work may see more significant benefits by learning question-conditioned visual attention from similar cues to those we used to define the spatial filter.

8.7 Testing Results and Analysis

Lastly, we select four representative approaches to evaluate on the test set and further interpret the performance of VLMs with:

1. Likelihood-based ranking
2. Coherence-based ranking
3. Coherence-based ranking with in-context learning
4. Coherence-based ranking with in-context learning and AGLA

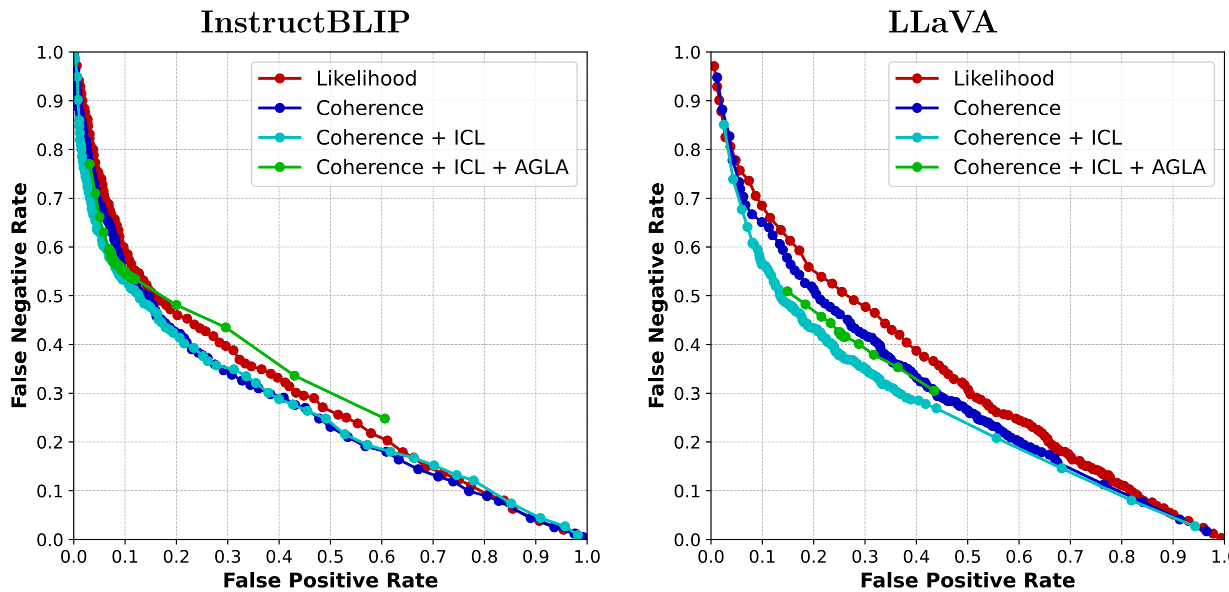


Figure 8.11: Mistake detection error tradeoff (DET) curves for selected approaches on the Ego4D-PMD test set.

We first report the accuracy, coherence, and other efficiency and reliability measurements for these approaches on the test set of Ego4D-PMD. We then take a closer look at the distribution of accuracy and coherence metrics on the test set, showing how these metrics can be used to characterize system behaviors and identify knowledge gaps.

8.7.1 Testing Results

We present the test set results for selected approaches in Figure 8.11, Table 8.8, and Table 8.9. The trends observed on the validation set mostly hold, with coherence-based question ranking and in-context learning in question generation bringing significant gains in accuracy, consistency, and verifiability. Meanwhile, integrating AGLA into VQA improves the efficiency and confidence of VLMs.

8.7.2 Visualizing Behaviors with Coherence Metrics

An additional benefit of our automated coherence metrics is the ability to audit the global and local reasoning behaviors of LLaVA under various strategies. In Figure 8.12, we visualize the distribution of decision error, relevance, and informativeness of the same four representative approaches applied to LLaVA. For each example, decision error is calculated by how far the VLM's success likelihood was from being 100% confident in the correct mistake de-

VLM	Ranking	ICL	Strategy	τ	Accuracy (%)	Consistency (%)	Verifiability (%)
InstructBLIP	Likelihood	✗	None	0.43	67.2	9.7	4.2
InstructBLIP	Coherence	✗	None	0.38	69.1	16.6	7.6
InstructBLIP	Coherence	✓	None	0.28	**69.3**	23.0	11.9
InstructBLIP	Coherence	✓	AGLA	0.29	67.7	**27.0**	**12.7**
LLaVA	Likelihood	✗	None	0.84	62.6	17.1	8.1
LLaVA	Coherence	✗	None	0.77	64.8	33.6	22.2
LLaVA	Coherence	✓	None	0.38	**68.9**	**46.7**	**35.3**
LLaVA	Coherence	✓	AGLA	0.99	67.1	44.4	30.5

Table 8.8: Multi-tiered coherence evaluation results on Ego4D-PMD test set for selected combinations of approaches to apply the LLaVA and InstructBLIP VLMs. Each result is reported for the maximum-accuracy mistake confidence threshold τ.

VLM	Ranking	ICL	Strat.	# Iter. ↓	Inf. Gain ↑	VQA Conf. ↑ (%)	ECE ↓ (%)	AURC ↓ (%)
I-BLIP	Likelihood	✗	None	4.10	0.336	76.4	**12.8**	30.2
I-BLIP	Coherence	✗	None	4.09	0.372	75.4	13.6	27.9
I-BLIP	Coherence	✓	None	4.31	0.457	75.4	16.2	**27.5**
I-BLIP	Coherence	✓	AGLA	**2.48**	**0.866**	**89.7**	30.9	35.4
LLaVA	Likelihood	✗	None	3.37	0.374	77.6	23.1	32.2
LLaVA	Coherence	✗	None	3.31	0.498	78.0	21.3	28.3
LLaVA	Coherence	✓	None	3.21	0.651	76.9	**22.0**	**23.7**
LLaVA	Coherence	✓	AGLA	**1.59**	**0.944**	**90.7**	32.8	24.7

Table 8.9: Explanatory dialog efficiency and decision confidence evaluation on Ego4D-PMD test set for selected combinations of approaches to apply the LLaVA and InstructBLIP (I-BLIP) VLMs.

tection decision. Relevance is calculated by the mean relevance of all generated questions. Informativeness is calculated by the maximum reference-adjusted informativeness among all confidence-filtered answers generated by the VLM, which can range from -1 to 1 (as discussed in Section 8.2.4.

The colors of points in these plots indicate various combinations of decision error, relevance, and informativeness in VLM outputs, highlighting common behaviors. In Figure 8.13, we provide several examples from LLaVA with coherence-based ranking, which displays a range of behaviors. Below, we further explain these behaviors and examples.

Correct and coherent points. Cyan points have low error with high informativeness and relevance, indicating correct decisions with coherent explanations. These are the best case examples from the model. Figure 8.13, Example A is one such case, where LLaVA correctly determines that the procedure "Pick up a sink brush from the kitchen slab" has been successfully completed, explaining it coherently and succinctly with a single question and answer about the location of the *sink brush*.

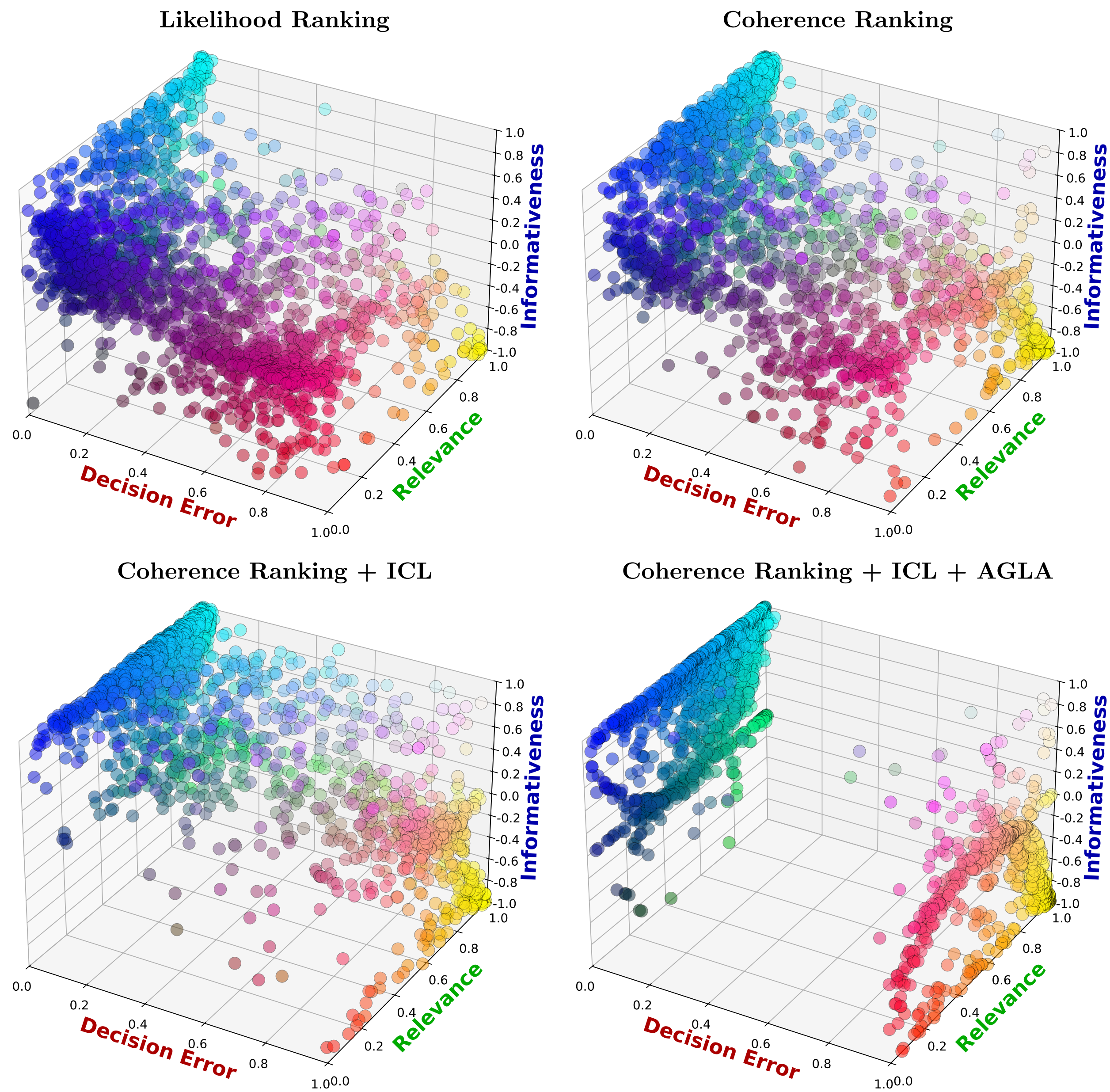


Figure 8.12: Visualization of decision error, relevance, and reference-adjusted informativeness for selected configurations of LLaVA applied to the Ego4D-PMD testing data. Informativeness is reference-adjusted to be negative when the NLI model used for evaluation is favoring the incorrect mistake detection decision (i.e., predicts success in a mistake case, or predicts mistake in a success case). Each data point represents a testing example, and its color indicates its position on each of the three color-coded axes.

Incorrect and incoherent points. Conversely, red to magenta points have high error, low informativeness, and low relevance, indicating incorrect decisions with incoherent explanations. These are the worst case examples from the model. Figure 8.13, Example B is one such case, where LLaVA incorrectly decides that the procedure "Pick a Christmas tree from the floor" was not successfully completed due to the person in the image not wearing a Santa suit or costume, an incoherent explanation for the decision.

Correct but incoherent points. Indigo to black points have low error, but low relevance and informativeness, indicating correct decisions without sufficient explanation. Figure 8.13, Example C, is an instance of this, where LLaVA correctly decides that the person in the image has not successfully completed the procedure "Open the box," but explains it by asking about the color of their shirt.

Coherent but incorrect points. White points have high error, relevance, and informativeness, indicating coherent explanations that do not lead to a correct decision. In other words, the information collected by the VLM should theoretically be sufficient to make a correct decision (according to our automated coherence metrics), but this did not occur. Figure 8.13, Example D shows one such case, where LLaVA incorrectly decides that the procedure "Open the fridge with your hand" was unsuccessful. While it correctly identified that *the fridge* in the image was open, it could not find the person's *hand* in the image, causing its incorrect decision. The VLM appears to be weighting the presence of the *hand* as a success condition for the procedure, but the key state change caused by this procedure is only that *the fridge* is open; whether or not the *hand* is present does not matter. This failure may be accountable to a deficiency in commonsense knowledge in LLaVA. The ability of this analysis to easily identify issues like this may be useful for future work in PMD and task guidance, as it enables the detection and thus the correction of system bugs.

Irrelevant but informative points. Blue points have low relevance but relatively high informativeness, indicating irrelevant questions that still yield informative answers. As shown in Figure 8.13, Example E, this does not necessarily indicate a failure of LLaVA, rather a terse explanation. In this example, LLaVA correctly determines that the procedure "Cut a plant with the sickle in your hand" has not been completed successfully. It reasonably explains this decision by asking whether the person is holding a *sickle* and responding with *No*. The question of whether the person is holding a *sickle* is deemed irrelevant by our metrics because if the answer were instead *Yes*, this would not provide sufficient information to conclude that the procedure was successful. However, since the answer was *No*, we do have sufficient information to conclude that the procedure is unsuccessful, despite the question being relatively indirect. Blue points may thus point to sufficient explanations which lack some detail or specificity.

Relevant but uninformative points. Green and yellow points have high relevance but low informativeness, indicating a failure to extract useful information in VQA. Green points have close to zero informativeness, indicating unsure responses in VQA. In Figure 8.13, Example F, LLaVA explains its decision about the procedure "Cut the guava on the cutting board with the knife in your hand" by asking whether *the guava* is on *the cutting board*. However, these objects are not present in the image and thus LLaVA's answer is not confident, causing it to respond *Unsure*, which causes zero informativeness. Despite this failure to answer the question, LLaVA still arrives at the correct conclusion that the procedure has not been successfully completed.

Meanwhile, yellow points have highly negative informativeness, indicating counterproductive responses in VQA that oppose the correct decision. As shown in Figure 8.13, Examples G and H, these cases typically occur when the VLM does not recognize an object in the image, or it recognizes an object that is not in the image. In Example G, LLaVA incorrectly decides that the procedure "Take a bottle of soda" is unsuccessful because it does not recognize that the person in the image is holding the *bottle of soda*, which is partly occluded. In Example H, LLaVA incorrectly decides that the procedure "Put the bottle in the cabinet" is successful because it hallucinates that a *bottle* is in *the cabinet*, despite neither object appearing in the image. The ability of this analysis to easily identify failures of visual perception in VLMs again may be useful for future work in this area.

In comparing the plots in Figure 8.12, we see a dramatic shift as coherence-based ranking and in-context learning are introduced. Specifically, we see much fewer red and black/indigo points, indicating much fewer cases of incoherent explanations and complete failures.

