Ph.D. DISSERTATION

Embodied Language Agents for Real-World Robotic Tasks: Visually-Grounded Interaction and Skill Learning

실세계 로봇 작업을 위한 체화된 언어 에이전트: 시각 기반 대화와 스킬 학습

February 2025

Interdisciplinary Program in Artificial Intelligence
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Gi-Cheon Kang

Ph.D. DISSERTATION

Embodied Language Agents for Real-World Robotic Tasks: Visually-Grounded Interaction and Skill Learning

실세계 로봇 작업을 위한 체화된 언어 에이전트: 시각 기반 대화와 스킬 학습

February 2025

Interdisciplinary Program in Artificial Intelligence
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Gi-Cheon Kang

Embodied Language Agents for Real-World Robotic Tasks: Visually-Grounded Interaction and Skill Learning

실세계 로봇 작업을 위한 체화된 언어 에이전트: 시각 기반 대화와 스킬 학습

지도교수 장 병 탁

이 논문을 공학박사학위논문으로 제출함

2025 년 02 월

서울대학교 대학원

협동과정 인공지능

강 기 천

강 기 천의 공학박사 학위논문을 인준함

2024 년 12 월

위 원 장	박 재 흥	(인)
부위원장	장 병 탁	(인)
위 원	김 건 희	(인)
위 원	최 종 현	(인)
위 원	김 진 화	(인)

Abstract

Building intelligent agents capable of understanding natural language instructions and performing various real-world tasks has been a longstanding goal of artificial intelligence. Advances in machine learning and natural language processing have significantly enhanced these agents' capabilities, enabling them to use language as *a vehicle for communication, reasoning, and learning*. Agents with such capabilities are often referred to as language agents. This dissertation explores *embodied language agents* that interpret human instructions and execute robotic tasks in a physical environment. Based on my previous works [Kang et al., 2023, 2024a,b], three challenges for embodied language agents are discussed, each building upon a core capability of language agents: 1) visually-grounded communication, 2) reasoning about underspecified instructions, and 3) learning robotic skills from natural language.

Visually-grounded communication is a core capability of embodied language agents, as it enables agents to effectively communicate with humans about their visual perceptions. We explore how embodied language agents can enhance their robustness and generalization capability for visually-grounded communication. To this end, we introduce a semi-supervised learning approach called generative self-training (GST). A key idea of GST is to generate artificial visual dialog data based on huge amounts of unlabeled images on the Web and use it for training. GST demonstrates significant performance improvements in both adversarial robustness and generalization to unseen data.

Natural language is inherently ambiguous and context-dependent. Thus, we discuss how embodied language agents can reason about underspecified in-

structions like "My device runs out of battery" and grasp desired objects (*e.g.,* charger) through human-robot interaction. Specifically, our proposed system generates questions for humans to disambiguate the target object in the scene and keeps updating its belief for each object candidate based on human answers. We call the process of evaluating how well each object candidate explains the current visual and dialogue context as *pragmatic inference*. Experimental results show that pragmatic inference improves the target object discovery and the task success rate of object grasping when given underspecified instruction.

The third topic discusses how language can be used as an interface for robot learning. The goal is to train language-conditioned robotic policies only through language supervision. We first present a language-based teleoperation system for robot data collection. Then, we introduce a vision-language-action (VLA) model that learns language-conditioned policies directly from the language supervision, which we call CLIP-RT (CLIP-based Robotics Transformer). Inspired by CLIP which uses natural language as a training signal, CLIP-RT extends this idea to robot learning. Specifically, CLIP-RT treats language supervision from humans (*e.g.,*'move the arm forward.") as supervision for robotic policies and learns to optimize the similarity between the language supervision and the robot's current state through contrastive learning. CLIP-RT demonstrates strong capabilities in learning novel robotic skills, outperforming the prior art.

Keywords: Deep learning, robot learning, human-robot interaction, visually-grounded dialog, pragmatic inference, language-conditioned policy learning
Student Number: 2020-36496

Contents

Abstract **i**

Chapter 1 Introduction **1**

1.1 Intelligent Agents: A Conceptual Framework 1

1.2 Language Agents 1

1.3 Embodied Language Agents 3

1.4 Organization of the Dissertation 6

Chapter 2 Background: Robots That Use Language **8**

2.1 Introduction 8

2.2 Communication 9

2.3 Reasoning and Planning 11

2.4 Learning and Control 12

2.5 Conclusion 13

Chapter 3 Visually-Grounded Communication **14**

3.1 Introduction 14

3.2 Related Work 16

3.2.1 Visual Question Answering 16

3.2.2 Visual Dialog . . . 17
3.2.3 Neural Dialog Generation . . . 17
3.3 Approach . . . 18
3.3.1 Preliminaries . . . 18
3.3.2 Generative Self-Training (GST) . . . 19
3.4 Experiments . . . 24
3.4.1 Visual Dialog Data . . . 24
3.4.2 Synthetic Data . . . 24
3.4.3 Evaluation Protocol . . . 24
3.4.4 Implementation . . . 25
3.5 Visual Dialog Results . . . 26
3.5.1 Comparison with State-of-the-Art . . . 26
3.5.2 Ablation Study . . . 28
3.5.3 Analysis on the Low-Data Regime . . . 30
3.6 Adversarial Robustness Analysis . . . 31
3.6.1 Adversarial Robustness Against Visual Attacks . . . 31
3.6.2 Adversarial Robustness Against Textual Attacks . . . 34
3.7 Human Evaluation . . . 36
3.8 Qualitative Analysis . . . 37
3.8.1 Comparison Between Silver and Gold Data . . . 37
3.8.2 A Visualization of Answer Predictions . . . 38
3.9 Discussions . . . 39
3.10 Conclusion . . . 39

Chapter 4 Reasoning about Underspecified Instructions 41

4.1 Introduction . . . 41
4.2 Related Work . . . 45

4.2.1 Language-Guided Object Grasping 45
4.2.2 Pragmatics 46
4.3 Approach 46
4.3.1 Background 46
4.3.2 Problem Statement 47
4.3.3 Pragmatic Object Grasping (PROGrasp) 48
4.4 Experimental Setup 51
4.4.1 Dataset 51
4.4.2 Robotic Platform 52
4.4.3 Compared Methods 52
4.5 Results and Discussions 53
4.5.1 Results on Offline Experiments 53
4.5.2 Results on Online Experiments 59
4.5.3 Qualitative Analysis 61
4.6 Conclusion 62

Chapter 5 Learning Robotic Skills from Natural Language 63

5.1 Introduction 63
5.2 Related Work 66
5.2.1 Collection of Real-World Robot Data 66
5.2.2 Language-Conditioned Robotic Policies 67
5.2.3 Vision-Language-Action (VLA) Models 67
5.3 Approach 68
5.3.1 Preliminaries 68
5.3.2 CLIP-based Robotics Transformer (CLIP-RT) 70
5.3.3 In-Domain Robot Data Collection 75
5.4 Experimental Setup 78

5.4.1 Common Tasks 78
5.4.2 Novel Tasks 81
5.4.3 Data 83
5.4.4 Compared Methods 84
5.4.5 Robotic Platform 85
5.5 Results and Discussions 85
5.5.1 Comparison with State-of-the-Art 86
5.5.2 Ablation Studies 88
5.6 Conclusion 89

Chapter 6 Concluding Remarks **90**
6.1 Summary of Methods and Contributions 90
6.2 Suggestions for Future Research 92
6.2.1 Lifelong Learning 92
6.2.2 Long-Horizon Task Execution 93

요약 **126**

List of Figures

Figure 1.1 A conceptual framework of the agent from Russell and Norvig [2016]. 2

Figure 3.1 An overview of generative self-training (GST). 19

Figure 3.2 A detailed architecture of our proposed model. We visualize the following: (a) an encoder-decoder model, where the encoder aggregates the given multimodal context and the decoder generates the target sentence; and (b) a more detailed view of the encoder. TRM and Co-TRM denote the transformer module and the co-attentional transformer module, respectively. $\oplus$ is the concatenation operation. 25

Figure 3.3 Adversarial robustness against FGSM attack on the VisDial v1.0 validation split. We report NDCG scores of each model. 32

Figure 3.4 A visualization of cosine similarities between clean and perturbed image features in both input and output levels. We employ the FGSM attack with $\epsilon = 0.1$ to corrupt the clean images. 33
Figure 3.5 A visualization of the gold and the silver data on the VisDial v1.0 validation split. 38
Figure 3.6 A visualization of answer predictions from the student and the teacher models. The red-colored text is an incorrect answer. The blue-colored text is not a ground-truth answer, but it seems correct or plausible. 40

Figure 4.1 Overview of interactive object grasping with intention-oriented instruction. The initial instruction does not contain the target object's category. 43
Figure 4.2 Illustration of the inference step in PROGrasp for $T = 1$. VG first performs object grounding using the dialogue history. Q-gen then selects the object candidate to ask and generates a question. After obtaining the response from the user, VG cooperates with A-int to determine the target object region. The object-grasping module finally grasps the object by computing the 3D coordinates of the target object. 48
Figure 4.3 The 86 categories of everyday objects used in the experiments. 52
Figure 4.4 A text prompt for foundation models. 58
Figure 4.5 Validation scores adjusting the hyperparameters, λ and T. 58

Figure 4.6 Visualization of PROGrasp's target object recovery. . . . 61
Figure 4.7 Visualization of the failure cases. 61

Figure 5.1 Overview of language-based teleoperation. 64
Figure 5.2 Overview of CLIP-RT. In practice, we add a simple text prompt to language instructions: *What motion should the robot arm perform to complete the instruction* {*instruction*}*?* 70
Figure 5.3 A list of predefined natural language supervisions. 74
Figure 5.4 A text prompt for language-based teleoperation. 76
Figure 5.5 A simplified 2D example of stochastic trajectory diversification. (a): a demonstration trajectory from the start s to the endpoint e, passing through a waypoint w_1. (b): a sampled trajectory generated by the diversification phase. (c)-(e): a visualization of the recovery phase. 77
Figure 5.6 An example of a task with "point to the blue cup". . . . 78
Figure 5.7 An example of a task with "pull out the tissue". 78
Figure 5.8 An example of a task with "place the green cup on the red box". 79
Figure 5.9 An example of a task with "pick up the banana". 79
Figure 5.10 An example of a task with "push the red dice to the blue dice". 79
Figure 5.11 An example of a task with "flip the yellow cup". 79
Figure 5.12 An example of a task with "knock over the blue cup". . 80
Figure 5.13 An example of a task with "slide the green car to the Piglet". 80

Figure 5.14 An example of a task with “move the blue cup on the yellow circle”. 80
Figure 5.15 An example of a task with “pour the dog food in the bowl”. 81
Figure 5.16 An example of a task with “draw a line from A to B”. . 81
Figure 5.17 An example of a task with “open the cabinet”. 81
Figure 5.18 An example of a task with “play with the car”. 82
Figure 5.19 An example of a task with “erase the whiteboard”. . . . 82
Figure 5.20 An example of a task with “close the laptop”. 82
Figure 5.21 An example of a task with “open the trashcan”. 82
Figure 5.22 An example of a task with “stamp next to the star”. . . 83
Figure 5.23 An example of a task with “hide the Pooh with the green cup”. 83
Figure 5.24 An example of a task with “hang the cup”. 83
Figure 5.25 A robotic platform used in the experiments. 85
Figure 5.26 Success rates on nine Common Tasks (top) and ten Novel Tasks (bottom). We conduct experiments using all compared methods on Common tasks and three models (OpenVLA, CLIP-RT, and CLIP-RT-Action) on Novel Tasks. The success rate for each task is measured by averaging the results of ten trials. Average success rates of all tasks are shown on the left for both Common and Novel task sets. Tasks are arranged from left to right based on the average number of steps per episode. The task on the right indicates that it requires more steps on average compared with the task on the left. 87

List of Tables

Table 3.1 Comparison with the state-of-the-art generative models on both the VisDial v0.9 and v1.0 validation datasets. ↑ indicates higher is better. ↓ indicates lower is better. NDCG is not supported in v0.9 dataset. † denotes that the models are re-implemented by the previous work [Gan et al., 2019]. The standard deviations of our proposed models are reported ± with three different initialized models. 27

Table 3.2 Ablation study on the VisDial v1.0 validation split. CPT denotes continued pre-training. 29

Table 3.3 Results of GST in the low-data regime. We report NDCG scores based on the VisDial v1.0 validation split. We assume a small subset of the gold VisDial data (∼30%) is available. 30

Table 3.4 Adversarial robustness results against the attacks on the dialog history. We apply two different dialog history attacks: a coreference attack and a random token attack. The models are evaluated on the VisDialConv dataset [Agarwal et al., 2020] with the NDCG metric. The standard deviations are reported ± with five different random seeds. 35

Table 3.5 Results of human evaluation on 100 generated answers. We ask five human judges to decide which of two responses from the student and teacher models is more accurate. 37

Table 4.1 Results on the offline experiments. Underlined scores indicate the performance of the runner-up method. 54

Table 4.2 The effect of pragmatic inference (PI). 56

Table 4.3 Comparison with the multimodal foundation model. . . . 56

Table 4.4 Analysis of communicative efficiency. ↓ indicates lower is better. 60

Table 4.5 Results on the online experiments. 60

Chapter 1

Introduction

1.1 Intelligent Agents: A Conceptual Framework

The concept of an intelligent agent was introduced in the early stage of artificial intelligence (AI). Russell and Norvig [2016] have established the conceptual framework of intelligent agents, as shown in Figure 1.1. They defined the agent as an entity that can perceive its environment through sensors and take actions in that environment through actuators. The agent function internally maps any perceptual input sequence to an action to maximize the expected performance measure (*e.g.,* task success). While contemporary AI agents largely follow this framework, advances in AI have expanded the boundaries of this classic concept. What has changed in the era of deep learning [LeCun et al., 2015]?