Meanwhile, when we introduce AGLA, we see a striking polarization of decision error, which comes at a tradeoff of more red points appearing in the plot. It is worth noting that introducing AGLA in LLaVA only changed the verifiability from 35.3% to 30.5%. Meanwhile, the average informativeness in the plots in Figure 8.12 drops more dramatically from 50.3% to 33.1%. Since verifiability is strictly positive, this suggests that while having similar numbers of insufficiently verified correct decisions, AGLA introduces a significant number of cases with negative informativeness, i.e., misleading information from the scene. Nonetheless, the improved efficiency and confidence from AGLA may be advantageous for some applications.

Ultimately, when choosing an approach to use in a task guidance system, we recommend weighing all of the factors we explored as appropriate for the setting: accuracy, coherence

Example A: Pick up a sink brush from the kitchen slab.

Label: ✓ **Predicted:** ✓

Explanation:
1. Is the sink brush in the person's hands? `Yes`

Example B: Pick a Christmas tree from the floor.

Label: ✓ **Predicted:** ✗

Explanation:
1. Is the person wearing a Santa suit? `No`
2. Is the person wearing a costume? `No`
3. Is the person wearing a uniform? `No`

Example C: Open the box.

Label: ✗ **Predicted:** ✗

Explanation:
1. Is the person wearing a white shirt? `No`
2. Is the person wearing a black shirt? `No`
3. Is the person wearing a blue shirt? `No`

Example D: Open the fridge with your hand.

Label: ✓ **Predicted:** ✗

Explanation:
1. Is the fridge open? `Yes`
2. Is the fridge open with the person's hand? `No`

Example E: Cut a plant with the sickle in your hand.

Label: ✗ **Predicted:** ✗

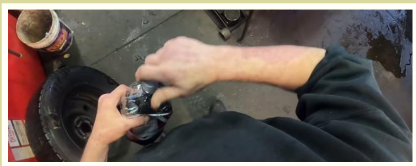
Explanation:
1. Is the person holding a sickle? `No`

Example F: Cut the guava on the cutting board with the knife in your hand.

Label: ✗ **Predicted:** ✗

Explanation:
1. Is the guava on the cutting board? `Unsure`

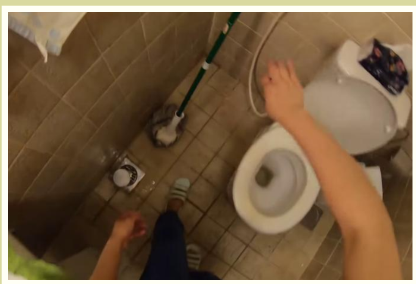
Example G: Take a bottle of soda.



Label: ✓ **Predicted:** ✗

Explanation:
1. Is the person holding a bottle of soda? `No`

Example H: Put the bottle in the cabinet.



Label: ✗ **Predicted:** ✓

Explanation:
1. Is the bottle in the cabinet? `Yes`

Figure 8.13: Sample outputs from LLaVA with coherence-based ranking, representing the range of behaviors visualized in Figure 8.12. Some images cropped for clarity and space.

(both raw and cascading metrics), efficiency, and reliability. For example, online applications may benefit from higher efficiency, even at a cost of accuracy or coherence. Meanwhile, high-risk applications may prioritize better calibration above other factors. Additional insights toward the fine-grained strengths and weaknesses of various approaches may be gained from breaking down these results by class label, mistake type, or verb and noun categories.

8.8 Summary of Findings

In this chapter, we adapted our multi-tiered accuracy and coherence evaluation framework for PCR to a new, practical problem of PMD. Specifically, we proposed automated coherence metrics leveraging fine-tuned NLI models, finding moderate correlations with human judgements of coherence. We then formulated PMD as an iterative question generation and answering problem, and drew from the previous chapters of this thesis to demonstrate that conditioning mistake detection with these questions and answers, prioritizing coherence in selecting questions to generate, and using in-context learning in question generation improved accuracy and coherence of VLMs in PMD. Lastly, we analyzed how various recent visual hallucination mitigation strategies, which aim to better align VLMs' visual attention to their prompts, impact performance, finding that they could improve VLMs' efficiency, confidence, and reliability at a possible cost of coherence. This first-of-its-kind study laid a foundation and yielded several valuable insights for future work in applying foundational VLMs to this difficult problem.

CHAPTER 9

Conclusions

This thesis has taken several steps to formalize, understand, and improve the coherence of foundational language models (LMs), including vision-and-language models (VLMs), in physical commonsense reasoning (PCR) tasks. Particularly, we extended PCR from what was traditionally viewed as a high-level classification task to a multi-faceted problem of identifying the most relevant aspects of noisy contexts, extracting physical state information about the environment based on those contexts, and making decisions about physical phenomena, e.g., physical plausibility, physical conversion of entities, and procedural execution mistakes. In this chapter, we review our breadth of research questions and findings about foundational LMs, as well as the limitations and future directions of this work.

9.1 Review of Research Questions

Despite the impressive capabilities and results of foundational LMs, incoherent behaviors, such as overfitting to superficial statistical cues and hallucination in language generation, hinder their utility in practical applications. To better understand how this problem manifests in applying LMs to reasoning tasks, we developed a multi-tiered evaluation paradigm for the coherence of reasoning in LMs. While reasoning-intensive tasks typically only evaluate the accuracy of end decisions from LMs, we proposed two concepts for more deeply evaluating the coherence of reasoning: consistency, which is the property of an LM's decision being supported by the most appropriate segments of a noisy language context, and verifiability, a PCR-specific property of these segments being associated with valid commonsense physical states from a surrounding environment.

To understand the consistency of LM decisions on reasoning-intensive classification tasks for natural language understanding (NLU), we annotated two existing benchmark datasets for dialog-based textual entailment and commonsense plausibility [269, 22]. We then fine-tuned various foundational LMs on the original datasets, and used our annotations to evaluate the consistency of their learned reasoning. Surprisingly, despite achieving very high accuracy on the end tasks, we found that fine-tuned LMs' reasoning was severely inconsistent, with predictions largely based on invalid evidence from the language context. Further, we found that transfer learning from larger relevant datasets, a common approach to improve performance on downstream tasks, did not remedy this issue.

To dive deeper into our notion of verifiability, we curated a novel, densely annotated benchmark called Tiered Reasoning for Intuitive Physics (`TRIP`), which presented a physical plausibility classification problem supported by multi-tiered reasoning chains from low-level physical states to conflicting sentences within stories, ultimately causing one story to be more plausible than the other. Using this dataset, we found that fine-tuning LMs on high-level end tasks often drove them to produce incoherent low-level reasoning chains. Interestingly, we were able to achieve the highest level of coherence by entirely excluding the end task objective from fine-tuning, and instead focusing on fine-tuning LMs to be coherent reasoners from the bottom up. Our analysis revealed several additional insights into the strengths and weaknesses of LMs' PCR capability.

Motivated by human cognitive psychology, we then explored heuristic-analytic approaches to apply foundational LMs to PCR. Specifically, we investigated how first making intuitive decisions then justifying them with lower-level reasoning chains in a top-down manner impacted the accuracy and coherence of PCR. Here, we found that in both fine-tuning and prompting LMs, a top-down reasoning direction enabled more consistent and verifiable reasoning chains to be generated, as these difficult reasoning steps benefited from being conditioned on higher-level intuitive decisions about procedural texts. More interestingly, we showed that this occurred because our cognitively motivated approach caused LMs' attention to be more faithful to the appropriate regions of language context in each step of reasoning.

We then extended our inquiry into multimodal settings with foundational multimodal representations and VLMs. First, we explored the capability of multimodal representations to capture physical states of objects in images, evaluating the similarity of visual representations of images to language descriptions of actions as well as simulated texts and images portraying action effects. Through this, we found that these representations best captured low-level physical state descriptions rather than general concepts of actions across images.

Inspired by these findings, we lastly adapted our notions of consistency and verifiability to the practical setting of procedural mistake detection (PMD), developing novel, reference-free, automated metrics powered by fine-tuned foundational LMs for natural language inference (NLI). We converted this PCR problem, previously only viewed as a classification task, to a multi-step reasoning task consisting of generating low-level questions and answers as well as the end task of making a final determination of whether some given procedure (described in text) has been successfully completed in a given egocentric video frame of someone performing the procedure. We performed a thorough set of experiments here, measuring how various interventions on text and visual inputs impacted the accuracy, coherence, efficiency, confidence, and reliability of VLMs on PMD. We found that for this problem, which has significantly more freedom than earlier settings in the reasoning chains that could validly be used to explain decisions, using this self-reflective dialog to condition VLMs' final PMD decision improves coherence at a small sacrifice of accuracy. Furthermore, we found that prioritizing coherence in generating questions and utilizing in-context learning to generate questions maximizes both coherence and accuracy. Interestingly, we found that various visual attention manipulation strategies for improving the coherence of visual processing in VLMs [245, 127, 8] had minimal impact on their accuracy and coherence, but instead improved the efficiency, confidence, and reliability of their explanations.

All in all, this thesis revealed a wealth of insights into the coherence of reasoning learned by foundational LMs. While significant progress has yet to be made in achieving truly coherent reasoning in foundational LMs, this work provided a suite of new tools for future work to interpret and improve reasoning in both text-based and practical multimodal settings. In future work, such interpretability will be essential to human users in building trust and achieving common ground with foundational LM agents in real-world task-oriented dialogs.

9.2 Limitations and Future Directions

While this work revealed many insights into the coherence of PCR in foundational LMs, the scope of these insights is largely limited to reasoning problems based on physical actions and states. On one hand, PCR has far-reaching impacts in many embodied AI application areas, and is sure to be an essential component of future solutions in these areas. On the other hand, there are many other aspects of reasoning, such as social commonsense, theory of mind, mathematical reasoning, and factual reasoning, that this thesis does not address despite prevalent incoherent behaviors in foundational LMs when applied to them. Future work on agents requiring these capabilities, e.g., in assisting humans in areas like writing, learning, and detecting disinformation, could benefit from similar deep inquiries into the coherence of foundational LMs.

Additionally, much of this work relies on the ability to densely annotate classification tasks with coherent reasoning chains. In many cases, this could be impractical, or even impossible to do comprehensively due to subjectivity. As such, it becomes beneficial to evaluate coherence in a softer, reference-free manner as we did in Chapter 8. This approach enables inspection of real-world systems applied to naturalistic data rather than requiring the use of meticulously curated benchmark data. While we made an initial attempt at quantifying coherence in one problem area, future work on other practical applications of LMs could certainly benefit from the development of automated metrics for various aspects. Additionally, future work may explore fine-tuning foundational LMs based on such automated metrics, e.g., using preference optimization techniques [182, 196], which was left unexplored in this thesis. The widespread availability of lightweight and specialized foundation models, e.g., LMs fine-tuned for natural language inference (NLI), can be useful for these efforts.

Lastly, much of this work relies on reasoning structures with limited flexibility or ability to recover from errors. For example, in Chapters 5 and 6, we forced LMs to reason in inflexible ways about PCR problems, without any capability to change intermediate reasoning steps after predicting them. While this improved the overall performance, it is possible that this causes unrecoverable cascading errors. Similarly, in Chapter 8, if hallucination or other errors occur when generating and answering questions toward determining the success of a procedure, they may severely degrade model performance. Future work on downstream systems employing reasoning strategies similar to those used in this work may further enhance the coherence of foundational LMs by exploring more robust, non-greedy strategies to generate reasoning chains. Furthermore, a particularly interesting inquiry could occur in developing LM-based agents that can anticipate their own incoherence, and communicate with or ask for help from a human-in-the-loop to recover from it.

APPENDIX A

Implementation Details and Supplementary Results for TRIP Baselines

In this appendix, we present several implementation details and supplementary results for the results presented in Chapter 5.

A.1 Fine-Tuning Details for TRIP Baselines

The RoBERTa, BERT, and DeBERTa models are built from HuggingFace's `Transformers` library [257], particularly their implementation for multiple-choice classification, and the pre-trained $\text{BERT}_{\text{LARGE}}$ parameters (336M), $\text{RoBERTa}_{\text{LARGE}}$ parameters (355M), and $\text{DeBERTa}_{\text{BASE}}$ parameters (140M) respectively. For all models, we use the AdamW optimizer [146]. Batch size is fixed at 1 story pair for all models, the maximum allowed by our available GPU memory. To select the optimizer learning rate and number of training epochs, all models are trained by grid search over these two, maximizing the validation set verifiability as defined in Section 4.2. Learning rate is selected from the set $\{1 \times 10^{-6}, 5 \times 10^{-6}, 1 \times 10^{-5}, 5 \times 10^{-5}, 1 \times 10^{-4}\}$, while the maximum number of epochs is fixed at 10. Ties are broken first by validation accuracy on the end plausibility classification task, then by selecting the model instance trained for fewer epochs (to avoid overfitting). The selected learning rate and number of epochs for each model presented in Chapter 5 are listed in Table A.1.

A.2 Supplementary Results for TRIP Baselines

Lastly, we provide additional results for the `TRIP` baselines presented in Chapter 5.[1]

[1]Note that the results in this appendix use a slightly simpler label space for `location` state classification, and thus are not directly comparable to the results presented in Chapter 5.

Table 5.1, All Losses

Model	Learning Rate	Epochs
BERT	5e-6	5
RoBERTa	1e-5	8
DeBERTa	5e-6	6

Table 5.1, Omit Conflict Detection Loss

Model	Learning Rate	Epochs
BERT	1e-6	2
RoBERTa	5e-6	9
DeBERTa	1e-6	4

Table 5.1, Omit Story Choice Loss

Model	Learning Rate	Epochs
BERT	5e-5	9
RoBERTa	1e-5	6
DeBERTa	5e-5	8

Table 5.1, Omit State Classification Loss

Model	Learning Rate	Epochs
BERT	1e-5	4
RoBERTa	1e-6	8
DeBERTa	5e-6	10

Table A.1: Selected learning rate (LR), number of training epochs, and validation verifiability and accuracy for all results presented in this thesis.

A.2.1 Conflict Detector Ablations

The Conflict Detector module takes in two types of inputs: 1) contextual embeddings of sentence-entity pairs, and 2) physical state logits from the Precondition and Effect Classifiers. To determine the impact of each, we present ablations omitting them for the best-performing instances from the previous section, i.e., those not considering story choice classification loss. Table A.2 presents these results for the validation set, while Table A.3 presents these results for the test set.

Without including the physical state inputs, we see a slight drop in consistency and verifiability of some models. For example, RoBERTa drops from 9.7% verifiability and 23.4% consistency to 4.6% and 17.7%, respectively. Meanwhile, DeBERTa increases from 8.0% verifiabiliy and 20.2% consistency to 11.4% and 24.5%. While RoBERTa seems to depend slightly on the predicted physical states in performing conflict detection, DeBERTa favors the contextual embedding.

Without including the contextual embeddings, we see a drastic drop across the board to below-random performance, with RoBERTa dropping to 0% verifiability and consistency, and DeBERTa to 2.3% and 6.6% respectively. This suggests that while forcing the model to track physical states enables greater explanation, they are not sufficient for models to learn conflict detection, or they are not incorporated successfully into the higher-level predictions. The contextual embedding, which is fine-tuned on physical state classification and conflict detection jointly, seems to be most powerful for solving the end task. Future work should further explore how to harness the rich information provided by the physical states to improve system performance and interpretability.

Contextual Embeddings + Physical States

Model	**Verif.** (%)	**Acc.** (%)	**Prec. F1** (%)	**Eff. F1** (%)	**Confl. F1** (%)
BERT	9.6	70.2	74.4	66.7	65.1
RoBERTa	**12.1**	**77.0**	72.3	62.7	70.9
DeBERTa	11.2	72.7	77.0	71.1	68.2

Contextual Embeddings Only

Model	**Verif.** (%)	**Acc.** (%)	**Prec. F1** (%)	**Eff. F1** (%)	**Confl. F1** (%)
BERT	**10.9**	72.7	75.9	69.3	66.7
RoBERTa	9.6	**76.1**	72.5	61.6	70.3
DeBERTa	9.9	**76.1**	77.3	71.3	68.6

Physical States Only

Model	**Verif.** (%)	**Acc.** (%)	**Prec. F1** (%)	**Eff. F1** (%)	**Confl. F1** (%)
BERT	0.6	54.7	60.5	59.9	51.1
RoBERTa	0.0	43.2	38.4	37.8	49.5
DeBERTa	2.2	58.1	81.0	79.0	53.0

Table A.2: Validation set performance of best models in Table 5.1 when ablating inputs to the Conflict Detector.

A.2.2 State Classification Performance by Attribute

Figure A.1 breaks down the F1 score for predicting precondition and effect states by attribute across the TRIP dataset. We find that for preconditions, openness and whether objects are running, i.e., activated, are best captured, and for effects, existence and consciousness are. Meanwhile, wetness and temperature are challenging for predicting both preconditions and effects.

Contextual Embeddings + Physical States

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
BERT	63.2	15.7	7.4
RoBERTa	**76.6**	23.4	9.7
DeBERTa	72.9	20.2	8.0

Contextual Embeddings Only

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
BERT	70.7	16.8	6.8
RoBERTa	**76.6**	17.7	4.6
DeBERTa	74.1	**24.5**	**11.4**

Physical States Only

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
BERT	56.1	3.4	0.3
RoBERTa	42.2	0.0	0.0
DeBERTa	59.3	6.6	2.3

Table A.3: Validation set performance of best models in Table 5.1 when ablating inputs to the Conflict Detector.

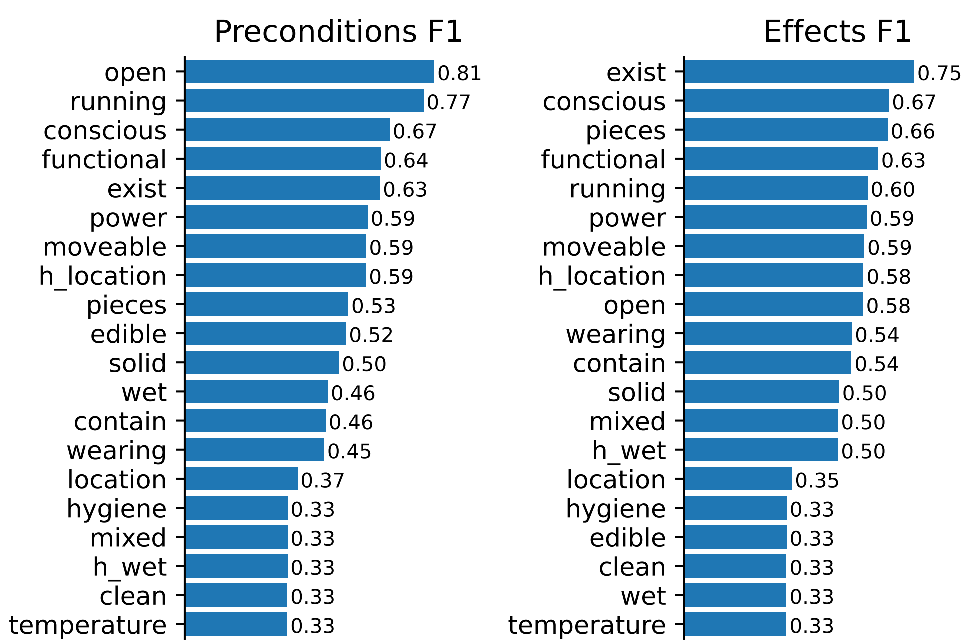


Figure A.1: Precision and recall of predictions for each attribute from our best RoBERTa model on the validation set.

APPENDIX B

Implementation Details and Supplementary Results for Cognitively Motivated Reasoning Strategies

In this appendix, we present several implementation details and supplementary results for the results presented in Chapter 6.

B.1 Tiered-ProPara Generation Details

Tiered-ProPara was generated by a simple matching process over the original dataset. We first re-split the dataset to distribute more stories into the testing and development sets from training set. Then, we applied a pairwise matching process to select two stories. Specifically, we enforce that the selected entity appears in both stories, but only converts to another entity in one of them. Also, the entity being converted must disappear after the conversion, while the entity converted must appear only after the conversion. The selected entity is provided as a known fact before the inference. Therefore, similar to `TRIP`, a system needs to predict which story has the conversion of the known entity given two stories, measured by *accuracy*. And *Consistency* is used to measure how often a system predicting the correct story and the sentence that has the conversion. Additionally, *Verifiability* measures how often a system identifies the fully correct reasoning chain, from story and sentence prediction to physical state prediction of understanding what entity has converted to. After our conversion of the dataset, there are 496 training instances, 206 development instances, and 213 testing instances.

B.2 Language Model Fine-Tuning Details

In Section 6.2, we proposed two fine-tuning approaches powered by CGLI [149], which predicts reasoning steps through task-specific layers. These layers are linear projections for the tiered tasks of selecting stories, sentences, and physical states. We preserve the *entity-aware* and *timestep-aware* encodings in CGLI. Specifically, given an *entity* E, we concatenate it with a story pair S to create a prompt $C =$ [CLS] E [SEP] S [SEP]. Following CGLI, C is then mapped with the embedding layer of the language model and summed with the *timestep* embedding. Finally, it is encoded by the language model to create a latent representation for task-specific classifications. We jointly optimize three cross-entropy losses for the story selection step, sentence selection step, and physical state prediction step ($\mathcal{L}_{story}+\mathcal{L}_{sentence}+\mathcal{L}_{state}$).

In `TRIP`, in order to make our proposed Focused CGLI (FCGLI) models focused on predicting explicit conflicts (defined in Appendix B.5) in stories, we do not optimize the physical state prediction loss for implicit conflicts. As such, the physical state information for each task instance consists of an entity, an attribute, effect state, and precondition state. The physical state loss is calculated by averaging four losses.

We selected the model for the test set by using the model with the highest validation verifiability for each task. We used a consistent set of hyper-parameters across tasks: a learning rate of 5e-6, a maximum of 10 training epochs, and a batch size of 1 (the maximum that could fit in GPU memory). The weight decay is set to be 0.01 for all parameters except for bias and LayerNorm.weight, and we use a warmup scheduler following CGLI. We report the performance by averaging three random runs. All experiments were performed on a single NVIDIA GeForce RTX 3090 Ti graphics card (24GB).

B.3 Statistical Significance Testing

While the performance gain in consistency and verifiability from our HAR strategies over baseline approaches was quite large in most cases, we performed McNemar's tests to measure the statistical significance of these gains [156]. We found that in the `TRIP` and Tiered-ProPara results presented in Table 6.1, the differences in consistency and verifiability between FCGLI-HAR and the FCGLI baseline were statistically significant ($p < 0.05$). This was also true for the differences in consistency and verifiability on both tasks between ICL-HAR and both ICL-U and ICL-CoT in Table 6.2, except for LLaMA on `TRIP`, where the consistency gain from ICL-HAR over either baseline was not significant. As expected, HAR indeed brings about statistically significant performance gains in verifiability of physical commonsense reasoning, while sometimes also significantly improving consistency.

InstructGPT

Approach	**Accuracy**	***Consistency***	***Verifiability***
ICL-U	70.9	40.7	7.1
ICL-CoT	75.0	40.7	10.8
ICL-HAR	72.6	**47.9**	**23.9**
PCICL-HAR	70.4	39.6	12.8

LLaMA

Approach	**Accuracy**	***Consistency***	***Verifiability***
ICL-U	70.4	42.3	14.8
ICL-CoT	74.6	42.3	19.7
ICL-HAR	55.6	**44.4**	**35.2**
PCICL-HAR	70.4	40.8	28.2

Table B.1: `TRIP` results of heuristic-analytic reasoning (HAR) strategies in in-context learning with LMs, including PCICL-HAR, the prompt-chaining alternative to ICL-HAR. As mentioned in Section 6.3.1, LLaMA is evaluated on a subset of `TRIP`, so in-context learning results on different LMs are not directly comparable.

B.4 HAR for Multi-Prompt In-Context Learning

On top of the chain-of-thought (CoT) implementation of HAR for in-context learning presented in this thesis, we experimented with a more strict form which chained multiple prompts together, one for each step of the reasoning tasks. This adds explicit structure that may make it more dependable. As shown in Figure B.1, each successive prompt is generated based on the previous higher-level prediction. In `TRIP` and Tiered-ProPara, LMs must first select one of two stories in which some phenomenon occurs (i.e., implausibility or conversion of entities). Then, the chosen story is used in a separate prompt to that LM and predict the sentence(s) where the phenomenon occurs. Lastly, given predicted sentence(s), LMs should predict the specific states underlying the phenomenon. While similar to the approach presented in this thesis, this approach enables us to completely remove irrelevant information from the context at each step of reasoning and rationalization, so that the model can focus only on the correct parts of the language context. While this restriction may help LMs ignore irrelevant context, this structure limits the general applicabilty of this approach compared to the more flexible chain-of-thought approach presented in this thesis.

As shown in Tables B.1 and B.2, prompt chaining for in-context learning with HAR (PCICL-HAR) also brings performance improvements over the ICR-U baseline in both `TRIP` and ProPara. In Tiered-ProPara, PCICL-HAR slightly exceeds HAR with chain-of-thought (ICL-HAR) in consistency when applied to both InstructGPT and LLaMA (up to 36.2% and 21.6% respectively), and in verifiability with LLaMA (up to 17.4%). This suggests that

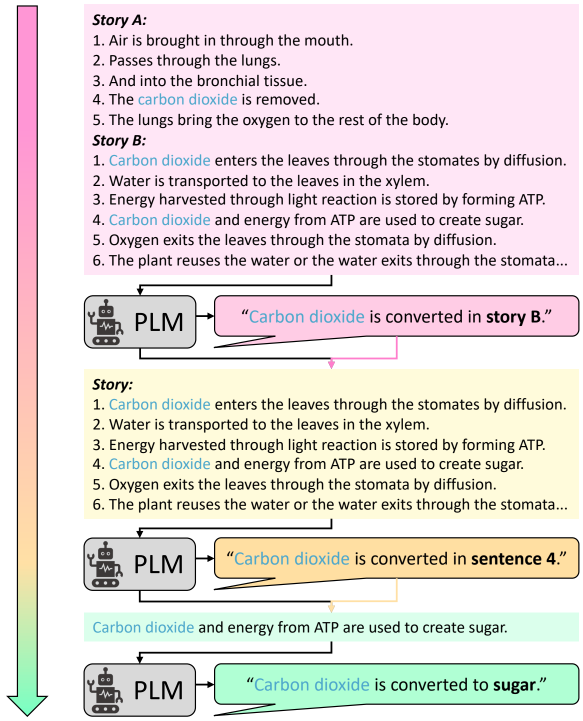


Figure B.1: Heuristic-analytic reasoning with prompt chaining for in-context learning with LMs on physical commonsense rationalization (PCICL-HAR). On Tiered-ProPara, the LM will first decide which story a conversion of an entity occurs in, then this will be used to refine the language prompt before asking the LM which sentence the conversion occurs in. Lastly, the chosen sentence will be used to predict the resulting entity after conversion.

in some cases, more explicitly structured HAR can be beneficial.

B.5 Implicit vs. Explicit Conflicts on TRIP

Plausibility conflicts between physical states in `TRIP` can take two forms. First, *explicit conflicts* like the one shown in Figure 6.4 exhibit direct disagreements in physical effect states of a particular entity after one sentence and its precondition states in a later sentence.[1] Other stories have *implicit conflicts* where no such disagreement exists, though the story may violate commonsense expectations.[2]

While our evaluation on LLaMA focused on explicit conflicts to ensure only the top-6 most common physical attributes could be used to rationalize plausibility conflicts, our

[1]For example, if *Mary put the cucumber on a plate and tossed the donut in the trash*, then in a later sentence, *Mary ate the donut*, the effect of the former sentence (i.e. *donut* is inedible) conflicts with the precondition of the latter sentence (i.e. *donut* is edible).

[2]For example, in one sentence, we may see that *Tom put the soup in the microwave*, which indirectly implies the *soup* should be heated up, then in a later sentence, *Tom ate the cold soup*.

InstructGPT			
Approach	**Accuracy**	***Consistency***	***Verifiability***
ICL-U	54.9	17.4	5.2
ICL-CoT	50.7	19.2	7.5
ICL-HAR	54.9	31.5	**20.7**
PCICL-HAR	54.5	**36.2**	18.8
LLaMA			
Approach	**Accuracy**	***Consistency***	***Verifiability***
ICL-U	51.2	3.8	1.4
ICL-CoT	57.3	9.4	4.2
ICL-HAR	41.8	17.8	13.1
PCICL-HAR	51.2	**21.6**	**17.4**

Table B.2: Tiered-ProPara results of heuristic-analytic reasoning (HAR) strategies in in-context learning with LMs, including PCICL-HAR, the prompt-chaining alternative to ICL-HAR.

evaluation on InstructGPT included both. As the connection between physical states and plausibility conflicts is unclear in implicit conflicts and our HAR strategies are intended to strengthen lower-level rationalization to support higher-level tasks, we may expect them to be especially beneficial for explicit conflicts. Table B.3 includes results for both types of conflicts on the fine-tuning and in-context learning approaches introduced so far. We see that implicit conflicts are more difficult to rationalize across the board, even with traditional chain-of-thought as applied in the ICL-CoT baseline. However, HAR indeed has a much more significant impact on `TRIP` examples with explicit conflicts, increasing verifiability from 10.0% to 36.1%, compared to an increase from 4.1% to 11.3% on implicit conflicts. This suggests that models' heuristic predictions of which story is plausible can indeed help improve performance on the analytic predictions of conflicting sentences and physical states, the latter of which has a large search space.

B.6 Language Model Prompt Details

Here, we include some extra details about how we prompted LMs in Chapter 6.

B.6.1 Automatic Exemplar Generation

We automatized exemplar generation for smoother and consistent experiments on TRIP and Tiered-ProPara.

Explicit Conflicts			
Approach	**Accuracy**	***Consistency***	***Verifiability***
ICL-U	71.7	42.2	10.0
ICL-CoT	73.9	41.7	16.7
ICL-HAR	72.2	**53.3**	**36.1**
PCICL-HAR	70.6	47.2	17.8
FCGLI	98.3	81.1	56.7
FCGLI-HAR	98.7	**89.4**	**68.9**

Implicit Conflicts			
Approach	**Accuracy**	***Consistency***	***Verifiability***
ICL-U	70.2	39.2	4.1
ICL-CoT	76.0	39.8	4.7
ICL-HAR	73.1	**42.1**	**11.1**
PCICL-HAR	70.2	31.6	7.6
FCGLI	88.7	50.2	9.3
FCGLI-HAR	89.5	**60.3**	**11.3**

Table B.3: `TRIP` results of heuristic-analytic reasoning (HAR) strategies in in-context learning with InstructGPT (top) and fine-tuning FCGLI (bottom) for explicit and implicit plausibility conflicts.