1.2 Language Agents

One of the most representative changes is arguably the capabilities of understanding and generating natural language. As large-scale text data [Conneau

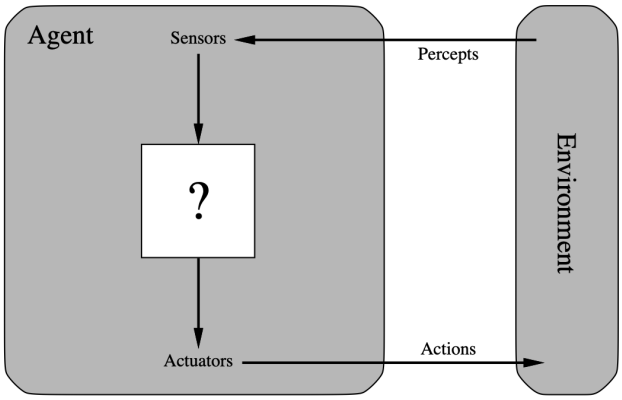


Figure 1.1: A conceptual framework of the agent from Russell and Norvig [2016].

et al., 2017] and advanced AI algorithms [Vaswani et al., 2017] become available, AI agents have significantly improved their ability to use language. Modern AI agents use language as *a vehicle for communication, reasoning, and learning*. They are referred to as language agents [Su, 2023, Ouyang et al., 2022].

Contemporary language agents like GPT-4 [OpenAI, 2023a], Gemini [Gemini et al., 2023], and HyperCLOVA X [Yoo et al., 2024] can communicate with humans on a wide range of topics through natural language. These capabilities significantly expand the breadth and depth of tasks that AI agents can perform, improving their adaptability to dynamic and complex environments. Additionally, using language for communication makes these agents more accessible to a broader range of users, including non-experts.

Language agents also use language to perform complex reasoning, such as arithmetic reasoning [Wei et al., 2022] and robot planning [Zeng et al., 2022]. The agents typically decompose complex reasoning problems into intermediate ones and explicitly represent their step-by-step reasoning results in language. All intermediate results specified in natural language are used to predict the final solutions. As a result, these capabilities lead to more accurate and justified

predictions compared with direct answering.

Another capability of language agents is learning from language supervision or feedback. Motivated by how humans learn from and teach each other, several studies [Radford et al., 2021, Cheng et al., 2023, Liu et al., 2023, Han et al., 2024] explore this capability by directly training AI agents with natural language. Compared with prevalent approaches that learn from numeric class labels or rewards [Ouyang et al., 2022], learning from language offers several benefits. First, language supervision contains rich signals about agent behaviors, helping agents avoid excessive trial and error [Cheng et al., 2023]. Second, it eliminates the need for designing a reward function, which requires careful consideration and is often time-consuming.

1.3 Embodied Language Agents

This dissertation focuses on *embodied language agents* — language agents that perform real-world robotic tasks using language. We describe the specific challenges for embodied language agents, which build upon the three capabilities of language agents: *communication, reasoning, and learning*. We further outline our approach to addressing each challenge.

The first challenge is visually-grounded communication. Since vision is a major modality of robot perception, the capability to continuously communicate with humans about visual inputs is an essential component for embodied language agents. We address this challenge on the visual dialog task [Das et al., 2017] where AI agents should answer a sequence of questions grounded in an image using dialog history as context. For example, the agent is expected to answer open-ended questions, such as "Is she wearing glasses?"

We introduce how embodied language agents can improve their robustness

and generalization capability for visually-grounded communication. One simple way to achieve this is to train the agents on diverse visual dialog data. To this end, we introduce a semi-supervised learning method for visually-grounded dialogue called generative self-training (GST). GST first trains models for answer generation (teacher) and question generation (questioner) using small amounts of human-labeled data. Next, the teacher and questioner alternatively generate the visual question and corresponding answer for large-scale unlabeled images on the Web via multimodal conditional text generation. Finally, GST trains another answer generation model, which we call the student, on the combination of human-labeled and machine-labeled visual dialog data.

The second challenge is reasoning about underspecified instructions. Humans often provide ambiguous or underspecified instructions. Thus, reasoning about instructions based on contextual information is an important challenge. Inspired by pragmatics [Goodman and Frank, 2016, Fried et al., 2022], where humans often convey their intended meanings by relying on context, we design a new task scenario to address this challenge, which we call pragmatic interactive object grasping (Pragmatic-IOG). In this task, a human first provides intention-oriented instruction like "I am thirsty." There is more than one object in the scene that meets the instructions. Embodied language agents should find all valid object candidates and ask questions for disambiguation. After interacting with human users, the agents should infer and grasp the target object.

We propose a modular approach for Pragmatic-IOG called pragmatic object grasping (PROGrasp). PROGrasp consists of four components: 1) visual grounding to identify object candidates, 2) question generation for instruction disambiguation, 3) answer interpretation to reason about the target object, and 4) object grasping to retrieve the target object. Furthermore, we present a reasoning method for target object discovery, which we call pragmatic inference.

This method measures how well each object candidate explains the observed data, including the visual scene and dialogue history. We use the modules for visual grounding and answer interpretation to implement pragmatic inference.

The third challenge is learning robotic skills from natural language. Learning robotic actions from natural language has significant potential, as it enables non-experts to intuitively train robots in their environments. To this end, we study how embodied language agents learn robotic skills (*e.g.,* open the cabinet) only from language guidance (*e.g.,* "move the arm to the left").

We first propose a language-based teleoperation method to collect robot demonstration data. Specifically, a human first provides natural language supervision at each time step, and large language models (LLMs) [OpenAI, 2023a, Gemini et al., 2023, Yoo et al., 2024] then translate language supervision into low-level robotic action based on the detailed text prompt. Robots finally move based on the action. After repeating this process, demonstration data is collected. To scale up the size of the collected demonstrations, we further propose a data augmentation method. It aims to collect alternative trajectory data for the original demonstration trajectories based on the heuristic algorithm.

We introduce a model that learns robotic policies directly from natural language supervision, which we call CLIP-RT (CLIP-based Robotics Transformer). Inspired by CLIP [Radford et al., 2021] that uses language as a training signal, we extend CLIP to robot learning. Specifically, we employ the pre-trained CLIP models and train them to optimize the pairwise similarity between natural language supervision (*e.g.,* "move the arm forward") and the robot's current state (*i.e.,* the observed visual scene and language instruction). In other words, CLIP-RT learns to predict actions specified in language, rather than directly predicting low-level robotic actions. At test time, CLIP-RT uses the pre-defined lookup table that maps the language action into a low-level robotic action.

1.4 Organization of the Dissertation

The remaining part of the dissertation is organized as follows.

Chapter 2 surveys related work at the intersection of natural language processing (NLP) and robotics, categorized into (1) communication, (2) reasoning and planning, and (3) learning and control.

Chapter 3 discusses visually-grounded communication. First, we describe the problem of visual dialog and the motivation of our proposed method. Then, we present the related work, including self-training and existing studies in visual dialog. Next, we introduce our proposed method, generative self-training (GST) for visually-grounded dialogue. Finally, we describe the experimental setup and results, including a comparison with state-of-the-art, adversarial robustness analysis, and qualitative analysis.

Chapter 4 discusses reasoning about underspecified instruction. We first illustrate the problem of interactive object grasping along with the motivation. We then describe our proposed task, Pragmatic-IOG. Next, we provide the related works consisting of language-guided object grasping and pragmatics. We delve into our proposed method, PROGrasp, with five components: visual grounding, question generation, answer interpretation, object grasping, and pragmatic inference. In experiments, we describe test data, evaluation protocols, and results. Finally, we discuss the quantitative and qualitative results.

Chapter 5 discusses learning robotic skills from natural language. We first introduce our motivation and an overview of our approach. Then, we provide related works regarding robot data collection and vision-language-action (VLA) models. Next, we elaborate on our proposed model, CLIP-RT, and the language-based teleoperation method. In experiments, we describe the experimental setup consisting of robotic tasks, data, compared methods, and a robotic platform.

We finally discuss the experimental results.

Chapter 6 concludes the dissertation by summarizing its content and highlighting the contributions. Finally, we present promising future directions: (1) lifelong learning and (2) long-horizon task execution.

Chapter 2

Background: Robots That Use Language

2.1 Introduction

Building robots that understand and use natural language has a long history, driven by the need for intuitive human-robot interaction. Early systems, such as Shakey the Robot [Nilsson et al., 1984], relied on rigid, predefined commands, offering basic language understanding. Advancements enabled service robots to process spoken instructions [Lopes and Teixeira, 2000, Dzifcak et al., 2009, Artzi and Zettlemoyer, 2013] like "Bring me a cup of coffee." However, these systems heavily relied on predefined grammar [Steedman, 1996] and lexical analysis to interpret instructions, making them unable to handle novel or ambiguous commands. Despite their limitations, these early efforts laid the groundwork for systems capable of handling more complex language and interacting in more dynamic environments.

Transformative advancements in both natural language processing (NLP)

and robotics have emerged, fueled by breakthroughs in machine learning. Deep learning [LeCun et al., 2015], particularly the rise of sequence-to-sequence models [Sutskever et al., 2014] and later transformer-based architectures [Vaswani et al., 2017], enabled robots to handle more nuanced linguistic inputs and respond more adaptively to context. Equipped with such capabilities, robots could not only follow commands but also engage in simple dialogues to clarify instructions [Thomason et al., 2019, Tellex et al., 2020]. Applications have expanded from structured environments to more dynamic, real-world scenarios, such as household assistants [Shridhar et al., 2020, Anderson et al., 2021].

More recently, the emergence of large language models (LLMs) [Brown et al., 2020, OpenAI, 2023a, Gemini et al., 2023, Yoo et al., 2024] has revolutionized the intersection of NLP and robotics. These models, trained on Internet-scale text corpora [wik, 2019, Conneau et al., 2017], allow robots to understand and generate language with advanced reasoning capabilities. These advancements drastically expand the scope of language-capable robots across three key areas: (1) communication, (2) reasoning and planning, and (3) learning and control. In the following section, we will discuss each area in detail.

2.2 Communication

Communicating with humans and other robots using natural language is one of the key capabilities for intelligent robots. This capability can be broadly categorized into single-turn and multi-turn interactions.

In single-turn communication, the robot processes a single instruction and performs the corresponding action without the need for follow-up clarification. For example, when given the command "Pick up the cup," the robot identifies and performs the action. Early systems like SHRDLU [Winograd, 1972]

were based on predefined commands and simple logic. However, modern systems, particularly those utilizing deep learning models [LeCun et al., 2015], have significantly enhanced the robot's ability to interpret and execute natural language commands in more dynamic environments. For example, in the field of robotic manipulation, extensive research has developed object-grasping systems that follow natural language instructions [Paul et al., 2017, Shridhar and Hsu, 2017, Venkatesh et al., 2021, Nguyen et al., 2020a, Kim et al., 2023]. Similarly, in robotic navigation, vision-and-language navigation (VLN) [Anderson et al., 2018b, Fried et al., 2018, Ku et al., 2020] has become a popular research area. These systems rely on language inputs to guide robots in complex environments. All of these studies assume that an initial, isolated instruction is enough to perform such tasks effectively.