B.6.1.1 TRIP

The lowest level task in TRIP dataset is annotated with symbolic states to represent precondition and effect states over 20 physical states. Therefore, we convert symbolic physical states to natural language. For stories with explicit conflict, we iterate through all combinations of entities and 20 attributes for the 2 conflicting sentences in order to find the conflicting physical states. For some conflicts that have different entity name for the same entity, we used the following algorithm to first find all possible conflicting entity pairs, and then iterate them for the conflicting entity pair which results in maximum cosine similarity (*argmax*) between GLoVe embedding vectors (50-dimensional) [186] and more specifically, the *argmax* function is defined by:

$$\underset{c \in candidates}{\arg\max} \frac{V^c_{GloVe(entity_1)} \cdot V^c_{GloVe(entity_2)}}{\|V^c_{GloVe(entity_1)}\| \|V^c_{GloVe(entity_2)}\|} \tag{B.1}$$

Physical state familiarization. To prime PLMs for the highly-dimensional physical state classification step of TRIP, all in-context learning experiments are prepended with a list of possible physical states (e.g. dirty, clean, unpowered, powered) and a 1-shot example for each one.[3] This process is called *familiarization*, and is essential to fully specify the expected outputs for the reasoning task and enable systematic evaluation, as the model will be more likely to predict physical states within this demonstrated space of labels. As Tiered-ProPara's low-level state space consists of entities mentioned in the given text, this is only necessary for TRIP.

Filtering TRIP to shorten prompts. Since LLaMA is limited to a context length of 2048 tokens, we create a filtered version of TRIP which only includes instances where annotated states involve only the top-6 most frequent physical attributes. The familiarization process mentioned above contains $\sim 70\%$ tokens in the full prompt (~ 3800 tokens). Therefore, we first perform a statistical analysis on the dataset and select 6 highest-frequency physical states for effect and precondition, namely *(no longer existent, existent), (broken, functional), (in pieces, whole), (turned off, turned on), (inedible, edible), (unpowered, powered).* After that, the prompt length is reduced to ~ 1600 tokens. As familiarization of physical states is used to familiarize the model with the classification space, we filtered the dataset to stories that only contain explicit conflicts between these high-frequency physical states.

B.6.1.2 Tiered-ProPara

The automatic exemplar generation process for Tiered-Propara is similar to the TRIP, but more straightforward. There's no familiarization stage for Tiered-Propara. After two-story prompt, we asked the model with a question: *What happened to [converted entity]?* The answer prompt in the demonstration is composed by filling ground truth labels (story, sentence, and the entity converted to) to the template.

B.6.2 Full Prompt Examples

From our in-context learning experiments, we include full example prompts used with InstructGPT for the in-context learning with heuristic-analytic reasoning (ICL-HAR) and unstructured in-context learning (ICL-U) strategies. Figures B.2 and B.3 show examples for ICL-HAR in TRIP and ProPara, respectively. Figures B.4, B.5, and B.6 show examples for the ICL-U in TRIP, while Figures B.7, B.8, and B.9 show examples for the ICL-U baseline in ProPara. For the ICL-CoT baseline, we simply append "Let's think step by step..." and zero-shot CoT generated by InstructGPT into the ICL-U prompting demonstrations before

[3] For example, *After Mary sliced the apple, what is the state of the apple? The apple is now in pieces.* and *Before Tom opened the door, what was the state of the door? The door was closed.* may be used to familiarize the model with the physical state classification space.

final predictions are made for each sub-task. We provide the full zero-shot CoT prompts we used for each sub-task in TRIP and Tiered-ProPara below:

- **TRIP, Story Selection:** Let's think step by step about which story is more plausible.
- **TRIP, Sentence Selection:** Let's think step by step about which sentences are conflicting in one story.
- **TRIP, Physical State Prediction:** Let's think step by step about which physical states are conflicting in two sentences in one story.
- **Tiered-Propara, Story Selection:** Let's think step by step about which story [entity] were converted in.
- **Tiered-Propara, Sentence Selection:** Let's think step by step about which sentence [entity] were converted in one story.
- **Tiered-Propara, Physical State Prediction:** Let's think step by step about what [entity] were converted to in one sentence in one story.

B.7 Attention Analysis Details

B.7.1 Self-Attention Weight Extraction

To enable our attention analysis for soft HAR in in-context learning, we used the *output_attentions* flag in the Transformers [257] library to extract the raw attentions computed during inference with LLaMA. An attention mask is applied to the attentions to remove the attentions associated with the demonstration prompt and special characters, only keeping the attentions associated with the tokens from the test prompt. We then summed up the attentions across each sentence in the test prompt, averaged across a subset of the generated tokens, and normalized by dividing the sum of the attentions. Here, we computed the token subset's average attention on a sentence of the story prompt, and we used it to measure the importance of a sentence to the model's reasoning outcomes. We then used these normalized weights to calculate the evaluation criteria proposed in Section 6.3.2.

B.7.2 Attentional Precision and Recall Details

Attentional precision and recall are calculated by converting normalized attention weights into binary measures of whether attention is correct. To do this, we check whether the average attention weight for the relevant segment of language context (i.e., the appropriate story or sentence(s)) exceeds a threshold. We calculate the average precision and recall over a set of 9 candidate thresholds centered around 0.1 (0.08 - 0.12) with an interval 0.005 because on average, there are about 10 sentences in a pair of stories, and all sentences' attention are normalized to a sum of 1.

Given this binary measure, we can then classify each PLM output into four combinations over whether its attention is faithful, and whether its intermediate predictions are coherent (i.e., consistent or verifiable): true positive, false positive, true negative, and false negative. We provide some examples here, assuming a static threshold of 0.09. In the story-level prediction example from Figure 6.6, we calculate the average sentence-wise attention in the story containing a conversion (i.e., story B), and compare it to the threshold. These values are $0.590/6 = 0.098$ and $0.837/6 = 0.140$ for ICL-U and ICL-HAR, respectively. Both values of ICL-U and ICL-HAR exceeded the threshold, but ICL-U didn't correctly identify the sentence that contains a conversion, while ICL-HAR did. Therefore, we classify the example from ICL-U as false negative and ICL-HAR as true positive. Similarly, in the physical state detection example from Figure 6.7, we calculate the average sentence-wise attention on the two conflicting sentences, which are sentences 1 and 5 in story A. These values are $(0.09 + 0.086)/2 = 0.088$ and $(0.213 + 0.154)/2 = 0.184$ for ICL-U and ICL-HAR, resp. Because ICL-HAR exceeded the threshold but ICL-U didn't, and ICL-HAR actually generated the correct response while ICL-U didn't, we classify the example from ICL-U as a true negative and ICL-HAR as a true positive.

TRIP - ICL-HAR

Familiarization Prompts - Physical States (6 of 80):
Physical state options: powered, edible, whole ...
Before Tom turned on the microwave, what was the state of the microwave? The microwave was powered.
Before Tom ate the cereal, what was the state of the cereal? The cereal was edible.
Before Tom cut the banana into slices, what was the state of the banana? The banana was whole.
...
Physical state options: unpowered, inedible, in pieces ...
After Tom unplugged the microwave, what is the state of the microwave? The microwave is now unpowered.
After Tom threw the cereal in the dustbin, what is the state of the cereal? The cereal is now inedible.
After Tom mashed the banana, what is the state of the banana? The banana is now in pieces.

Demonstration Prompts - Physical States (1 of 4):
Story A:
1. Mary went to the fridge.
2. Mary took out a bowl from the fridge.
3. The bowl had a cucumber and a donut in it.
4. Mary put the cucumber on a plate and tossed the donut in the trash.
5. Mary ate the donut.

Story B:
1. Mary went to the fridge.
2. Mary took out a bowl from the fridge.
3. The bowl had a cucumber and a donut in it.
4. Mary put the cucumber on the counter.
5. Mary ate the donut.

Story B is more plausible.
In Story A, sentences 4 and 5 conflict with each other.
For sentence 4:
After Mary put the cucumber on a plate and tossed the donut in the trash, what is the state of the donut?
The donut is now inedible.
For sentence 5:
Before Mary ate the donut, what was the state of the donut? The donut was edible.

Test Prompt:
Story A:
1. Ann opened the toolbox and realized that someone took away all of her tools.
2. Ann took out an axe, a pair of scissors, rope and a few other things.
3. Ann put these items in a box and placed them in the trunk of her car.
4. Ann fastened her bicycle to the bicycle rack on her car.
5. Ann went inside the house to get the rest of her gear to put in the car.

Story B:
1. Ann opened the toolbox to get a few things for her camping trip.
2. Ann took out an axe, a pair of scissors, rope and a few other things.
3. Ann put these items in a box and placed them in the trunk of her car.
4. Ann fastened her bicycle to the bicycle rack on her car.
5. Ann went inside the house to get the rest of her gear to put in the car.

LLM Sample Output:
Story B is more plausible.
In Story A, sentences 1 and 2 conflict with each other.
For sentence 1:
After Ann opened the toolbox and realized that someone took away all of her tools, what are the state of the tools? The tools are now no longer existent.
For sentence 2:
Before Ann took out an axe, a pair of scissors, rope and a few other things, what were the state of the things? The things were existent.

Figure B.2: Sample prompt used for in-context learning with heuristic-analytic reasoning (ICL-HAR) on TRIP. Some redundant sections of the prompt are omitted (indicated with "..." and in section headings).

Tiered-ProPara - ICL-HAR

Demonstration Prompts (1 of 4):
Story A:
1. Wind and water erode rocks.
2. The wind and water pick up sediment that came off the rocks.
3. Sediment is carried to an area by the wind, water, air, or ice.
4. Sediment is deposited on the same spot many times.
5. The sediment piles on top of other sediment for a long time.
6. The sediment near the bottom is compressed by the weight of newer sediment.
7. The sediment becomes sedimentary rock as it is pushed together by the heavy weight.
Story B:
1. Waves hit the coast.
2. Waves have energy that pushes the water and sediment.
3. The sediment can help erode land like sandpaper.
4. Water washes the sediment back.
5. Some sediment is left as sand.
What happened to sediment?
Sediment is converted in story A.
In story A, sediment is converted in sentence 7.
After the sediment becomes sedimentary rock as it is pushed together by the heavy weight, sediment is converted to sedimentary rock.

Test Prompt:
Story A:
1. Coal is heated in the boiler.
2. The water tank over the boiler is heated.
3. Creates steam.
4. The steam is funneled to the piston.
5. Piston uses the steam as energy.
6. The piston causes the crankshaft to move.
Story B:
1. Plants obtain water through the soil they are growing in.
2. The plants roots absorb the water from the soil.
3. Transport the water to the parts of the plant where the water is needed.
4. The plant uses the water as part of the photosynthesis process.
5. The plant creates food, called glucose, for itself.
6. The plant uses water to circulate the glucose around the plant to deliver the glucose to where it's needed.
What happened to water?

LLM Sample Output:
Water is converted in story A.
In story A, water is converted in sentence 3.
After creates steam, water is converted to steam.

Figure B.3: Sample prompt used for in-context learning with heuristic-analytic reasoning (ICL-HAR) on Tiered-ProPara. Some redundant sections of the prompt are omitted (indicated with "..." and in section headings).

TRIP - ICL-U Baseline - Plausible Story Selection

Demonstration Prompts (1 of 4):
Story A:
1. Mary went to the fridge.
2. Mary took out a bowl from the fridge.
3. The bowl had a cucumber and a donut in it.
4. Mary put the cucumber on a plate and tossed the donut in the trash.
5. Mary ate the donut.

Story B:
1. Mary went to the fridge.
2. Mary took out a bowl from the fridge.
3. The bowl had a cucumber and a donut in it.
4. Mary put the cucumber on the counter.
5. Mary ate the donut.

Story B is more plausible.

Test Prompt:
Story A:
1. Ann opened the toolbox and realized that someone took away all of her tools.
2. Ann took out an axe, a pair of scissors, rope and a few other things.
3. Ann put these items in a box and placed them in the trunk of her car.
4. Ann fastened her bicycle to the bicycle rack on her car.
5. Ann went inside the house to get the rest of her gear to put in the car.

Story B:
1. Ann opened the toolbox to get a few things for her camping trip.
2. Ann took out an axe, a pair of scissors, rope and a few other things.
3. Ann put these items in a box and placed them in the trunk of her car.
4. Ann fastened her bicycle to the bicycle rack on her car.
5. Ann went inside the house to get the rest of her gear to put in the car.

LLM Sample Output:
Story B is more plausible.

Figure B.4: Sample prompt used for plausible story selection step of unstructured in-context learning (ICL-U) baseline on TRIP. Some redundant sections of the prompt are omitted (indicated with "..." and in section headings).

TRIP - ICL-U Baseline - Conflicting Sentence Selection

Demonstration Prompts (1 of 4):
Story A:
1. Mary went to the fridge.
2. Mary took out a bowl from the fridge.
3. The bowl had a cucumber and a donut in it.
4. Mary put the cucumber on a plate and tossed the donut in the trash.
5. Mary ate the donut.

Story B:
1. Mary went to the fridge.
2. Mary took out a bowl from the fridge.
3. The bowl had a cucumber and a donut in it.
4. Mary put the cucumber on the counter.
5. Mary ate the donut.

Sentences 4 and 5 conflict with each other in story A.

Test Prompt:
Story A:
1. Ann opened the toolbox and realized that someone took away all of her tools.
2. Ann took out an axe, a pair of scissors, rope and a few other things.
3. Ann put these items in a box and placed them in the trunk of her car.
4. Ann fastened her bicycle to the bicycle rack on her car.
5. Ann went inside the house to get the rest of her gear to put in the car.

Story B:
1. Ann opened the toolbox to get a few things for her camping trip.
2. Ann took out an axe, a pair of scissors, rope and a few other things.
3. Ann put these items in a box and placed them in the trunk of her car.
4. Ann fastened her bicycle to the bicycle rack on her car.
5. Ann went inside the house to get the rest of her gear to put in the car.

LLM Sample Output:
Sentences 1 and 2 conflict with each other in story A.

Figure B.5: Sample prompt used for conflicting sentence selection step of unstructured in-context learning (ICL-U) baseline on TRIP. Some redundant sections of the prompt are omitted (indicated with "..." and in section headings).

TRIP - ICL-U Baseline - Physical State Prediction

Familiarization Prompts - Physical States (6 of 80):

Physical state options: powered, edible, whole ...
Tom turned on the microwave. Before, what was the state of the microwave? The microwave was powered.
Tom ate the cereal. Before, what was the state of the cereal? The cereal was edible.
Tom cut the banana into slices. Before, what was the state of the banana? The banana was whole.
...
Physical state options: unpowered, inedible, in pieces ...
Tom unplugged the microwave. After, what is the state of the microwave? The microwave is now unpowered.
Tom threw the cereal in the dustbin. After, what is the state of the cereal? The cereal is now inedible.
Tom mashed the banana. After, what is the state of the banana? The banana is now in pieces.

Demonstration Prompts (1 of 4):

Story A:
1. Mary went to the fridge.
2. Mary took out a bowl from the fridge.
3. The bowl had a cucumber and a donut in it.
4. Mary put the cucumber on a plate and tossed the donut in the trash.
5. Mary ate the donut.

Story B:
1. Mary went to the fridge.
2. Mary took out a bowl from the fridge.
3. The bowl had a cucumber and a donut in it.
4. Mary put the cucumber on the counter.
5. Mary ate the donut.

After, what is the state of the donut? The donut is now inedible.
Before, what was the state of the donut? The donut was edible.

Test Prompt:

Story A:
1. Ann opened the toolbox and realized that someone took away all of her tools.
2. Ann took out an axe, a pair of scissors, rope and a few other things.
3. Ann put these items in a box and placed them in the trunk of her car.
4. Ann fastened her bicycle to the bicycle rack on her car.
5. Ann went inside the house to get the rest of her gear to put in the car.

Story B:
1. Ann opened the toolbox to get a few things for her camping trip.
2. Ann took out an axe, a pair of scissors, rope and a few other things.
3. Ann put these items in a box and placed them in the trunk of her car.
4. Ann fastened her bicycle to the bicycle rack on her car.
5. Ann went inside the house to get the rest of her gear to put in the car.

LLM Sample Output:

After, what are the state of the tools? The tools are now no longer existent.
Before, what were the state of the things? The things were existent.

Figure B.6: Sample prompt used for physical state prediction step of unstructured in-context learning (ICL-U) baseline on TRIP. Some redundant sections of the prompt are omitted (indicated with "..." and in section headings).

Tiered-ProPara - ICL-U Baseline - Conversion Story Selection

Demonstration Prompts (1 of 4):
Story A:
1. Wind and water erode rocks.
2. The wind and water pick up sediment that came off the rocks.
3. Sediment is carried to an area by the wind, water, air, or ice.
4. Sediment is deposited on the same spot many times.
5. The sediment piles on top of other sediment for a long time.
6. The sediment near the bottom is compressed by the weight of newer sediment.
7. The sediment becomes sedimentary rock as it is pushed together by the heavy weight.
Story B:
1. Waves hit the coast.
2. Waves have energy that pushes the water and sediment.
3. The sediment can help erode land like sandpaper.
4. Water washes the sediment back.
5. Some sediment is left as sand.
What happened to sediment?
Sediment is converted in story A.

Test Prompt:
Story A:
1. Coal is heated in the boiler.
2. The water tank over the boiler is heated.
3. Creates steam.
4. The steam is funneled to the piston.
5. Piston uses the steam as energy.
6. The piston causes the crankshaft to move.
Story B:
1. Plants obtain water through the soil they are growing in.
2. The plants roots absorb the water from the soil.
3. Transport the water to the parts of the plant where the water is needed.
4. The plant uses the water as part of the photosynthesis process.
5. The plant creates food, called glucose, for itself.
6. The plant uses water to circulate the glucose around the plant to deliver the glucose to where it's needed.
What happened to water?

LLM Sample Output:
Water is converted in story A.

Figure B.7: Sample prompt used for conversion story selection step of unstructured in-context learning (ICL-U) baseline on Tiered-ProPara. Some redundant sections of the prompt are omitted (indicated with "..." and in section headings).

Tiered-ProPara - ICL-U Baseline - Conversion Sentence Selection

Demonstration Prompts (1 of 4):

Story A:
1. Wind and water erode rocks.
2. The wind and water pick up sediment that came off the rocks.
3. Sediment is carried to an area by the wind, water, air, or ice.
4. Sediment is deposited on the same spot many times.
5. The sediment piles on top of other sediment for a long time.
6. The sediment near the bottom is compressed by the weight of newer sediment.
7. The sediment becomes sedimentary rock as it is pushed together by the heavy weight.

Story B:
1. Waves hit the coast.
2. Waves have energy that pushes the water and sediment.
3. The sediment can help erode land like sandpaper.
4. Water washes the sediment back.
5. Some sediment is left as sand.

What happened to sediment?
Sediment is converted in sentence 7 in story A.

Test Prompt:

Story A:
1. Coal is heated in the boiler.
2. The water tank over the boiler is heated.
3. Creates steam.
4. The steam is funneled to the piston.
5. Piston uses the steam as energy.
6. The piston causes the crankshaft to move.

Story B:
1. Plants obtain water through the soil they are growing in.
2. The plants roots absorb the water from the soil.
3. Transport the water to the parts of the plant where the water is needed.
4. The plant uses the water as part of the photosynthesis process.
5. The plant creates food, called glucose, for itself.
6. The plant uses water to circulate the glucose around the plant to deliver the glucose to where it's needed.

What happened to water?

LLM Sample Output:

Water is converted in sentence 3 in story A.

Figure B.8: Sample prompt used for conversion sentence selection step of unstructured in-context learning (ICL-U) baseline on Tiered-ProPara. Some redundant sections of the prompt are omitted (indicated with "..." and in section headings).

Tiered-ProPara - ICL-U Baseline - Physical State Prediction

Demonstration Prompts (1 of 4):

Story A:
1. Wind and water erode rocks.
2. The wind and water pick up sediment that came off the rocks.
3. Sediment is carried to an area by the wind, water, air, or ice.
4. Sediment is deposited on the same spot many times.
5. The sediment piles on top of other sediment for a long time.
6. The sediment near the bottom is compressed by the weight of newer sediment.
7. The sediment becomes sedimentary rock as it is pushed together by the heavy weight.

Story B:
1. Waves hit the coast.
2. Waves have energy that pushes the water and sediment.
3. The sediment can help erode land like sandpaper.
4. Water washes the sediment back.
5. Some sediment is left as sand.

What happened to sediment?
Sediment is converted to sedimentary rock.

Test Prompt:

Story A:
1. Coal is heated in the boiler.
2. The water tank over the boiler is heated.
3. Creates steam.
4. The steam is funneled to the piston.
5. Piston uses the steam as energy.
6. The piston causes the crankshaft to move.

Story B:
1. Plants obtain water through the soil they are growing in.
2. The plants roots absorb the water from the soil.
3. Transport the water to the parts of the plant where the water is needed.
4. The plant uses the water as part of the photosynthesis process.
5. The plant creates food, called glucose, for itself.
6. The plant uses water to circulate the glucose around the plant to deliver the glucose to where it's needed.

What happened to water?

LLM Sample Output:

Water is converted to steam.

Figure B.9: Sample prompt used for conversion entity prediction step of unstructured in-context learning (ICL-U) baseline on Tiered-ProPara. Some redundant sections of the prompt are omitted (indicated with "..." and in section headings).

APPENDIX C

Implementation Details for Procedural Mistake Detection Strategies

In this appendix, we present several implementation details for the results presented in Chapter 8.

C.1 Coherence Metrics Human Study Details

To collect the human judgements reported in Chapter 8.2 when introducing our proposed metrics for relevance and informativeness, we randomly sampled the outputs for 50 iterations of LLaVA's self-dialog from two combinations of evaluated approaches:

- Likelihood-based question ranking
- Coherence-based question ranking augmented with question candidates from in-context learning

For both relevance and informativeness annotation, we provided the following background for the task:

> *Imagine you just had eye surgery, and are currently unable to see. You're performing a task you're familiar with, but need help to determine whether you successfully completed it. You video call a friend (who is unfamiliar with the task) and show them what you're working on. You then ask them some yes/no questions to figure out whether you successfully completed the task.*

Relevance annotation instructions and example. Annotators are provided the following instructions for annotating relevance:

> *For each annotation task, you will be given the following information:*

- *A **sentence** describing the procedure you're trying to perform.*
- *An optional list of **previous questions** you already asked, and their **answers**.*
- *A **potential next question** you could ask your friend.*

*You must rate how **relevant** the potential next question is. By relevant, we mean: **given the previous questions and answers, how helpful could an answer to this question be in determining whether you successfully completed the task?***

We recruit 5 annotators, each of which annotates 10 samples. One sample is listed below:

Sentence: *Drop the bowls on the table with your hand*

Previous questions and answers:

1. *Are the bowls on the table? (Answer: Yes)*
2. *Is the person holding the bowls in their hand? (Answer: No)*

Potential next question: *Is the person about to drop the bowls on the table?*

Your rating:

- *1 (very irrelevant)*
- *2 (slightly irrelevant)*
- *3 (neutral; may or may not be relevant)*
- *4 (slightly relevant)*
- *5 (very relevant)*
- *Instructions Unclear*

Informativeness annotation instructions and example. Annotators are provided the following instructions for annotating informativeness:

For each annotation task, you will be given the following information:

- *A **sentence** describing the procedure you're trying to perform.*
- *A list of **questions** you asked your friend, and their **answers**.*

*You must rate how **informative** the questions and answers are. By informative, we mean: **based on all the information you have, how sure are you about whether you succeeded?***

We recruit 5 annotators, each of which annotates 10 samples. One sample is listed below:

Sentence: *Clean the bowl*

Previous questions and answers: *None*

Last question: *Is there a bowl in the image?* ***Last answer:*** *Yes*

Your rating:

- *1 (very uninformative/unsure)*
- *2 (slightly uninformative/unsure)*
- *3 (neutral; may or may not be relevant)*
- *4 (slightly informative/sure)*
- *5 (very informative/sure)*
- *Instructions Unclear*

C.2 Self-Dialog Prompt Templates

When prompting vision-and-language models (VLMs) to generate questions, we use the following prompt template for the bottom-up approach:

> *This is a photo of someone working on the procedure "⟨procedural text⟩". I will ask a series of different yes/no questions to gather information about the scene, then use it to determine whether the person has successfully executed the procedure. The goal is to extract as much relevant information as possible from the scene, so I will not repeat questions.*
>
> *Q:*

For the top-down approach, we use the following prompt template:

> *This is a photo of someone working on the procedure "⟨procedural text⟩". I will ask a series of different yes/no questions to determine whether the person has successfully executed the procedure, then explain why based on information about the scene. The goal is to extract as much relevant information as possible from the scene, so I will not repeat questions.*
>
> *Q:*

Once a question is generated, we append it to the prompt, then prompt the VLM again with "A:" to elicit an answer. Questions and answers are generated iteratively until the stopping criteria described in Chapter 8.4 are met. To prompt the VLM to judge the success of a procedure, we use the following prompt:

> *Based on the image and above information, has the procedure "⟨procedural text⟩" been successfully executed?*

C.3 Rephrasing Prompt Details

As discussed in Chapter 8.2.1, we use a fine-tuned NLI model to judge the success of procedures given questions and answers. In order to convert questions and answers into declarative statements to pass into the NLI model, we prompt a VLM with the following 10 in-context demonstrations of rephrasing before prompting it to rephrase a question and answer for the task at hand:

1. ***Question:*** *Is there a bowl on the table?*
   Answer: *Yes*
   Statement: *There is a bowl on the table.*

2. ***Question:*** *Are the eggs cracked?*
   Answer: *No*
   Statement: *The eggs are not cracked.*

3. ***Question:*** *Does the cardboard box look open?*
   Answer: *Yes*
   Statement: *The cardboard box looks open.*

4. ***Question:*** *Are there any leaves outside of the basket?*
   Answer: *No*
   Statement: *There are not any leaves outside of the basket.*

5. ***Question:*** *Is the orange peeled?*
   Answer: *Yes*
   Statement: *The orange is peeled.*

6. ***Question:*** *Is the mug empty?*
   Answer: *No*
   Statement: *The mug is not empty.*

7. ***Question:*** *Are there hedge trimmers in the image?*
   Answer: *Yes*
   Statement: *There are hedge trimmers in the image.*

8. ***Question:*** *Has the light switch been turned on?*
   Answer: *No*
   Statement: *The light switch has not been turned on.*

9. ***Question:*** *Does the table have any cups on it?*
   Answer: *Yes*
   Statement: *The table has cups on it.*

10. ***Question:*** *Is the cabinet closed?*
    Answer: *No*
    Statement: *The cabinet is not closed.*

C.4 Example Questions for In-Context Learning in Question Generation

As discussed in Chapter 8.5.2, we condition VLMs with sets of human-written questions for 20 procedures from the Ego4D for Procedural Mistake Detection (Ego4D-PMD) dataset. The annotated procedures (underlined) and questions (italicized) are listed below:

1. <u>Soak the sponge in a soapy water with your hands</u>
   (a) *Is there a sponge?*
   (b) *Is the sponge in water?*
   (c) *Is the water soapy?*
2. <u>Open the bottle</u>
   (a) *Is there a bottle in the image?*
   (b) *Is the bottle open?*
   (c) *Does the bottle have a lid on it?*
3. <u>Take the baking tray away from the table</u>
   (a) *Can you see a baking tray?*
   (b) *Is the baking tray on the table?*
   (c) *Is the baking tray picked up by someone?*

4. Turn on a torch light
    (a) *Is there a torch light in the photo?*
    (b) *Is the torch light powered on?*
    (c) *Is the torch light lit up?*
5. Fold the right edge of the wrapper
    (a) *Is there a wrapper in the image?*
    (b) *Is the wrapper completely flat?*
    (c) *Is the right edge of the wrapper folded?*
6. Pour the water into the blue container
    (a) *Do you see a blue container anywhere?*
    (b) *Is there water in the blue container?*
    (c) *Is the blue container empty?*
7. Paint the patio with the paint brush
    (a) *Is this a photo of a patio?*
    (b) *Is the patio painted?*
    (c) *Is someone holding a paint brush?*
8. Spread the black peas on the salad with the spoon in your hand
    (a) *Is there a salad?*
    (b) *Are there black peas on the salad?*
    (c) *Is there a spoon in someone's hand?*
9. Scoop paint from the pallet on the table with the paint brush
    (a) *Do you see a paint brush and a paint palette?*
    (b) *Is there paint on the paint brush?*
    (c) *Is the paint brush in someone's hand?*
10. Wash the car with a sponge in your hand
    (a) *Do you see a car?*
    (b) *Is the car clean?*
    (c) *Is the sponge being held?*
11. Pick the scrubber from the sink
    (a) *Do you see a scrubber somewhere?*
    (b) *Is the scrubber in the sink?*
    (c) *Is the scrubber in someone's hand?*

12. Peel the onion
    (a) *Is there an onion in the image?*
    (b) *Is the onion's skin removed?*
    (c) *Is the onion peeled?*
13. Put the dirt in the dust bin
    (a) *Is there a dust bin?*
    (b) *Is there dirt in the dust bin?*
    (c) *Is there any dirt outside of the dust bin?*
14. Cut dough into two
    (a) *Do you see any dough?*
    (b) *Is the dough in two pieces?*
    (c) *Is the dough whole?*
15. Break the walnut with the nutcracker in your hand
    (a) *Do you see a walnut?*
    (b) *Is the walnut cracked?*
    (c) *Is there a nut cracker in someone's hand?*
16. Turn off the tap
    (a) *Is there a tap in the photo?*
    (b) *Is the water running?*
    (c) *Is the faucet switched off?*
17. Heat the edge of the bag with the lighter
    (a) *Do you see a bag and a lighter?*
    (b) *Is there a flame coming from the lighter?*
    (c) *Is the lighter near the bag?*
18. Close the fridge
    (a) *Is there a fridge?*
    (b) *Is the fridge open?*
    (c) *Can you see inside the fridge?*
19. Chop green beans with a knife on the chopping board
    (a) *Do you see green beans on a cutting board?*
    (b) *Are the green beans sliced?*
    (c) *Is someone using a knife?*

20. Drop the brush in your hand on the oven

    (a) *Is there a brush in the scene?*
    (b) *Is there an oven?*
    (c) *Is the brush on the oven?*

BIBLIOGRAPHY

[1] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Qin Cai, Martin Cai, Caio César Teodoro Mendes, Weizhu Chen, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Yen-Chun Chen, Yi-Ling Chen, Parul Chopra, Xiyang Dai, Allie Del Giorno, Gustavo de Rosa, Matthew Dixon, Ronen Eldan, Victor Fragoso, Dan Iter, Mei Gao, Min Gao, Jianfeng Gao, Amit Garg, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Jamie Huynh, Mojan Javaheripi, Xin Jin, Piero Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Mahoud Khademi, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Ce Liu, Mengchen Liu, Weishung Liu, Eric Lin, Zeqi Lin, Chong Luo, Piyush Madan, Matt Mazzola, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Xin Wang, Lijuan Wang, Chunyu Wang, Yu Wang, Rachel Ward, Guanhua Wang, Philipp Witte, Haiping Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Sonali Yadav, Fan Yang, Jianwei Yang, Ziyi Yang, Yifan Yang, Donghan Yu, Lu Yuan, Chengruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv: 2404.14219*, 2024.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv: 2303.08774*, 2023.