Multi-turn communication, on the other hand, involves an ongoing dialogue where the robot seeks clarification or elaborates on tasks to refine its understanding. For instance, when asked to "Pick up the cup," a multi-turn system might ask, "Do you mean the blue cup on the table?" This iterative process allows the robot to resolve uncertainties and ensure task alignment, which is crucial for tasks that are more complex or require contextual understanding. Recent work in object grasping has explored interactive systems where robots ask for more information to disambiguate target objects [Shridhar and Hsu, 2018, Hatori et al., 2018, Zhang et al., 2021, Yang et al., 2022, Mo et al., 2022]. In navigation, cooperative vision-and-dialog navigation (CVDN) [Thomason et al., 2020] assumes a scenario where navigation agents receive an underspecified or ambiguous command in indoor environments, such as "Go to the room with the bed." After a few question-answer exchanges, the agents are expected to navigate the space based on the dialog history. These studies showcase the increasing importance of dialogue and context for robots to perform tasks ac-

curately and efficiently.

2.3 Reasoning and Planning

Reasoning and planning are crucial capabilities, enabling robots to tackle complex tasks through informed decision-making. Reasoning encompasses diverse cognitive capabilities, including commonsense reasoning [Chen et al., 2020b], spatio-temporal reasoning [Huang et al., 2024], and pragmatic reasoning [Kang et al., 2024a]. Recent advancements in reasoning have been largely influenced by the capabilities of large language models (LLMs) [Brown et al., 2020, OpenAI, 2023a, Gemini et al., 2023, Yoo et al., 2024] and advanced reasoning algorithms [Wei et al., 2022, Yao et al., 2024]. One significant change in LLM-based reasoning is using language as a vehicle for thought [Yu et al., 2023]. For example, Socratic Models [Zeng et al., 2022] compose multiple pre-trained models (*e.g.,* LLMs, Vision-Language Models, and Audio-Language Models) and make them exchange information with each other for multimodal reasoning. The pre-trained models directly use natural language as the intermediate representation by which the modules exchange information with each other.

Large language models (LLMs) have demonstrated their excellence in task planning. Typically, LLMs take high-level instructions (*e.g.,* "Move the blocks on the empty bowl") as inputs and translate them into a sequence of action primitives [Huang et al., 2022a] or Python codes that result in actions [Liang et al., 2023]. Other methods adjust plans based on observations [Ahn et al., 2022, Wu et al., 2023, Song et al., 2023] or execution failures [Huang et al., 2022b, Shin et al., 2024]. Such adaptive planning improves flexibility and robustness in task execution, enabling robots to better respond to changing conditions and unexpected scenarios. Furthermore, some studies have explored LLMs

as task and motion planners, making them generate low-level robotic trajectories [Kwon et al., 2023, Mandi et al., 2024].

2.4 Learning and Control

Significant efforts have been devoted to developing robotic systems that learn to execute natural language instructions. In the context of robotic manipulation, some studies have trained language-conditioned visuomotor policies through offline reinforcement learning algorithms [Kostrikov et al., 2021, Meng et al., 2023] or imitation learning [Stepputtis et al., 2020, Lynch and Sermanet, 2020, Shridhar et al., 2022, Mees et al., 2022]. The advent of large-scale robot data [Jang et al., 2022, Padalkar et al., 2023] has accelerated progress in imitation learning-based methods. More recently, a line of research directly trains pre-trained vision-language models (VLMs) to predict robotic actions, often referred to as vision-language-action (VLA) models. By leveraging rich semantic knowledge of VLMs, VLA models demonstrate more generalizable and effective language-conditioned policies [Brohan et al., 2022, 2023, Kim et al., 2024, Belkhale et al., 2024, Kang et al., 2024b] across diverse robotic tasks.

In the context of robot navigation, researchers have studied embodied instruction following agents in the context of vision-and-language navigation (VLN) [Anderson et al., 2018b, Ku et al., 2020, Thomason et al., 2020]. VLN agents should navigate 3D indoor environments by following natural language instructions. They were mostly developed in simulation environments [Chang et al., 2017], but some works have attempted to transfer them to real-world environments [Anderson et al., 2021, Krantz and Lee, 2022] or continuous 3D environments [Krantz et al., 2020, 2021].

2.5 Conclusion

We have categorized the existing literature on language-capable robots into three areas: (1) communication, (2) reasoning and planning, and (3) learning and control. This survey highlights the advancements and challenges within these three areas, providing a comprehensive overview of the current state of research on language-capable robots. In the following chapters, we will discuss how embodied language agents can bridge the gap between each area.

Chapter 3

Visually-Grounded Communication

3.1 Introduction

Extensive studies have focused on developing interactive agents that can "see" and "communicate" [Das et al., 2017, De Vries et al., 2017, Kim et al., 2019] due to their popularity in many real-world applications (*e.g.,* interacting with humanoid robots or assisting visually impaired person). Notably, Visual Dialog (VisDial) [Das et al., 2017] serves as a testbed for developing these capabilities, requiring a dialog agent to answer a sequence of visually-grounded questions. For example, the agent should answer open-ended questions like *"Is she wearing glasses?"* and *"What color is it?"*. The VisDial requires a deep understanding of visual perception and linguistic semantics, with a primary focus on effectively grounding the two.

Prior works in the VisDial have trained the dialog agents solely on the VisDial data via supervised learning [Lu et al., 2017, Seo et al., 2017, Kottur

et al., 2018, Niu et al., 2019, Schwartz et al., 2019, Guo et al., 2019, Gan et al., 2019, Kang et al., 2019, Nguyen et al., 2020b, Kang et al., 2021] or employed self-supervised pre-training [Murahari et al., 2020, Wang et al., 2020b, Chen et al., 2022] before training with the VisDial data. In other words, all existing studies rely on the VisDial data collected by humans and have no control over this supervision. We thus ask: *How can the agent improve its robustness and generalization capabilities beyond what it learns from the static, human-labeled visual dialog data?* Prior works have demonstrated that semi-supervised learning improves generalization in both image [Zoph et al., 2020] and text classification [Du et al., 2021]. Accordingly, we consider semi-supervised learning (SSL) as an approach to addressing our research question.

Let us assume that we obtain huge amounts of unlabeled images. SSL for the VisDial can be applied to generate synthetic dialogue data for the unlabeled images and train the agent with the synthetic data. However, there are two critical problems with this approach. First, the target output for the VisDial (*i.e.,* multi-turn visual QA data) is complex compared with the classification tasks [Zoph et al., 2020, Du et al., 2021]. Second, even if SSL results in synthetic dialog data via text generation, the synthetic data may be noisy, containing irrelevant questions or incorrect answers. A robust training method is required to leverage such noisy synthetic dialog datasets.

Inspired by self-training [Zoph et al., 2020, Du et al., 2021, Lee et al., 2013, Berthelot et al., 2019, Sohn et al., 2020, Xie et al., 2020a,b, He et al., 2020] that have mainly been studied in classification tasks [Xie et al., 2020b, Zoph et al., 2020, Sohn et al., 2020, Du et al., 2021], we extend the idea of self-training to the task of visually-grounded dialogue. To this end, we propose a new learning strategy, which we call *generative self-training* (GST) that aims to generate the synthetic visual dialog data and utilizes the data for training. GST first trains

the teacher model (answerer) and the visual question generation model (questioner) using human-labeled VisDial data. It then retrieves unlabeled images from a Web image dataset, Conceptual 12M [Changpinyo et al., 2021]. Next, the questioner and the teacher alternatively generate a sequence of visual QA pairs for unlabeled images. Finally, the student learns the synthetic and the original VisDial data. We also introduce two methods: perplexity-based data selection (PPL) and multimodal consistency regularization (MCR) to effectively learn the noisy dialogue data. As a result, GST effortlessly scales up the size of training data (1.2M QA pairs $\rightarrow$ 12.9M QA pairs).

Our key contributions are four-fold. First, we propose a semi-supervised learning method called generative self-training (GST) to advance the general understanding of visually-grounded dialogue. Second, experiments show that GST achieves new state-of-the-art performance on the VisDial v1.0 and v0.9 datasets at the time of publication. Third, we conduct an adversarial robustness analysis to identify the robustness of GST. We observe that GST significantly improves the robustness compared with the baseline models against all types of visual and textual adversarial attacks, boosting performance. Finally, we perform a qualitative analysis by visualizing the synthetic data and answer predictions from different models.

3.2 Related Work

3.2.1 Visual Question Answering

Visual Question Answering (VQA) [Antol et al., 2015, Goyal et al., 2017] is a pivotal task at the intersection of computer vision and natural language processing. This task requires a holistic understanding of an input image and natural language question grounded to the image. Extensive studies have focused on

learning effective joint representations of images and questions [Yang et al., 2016, Kim et al., 2016, 2017, Anderson et al., 2018a, Yu et al., 2018, Kim et al., 2018, 2020]. However, these methods inherently involve a *single* round of interaction, treating each question as an independent query. This limitation highlights the need for models capable of handling multi-turn interactions while maintaining context.

3.2.2 Visual Dialog

Visual Dialog (VisDial) [Das et al., 2017] requires a dialog agent to answer a sequence of image-grounded questions using the dialog history. Prior works have explored diverse attention mechanisms [Lu et al., 2017, Seo et al., 2017, Kottur et al., 2018, Niu et al., 2019, Schwartz et al., 2019, Guo et al., 2019, Gan et al., 2019, Kang et al., 2019, Nguyen et al., 2020b] considering the grounding of the image, dialog history, and question. A line of research [Zheng et al., 2019, Kang et al., 2021] explicitly builds the semantic structures of the dialog inspired by graph neural networks [Scarselli et al., 2008]. More recently, a line of research [Murahari et al., 2020, Wang et al., 2020b, Chen et al., 2022] has employed self-supervised pre-training to leverage the knowledge of related vision-and-language datasets [Sharma et al., 2018, Antol et al., 2015, Zhu et al., 2015]. All of them have relied on human-labeled VisDial data. However, our approach is based on semi-supervised learning and actively generates the synthetic visual dialog data for training.

3.2.3 Neural Dialog Generation

Neural dialogue generation is a vibrant research area in natural language processing [Zhang et al., 2020, Shang et al., 2015, Li et al., 2016, Serban et al., 2017, Saleh et al., 2020, Li et al., 2017a, Wang et al., 2020a, Huang et al., 2020].

Our approach is similar to neural dialogue generation in that the model should generate a corresponding response based on the dialog history and the current utterance. However, we aim to generate *visually-grounded* dialogue and thus focus on the image-groundedness of visual questions and the semantic correctness of corresponding answers. On the other hand, neural dialogue generation considers diverse aspects when evaluating the systems: specificity, response-relatedness [See et al., 2019], interestingness [Mehri and Eskenazi, 2020], and diversity [Li et al., 2016].

3.3 Approach

3.3.1 Preliminaries

Visual Dialog. The visual dialog (VisDial) data [Das et al., 2017] contains an image v and a corresponding dialog $d = \{\underbrace{c}_{d_0}, \underbrace{(q_1, a_1)}_{d_1}, \cdots, \underbrace{(q_T, a_T)}_{d_T}\}$ where c is an image caption. T is the number of rounds for each dialog. At each round t, a dialog agent takes a triplet $(v, d_{<t}, q_t)$ as an input, consisting of the image, the dialog history, and a visual question. $d_{<t}$ denotes all dialog rounds before the t-th round. The agent is then expected to predict a ground-truth answer a_t. There are two broad classes of methods in the VisDial: *generative* and *discriminative*. Generative models aim to generate the ground-truth answer by maximizing the log-likelihood of a_t. In contrast, discriminative models learn to retrieve the ground-truth answer from a set of answer candidates $a_t \in \{a_t^1, \cdots, a_t^{100}\}$. Our focus is generative models since they do not need pre-defined answer candidates and are thus more practical for real-world applications.

Self-Training. There exists a labeled dataset $L = \{(x_n, y_n)\}_{n=1}^N$ and an unlabeled dataset $U = \{\tilde{x}_m\}_{m=1}^M$. Typically, self-training trains a teacher model

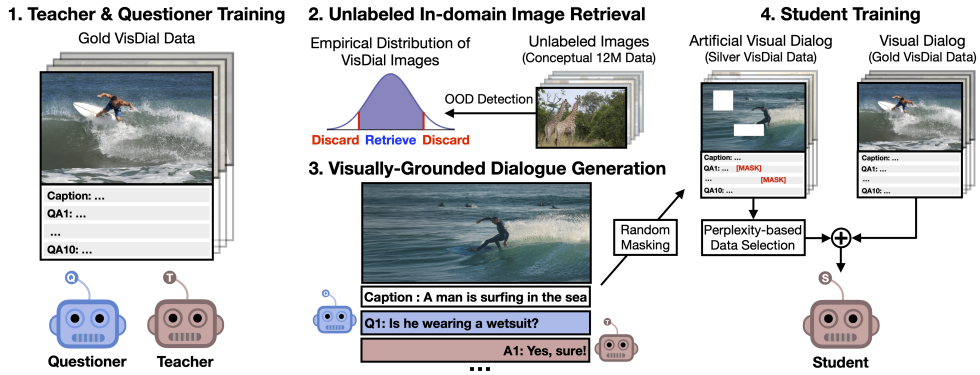


Figure 3.1: An overview of generative self-training (GST).