[3] Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. Fine-grained Analysis of Sentence Embeddings Using Auxiliary Prediction Tasks. *arXiv: 1608.04207*, 2017.

[4] Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. Analyzing the behavior of visual question answering models. In Jian Su, Kevin Duh, and Xavier Carreras, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1955–1960, Austin, TX, USA, 2016. Association for Computational Linguistics.

[5] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron

David, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Daniel Ho, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Eric Jang, Rosario Jauregui Ruano, Kyle Jeffrey, Sally Jesmonth, Nikhil Jayant Joshi, Ryan C. Julian, Dmitry Kalashnikov, Yuheng Kuang, Kuang-Huei Lee, Sergey Levine, Yao Lu, Linda Luu, Carolina Parada, Peter Pastor, Jornell Quiambao, Kanishka Rao, Jarek Rettinghouse, Diego M Reyes, Pierre Sermanet, Nicolas Sievers, Clayton Tan, Alexander Toshev, Vincent Vanhoucke, F. Xia, Ted Xiao, Peng Xu, Sichun Xu, and Mengyuan Yan. Do as I can, not as I say: Grounding language in robotic affordances. In *Conference on Robot Learning*, 2022.

[6] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikoł aj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 23716–23736. Curran Associates, Inc., 2022.

[7] Aida Amini, Antoine Bosselut, Bhavana Dalvi, Yejin Choi, and Hannaneh Hajishirzi. Procedural reading comprehension with attribute-aware context flow. In *Conference on Automated Knowledge Base Construction*, 2020.

[8] Wenbin An, Feng Tian, Sicong Leng, Jiahao Nie, Haonan Lin, QianYing Wang, Guang Dai, Ping Chen, and Shijian Lu. AGLA: Mitigating object hallucinations in large vision-language models with assembly of global and local attention. *arXiv: 2406.12718*, 2024.

[9] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-Language Navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018. IEEE.

[10] Thomas Anthony, Zheng Tian, and David Barber. Thinking fast and slow with deep learning and tree search. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[11] Aristotle. *Prior Analytics.* Hackett Publishing, 1989.

[12] Stéphane Aroca-Ouellette, Cory Paik, Alessandro Roncone, and Katharina Kann. PROST: Physical reasoning about objects through space and time. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4597–4608, Online, 2021. Association for Computational Linguistics.

[13] Akari Asai and Hannaneh Hajishirzi. Logic-guided data augmentation and regularization for consistent question answering. In *Proceedings of the 58th Annual Meeting of*

the Association for Computational Linguistics, Online, 2020. Association for Computational Linguistics.

[14] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

[15] Anton Bakhtin, Laurens van der Maaten, Justin Johnson, Laura Gustafson, and Ross Girshick. Phyre: A new benchmark for physical reasoning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[16] Yuwei Bao, Keunwoo Yu, Yichi Zhang, Shane Storks, Itamar Bar-Yossef, Alex de la Iglesia, Megan Su, Xiao Zheng, and Joyce Chai. Can foundation models watch, talk and guide you step by step to make a cake? In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12325–12341, Singapore, 2023. Association for Computational Linguistics.

[17] Peter W. Battaglia, Jessica B. Hamrick, and Joshua B. Tenenbaum. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332, 2013.

[18] Yonatan Belinkov, Sebastian Gehrmann, and Ellie Pavlick. Interpretability and analysis in neural NLP. In Agata Savary and Yue Zhang, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 1–5, Online, 2020. Association for Computational Linguistics.

[19] Yonatan Belinkov, Adam Poliak, Stuart Shieber, Benjamin Van Durme, and Alexander Rush. Don't take the premise for granted: Mitigating artifacts in natural language inference. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 877–891, Florence, Italy, 2019. Association for Computational Linguistics.

[20] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. In *Advances in Neural Information Processing Systems*, volume 13, 2000.

[21] Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

[22] Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen-tau Yih, and Yejin Choi. Abductive commonsense reasoning. In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020.

[23] Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian. Experience grounds language. In Bonnie Webber, Trevor

Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8718–8735, Online, 2020. Association for Computational Linguistics.

[24] Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

[25] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

[26] Joan Bliss. Commonsense reasoning about the physical world. *Studies in Science Education*, 44(2):123–155, 2008.

[27] Dan Bohus, Sean Andrist, Nick Saw, Ann Paradiso, Ishani Chakraborty, and Mahdi Rad. Sigma: An open-source interactive system for mixed-reality task assistance research - extended abstract. In *2024 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE, 2024.

[28] Grady Booch, Francesco Fabiano, Lior Horesh, Kiran Kate, Jonathan Lenchner, Nick Linck, Andreas Loreggia, Keerthiram Murgesan, Nicholas Mattei, Francesca Rossi, and Biplav Srivastava. Thinking fast and slow in AI. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(17):15042–15046, 2021.

[29] George Boole. *An Investigation of the Laws of Thought: On Which Are Founded the Mathematical Theories of Logic and Probabilities*. Dover Publications, 1854.

[30] Antoine Bosselut, Omer Levy, Ari Holtzman, Corin Ennis, Dieter Fox, and Yejin Choi. Simulating Action Dynamics with Neural Process Networks. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018.

[31] Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. COMET: Commonsense transformers for automatic knowledge graph construction. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy, 2019. Association for Computational Linguistics.

[32] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, 2015. Association for Computational Linguistics.

[33] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[34] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020.

[35] Eugene Charniak, Yasemin Altun, Rodrigo de Salvo Braz, Benjamin Garrett, Margaret Kosmala, Tomer Moscovich, Lixin Pang, Changhee Pyo, Ye Sun, Wei Wy, Zhongfa Yang, Shawn Zeiler, and Lisa Zorn. Reading comprehension programs in a statistical-language-processing class. In *ANLP-NAACL 2000 Workshop: Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding Systems*, 2000.

[36] Stanley F Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, 13(4):359–394, 1999.

[37] Yangyi Chen, Karan Sikka, Michael Cogswell, Heng Ji, and Ajay Divakaran. Measuring and improving chain-of-thought reasoning in vision-language models. In *Proceedings of the 2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Mexico City, Mexico, 2024. Association for Computational Linguistics.

[38] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In Dekai Wu, Marine Carpuat, Xavier Carreras, and Eva Maria Vecchi, editors, *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, 2014. Association for Computational Linguistics.

[39] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4069–4082, Hong Kong, China, 2019. Association for Computational Linguistics.

[40] Andy Coenen, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda Viégas, and Martin Wattenberg. Visualizing and measuring the geometry of BERT. In *Advances in Neural Information Processing Systems*, volume 32, Red Hook, NY, USA, 2019. Curran Associates Inc.

[41] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

[42] Gautier Dagan, Frank Keller, and Alex Lascarides. Learning the effects of physical actions in a multi-modal environment. In Andreas Vlachos and Isabelle Augenstein, editors, *Findings of the Association for Computational Linguistics: EACL 2023*, pages 133–148, Dubrovnik, Croatia, 2023. Association for Computational Linguistics.

[43] Ido Dagan, Oren Glickman, and Bernardo Magnini. The PASCAL Recognising Textual Entailment Challenge. In Joaquin Quiñonero-Candela, Ido Dagan, Bernardo Magnini,

and Florence d'Alché-Buc, editors, *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, volume 3944, pages 177–190. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[44] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 49250–49267. Curran Associates, Inc., 2023.

[45] Wenliang Dai, Zihan Liu, Ziwei Ji, Dan Su, and Pascale Fung. Plausible may not be faithful: Probing object hallucination in vision-language pre-training. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2136–2148, Dubrovnik, Croatia, 2023. Association for Computational Linguistics.

[46] Bhavana Dalvi, Lifu Huang, Niket Tandon, Wen-tau Yih, and Peter Clark. Tracking state changes in procedural text: a challenge dataset and models for process paragraph comprehension. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1595–1604, New Orleans, LA, USA, 2018. Association for Computational Linguistics.

[47] Bhavana Dalvi, Peter Jansen, Oyvind Tafjord, Zhengnan Xie, Hannah Smith, Leighanna Pipatanangkura, and Peter Clark. Explaining answers with entailment trees. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7358–7370, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

[48] Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. Analyzing transformers in embedding space. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16124–16170, Toronto, Canada, 2023. Association for Computational Linguistics.

[49] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied Question Answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018. IEEE.

[50] Herbert Alan Davidson. *Alfarabi, Avicenna, and Averroes on Intellect: Their Cosmologies, Theories of the Active Intellect, and Theories of Human Intellect*. Oxford University Press on Demand, 1992.

[51] Ernest Davis. Logical Formalizations of Commonsense Reasoning: A Survey. *Journal of Artificial Intelligence Research*, 59:651–723, 2017.

[52] Ernest Davis. Benchmarks for automated commonsense reasoning: A survey. *ACM Comput. Surv.*, 56(4), 2023.

[53] Ernest Davis and Gary Marcus. Commonsense reasoning and commonsense knowledge in artificial intelligence. *Commun. ACM*, 58(9):92–103, 2015.

[54] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[55] Defense Advanced Research Projects Agency (DARPA). Broad Agency Announcement: Perceptually-enabled Task Guidance (PTG), 2021. BAA Release Date: March 5, 2021.

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

[57] Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online, 2020. Association for Computational Linguistics.

[58] Mingyu Ding, Zhenfang Chen, Tao Du, Ping Luo, Josh Tenenbaum, and Chuang Gan. Dynamic visual reasoning by learning differentiable physics models from video and language. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 887–899. Curran Associates, Inc., 2021.

[59] Y. Ding, J. Liu, J. Xiong, and Y. Shi. Revisiting the evaluation of uncertainty estimation and its application to explore model complexity-uncertainty trade-off. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 22–31, Los Alamitos, CA, USA, 2020. IEEE Computer Society.

[60] George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC '04)*, Lisbon, Portugal, 2004. European Language Resources Association (ELRA).

[61] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[62] Yuqing Du, Ksenia Konyushkova, Misha Denil, Akhil Raju, Jessica Landon, Felix Hill, Nando de Freitas, and Serkan Cabi. Vision-language models as success detectors. In Sarath Chandar, Razvan Pascanu, Hanie Sedghi, and Doina Precup, editors, *Proceedings of The 2nd Conference on Lifelong Learning Agents*, volume 232 of *Proceedings of Machine Learning Research*, pages 120–136. PMLR, 2023.

[63] Nouha Dziri, Ehsan Kamalloo, Kory Mathewson, and Osmar Zaiane. Evaluating coherence in dialogue systems using entailment. 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 3806–3812, Minneapolis, MN, USA, 2019. Association for Computational Linguistics.

[64] Allyson Ettinger, Ahmed Elgohary, Colin Phillips, and Philip Resnik. Assessing Composition in Sentence Vector Representations. In *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, NM, USA, 2018. Association for Computational Linguistics.

[65] J St BT Evans and Peter C Wason. Rationalization in a reasoning task. *British Journal of Psychology*, 67(4):479–486, 1976.

[66] Jonathan St B. T. Evans. Heuristic and analytic processes in reasoning. *British Journal of Psychology*, 75(4):451–468, 1984.

[67] Jonathan St BT Evans. Intuition and reasoning: A dual-process perspective. *Psychological Inquiry*, 21(4):313–326, 2010.

[68] Jonathan St.B.T. Evans. In two minds: dual-process accounts of reasoning. *Trends in Cognitive Sciences*, 7(10):454–459, 2003.

[69] Jonathan St.B.T. Evans. Dual-process theories of reasoning: Contemporary issues and developmental applications. *Developmental Review*, 31(2):86–102, 2011. Special Issue: Dual-Process Theories of Cognitive Development.

[70] Maxwell Forbes and Yejin Choi. Verb physics: Relative physical knowledge of actions and objects. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 266–276, Vancouver, Canada, 2017. Association for Computational Linguistics.

[71] Maxwell Forbes, Ari Holtzman, and Yejin Choi. Do neural language representations learn physical commonsense? In *Proceedings of the 41st Annual Conference of the Cognitive Science Society*, 2019.

[72] Marianna B. Ganapini, Murray Campbell, Francesco Fabiano, Lior Horesh, Jon Lenchner, Andrea Loreggia, Nicholas Mattei, Taher Rahgooy, Francesca Rossi, Biplav Srivastava, and Brent Venable. Combining fast and slow thinking for human-like and efficient navigation in constrained environments. In *16th International Workshop on Neural-Symbolic Learning and Reasoning*, 2022.

[73] Roy Ganz, Yair Kittenplon, Aviad Aberdam, Elad Ben Avraham, Oren Nuriel, Shai Mazor, and Ron Litman. Question aware vision transformer for multimodal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13861–13871, 2024.

[74] Qiaozi Gao, Malcolm Doering, Shaohua Yang, and Joyce Chai. Physical causality of action verbs in grounded language understanding. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1814–1824, Berlin, Germany, 2016. Association for Computational Linguistics.

[75] Qiaozi Gao, Shaohua Yang, Joyce Chai, and Lucy Vanderwende. What action causes this? towards naive physical action-effect prediction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 934–945, Melbourne, Australia, 2018. Association for Computational Linguistics.

[76] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv: 2312.11805*, 2023.

[77] Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235, Singapore, 2023. Association for Computational Linguistics.

[78] Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics.

[79] Mor Geva, Yoav Goldberg, and Jonathan Berant. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China, 2019. Association for Computational Linguistics.

[80] Mor Geva, Ankit Gupta, and Jonathan Berant. Injecting numerical reasoning skills into language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 946–958, Online, 2020. Association for Computational Linguistics.

[81] Daniel Gildea and Daniel Jurafsky. Automatic labeling of semantic roles. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 512–520, Hong Kong, 2000. Association for Computational Linguistics.

[82] John J. Godfrey and Edward Holliman. Switchboard-1 release 2. Linguistic Data Consortium, 1997.

[83] Andrew S. Gordon. Commonsense Interpretation of Triangle Behavior. In *Thirtieth AAAI Conference on Artificial Intelligence*, Phoenix, AZ, USA, 2016. AAAI Press.

[84] Jonathan Gordon and Benjamin Van Durme. Reporting bias and knowledge acquisition. In *Proceedings of the 2013 Workshop on Automated Knowledge Base Construction*, AKBC '13, pages 25–30, New York, NY, USA, 2013. Association for Computing Machinery.

[85] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4D: Around the World in 3,000 Hours of Egocentric Video. In *IEEE/CVF Computer Vision and Pattern Recognition (CVPR)*, 2022.

[86] Ralph Grishman and Beth Sundheim. Message Understanding Conference-6: A Brief History. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*, COLING '96, pages 466–471, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics.

[87] Lin Guan, Karthik Valmeekam, Sarath Sreedharan, and Subbarao Kambhampati. Leveraging pre-trained large language models to construct and utilize world models for model-based task planning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[88] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. HallusionBench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[89] Aditya Gupta and Greg Durrett. Effective use of transformer networks for entity tracking. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 759–769, Hong Kong, China, 2019. Association for Computational Linguistics.

[90] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, LA, USA, 2018. Association for Computational Linguistics.

[91] Lovisa Hagström and Richard Johansson. What do models learn from training on more than text? Measuring visual commonsense knowledge. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 252–261, Dublin, Ireland, 2022. Association for Computational Linguistics.

[92] Adi Haviv, Ido Cohen, Jacob Gidron, Roei Schuster, Yoav Goldberg, and Mor Geva. Understanding transformer memorization recall through idioms. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 248–264, Dubrovnik, Croatia, 2023. Association for Computational Linguistics.

[93] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. *arXiv: 2006.03654*, 2021.

[94] Lisa Anne Hendricks and Aida Nematzadeh. Probing image-language transformers for verb understanding. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3635–3644, Online, 2021. Association for Computational Linguistics.

[95] Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. Linearity of relation decoding in transformer language models. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.

[96] John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, MN, USA, 2019. Association for Computational Linguistics.

[97] Lynette Hirschman, Marc Light, Eric Breck, and John D. Burger. Deep read: A reading comprehension system. In *Proceedings of the 37th Annual Meeting of the*

Association for Computational Linguistics, pages 325–332, College Park, MD, USA, 1999. Association for Computational Linguistics.

[98] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, 1997.

[99] Yifan Hou, Jiaoda Li, Yu Fei, Alessandro Stolfo, Wangchunshu Zhou, Guangtao Zeng, Antoine Bosselut, and Mrinmaya Sachan. Towards a mechanistic interpretation of multi-step reasoning capabilities of language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4902–4919, Singapore, 2023. Association for Computational Linguistics.

[100] Wenyue Hua and Yongfeng Zhang. System 1 + system 2 = better world: Neural-symbolic chain of logic reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 601–612, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics.

[101] Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 9118–9147, Baltimore, MD, USA, 2022. PMLR.

[102] Drew A Hudson and Christopher D Manning. GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering. In *2019 IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 2019. IEEE.

[103] Taichi Iki and Akiko Aizawa. Effect of visual extensions on natural language understanding in vision-and-language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2189–2196, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

[104] Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. Adversarial example generation with syntactically controlled paraphrase networks. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, LA, USA, 2018. Association for Computational Linguistics.

[105] Alon Jacovi, Swabha Swayamdipta, Shauli Ravfogel, Yanai Elazar, Yejin Choi, and Yoav Goldberg. Contrastive explanations for model interpretability. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1597–1611, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

[106] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, 2019. Association for Computational Linguistics.

[107] Harsh Jhamtani and Peter Clark. Learning to explain: Datasets and models for identifying valid reasoning chains in multihop question-answering. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 137–150, Online, 2020. Association for Computational Linguistics.

[108] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12), 2023.

[109] Robin Jia and Percy Liang. Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pages 2021–2031, Copenhagen, Denmark, 2017. Association for Computational Linguistics.

[110] Woojeong Jin, Dong-Ho Lee, Chenguang Zhu, Jay Pujara, and Xiang Ren. Leveraging visual knowledge in language tasks: An empirical study on intermediate pre-training for cross-modal knowledge transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2750–2762, Dublin, Ireland, 2022. Association for Computational Linguistics.

[111] Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng LYU, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. CLadder: A benchmark to assess causal reasoning capabilities of language models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[112] Thorsten Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European Conference on Machine Learning*, pages 137–142. Springer, 1998.

[113] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017.

[114] Daniel Kahneman. *Thinking, Fast and Slow*. MacMillan, New York, 2011.

[115] Ivan Kapelyukh, Vitalis Vosylius, and Edward Johns. DALL-E-Bot: Introducing web-scale diffusion models to robotics. *IEEE Robotics and Automation Letters (RA-L)*, 2023.

[116] Mehran Kazemi, Najoung Kim, Deepti Bhatia, Xin Xu, and Deepak Ramachandran. LAMBADA: Backward chaining for automated reasoning in natural language. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6547–6568, Toronto, Canada, 2023. Association for Computational Linguistics.

[117] Wonjae Kim, Bokyung Son, and Ildoo Kim. ViLT: Vision-and-language transformer without convolution or region supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021.

[118] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, volume 35, 2022.

[119] Nikhil Krishnaswamy and James Pustejovsky. Multimodal semantic simulations of linguistically underspecified motion events. In *Spatial Cognition X*, pages 177–197. Springer, 2016.

[120] Nikhil Krishnaswamy and James Pustejovsky. VoxSim: A visual platform for modeling motion language. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 54–58, Osaka, Japan, 2016. The COLING 2016 Organizing Committee.

[121] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40:e253, 2017.

[122] George Lakoff. Linguistics and natural logic. *Synthese*, 22(1):151–271, 1970.

[123] Mirella Lapata. Probabilistic text structuring: Experiments with sentence ordering. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan, 2003. Association for Computational Linguistics.

[124] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62, 2022.

[125] Yoonjoo Lee, Kyungjae Lee, Sunghyun Park, Dasol Hwang, Jaehyeon Kim, Anthony Moi, Hong-in Lee, and Moontae Lee. QASA: Advanced question answering on scientific articles. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

[126] Teesid Leelasawassuk, Dima Damen, and Walterio Mayol-Cuevas. Automated capture and delivery of assistive task guidance with an eyewear computer: the GlaciAR system. In *Proceedings of the 8th Augmented Human International Conference*, AH '17, Silicon Valley, CA, USA, 2017. Association for Computing Machinery.

[127] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13872–13882, 2024.

[128] Hector J. Levesque, Ernest Davis, and Leora Morgenstern. The Winograd schema challenge. In *Proceedings of the Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning (KR2012)*, Rome, Italy, 2012. AAAI Press.

[129] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, 2020. Association for Computational Linguistics.

[130] Belinda Z. Li, Maxwell Nye, and Jacob Andreas. Language modeling with latent situations. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12556–12571, Toronto, Canada, 2023. Association for Computational Linguistics.

[131] Fangjun Li, David C. Hogg, and Anthony G. Cohn. Exploring the GLIDE model for human action effect prediction. In Patrizia Paggio, Albert Gatt, and Marc Tanti, editors, *Proceedings of the 2nd Workshop on People in Vision, Language, and the Mind*, pages 1–5, Marseille, France, 2022. European Language Resources Association.

[132] Jiwei Li, Will Monroe, and Dan Jurafsky. Understanding neural networks through representation erasure. *arXiv: 1612.08220*, 2016.

[133] 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 *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

[134] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Proceedings of the 39th International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022.

[135] Tao Li, Vivek Gupta, Maitrey Mehta, and Vivek Srikumar. A logic-driven framework for consistency of neural models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, 2019. Association for Computational Linguistics.

[136] 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 *European Conference on Computer Vision*, pages 121–137. Springer, 2020.

[137] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305, Singapore, 2023. Association for Computational Linguistics.

[138] Yong-Lu Li, Yue Xu, Xinyu Xu, Xiaohan Mao, Yuan Yao, Siqi Liu, and Cewu Lu. Beyond object recognition: A new benchmark towards object concept learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20029–20040, 2023.

[139] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26296–26306, 2024.

[140] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916. Curran Associates, Inc., 2023.

[141] Ruibo Liu, Jason Wei, Shixiang Shane Gu, Te-Yen Wu, Soroush Vosoughi, Claire Cui, Denny Zhou, and Andrew M. Dai. Mind's eye: Grounded language model reasoning through simulation. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

[142] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv: 1907.11692*, 2019.

[143] Adian Liusie, Vatsal Raina, Vyas Raina, and Mark Gales. Analyzing biases to spurious correlations in text classification tasks. In Yulan He, Heng Ji, Sujian Li, Yang Liu, and Chua-Hui Chang, editors, *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 78–84, Online, 2022. Association for Computational Linguistics.

[144] Lajanugen Logeswaran, Honglak Lee, and Dragomir Radev. Sentence ordering and coherence modeling using recurrent neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-18)*, New Orleans, LA, USA, 2018.

[145] Jieyi Long. Large language model guided tree-of-thought. *arXiv: 2305.08291*, 2023.

[146] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018.

[147] Yujie Lu, Wanrong Zhu, Xin Wang, Miguel Eckstein, and William Yang Wang. Imagination-augmented natural language understanding. In *Proceedings of the 2022*

Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 4392–4402, Seattle, WA, USA, 2022. Association for Computational Linguistics.

[148] Hans Peter Luhn. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1(4):309–317, 1957.

[149] Kaixin Ma, Filip Ilievski, Jonathan Francis, Eric Nyberg, and Alessandro Oltramari. Coalescing global and local information for procedural text understanding. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1534–1545, Gyeongju, Republic of Korea, 2022. International Committee on Computational Linguistics.

[150] Bill MacCartney and Christopher D. Manning. Natural Logic for Textual Inference. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 193–200, Prague, Czech Republic, 2007. Association for Computational Linguistics.

[151] Cheahuychou Mao. *Understanding language through visual imagination*. PhD thesis, Massachusetts Institute of Technology, 2019.

[152] Gary Marcus. The next decade in AI: Four steps towards robust artificial intelligence. *arXiv: 2002.06177*, 2020.

[153] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 1(2), 1993.

[154] John McCarthy. Programs with Common Sense. In *Semantic Information Processing*, pages 403–418. MIT Press, 1968.

[155] Tom McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, 2019. Association for Computational Linguistics.

[156] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

[157] Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, volume 35, 2022.

[158] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the 1st International Conference on Learning Representations (ICLR 2013)*, Scottsdale, AZ, USA, 2013.

[159] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed Representations of Words and Phrases and their Compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26, pages 3111–3119. Curran Associates, Inc., 2013.

[160] Hilary E Miller, Heather L Kirkorian, and Vanessa R Simmering. Using eye-tracking to understand relations between visual attention and language in children's spatial skills. *Cognitive Psychology*, 117:101264, 2020.

[161] Junghyun Min, R. Thomas McCoy, Dipanjan Das, Emily Pitler, and Tal Linzen. Syntactic Data Augmentation Increases Robustness to Inference Heuristics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, Online, 2020. Association for Computational Linguistics.

[162] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, Xiao Wang, Xiaohua Zhai, Thomas Kipf, and Neil Houlsby. Simple open-vocabulary object detection. In *European Conference on Computer Vision*, page 728–755, Berlin, Heidelberg, 2022. Springer-Verlag.

[163] Marvin Minsky. Commonsense-based interfaces. *Communications of the ACM*, 43(8):66–73, 2000.

[164] Roshanak Mirzaee, Hossein Rajaby Faghihi, Qiang Ning, and Parisa Kordjamshidi. SPARTQA: A textual question answering benchmark for spatial reasoning. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4582–4598, Online, 2021. Association for Computational Linguistics.

[165] Hosein Mohebbi, Willem Zuidema, Grzegorz Chrupała, and Afra Alishahi. Quantifying context mixing in transformers. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3378–3400, Dubrovnik, Croatia, 2023. Association for Computational Linguistics.

[166] Robert C. Moore. The Role of Logic in Knowledge Representation and Commonsense Reasoning. In *Proceedings of the Second AAAI Conference on Artificial Intelligence (AAAI '82)*, AAAI'82, pages 428–433, Pittsburgh, PA, USA, 1982. AAAI Press.

[167] Augustus De Morgan. *Formal Logic: Or, The Calculus of Inference, Necessary and Probable*. Taylor and Walton, 1847.

[168] Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. A corpus and cloze evaluation for deeper understanding of commonsense stories. In Kevin Knight, Ani Nenkova, and

Owen Rambow, editors, *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, CA, USA, 2016. Association for Computational Linguistics.

[169] Nasrin Mostafazadeh, Aditya Kalyanpur, Lori Moon, David Buchanan, Lauren Berkowitz, Or Biran, and Jennifer Chu-Carroll. GLUCOSE: GeneraLized and COntextualized story explanations. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4569–4586, Online, 2020. Association for Computational Linguistics.

[170] W James Murdoch, Peter J Liu, and Bin Yu. Beyond word importance: Contextual decomposition to extract interactions from LSTMs. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018.

[171] Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using Bayesian binning. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 29, 2015.

[172] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

[173] Soroush Nasiriany, Fei Xia, Wenhao Yu, Ted Xiao, Jacky Liang, Ishita Dasgupta, Annie Xie, Danny Driess, Ayzaan Wahid, Zhuo Xu, Quan Vuong, Tingnan Zhang, Tsang-Wei Edward Lee, Kuang-Huei Lee, Peng Xu, Sean Kirmani, Yuke Zhu, Andy Zeng, Karol Hausman, Nicolas Heess, Chelsea Finn, Sergey Levine, and Brian Ichter. PIVOT: Iterative visual prompting elicits actionable knowledge for VLMs. *arXiv: 2402.07872*, 2024.

[174] Hwee Tou Ng, Leong Hwee Teo, and Jennifer Lai Pheng Kwan. A Machine Learning Approach to Answering Questions for Reading Comprehension Tests. In *2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 124–132, Hong Kong, China, 2000. Association for Computational Linguistics.

[175] Nguyen Nguyen, Jing Bi, Ali Vosoughi, Yapeng Tian, Pooyan Fazli, and Chenliang Xu. Oscar: Object state captioning and state change representation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, Mexico City, Mexico, 2024. Association for Computational Linguistics.

[176] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial NLI: A new benchmark for natural language understanding. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online, 2020. Association for Computational Linguistics.

[177] Timothy Niven and Hung-Yu Kao. Probing Neural Network Comprehension of Natural Language Arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, Florence, Italy, 2019. Association for Computational Linguistics.

[178] Geoffrey Nunberg. Position Paper on Common-Sense and Formal Semantics. In *Proceedings of the 1987 Workshop on Theoretical Issues in Natural Language Processing (TINLAP '87)*, TINLAP '87, pages 129–133, Las Cruces, NM, USA, 1987. Association for Computational Linguistics.