$P_{\mathcal{T}}$ on the labeled dataset L. The teacher then predicts the pseudo label $\tilde{y}$ for the unlabeled data $\tilde{x}$, constructing the synthetic dataset $\tilde{L} = \{(\tilde{x}_m, \tilde{y}_m)\}_{m=1}^{M}$. Finally, a student model $P_{\mathcal{S}}$ is trained on the combination of the original and synthetic datasets $L \cup \tilde{L}$. Many variants have been studied on this setup: (1) selecting the subset of the synthetic dataset [He et al., 2020, Xie et al., 2020b, Sohn et al., 2020], (2) adding noise to inputs [Zoph et al., 2020, He et al., 2020, Xie et al., 2020b,a, Sohn et al., 2020], and (3) iterating the above setup multiple times [He et al., 2020, Xie et al., 2020b].

3.3.2 Generative Self-Training (GST)

Figure 3.1 illustrates an overview of GST. There is a human-labeled VisDial data $L = \{(v_n, d_n)\}_{n=1}^{N}$ where v_n is a given image, and each dialog $d_n = \{\underbrace{c_n}_{d_{n,0}}, \underbrace{(q_{n,1}, a_{n,1})}_{d_{n,1}}, \cdots, \underbrace{(q_{n,T}, a_{n,T})}_{d_{n,T}}\}$ consists of an image caption c and T rounds of QA pairs. GST first trains a teacher $P_{\mathcal{T}}$ and a questioner $P_{\mathcal{Q}}$ with the labeled dataset L via supervised learning. It then retrieves unlabeled images $U = \{\tilde{v}_m\}_{m=1}^{M}$ from the Conceptual 12M dataset [Changpinyo et al., 2021] using a simple outlier detection model, the multivariate normal distribution. Next,

the questioner and the teacher generate the visually-grounded dialog $\tilde{d}$ for the unlabeled image $\tilde{v}$ via multimodal conditional text generation, finally yielding a synthetic dialog dataset $\tilde{L} = \{(\tilde{v}_m, \tilde{d}_m)\}_{m=1}^{M}$. We call this dataset the *silver VisDial* data to distinguish it from the human-labeled VisDial dataset [Das et al., 2017] (short for the *gold VisDial* data). Finally, a student $P_{\mathcal{S}}$ is trained on a combination of the gold and the silver VisDial data while applying perplexity-based data selection (PPL) and multimodal consistency regularization (MCR) to the silver VisDial data.

Teacher and Questioner Training. GST first trains the answer generator — the teacher model $P_{\mathcal{T}}$ on the gold VisDial data. Specifically, the teacher learns to generate the ground-truth answer $a_t = (w_{t,1}, \cdots, w_{t,S})$, given the context $c_t \triangleq (v, d_{<t}, q_t)$ comprising the image, the dialog history, and the question. We minimize the negative log-likelihood of the ground-truth answer:

$$\begin{aligned} \mathcal{L}_{\mathcal{T}} &= -\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T}\log P_{\mathcal{T}}(a_{n,t}|c_{n,t}) \\ &= -\frac{1}{NTS}\sum_{n=1}^{N}\sum_{t=1}^{T}\sum_{s=1}^{S}\log P_{\mathcal{T}}(w_s|c_{n,t}, w_{<s}) \end{aligned} \tag{3.1}$$

where N, T, and S indicate the number of data tuples in gold VisDial data, dialog rounds, and the sequence length of the ground-truth answer, respectively. $w_{<s}$ denotes all word tokens before the s-th token in the answer sequence. Similar to the teacher, the questioner is trained to generate the question at round t, given the image and the dialog history until round $t-1$ (*i.e.,* $P_{\mathcal{Q}}(q_t|v, d_{<t})$). The questioner also learns to minimize the negative log-likelihood of the question. Note that the teacher and the questioner are trained separately to prevent possible unintended co-adaptation [Kim et al., 2019]. Both the teacher and the questioner are based on encoder-decoder architecture, where an encoder

aggregates the context, and a decoder generates the target sequence. We implement the models by integrating a pre-trained vision-and-language encoder, ViLBERT [Lu et al., 2019], with the transformer decoder [Rothe et al., 2020].

Unlabeled In-Domain Image Retrieval (IIR). Inspired by the work [Du et al., 2021], GST retrieves in-domain image data from the unlabeled image dataset [Changpinyo et al., 2021] using an out-of-distribution (OOD) detection model. Specifically, we extract the D dimensional feature vector for each image in the gold VisDial dataset by using the Vision Transformer (ViT) [Dosovitskiy et al., 2021] in the CLIP model [Radford et al., 2021], yielding a feature matrix for the entire images $\mathbf{X} = (X_1, \cdots, X_N)^\top \in \mathbb{R}^{N \times D}$. Based on the matrix, we build the multivariate normal distribution whose dimension is D, *i.e.,* $\mathbf{X} \sim \mathcal{N}_D(\mu, \Sigma)$. We regard this normal distribution as the empirical distribution of the gold VisDial images and perform OOD detection by identifying the probability of each feature vector for the unlabeled image. Consequently, the top-M unlabeled images are retrieved out of 12 million images ($M \approx 3.6$ million).

Visually-Grounded Dialog Generation. Regarding the retrieved images $U = \{\tilde{v}_m\}_{m=1}^M$, the goal of this step is to generate the visually-grounded dialogs $\{\tilde{d}_m\}_{m=1}^M$ where each dialog $\tilde{d}$ consists of the image caption and T rounds of QA pairs. In an actual implementation, we use the image captions in the Conceptual 12M dataset [Changpinyo et al., 2021] and thus do not generate the captions. The QA pairs are sequentially generated. The questioner generates the question $\tilde{q}_t$, given the image $\tilde{v}$, the caption $\tilde{c}$, and the generated QA pairs until round $t-1$. Next, the teacher produces the corresponding answer $\tilde{a}_t$ based on the image $\tilde{v}$, the dialog history $\tilde{d}_{<t}$, and the generated question $\tilde{q}_t$. As a consequence, GST yields the silver VisDial dataset $\tilde{L} = \{(\tilde{v}_m, \tilde{d}_m)\}_{m=1}^M$.

Student Training. The student $P_{\mathcal{S}}$ is trained on the combination of the silver and the gold VisDial data. According to prior works in self-training [Xie et al., 2020b, He et al., 2020, Sohn et al., 2020, Zoph et al., 2020], selectively utilizing the samples in the synthetic dataset is a common approach since the confidence of the teacher model's predictions varies depending on the sample. To this end, we introduce a simple yet effective data selection method for sequence generation, perplexity-based data selection (PPL). PPL utilizes the answer data whose perplexity of the teacher is below a certain threshold. Perplexity is the exponentiated average negative log-likelihood of a sequence; the lower, the better. We hypothesize that PPL, albeit noisy, can be an indicator of whether the generated answer is correct or not, as in [Shakeri et al., 2020]. Furthermore, inspired by the consistency regularization [Xie et al., 2020a, Sohn et al., 2020], we also propose the multimodal consistency regularization (MCR) to improve the generalization capability of the student. MCR encourages the student to yield predictions similar to the teacher's predictions even when the student is provided with perturbed multimodal inputs. A loss function for the student is:

$$\begin{aligned}\mathcal{L}_{\mathcal{S}} = &-\frac{1}{MT}\sum_{m=1}^{M}\sum_{t=1}^{T}\mathbb{1}(\text{PPL}(\tilde{a}_{m,t}) < \tau)\log \underbrace{P_{\mathcal{S}}(\tilde{a}_{m,t}|\mathcal{M}(\tilde{c}_{m,t}))}_{\text{MCR}} \\ &-\frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T}\log P_{\mathcal{S}}(a_{n,t}|c_{n,t}) \\ &\text{where}\ \ \text{PPL}(\tilde{a}_t) = \exp\left\{-\frac{1}{S}\sum_{s=1}^{S}\log P_{\mathcal{T}}(\tilde{w}_s|\tilde{c}_t, \tilde{w}_{<s})\right\}\end{aligned} \tag{3.2}$$

where M, $\mathbb{1}$, and τ denote the number of data tuples in silver VisDial data, indicator function, and selection threshold, respectively. $\tilde{c}_{m,t} \triangleq (\tilde{v}_m, \tilde{d}_{m,<t}, \tilde{q}_{m,t})$ is the context for the silver VisDial data. The loss function consists of the losses for the silver and the gold VisDial data. PPL and MCR are applied to the loss of the silver VisDial data. PPL is implemented by the indicator function

above, selecting the synthetic answers whose perplexity of the teacher is below the threshold τ. It implies that unselected answers are ignored during training. The teacher's perplexity of each answer is computed in the dialog generation step. Next, $\mathcal{M}$ denotes the stochastic function for MCR that injects perturbations to the input space of the student. We implement the stochastic function by randomly masking 15% of image regions and word tokens [Lu et al., 2019]. Masked image regions have their image features zeroed out, and the masked word tokens are replaced with a special `[MASK]` token. MCR induces minimizing the distance between the *perturbed* (*i.e.,* masked) predictions from the student and the *unperturbed* predictions (*i.e.,* $\tilde{a}_{m,t}$) from the teacher. We believe MCR makes the student robust to the input noise, and PPL encourages the student to maintain a low entropy (*i.e.,* confident) in noisy data training. The student and the teacher share the same model architecture.

Iterative Training. We employ the concept of iterative training [Xie et al., 2020b, He et al., 2020], which repeats GST several times. The iterative training treats the student model at i-th iteration as a teacher model at $(i+1)$-th iteration and generates new synthetic data to train a new student. In other words, the iterative training repeats the third and fourth steps in Figure 3.1, where the silver data accumulates as the iteration proceeds. The student model at each iteration is trained with the accumulated silver and gold data [Xie et al., 2020b, He et al., 2020]. Unless stated otherwise, the student model is trained with three iterations.

3.4 Experiments

3.4.1 Visual Dialog Data

We evaluate our proposed approach on the VisDial v1.0 and v0.9 datasets [Das et al., 2017], collected by the AMT chatting between two workers about MS-COCO images [Lin et al., 2014]. Each dialog consists of a caption from COCO and a sequence of ten QA pairs. The VisDial v0.9 dataset has 83k dialogs on COCO-train and 40k dialogs on COCO-validation images. More recently, [Das et al., 2017] released additional 10k dialogs on Flickr images to use them as validation and test splits for the VisDial v1.0 dataset. As a result, the VisDial v1.0 dataset contains 123k, 2k, and 8k dialogs as train, validation, and test split. This dataset is based on a Creative Commons Attribution 4.0 International License.

3.4.2 Synthetic Data

The size of the silver VisDial data (*i.e.,* M) is 3.6M which is 30x larger than that of the gold VisDial data ($N = 0.12$M). Note that the silver VisDial data contains approximately 36M QA pairs since each dialog contains 10 QA pairs. 11.7M QA pairs out of 36M ($\sim$32%) are actually utilized after applying perplexity-based data selection. Consequently, the total amount of the training data is nearly 12.9M QA pairs, combining the silver data (11.7M QA pairs) with the original gold data (1.2M QA pairs).

3.4.3 Evaluation Protocol

We follow the standard evaluation protocol [Das et al., 2017] for model evaluation. The visual dialog models for both generative and discriminative tasks have been evaluated by the retrieval-based evaluation metrics: mean reciprocal

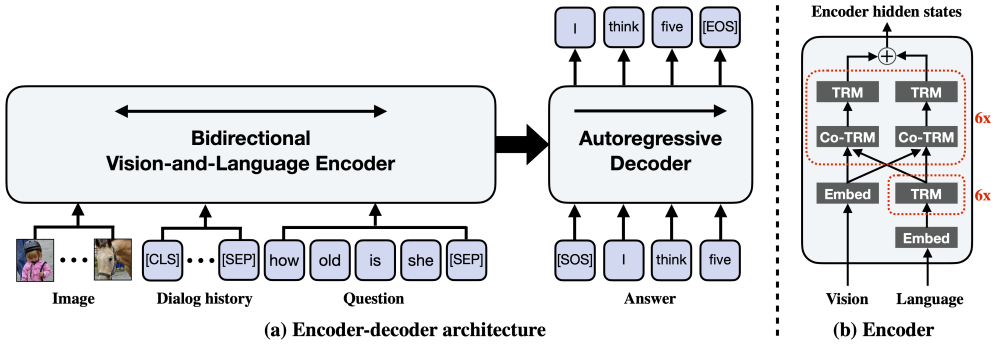


Figure 3.2: A detailed architecture of our proposed model. We visualize the following: (a) an encoder-decoder model, where the encoder aggregates the given multimodal context and the decoder generates the target sentence; and (b) a more detailed view of the encoder. TRM and Co-TRM denote the transformer module and the co-attentional transformer module, respectively. $\oplus$ is the concatenation operation.

rank (MRR), recall@k (R@k), mean rank (Mean), and normalized discounted cumulative gain (NDCG). Specifically, all dialogs in the VisDial contain a list of 100 answer candidates for each visual question, and there is one ground-truth answer in the answer candidates. The model sorts the answer candidates by the log-likelihood scores and then is evaluated by the four different metrics. MRR, R@k, and Mean consider the rank of the single ground-truth answer, while NDCG[1] considers all relevant answers from the 100-answers list by using the densely annotated relevance scores for all answer candidates. NDCG is regarded as the primary evaluation metric.

3.4.4 Implementation

As shown in Figure 3.2, we integrate the vision-and-language encoder [Lu et al., 2019] with the transformer decoder for sequence generation [Rothe et al., 2020] to train the teacher, the questioner, and the student. The decoder has 12 layers

[1]https://visualdialog.org/challenge/2019#evaluation

of transformer blocks, with each block having 12 attention heads and a hidden size of 768. The maximum sequence length of the encoder and the decoder is 256 and 25, respectively. We extract the feature vectors of the input images by using the Faster R-CNN [Ren et al., 2015, Anderson et al., 2018a] pre-trained on Visual Genome [Krishna et al., 2017]. The number of bounding boxes for each image is fixed to 36. We set the threshold for PPL τ to 50. We train on one A100 GPU with a batch size of 72 for 70 epochs. Training time takes about 3 days. We use the Adam optimizer [Kingma and Ba, 2014] with an initial learning rate 1e-5. The learning rate is warmed up to 2e-5 until 10k iterations and linearly decays to 1e-5. In visually-grounded dialog generation, the questioner and the teacher decode the sequences using the top-k sampling [Fan et al., 2018, Holtzman et al., 2018, Radford et al., 2019] with $k = 7$ and the temperature of 0.7. We use the top-k sampling since its computation is cheap yielding accurate and diverse sequences. Furthermore, we apply the 4-gram penalty [Paulus et al., 2018, Klein et al., 2017] when generating visual questions to ensure that no 4-gram appears twice in the questions for each dialog.

3.5 Visual Dialog Results

3.5.1 Comparison with State-of-the-Art

We compare GST with the state-of-the-art approaches on the validation set of the VisDial v1.0 and v0.9 datasets: UTC [Chen et al., 2022], MITVG [Chen et al., 2021], VD-BERT [Wang et al., 2020b], LTMI [Nguyen et al., 2020b], KBGN [Jiang et al., 2020a], DAM [Jiang et al., 2020b], ReDAN [Gan et al., 2019], DMRM [Chen et al., 2020a], Primary [Guo et al., 2019], RvA [Niu et al., 2019], CorefNMN [Kottur et al., 2018], CoAtt [Wu et al., 2018], HCIAE [Lu et al., 2017], and MN [Das et al., 2017]. We used the validation splits for evaluation since all previous studies benchmarked the models on those splits. As

Table 3.1: Comparison with the state-of-the-art generative models on both the VisDial v0.9 and v1.0 validation datasets. ↑ indicates higher is better. ↓ indicates lower is better. NDCG is not supported in v0.9 dataset. † denotes that the models are re-implemented by the previous work [Gan et al., 2019]. The standard deviations of our proposed models are reported ± with three different initialized models.

	VisDial v0.9 (val)					VisDial v1.0 (val)					
Model	MRR↑	R@1↑	R@5↑	R@10↑	Mean↓	NDCG↑	MRR↑	R@1↑	R@5↑	R@10↑	Mean↓
MN† [Das et al., 2017]	52.59	42.29	62.85	68.88	17.06	51.86	47.99	38.18	57.54	64.32	18.60
HCIAE† [Lu et al., 2017]	53.86	44.06	63.55	69.24	16.01	59.70	49.07	39.72	58.23	64.73	18.43
CoAtt† [Wu et al., 2018]	55.78	46.10	65.69	71.74	14.43	59.24	49.64	40.09	59.37	65.92	17.86
CorefNMN [Kottur et al., 2018]	53.50	43.66	63.54	69.93	15.69	-	-	-	-	-	-
RvA [Niu et al., 2019]	55.43	45.37	65.27	72.97	**10.71**	-	-	-	-	-	-
Primary [Guo et al., 2019]	-	-	-	-	-	-	49.01	38.54	59.82	66.94	16.60
DMRM [Chen et al., 2020a]	55.96	46.20	66.02	72.43	13.15	-	50.16	40.15	60.02	67.21	15.19
ReDAN [Gan et al., 2019]	-	-	-	-	-	60.47	50.02	40.27	59.93	66.78	17.40
DAM [Jiang et al., 2020b]	-	-	-	-	-	60.93	50.51	40.53	60.84	67.94	16.65
KBGN [Jiang et al., 2020a]	-	-	-	-	-	60.42	50.05	40.40	60.11	66.82	17.54
LTMI [Nguyen et al., 2020b]	-	-	-	-	-	63.58	50.74	40.44	61.61	69.71	14.93
VD-BERT [Wang et al., 2020b]	55.95	46.83	65.43	72.05	13.18	-	-	-	-	-	-
MITVG [Chen et al., 2021]	<u>56.83</u>	<u>47.14</u>	<u>67.19</u>	<u>73.72</u>	<u>11.95</u>	61.47	51.14	41.03	61.25	68.49	<u>14.37</u>
UTC [Chen et al., 2022]	-	-	-	-	-	<u>63.86</u>	<u>52.22</u>	<u>42.56</u>	<u>62.40</u>	<u>69.51</u>	15.67
Student (ours)	**60.03**±.18	**50.40**±.15	**70.74**±.09	**77.15**±.13	12.13±.18	**65.47**±.14	**53.19**±.11	**43.08**±.10	**64.09**±.05	**71.51**±.13	**14.34**±.15

shown in Table 3.1, GST significantly outperforms all compared methods on all evaluation metrics. Compared with the state-of-the-art model, the student model improves MRR 3.20% (56.83 → 60.03) and R@1 3.26% (47.14 → 50.40) on the VisDial v0.9 dataset. The improvement is consistently observed on the VisDial v1.0 dataset, boosting NDCG 1.61% (63.86 → 65.47) and MRR 0.97% (52.22 → 53.19). Moreover, it is noticeable that recent strong models (*i.e.,* UTC, MITVG, and VD-BERT) are also built based on the pre-trained weights of ViLBERT [Lu et al., 2019], transformer [Vaswani et al., 2017], and BERT [Devlin et al., 2019], respectively.

3.5.2 Ablation Study

We perform an ablation study to illustrate the effect of each component in GST. We report the performance of four ablative models: student w/o PPL, student w/o MCR, student w/o IIR, and teacher w/ CPT. Student w/o PPL denotes the model that utilizes all generated QA pairs without applying the perplexity-based data selection. Student w/o MCR does not inject noises into the inputs of the student model. Student w/o IIR utilizes the entire CC12M [Changpinyo et al., 2021] images to generate the silver VisDial data without applying in-domain image retrieval. Lastly, the teacher with continued pre-training (CPT) continues to perform pre-training with image-caption pairs in the silver VisDial data. CPT is proposed to identify the effect of utilizing additional vision-and-language data. Specifically, masked language modeling loss and masked image region loss are optimized by following ViLBERT [Lu et al., 2019].

In Table 3.2, we observe all components (*i.e.,* PPL, MCR, and IIR) play a significant role in boosting the performance. Notably, by comparing the student model with the student w/o IIR, we find that utilizing all the Web images does not contribute to an accurate answer prediction. Moreover, we observe that

Table 3.2: Ablation study on the VisDial v1.0 validation split. CPT denotes continued pre-training.

Model	PPL	MCR	IIR	Iteration	VisDial v1.0 (val)					
					NDCG↑	MRR↑	R@1↑	R@5↑	R@10↑	Mean↓
Teacher				0	64.50	52.06	42.04	62.92	71.06	14.54
Teacher (w/ CPT)			✓	0	63.59	51.70	41.99	61.88	68.62	16.21
Student (iter1, w/o PPL)		✓	✓	1	63.96	52.33	42.68	62.52	69.47	15.56
Student (iter1, w/o MCR)	✓		✓	1	63.71	52.49	42.56	62.87	70.00	15.21
Student (iter1, w/o IIR)	✓	✓		1	64.57	52.33	42.10	63.46	**71.54**	**14.31**
Student (iter1)	✓	✓	✓	1	65.06	52.84	42.74	<u>63.66</u>	71.30	14.60
Student (iter2)	✓	✓	✓	2	<u>65.46</u>	<u>53.04</u>	**43.15**	63.63	71.00	14.73
Student (iter3)	✓	✓	✓	3	**65.47**	**53.19**	<u>43.08</u>	**64.09**	<u>71.51</u>	<u>14.34</u>

Table 3.3: Results of GST in the low-data regime. We report NDCG scores based on the VisDial v1.0 validation split. We assume a small subset of the gold VisDial data (∼30%) is available.

	NDCG				
Model	1%	5%	10%	20%	30%
Teacher	27.64	50.04	54.46	57.14	60.67
Student	**38.73** (**+11.09**)	**56.60** (**+6.56**)	**58.62** (**+4.16**)	**60.92** (**+3.78**)	**63.09** (**+2.42**)

CPT results in a considerable drop in performance. We conjecture that it is due to low-precision image captions in the CC12M dataset, as mentioned in the paper [Changpinyo et al., 2021]. However, the student still shows competitive performance even if it also utilizes the captions in the dialog history.

3.5.3 Analysis on the Low-Data Regime

Is GST also helpful when gold data is extremely scarce? We investigate it to identify the effect of GST in the low-data regime. We assume that only a small subset of the gold VisDial data (1%, 5%, 10%, 20%, and 30%) is available. Therefore, the size of the gold data is $0.01N$, $0.05N$, $0.1N$, $0.2N$, and $0.3N$, respectively. We first train the teacher and the questioner on such scarce data, and then these two agents generate a new silver VisDial data for unlabeled images in the Conceptual 12M dataset [Changpinyo et al., 2021] with size $5N$. The student is then trained on the newly generated silver VisDial data and the small amount of the gold VisDial data. The student is based on a single iterative training, and PPL and MCR are still applied in this experiment. In Table 3.3, GST yields huge improvements on both metrics, especially NDCG, boosting up to 11.09 absolute points compared with the teacher. We observe that the smaller the amount of gold data, the larger the performance gap between the teacher and the student on NDCG. It implies that GST is helpful, especially

when gold data is scarce. We believe these results are particularly remarkable in other dialog-based tasks [Thomason et al., 2020, Alamri et al., 2019, Rashkin et al., 2019, Li et al., 2017b] since they are based on small-scaled datasets, and scaling up the size of the human-dialog datasets is laborious and expensive.

3.6 Adversarial Robustness Analysis

We introduce a comprehensive evaluation setup for adversarial robustness in the VisDial. Specifically, we propose three different adversarial attacks: (1) the FGSM attack, (2) a coreference attack, and (3) a random token attack. The FGSM attack perturbs input visual features, and the others attack the dialog history (*i.e.,* textual inputs).

Baselines. We compare our student model against three ablative baselines: (1) the teacher model, (2) the student model utilizing the entire CC12M images without applying the in-domain image retrieval (*i.e.,* student w/o IIR), and (3) the student model without multimodal consistency regularization (*i.e.,* student w/o MCR).