[179] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. In *Proceedings of the 10th International Conference on Learning Representations*, 2022.

[180] Jennifer Ockerman and Amy Pritchett. A review and reappraisal of task guidance: Aiding workers in procedure following. *International Journal of Cognitive Ergonomics*, 4(3):191–212, 2000.

[181] Jennifer J Ockerman and Amy R Pritchett. Preliminary investigation of wearable computers for task guidance in aircraft inspection. In *Digest of Papers. Second International Symposium on Wearable Computers (Cat. No. 98EX215)*, pages 33–40. IEEE, 1998.

[182] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744, 2022.

[183] Rohith Peddi, Shivvrat Arya, Bharath Challa, Likhitha Pallapothula, Akshay Vyas, Jikai Wang, Qifan Zhang, Vasundhara Komaragiri, Eric Ragan, Nicholas Ruozzi, Yu Xiang, and Vibhav Gogate. CaptainCook4D: A dataset for understanding errors in procedural activities. *arXiv: 2312.14556*, 2023.

[184] Charles S. Peirce. A Theory of Probable Inference. In Charles S. Peirce, editor, *Studies in Logic by Members of the Johns Hopkins University*, pages 126–181. Little, Brown, and Company, Boston, MA, USA, 1883.

[185] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv: 2306.14824*, 2023.

[186] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, 2014. Association for Computational Linguistics.

[187] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, LA, USA, 2018. Association for Computational Linguistics.

[188] Matthew E. Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Brussels, Belgium, 2018. Association for Computational Linguistics.

[189] Preston Pierce. Software verification and validation. In *IEEE Technical Applications Conference. Northcon/96. Conference Record*, pages 265–268, 1996.

[190] Adam Poliak, Aparajita Haldar, Rachel Rudinger, J. Edward Hu, Ellie Pavlick, Aaron Steven White, and Benjamin Van Durme. Collecting diverse natural language inference problems for sentence representation evaluation. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 67–81, Brussels, Belgium, 2018. Association for Computational Linguistics.

[191] Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. Hypothesis Only Baselines in Natural Language Inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, New Orleans, LA, USA, 2018. Association for Computational Linguistics.

[192] Vasin Punyakanok, Dan Roth, Wen-tau Yih, and Dav Zimak. Semantic role labeling via integer linear programming inference. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 1346–1352, Geneva, Switzerland, 2004. Association for Computational Linguistics.

[193] 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 *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[194] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI*, 2018.

[195] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[196] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[197] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

[198] Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy, 2019. Association for Computational Linguistics.

[199] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv: 2204.06125*, 2022.

[200] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[201] Arvin Christopher C Reyes, Neil Patrick A Del Gallego, and Jordan Aiko P Deja. Mixed reality guidance system for motherboard assembly using tangible augmented reality. In *Proceedings of the 2020 4th International Conference on Virtual and Augmented Reality Simulations*, pages 1–6, 2020.

[202] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[203] Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online, 2020. Association for Computational Linguistics.

[204] Kyle Richardson, Ronen Tamari, Oren Sultan, Dafna Shahaf, Reut Tsarfaty, and Ashish Sabharwal. Breakpoint transformers for modeling and tracking intermediate beliefs. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9703–9719, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics.

[205] Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. Choice of Plausible Alternatives: An Evaluation of Commonsense Causal Reasoning. In *AAAI Spring Symposium on Logical Formalizations of Commonsense Reasoning*, Stanford, CA, USA, 2011.

[206] Paul Roit, Johan Ferret, Lior Shani, Roee Aharoni, Geoffrey Cideron, Robert Dadashi, Matthieu Geist, Sertan Girgin, Leonard Hussenot, Orgad Keller, Nikola Momchev, Sabela Ramos Garea, Piotr Stanczyk, Nino Vieillard, Olivier Bachem, Gal Elidan,

Avinatan Hassidim, Olivier Pietquin, and Idan Szpektor. Factually consistent summarization via reinforcement learning with textual entailment feedback. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6252–6272, Toronto, Canada, 2023. Association for Computational Linguistics.

[207] Ronald Rosenfeld. Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE*, 88(8):1270–1278, 2000.

[208] Roni Rosenfeld and Philip Clarkson. Statistical language modeling using the CMU-Cambridge toolkit, 1997.

[209] Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. Gender Bias in Coreference Resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2018)*, pages 8–14, New Orleans, LA, USA, 2018. Association for Computational Linguistics.

[210] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[211] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. In *Proceedings of the 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing at NeurIPS 2019*, Vancouver, Canada, 2019.

[212] Maarten Sap, Ronan LeBras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Inteligence*, Honolulu, HI, USA, 2019. AAAI Press.

[213] Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China, 2019. Association for Computational Linguistics.

[214] Maarten Sap, Vered Shwartz, Antoine Bosselut, Yejin Choi, and Dan Roth. Commonsense reasoning for natural language processing. In Agata Savary and Yue Zhang, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 27–33, Online, 2020. Association for Computational Linguistics.

[215] Roy Schwartz, Maarten Sap, Ioannis Konstas, Leila Zilles, Yejin Choi, and Noah A. Smith. The effect of different writing tasks on linguistic style: A case study of the ROC story cloze task. In Roger Levy and Lucia Specia, editors, *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 15–25, Vancouver, Canada, 2017. Association for Computational Linguistics.

[216] Claude Elwood Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[217] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. ALFRED: A benchmark for interpreting grounded instructions for everyday tasks. In *The IEEE Conference on Computer Vision and Pattern Recognition*, Online, 2020.

[218] Chenglei Si, Dan Friedman, Nitish Joshi, Shi Feng, Danqi Chen, and He He. Measuring inductive biases of in-context learning with underspecified demonstrations. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11289–11310, Toronto, Canada, 2023. Association for Computational Linguistics.

[219] Chandan Singh, Jeevana Priya Inala, Michel Galley, Rich Caruana, and Jianfeng Gao. Rethinking interpretability in the era of large language models. *arXiv: 2402.01761*, 2024.

[220] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. ProgPrompt: program generation for situated robot task planning using large language models. *Autonomous Robots*, 47(8):999–1012, 2023.

[221] Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.

[222] Charles Spearman. The proof and measurement of association between two things. *American Journal of Psychology*, 15:72–101, 1904.

[223] Evangelia Spiliopoulou, Artidoro Pagnoni, Yonatan Bisk, and Eduard Hovy. EvEntS ReaLM: Event reasoning of entity states via language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1982–1997, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics.

[224] Tejas Srinivasan, Jack Hessel, Tanmay Gupta, Bill Yuchen Lin, Yejin Choi, Jesse Thomason, and Khyathi Chandu. Selective "selective prediction": Reducing unnecessary abstention in vision-language reasoning. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12935–12948, Bangkok, Thailand, 2024. Association for Computational Linguistics.

[225] Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7035–7052, Singapore, 2023. Association for Computational Linguistics.

[226] Alessandro Stolfo, Zhijing Jin, Kumar Shridhar, Bernhard Schoelkopf, and Mrinmaya Sachan. A causal framework to quantify the robustness of mathematical reasoning with language models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 545–561, Toronto, Canada, 2023. Association for Computational Linguistics.

[227] Shane Storks, Qiaozi Gao, and Joyce Y Chai. Recent advances in natural language inference: A survey of benchmarks, resources, and approaches. *arXiv:1904.01172*, 2020.

[228] Shane Storks, Qiaozi Gao, Yichi Zhang, and Joyce Chai. Tiered reasoning for intuitive physics: Toward verifiable commonsense language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4902–4918, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

[229] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun, editors, *Proceedings of the 2nd International Conference on Learning Representations*, Banff, Canada, 2014.

[230] Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. ProofWriter: Generating implications, proofs, and abductive statements over natural language. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3621–3634, Online, 2021. Association for Computational Linguistics.

[231] Hao Tan and Mohit Bansal. LXMERT: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2019.

[232] Niket Tandon, Keisuke Sakaguchi, Bhavana Dalvi, Dheeraj Rajagopal, Peter Clark, Michal Guerquin, Kyle Richardson, and Eduard Hovy. A dataset for tracking entities in open domain procedural text. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6408–6417, Online, 2020. Association for Computational Linguistics.

[233] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT Rediscovers the Classical NLP Pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, Florence, Italy, 2019. Association for Computational Linguistics.

[234] Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. What do you learn from context? Probing for sentence structure in contextualized word representations. In *Proceedings of the 6th International Conference on Learning Representations (ICLR 2018)*, Vancouver, Canada, 2018.

[235] Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.

[236] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. *arXiv: 2302.13971*, 2023.

[237] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *arXiv: 2307.09288*, 2023.

[238] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. In *Advances in Neural Information Processing Systems*, volume 34, pages 200–212, 2021.

[239] Takeo Tsujii and Shigeru Watanabe. Neural correlates of dual-task effect on belief-bias syllogistic reasoning: a near-infrared spectroscopy study. *Brain research*, 1287:118–125, 2009.

[240] Shagun Uppal, Vivek Gupta, Avinash Swaminathan, Haimin Zhang, Debanjan Mahata, Rakesh Gosangi, Rajiv Ratn Shah, and Amanda Stent. Two-step classification using recasted data for low resource settings. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, Suzhou, China, 2020. Association for Computational Linguistics.

[241] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[242] Ramakrishna Vedantam, Xiao Lin, Tanmay Batra, C Lawrence Zitnick, and Devi Parikh. Learning common sense through visual abstraction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2542–2550, 2015.

[243] Elena Voita, Rico Sennrich, and Ivan Titov. The bottom-up evolution of representations in the transformer: A study with machine translation and language modeling objectives. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4396–4406, Hong Kong, China, 2019. Association for Computational Linguistics.

[244] Ellen M. Voorhees. The TREC-8 Question Answering Track. *Natural Language Engineering*, 7:361–378, 2000.

[245] David Wan, Jaemin Cho, Elias Stengel-Eskin, and Mohit Bansal. Contrastive region guidance: Improving grounding in vision-language models without training. In *Proceedings of the 2024 European Conference on Computer Vision*, 2024.

[246] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[247] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, Brussels, Belgium, 2018. Association for Computational Linguistics.

[248] Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

[249] Xin Wang, Taein Kwon, Mahdi Rad, Bowen Pan, Ishani Chakraborty, Sean Andrist, Dan Bohus, Ashley Feniello, Bugra Tekin, Felipe Vieira Frujeri, Neel Joshi, and Marc Pollefeys. Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20270–20281, 2023.

[250] Xuan Wang, SK Ong, and Andrew Yeh-Ching Nee. Multi-modal augmented-reality assembly guidance based on bare-hand interface. *Advanced Engineering Informatics*, 30(3):406–421, 2016.

[251] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

[252] Peter C Wason and J St BT Evans. Dual processes in reasoning? *Cognition*, 3(2):141–154, 1974.

[253] Albert Webson and Ellie Pavlick. Do prompt-based models really understand the meaning of their prompts? In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2300–2344, Seattle, WA, USA, 2022. Association for Computational Linguistics.

[254] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[255] Sean Welleck, Jason Weston, Arthur Szlam, and Kyunghyun Cho. Dialogue natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, Florence, Italy, 2019. Association for Computational Linguistics.

[256] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, LA, USA, 2018. Association for Computational Linguistics.

[257] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, 2020. Association for Computational Linguistics.

[258] Tongshuang Wu, Ellen Jiang, Aaron Donsbach, Jeff Gray, Alejandra Molina, Michael Terry, and Carrie J Cai. Promptchainer: Chaining large language model prompts through visual programming. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pages 1–10, 2022.

[259] Jie Yang and Yue Zhang. NCRF++: An open-source neural sequence labeling toolkit. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2018.

[260] Yue Yang, Wenlin Yao, Hongming Zhang, Xiaoyang Wang, Dong Yu, and Jianshu Chen. Z-LaVI: Zero-shot language solver fueled by visual imagination. In *Proceedings*

of the 2022 Conference on Empirical Methods in Natural Language Processing, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics.

[261] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822. Curran Associates, Inc., 2023.

[262] Kayo Yin and Graham Neubig. Interpreting language models with contrastive explanations. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 184–198, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics.

[263] Tianhe Yu, Ted Xiao, Austin Stone, Jonathan Tompson, Anthony Brohan, Su Wang, Jaspiar Singh, Clayton Tan, Dee M, Jodilyn Peralta, Brian Ichter, Karol Hausman, and Fei Xia. Scaling robot learning with semantically imagined experience. *arXiv: 2302.11550*, 2023.

[264] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

[265] Tian Yun, Chen Sun, and Ellie Pavlick. Does vision-and-language pretraining improve lexical grounding? In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4357–4366, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

[266] Eric Zelikman, Jesse Mu, Noah D. Goodman, and Yuhuai Tony Wu. STaR: Self-taught reasoner bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[267] Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium, 2018. Association for Computational Linguistics.

[268] Rowan Zellers, Ari Holtzman, Matthew Peters, Roozbeh Mottaghi, Aniruddha Kembhavi, Ali Farhadi, and Yejin Choi. PIGLeT: Language grounding through neuro-symbolic interaction in a 3D world. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2040–2050, Online, 2021. Association for Computational Linguistics.

[269] Chen Zhang and Joyce Y. Chai. Towards Conversation Entailment: An Empirical Investigation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP 2010)*, pages 756–766, Cambridge, MA, USA, 2010. Association for Computational Linguistics.

[270] Chenyu Zhang, Benjamin Van Durme, Zhuowan Li, and Elias Stengel-Eskin. Visual commonsense in pretrained unimodal and multimodal models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5321–5335, Seattle, WA, USA, 2022. Association for Computational Linguistics.

[271] Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, and Yinfei Yang. Ferret-v2: An improved baseline for referring and grounding with large language models. *arXiv: 2404.07973*, 2024.

[272] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. VinVL: Making visual representations matter in vision-language models. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[273] Sheng Zhang, Rachel Rudinger, Kevin Duh, and Benjamin Van Durme. Ordinal Common-Sense Inference. *Transactions of the Association for Computational Linguistics*, 5:379–395, 2017.

[274] Yichi Zhang, Jiayi Pan, Yuchen Zhou, Rui Pan, and Joyce Chai. Grounding visual illusions in language: Do vision-language models perceive illusions like humans? In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5718–5728, Singapore, 2023. Association for Computational Linguistics.

[275] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. ERNIE: Enhanced language representation with informative entities. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy, 2019. Association for Computational Linguistics.

[276] Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. Explainability for large language models: A survey. *ACM Trans. Intell. Syst. Technol.*, 15(2), 2024.

[277] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc., 2023.

[278] Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. "going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3363–3369, Hong Kong, China, 2019. Association for Computational Linguistics.

[279] Ben Zhou, Kyle Richardson, Qiang Ning, Tushar Khot, Ashish Sabharwal, and Dan Roth. Temporal reasoning on implicit events from distant supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1361–1371, Online, 2021. Association for Computational Linguistics.

[280] Kaiwen Zhou, Kwonjoon Lee, Teruhisa Misu, and Xin Wang. ViCor: Bridging visual understanding and commonsense reasoning with large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10783–10795, Bangkok, Thailand, 2024. Association for Computational Linguistics.