3.6.1 Adversarial Robustness Against Visual Attacks

The Fast Gradient Signed Method (FGSM) [Goodfellow et al., 2015] is a white-box attack that perturbs the visual inputs based on the gradients of the loss with respect to the visual inputs. Formally,

$$\text{FGSM}(x) = x + \epsilon \cdot \text{sign}(\nabla_x \mathcal{L}(x, y)) \tag{3.3}$$

where x and y denote the visual inputs and the corresponding ground-truth labels, respectively. ϵ is a hyperparameter that adjusts the intensity of perturbations. However, different from the above setup, each question in the VisDial can have one or more relevant answers in the list of answer candidates. We thus

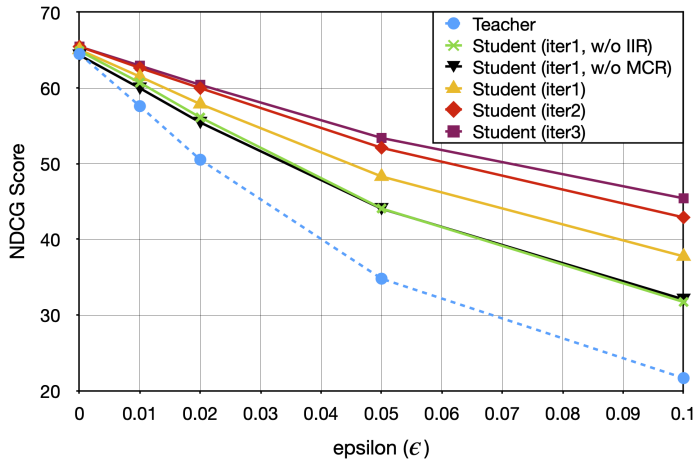


Figure 3.3: Adversarial robustness against FGSM attack on the VisDial v1.0 validation split. We report NDCG scores of each model.

define the FGSM attack for the VisDial as follows:

$$\text{FGSM}(v) = v + \epsilon \cdot \text{sign}(\sum_{c=1}^{C} r(a_{t,c}) \cdot \nabla_v \mathcal{L}(c_t, a_{t,c})) \tag{3.4}$$

where $C = 100$ and $r(\cdot)$ denote the number of answer candidates and a function that returns the human-annotated relevance scores for each answer candidate, respectively. The relevance scores range from 0 to 1. c_t and $a_{t,c}$ are the context (*i.e.*, $c_t \triangleq (v, d_{<t}, q_t)$) and the c-th answer candidate, respectively. Equation 3.4 indicates that the gradients of the loss for all relevant answers are considered for the FGSM attack.

As shown in Figure 3.3, we validate the models with four different epsilon values $\epsilon \in \{0.01, 0.02, 0.05, 0.1\}$. The student model shows very significant improvements in NDCG compared with the teacher model. Specifically, the performance gap between the student model with three iterations (*i.e.*, student-iter3) and the teacher model widens up to 23.83 absolute points (21.60 $\rightarrow$ 45.43) when ϵ is 0.1. It illustrates that GST makes the visual dialog model robust

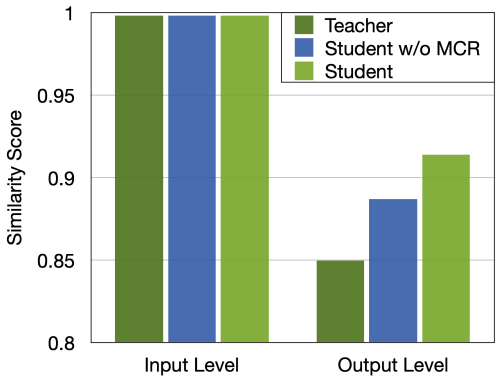


Figure 3.4: A visualization of cosine similarities between clean and perturbed image features in both input and output levels. We employ the FGSM attack with $\epsilon = 0.1$ to corrupt the clean images.

against the FGSM attack even though the student model is not optimized for adversarial robustness. Furthermore, we can identify the efficacy of the iterative training as the intensity of the perturbations increases. The NDCG scores are boosted from 37.82% (iter1) to 45.43% (iter3) at $\epsilon = 0.1$. Finally, in-domain image retrieval (IIR) and multimodal consistency regularization (MCR) boost adversarial robustness in the FGSM attack. It implies: (1) the additional use of the discarded images along with the synthetic dialog does not bring any gains and (2) learning perturbed multimodal inputs improves the robustness.

We hypothesize that MCR, combined with learning from diverse Web images, encourages the model to learn more robust, invariant image representations that are less sensitive to adversarial perturbations. To validate this, we conduct an analysis comparing the student model (with MCR) with two ablative models: (1) the student model without MCR, and (2) the teacher model, which has no access to Web images or MCR. For each model, we inject adversarial perturbations generated by the FGSM attack [Goodfellow et al., 2015] with $\epsilon = 0.1$ into the input images and compute two types of similarity: (1) the input-

level similarity between clean and perturbed images, and (2) the output-level similarity between corresponding clean and perturbed outputs. We compute the feature-level similarities by using cosine similarity, and the output-level features are derived from each model's encoder (see Figure 3.2). In Figure 3.4, all input-level cosine similarities are close to one, indicating that only small perturbations were injected into the input images. However, in terms of the output-level similarity, the student model exhibits a significantly higher similarity score compared to both baselines. This demonstrates that the combination of MCR and the use of unlabeled Web images reduces the student model's sensitivity to adversarial noise, which may explain its improved adversarial robustness.

3.6.2 Adversarial Robustness Against Textual Attacks

We also study the adversarial robustness against textual attacks to illustrate the effect of GST. We chose to perturb the dialog history because it contains useful information to answer the given question (*e.g.,* cues for pronoun). However, according to recent studies [Agarwal et al., 2020, Kang et al., 2021] in the VisDial, not all questions require the dialog history to respond with the correct answers. So the work [Agarwal et al., 2020] has proposed a challenging subset of the VisDial validation dataset called the VisDialConv. The VisDialConv dataset only contains questions that necessarily require the dialog history to answer (*e.g.,* can you tell what it is for?). The crowd-workers conducted a manual inspection to select such *context-dependent* questions.

Based on the VisDialConv dataset, we apply two different black-box attacks. First, we propose the coreference attack, which substitutes the noun phrases or pronouns in the dialog history with their synonyms to fool the VisDial models. Specifically, we leverage the off-the-shelf neural coreference resolution tool[2] and

[2]https://github.com/huggingface/neuralcoref based on Clark and Manning [2016].

Table 3.4: Adversarial robustness results against the attacks on the dialog history. We apply two different dialog history attacks: a coreference attack and a random token attack. The models are evaluated on the VisDialConv dataset [Agarwal et al., 2020] with the NDCG metric. The standard deviations are reported ± with five different random seeds.

Model	No Attack	Coreference Attack	Random Token Attack			
			10%	20%	30%	40%
Teacher	56.55	52.60	54.69±1.12	52.86±0.79	49.41±2.09	45.04±2.28
Student (iter1, full)	58.53	54.26	56.59±1.37	54.55±1.15	50.98±2.06	46.56±1.96
Student (iter1)	58.63	54.34	55.59±0.88	54.26±1.54	51.04±2.39	47.04±2.03
Student (iter2)	56.92	52.69	55.59±0.88	53.57±1.40	49.95±1.91	46.82±2.02
Student (iter3)	**59.30**	**55.44**	**57.25**±0.91	**55.10**±1.50	**52.11**±2.75	**48.00**±2.90

find words in the dialog history that refer to objects such as those mentioned in a given question. We also borrow the counter-fitting word embeddings [Mrkšić et al., 2016] similar to textfooler [Jin et al., 2020] to retrieve the synonyms. We greedily substitute the words with the synonyms with a minimum cosine distance in the embedding space since we observe that the other synonyms harm the original semantics of the dialog history. In Table 3.4, the student-iter3 model outperforms the teacher model on NDCG by a large margin (2.84%, 52.60 $\rightarrow$ 55.44) in the coreference attack. Furthermore, we do not see any merit in utilizing the entire CC12M [Changpinyo et al., 2021] images and the corresponding synthetic dialog data, comparing the student-iter1-full with the student-iter1.

The random token attack randomly replaces the word or sub-word tokens in the dialog history with a special `[MASK]` token. The pre-trained $\text{BERT}_{\texttt{BASE}}$ model [Devlin et al., 2019] then recovers the masked tokens with masked language modeling (MLM) similar to BERT-ATTACK [Li et al., 2020]. Finally, the perturbed dialog history is fed into the visual dialog models. We conduct this experiment by adjusting the probability of random masking up to 40%. As shown in Table 3.4, we evaluate each model with five random seeds and report the arithmetic mean and the standard deviations. The results demonstrate that GST is relatively robust against the random token attack compared with the baseline models.

3.7 Human Evaluation

Does GST expand the breath of the task that visual dialog models can perform? We randomly sample 20 in-domain images close to the cut-off point of in-domain image retrieval. Then, we generate 100 open-ended visual questions for human evaluation. The student and teacher models generate corresponding answers to the questions. By following evaluation protocols defined by Li et al.

Table 3.5: Results of human evaluation on 100 generated answers. We ask five human judges to decide which of two responses from the student and teacher models is more accurate.

Setting	Student Win	Student Lose	Tie
Multi-turn	0.39 (195/500)	0.13 (63/500)	0.48 (242/500)

[2016], we ask 5 human judges to decide which of the two answers for each question is more accurate. Ties are permitted, and the order of two responses were randomly shuffled. Ties include: (1) both responses are identical, (2) both responses are incorrect, and (3) both responses are correct, but it is hard to decide which one is more accurate. As shown in Table 3.5, the student model shows significant improvements in the winning rate compared with the teacher model. This indicates that the student model generates more accurate answers to visual questions across a broader range of images.

3.8 Qualitative Analysis

3.8.1 Comparison Between Silver and Gold Data

For qualitative analysis of the silver data, we visualize the generated conversations from our proposed models and the ones from humans. We excerpt the human conversation from the VisDial v1.0 validation dataset, and the questioner and the student generate the machine conversation using the image and the caption in the validation data. As shown in Figure 3.5, diverse visual questions are generated in the silver VisDial data. For example, in D10 of the last example, the questioner asks about "a car" not mentioned by the human questioner and not even presented in the image caption. The student responds correctly to the question. Likewise, from D3 to D6 in the first example, the questioner deals with "a cell phone," whereas the human questioner deals with different

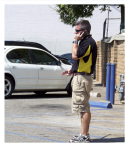
Image & Caption	Human Conversation (Gold VisDial)	Machine Conversation (Silver VisDial)
 A man is talking on a cell phone in a parking lot	D1 : Is it daytime? Yes D2 : Is this is a busy full parking lot? No D3 : Is the man dressed for work? No D4 : What is he wearing? Polo shirt and shorts D5 : Are there other people around? No D6 : Do you see any cars? Yes D7 : Are there a lot of cars? No D8 : How many cars do you think there are? 3 D9 : What colors are the cars? White and black D10 : Is the man wearing sunglasses? Yes	D1 : Is the man wearing a hat? No D2 : Is the man dressed for cold or warm weather? Warm weather D3 : Is the man talking on a cell phone? Yes he is D4 : Is it a flip phone? Yes it is D5 : What color is the phone? Black and silver D6 : Is the man holding it in his hand? Yes it is D7 : Are there any cars in the parking lot? Yes D8 : Is he waiting for someone to get on the car? No just standing D9 : Is the man looking at the camera? No he is looking off to the side D10 : Are there any buildings in the background? \ There is a brick building behind him
 A woman rides her bike on a sidewalk next to a fence	D1 : Is it daytime? Yes D2 : Is the sun out? Yes D3 : How old is the woman? 24 D4 : Is she alone? Yes D5 : Is she wearing sunglasses? No D6 : Is she carrying anything else with her? No D7 : Is she wearing sneakers? Yes D8 : What color is her shirt? Wearing a black jacket D9 : Is she smiling? No D10 : Where is she going? West on the road	D1 : Is this a color photo? Yes, it is a color photo D2 : Is it daytime? Yes D3 : What is the weather like? Looks like sunny D4 : What color is the woman's hair? Hair is brown D5 : How old does she seem? Around 35 D6 : Is she wearing a helmet? No helmet seen D7 : Is she wearing jeans? No D8 : Is she wearing glasses? No D9 : What color is her bike? **Bike is blue in color** D10 : Are there any cars? Yes, i see 1 car

Figure 3.5: A visualization of the gold and the silver data on the VisDial v1.0 validation split.

topics. However, the student sometimes fails to generate correct answers (*i.e.,* the red-colored text), showing the importance of more precise visual grounding.

3.8.2 A Visualization of Answer Predictions

We visualize the ground-truth answer (*i.e.,* the gold answer) and the answer predictions from the student and the teacher models given the same context. As shown in Figure 3.6, the student model indeed produces correct answers compared with the teacher model. Moreover, both models produce many correct or plausible answers, although the predicted answers differ from the gold answers (see the blue-colored text). For instance, for the last question in the third example (*i.e., Is she wearing a bathing suit?*), the student answers "wetsuit" to the question, although the ground-truth answer is "no". We conjecture that the ability to generate such flexible answers is evaluated as a high NDCG performance; NDCG considers all relevant responses in the answer candidates.

3.9 Discussions

We develop the teacher, the questioner, and the student models on top of ViLBERT [Lu et al., 2019], which leverages vision-and-language pre-training. Thus, the teacher can be understood as a typical model that follows the pretrain-then-transfer learning strategy mentioned in the introduction. On the other hand, the student leverages both pre-training and generative self-training. By comparing the student with the teacher, we identify that self-supervised pre-training and GST provide complementary modeling capabilities.

One of the major limitations of our approach is the learning efficiency of the student model. We demonstrate the effectiveness of our proposed method, but there can be more efficient ways to improve the visual dialog model. For example, our method generates the dialog data without considering the difficulty of the question. We believe that competency-aware or curriculum-based visual dialog generation can make our proposed self-training algorithm more efficient and powerful. We will leave it as a future work.

3.10 Conclusion

We propose a semi-supervised learning approach for the VisDial, called GST, that generates a synthetic visual dialog dataset for unlabeled Web images via multimodal conditional text generation. GST achieves the new state-of-the-art performance on the VisDial v1.0 and v0.9 datasets. Moreover, we demonstrate the efficacy of GST in the low-data regime and adversarial robustness analysis. Finally, GST produces diverse dialogs compared with the human dialog. We believe the idea of GST is generally applicable to other multimodal generative domains and expect GST to open the door to leveraging unlabeled images for visually-grounded interaction.

Image & Caption	Question	GT Answer	Student Model	Teacher Model
 The 2 children's teams are playing soccer against each other	Q1 : How many children are there?	3	3	4
	Q2 : What is the time of the day?	I can't tell that	Daytime	Daytime
	Q3 : Can you see a soccer ball?	Yes	Yes	Yes
	Q4 : What is the children wearing?	Soccer shirts and pants	Uniforms	Uniforms
	Q5 : What is the color of the ball?	White and red	White and red	White with red
	Q6 : Can you see a spectator?	No	No	No
	Q7 : Are they happy enjoying?	Yes	Yes	Yes
	Q8 : What is the color of the hair or they are wearing a helmet?	No	Brown	They all have brown hair
	Q9 : Can you see the sky?	No	No	No
	Q10 : Is there any female?	No	No	No
 2 men and a woman are on slopes wearing their skies	Q1 : Is there any snow on the ground?	Yes	Yes there is	Yes tons
	Q2 : Are they dressed for skiing?	Yes	Yes	Yes
	Q3 : Color shirt is the woman wearing?	Blue and white	Blue and white jacket	Black
	Q4 : Is there any trees around?	There are	Yes	Yes
	Q5 : What kind of day is it?	A snowy day	Sunny	Sunny
	Q6 : Are the man taller than the woman?	Yes, they are	Yes	Yes
	Q7 : How many hats are in the picture?	3	3 hats	2
	Q8 : Are they all wearing gloves?	Yes, they are all	Yes	Only two of them are
	Q9 : Do they seem to be enjoy the day?	Yes	Yes, they do	Yes, they do
	Q10 : What time of day is it?	Morning	Daytime	Noon
 A man is teaching a woman how to ride a surfboard	Q1 : Is the picture in color?	Yes	Yes	Yes
	Q2 : Are they the only ones in the water?	No	No	No
	Q3 : Can you see the beach?	Yes	Yes	Yes
	Q4 : Are there people on the beach?	Yes	Yes	Yes
	Q5 : How old is the man?	20's	20's	20's
	Q6 : What color is his hair?	Brown	Brown	Black
	Q7 : How many hats are in the picture?	Early 20's	20s	Same
	Q8 : What color is her hair?	Brown	Dark brown	Brown
	Q9 : What color is the surfboard?	White	White	White
	Q10 : Is she wearing a bathing suit?	No	Wetsuit	Yes
 A young man performs a trick on a skateboard	Q1 : Is the man wearing a helmet?	No	No	No
	Q2 : Is this outdoors?	Yes	Yes	Yes
	Q3 : Is it a sunny day?	Yes	Yes	Yes
	Q4 : Is the man wearing a hat?	No	No	No
	Q5 : Can you see graffiti?	No	No	No
	Q6 : Is he wearing elbow pads?	No	No	No
	Q7 : Is he wearing shorts?	No	No	No, jeans
	Q8 : Can you see anyone else?	Yes	Yes	Yes, but they are blurry
	Q9 : What color is his board?	Black	Black	Black
	Q10 : What color is his shirt?	White with brown spots	White	White
 A woman rides her bike on a sidewalk next to a fence	Q1 : Is it daytime?	Yes	Yes	Yes
	Q2 : Is the sun out?	Yes	Yes	Yes
	Q3 : How old is the woman?	24	20s	Late twenties
	Q4 : Is she alone?	Yes	Yes	Yes
	Q5 : Is she wearing sunglasses?	No	No	No
	Q6 : Is she carrying anything else with her?	No	A bag	A purse
	Q7 : Is she wearing sneakers?	Yes	Yes	No
	Q8 : What color is her shirt?	Wearing a black jacket	Black	Black
	Q9 : Is she smiling?	No	Yes	No
	Q10 : Where is she going?	West on the road	Not sure	She's at a park
 A man walks down the street, pass a yellow fire hydrant	Q1 : Is the man young?	I'd say early 20s	Yes	Yes
	Q2 : Is he dressed casual?	Yes, he is	Yes, t-shirt and jeans	Yes, jeans, a polo shirts
	Q3 : Is his hair short or longish?	It is shoulder length	It is medium length	It's pretty short
	Q4 : What color is his shirt?	It is medium brown	Navy blue	Blue with white writing
	Q5 : Is he wearing jeans?	Yes, he is	Yes, he is	Yes, he is
	Q6 : Is this in the city?	Looks like a suburb area	Looks like it	Yes, it looks like it
	Q7 : Are there any trees around?	Yes, 1 tree	A few in the background	Yes, I can see one
	Q8 : How about cars?	No cars	I don't see any cars	No cars are in the photo
	Q9 : Are there any other people?	No, he is alone	No other people	No
	Q10 : Is it sunny?	Yes, it is	I can't see the sky, but it is daytime	I can't really tell

Figure 3.6: A visualization of answer predictions from the student and the teacher models. The red-colored text is an incorrect answer. The blue-colored text is not a ground-truth answer, but it seems correct or plausible.

Chapter 4

Reasoning about Underspecified Instructions

4.1 Introduction

Advances in robotics and artificial intelligence are increasingly bringing robots into our daily lives, such as household assistants [Black et al., 2024]. To get closer to a broader range of users like non-experts, robots interact with humans and make decisions based on the interaction. Natural language is a preferred interface for human-robot interaction because of its intuitive and accessible nature [Kollar et al., 2010]. However, it is inherently ambiguous and largely depends on context [Piantadosi et al., 2012, Fried et al., 2022], which potentially leads to misunderstandings, especially for robots that lack contextual awareness. Therefore, reasoning about ambiguous and underspecified language instructions by leveraging contextual information is an important challenge for robots.

A line of research in human-robot interaction [Shridhar and Hsu, 2018, Hatori et al., 2018, Zhang et al., 2021, Yang et al., 2022, Mo et al., 2022] has

addressed the challenge in the context of interactive object grasping (IOG). A typical scenario of IOG starts mentioning the target object, such as "Give me the plastic bottle", but there is more than one object in the scene that meets the instruction. The robot should disambiguate the target object by asking questions to the interlocutor and then perform object grasping.

While the progress of IOG is exciting, the current scenario limits the ability of robots to understand beyond the literal meaning of natural language instructions. Specifically, instructions in the existing scenario clearly specify the category of the target object (*e.g.,* bottle). In other words, current IOG systems may work properly when the target object's category is given. However, we humans often convey our *intended meanings* by relying on context for streamlined communication and reduced cognitive load [Searle and Searle, 1969, Frank and Goodman, 2012]. For example, when we need some water and want our interlocutor to bring it, we briefly say "I am thirsty." The interlocutor then enriches the literal meaning of such underspecified instruction based on various types of shared context (*e.g.,* visual information and dialogue context) and reason about context-appropriate behavior. This ability to interpret and use language in context to achieve goals is known as pragmatics [Fried et al., 2022, Goodman and Frank, 2016, Smith et al., 2013].

We argue that the next-generation robotic system should have pragmatic reasoning ability — capture the user's intention with contextual information and achieve the desired goal. To this end, we introduce a new task, Pragmatic-IOG, to study the pragmatic reasoning behavior of embodied language agents. As shown in Figure 4.1, we consider a scenario where a human user begins a conversation with an *intention-oriented* instruction like "My device runs out of battery." The robot should then find all valid object candidates (*e.g.,* the red-colored object regions in Figure 4.1) via visual grounding [Yu et al., 2016] and

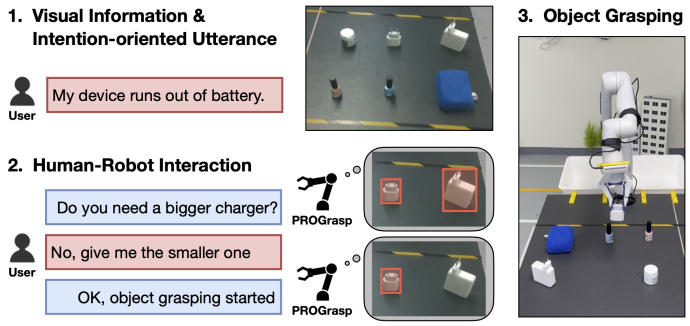


Figure 4.1: Overview of interactive object grasping with intention-oriented instruction. The initial instruction does not contain the target object's category.

ask a question for disambiguation. After receiving the user's response, the robot should pinpoint the target object and grasp the desired object. To study this problem, we collect a new dataset called Intention-oriented Multi-modal Dialogue (IM-Dial). The IM-Dial dataset contains 800 images and 500 human-to-human dialogue data regarding 86 categories of everyday objects. The dialogue consists of intention-oriented instruction and a series of question-and-answer pairs for target object discovery.

We propose a new robotic system that can reason about intention-oriented natural language instructions of the user and grasp desired objects through human-robot interaction called **PR**agmatic **O**bject **Grasp**ing (PROGrasp). PROGrasp consists of four modules: (1) a visual grounding module (VG) that predicts the region coordinates of valid objects, (2) a question generation module (Q-gen) that learns to generate questions to identify the user's intention, (3) an answer interpretation module (A-int) that interprets the human user's response given the multi-modal context, and (4) an object-grasping module (OG) to pick up the inferred object.

PROGrasp trains VG, Q-gen, and A-int on the IM-Dial data. After train-

ing, PROGrasp performs our proposed task by interacting with the human user. Specifically, VG first predicts a set of object region candidates given the intention-oriented instruction. Q-gen then generates a question, and the user responds to the question. Next, PROGrasp determines the target region among the region candidates based on how well each candidate region explains the visual and dialogue context, which we call *pragmatic inference*. We implement pragmatic inference as a multi-agent reasoning of VG and A-int where VG evaluates the likelihood of each region candidate given the visual and dialogue context, and A-int rescores alternative region candidates by interpreting the user's response. Finally, OG computes the 3D coordinates of the inferred region and performs object grasping.

We conduct offline and online experiments on the IM-Dial dataset. In offline experiments, we study how well PROGrasp identifies the target object. PROGrasp significantly improves the accuracy of offline experiments by 35% compared with the baselines. Moreover, PROGrasp outperforms the powerful multimodal foundation model [OpenAI, 2023b] on validation data. In online experiments, we use a physical robot arm to evaluate the success rate of object grasping. PROGrasp boosts the success rate by 17%. Furthermore, our system efficiently identifies the target object through fewer interactions than baselines. Finally, we perform qualitative analysis, visualizing diverse samples.

Our contributions are three-fold. First, we propose an interactive object-grasping system (*i.e.,* PROGrasp) that capably understands the human user's intention and grasps the desired object through dialogue. Second, we introduce a new task Pragmatic-IOG, along with a novel dataset called Intention-oriented Multi-modal Dialogue (IM-Dial). Third, through extensive experiments, our robotic system validates its (1) efficacy in both offline and online experiments and (2) efficiency when identifying the target object via pragmatic inference.

4.2 Related Work

4.2.1 Language-Guided Object Grasping

There has been extensive research on developing object-grasping systems that can understand natural language. Some studies [Paul et al., 2017, Shridhar and Hsu, 2017, Venkatesh et al., 2021, Nguyen et al., 2020a, Kim et al., 2023] make robots manipulate objects only with initial language instruction, assuming that the instruction is enough to identify the desired object. However, natural language is inherently ambiguous [Piantadosi et al., 2012]. Therefore, a line of research [Shridhar and Hsu, 2018, Hatori et al., 2018, Zhang et al., 2021, Yang et al., 2022, Mo et al., 2022], which we call interactive object grasping (IOG), considers the scenario where robots need more information to disambiguate the target object. They typically generate questions and perform object grasping based on the response from a human user. Our approach belongs to IOG, but it differs from previous studies in two aspects. First, prior work in IOG considers a scenario where the category of the target object is clearly specified. However, we design a task scenario that requires robots to understand the semantic meaning and, by extension, *the intended meaning* of the user's instruction. Accordingly, intention-oriented instructions focus on the user's intention without specifying the category of the target object. Second, previous studies define the format of either questions or responses. Specifically, the questions are fixed (*e.g.,* "Which one?") [Hatori et al., 2018, Whitney et al., 2017] or based on templates [Shridhar and Hsu, 2018, Zhang et al., 2021, Yang et al., 2022, Mo et al., 2022]. The answer formats are also binary [Hatori et al., 2018], a single word [Whitney et al., 2017], or based on a pre-defined pool [Shridhar and Hsu, 2018, Zhang et al., 2021, Yang et al., 2022, Mo et al., 2022]. However, PROGrasp does not impose any constraints on the format of the questions and responses. Our Q-gen

generates unconstrained questions without relying on any templates, and A-int understands various types of responses, enabling non-expert users to interact with the robot more naturally.

4.2.2 Pragmatics

The study of how linguistic meaning is affected by context [Grice, 1975, Searle and Searle, 1969, Frank and Goodman, 2012], known as pragmatics, has a long history of research. According to the work [Fried et al., 2022], there are four kinds of well-studied tasks in the field of pragmatics: reference games [Frank and Goodman, 2012, Monroe et al., 2017], image captioning [Andreas and Klein, 2016, Cohn-Gordon et al., 2018], instruction following [Chen and Mooney, 2011, Anderson et al., 2018b], and grounded dialogue [De Vries et al., 2017, Kim et al., 2019, Chai et al., 2014, Kang et al., 2023]. Our work belongs to the last category, but it is the first work that integrates grounded goal-oriented dialogue into a real-world robot arm for object grasping. Regarding computational modeling, PROGrasp shares the spirit with the Rational Speech Acts (RSA) [Frank and Goodman, 2012, Goodman and Frank, 2016]. We propose a multi-agent reasoning method (*i.e.,* pragmatic inference) to identify target objects accurately and efficiently by interpreting the human user's response.

4.3 Approach

4.3.1 Background

As the Web-scale data sources [Changpinyo et al., 2021, Schuhmann et al., 2021] are publicly available, finetuning the model pre-trained on such datasets to the specific task has become a de facto standard strategy in AI. Accordingly, there has been a lot of multi-modal pre-training methods [Wang et al., 2022, Lu et al., 2019, Tan and Bansal, 2019] trained on the large-scale image-

text pairs. We employ a simple yet powerful multi-modal sequence-to-sequence model, OFA [Wang et al., 2022], since it can cover various multi-modal tasks with a unified architecture. OFA is pre-trained on a wide range of multi-modal and uni-modal datasets with sequence-to-sequence learning [Sutskever et al., 2014, Vaswani et al., 2017]. The learning objective of OFA is to optimize:

$$\underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{|y|} \log P_\theta(y_i | y_{<i}, v, x) \tag{4.1}$$

where x and y denote the input and target sequences, respectively. $y_{<i}$ denotes all tokens before the i-th token in the target sequence, and v is visual information. P_θ represents the probability derived from the model with the parameters θ. The encoder encodes x and v and conveys the hidden states to the decoder. The decoder predicts the next token y_i given a set of preceding tokens $y_{<i}$ and the hidden states of the encoder. We train our proposed modules in PROGrasp (*i.e.,* VG, Q-gen, and A-int) by finetuning OFA on the IM-Dial dataset. More details are discussed in the following Section.

4.3.2 Problem Statement

The goal of Pragmatic-IOG is to discover the target object's region coordinates $r^* = \langle x_1, y_1, x_2, y_2 \rangle$ in 2D images through human-robot interaction and pick up the object. We assume that a human user initially provides intention-oriented natural language instruction ℓ, and there is one target object in the given 2D image $\mathcal{I}$. An agent should ask a natural language question q to the user to identify the target object, and the user should provide an answer a. The dialogue history $\mathcal{D}$ is initialized to ℓ, and question and answer pairs at each round are appended to the dialogue history. The agent should predict the region coordinates $\hat{r}$ after T rounds of dialogue. Finally, it performs object grasping, computing the 3D coordinates of the target object based on $\hat{r}$ and the point cloud.

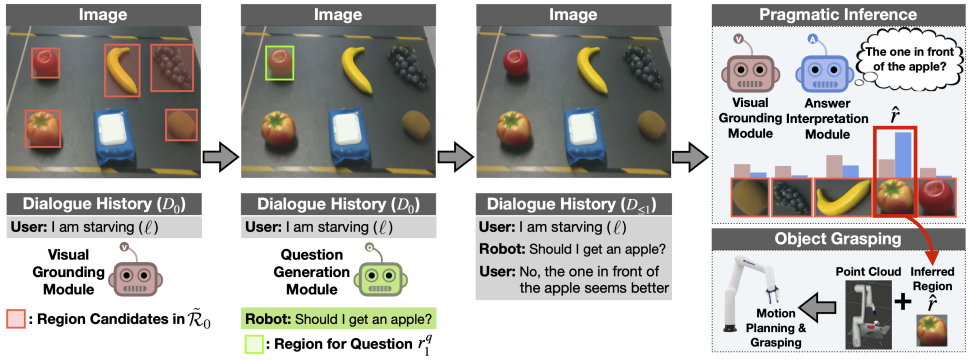


Figure 4.2: Illustration of the inference step in PROGrasp for $T = 1$. VG first performs object grounding using the dialogue history. Q-gen then selects the object candidate to ask and generates a question. After obtaining the response from the user, VG cooperates with A-int to determine the target object region. The object-grasping module finally grasps the object by computing the 3D coordinates of the target object.

4.3.3 Pragmatic Object Grasping (PROGrasp)

Visual Grounding Module. The visual grounding module (VG) aims to localize objects based on the multi-modal context. Specifically, the Intention-oriented Multi-modal Dialogue (IM-Dial) dataset contains an image $\mathcal{I}$, a visually-grounded dialogue $\mathcal{D} = \{\underbrace{\ell}_{\mathcal{D}_0}, \underbrace{(q_1, a_1)}_{\mathcal{D}_1}, \cdots, \underbrace{(q_N, a_N)}_{\mathcal{D}_N}\}$, and region labels $\mathcal{R} = \{\mathcal{R}_0, \cdots, \mathcal{R}_N\}$ where each element $\mathcal{R}_n = \{r_i^{vg}\}_{i=1}^{|\mathcal{R}_n|}$ is a set of object regions that VG should predict given the dialogue history $\mathcal{D}_{\leq n}$. $\mathcal{D}_{\leq n}$ denotes all dialogue data up to and including the n-th round. Consequently, VG is trained to maximize the log-likelihood of the ground-truth regions:

$$\underset{\theta}{\operatorname{argmax}} \sum_{n=0}^{N} \log P_{\mathcal{V}_\theta}(\mathcal{R}_n | \mathcal{I}, \mathcal{D}_{\leq n}). \tag{4.2}$$

PROGrasp regards the concatenation of the region coordinates in $\mathcal{R}_n$ as the target sequence in sequence-to-sequence learning and trains VG on top of the pre-trained OFA model [Wang et al., 2022].

Question Generation and Answer Interpretation Modules. PROGrasp trains the question generation (Q-gen) and answer interpretation (A-int) modules to produce the instructions from the human questioner and the human answerer, respectively. First, Q-gen learns to generate human-annotated questions in the IM-Dial dataset, given an input image, the dialogue history, and an object region for questioning as follows:

$$\underset{\theta}{\operatorname{argmax}} \sum_{n=1}^{N} \log P_{\mathcal{Q}_\theta}(q_n | \mathcal{I}, \mathcal{D}_{\leq n-1}, r_n^q) \tag{4.3}$$

where the object region r_n^q is annotated by the human questioner when collecting the IM-Dial dataset. Regarding sequence-to-sequence learning, the target sequence is the question q_n, and the triplet $(\mathcal{I}, \mathcal{D}_{\leq n-1}, r_n^q)$ is fed into the encoder. Next, the answer interpretation module (A-int), which is a proxy for the human user, learns to generate the response of the human answerer. It is optimized by maximizing the log-likelihood of the ground-truth answer:

$$\underset{\theta}{\operatorname{argmax}} \sum_{n=1}^{N} \log P_{\mathcal{A}_\theta}(a_n | \mathcal{I}, r^*, q_n). \tag{4.4}$$

Note that A-int takes the ground-truth region coordinates r^* during training and implicitly learns the semantic alignment between r^* and the question-and-answer pair (q_n, a_n). Moreover, we assume that the answer distribution is independent of the dialogue history (*i.e.,* $P(a_n|\mathcal{I}, \mathcal{D}_{\leq n-1}, r^*, q_n) = P(a_n|\mathcal{I}, r^*, q_n)$) by following Lee et al. [2018]. We also implement both Q-gen and A-int by finetuning the pre-trained OFA model [Wang et al., 2022].

Inference. The inference step of PROGrasp is described in Algorithm 1 and Figure 4.2. PROGrasp obtains an image $\mathcal{I}$ from a camera. A human user provides intention-oriented instruction ℓ. PROGrasp proceeds T rounds of dialogue in the inference step, and T is a hyperparameter. VG (*i.e,* $P_{\mathcal{V}_\theta}$) first predicts a

Algorithm 1 Pragmatic Object Grasping

Require: Modules for VG ($P_{\mathcal{V}_\theta}$), Q-gen ($P_{\mathcal{Q}_\theta}$), A-int ($P_{\mathcal{A}_\theta}$)
Require: Module for object grasping ($\mathcal{O}$)
Require: 2D RGB image $\mathcal{I}$ and dialogue history $\mathcal{D} \leftarrow \{\ell\}$
Require: A human user to interact with the robotic system
Require: The empty set of object regions $\mathcal{R} = \emptyset$
1: **for** $t \leftarrow 1$ to T **do**
2: $\quad \tilde{\mathcal{R}}_{t-1} \leftarrow P_{\mathcal{V}_\theta}(\cdot|\mathcal{I}, \mathcal{D}_{\leq t-1})$ where $\tilde{\mathcal{R}}_{t-1} = \left\{r_1, \cdots, r_{|\tilde{\mathcal{R}}|}\right\}$
3: $\quad \mathcal{R} \leftarrow \mathcal{R} \cup \tilde{\mathcal{R}}_{t-1}$
4: $\quad \tilde{q}_t \leftarrow P_{\mathcal{Q}_\theta}(\cdot|\mathcal{I}, \mathcal{D}_{\leq t-1}, r_t^q)$ where $r_t^q \sim \mathcal{R}$
5: $\quad$ The user provides an answer $\tilde{a}_t$ to the question $\tilde{q}_t$
6: $\quad \mathcal{D} \leftarrow \mathcal{D} \cup \{\tilde{q}_t, \tilde{a}_t\}$
7: $\quad \hat{r} \leftarrow \underset{r \in \mathcal{R}}{\text{argmax}} \log(P_{\mathcal{A}_\theta}(\tilde{a}_t|\mathcal{I}, r, \tilde{q}_t)^\lambda \cdot P_{\mathcal{V}_\theta}(r|\mathcal{I}, \mathcal{D}_{\leq t})^{1-\lambda})$
8: **end for**
9: Grasp the object $\mathcal{O}(\hat{r}, \mathcal{P})$ with $\hat{r}$ and the point cloud $\mathcal{P}$

set of object regions $\tilde{\mathcal{R}}$ and saves it to the superset $\mathcal{R}$. PROGrasp accumulates the object region candidates predicted in each round of dialogue to maximize the probability of having the target object's region coordinates in the set $\mathcal{R}$. PROGrasp then samples an object region r_t^q from $\mathcal{R}$. The image, the dialogue history, and the sampled object region are fed into Q-gen (*i.e.,* $P_{\mathcal{Q}_\theta}$) to produce a question (*e.g.,* "Should I get a banana?" in Figure 4.2). Next, the human user answers the question by checking whether the object mentioned in the question corresponds to the target object. After receiving the user's response, PROGrasp saves the question-and-answer pairs to the dialogue history and evaluates each object region candidate r in the set $\mathcal{R}$. In the evaluation, VG computes the likelihood of each region candidate given the multi-modal context, and A-int rescores each candidate r whether it describes the question and the user's response $(\tilde{q}_t, \tilde{a}_t)$. In other words, VG cooperates with A-int to determine the best region $\hat{r}$ for the target by evaluating how well each candidate explains the visual and dialogue context. We call it *pragmatic inference* (see Figure 4.2). In line 7

at Algorithm 1, λ is a rationality parameter [Monroe et al., 2017, Fried et al., 2018, Shen et al., 2019] in the range $[0, 1]$ that indicates the relative importance of the evaluation from A-int in pragmatic inference.

Object Grasping. The object-grasping module (OG) first computes the 3D coordinates of the predicted object using the 2D region coordinates $\hat{r}$ and the point cloud $\mathcal{P}$. Specifically, OG matches the 2D object region with the point cloud on the identical resolution and then segments points inside the region. We employ the RANSAC [Schnabel et al., 2007] to remove the table plane from the segmented points. The 3D target coordinates are computed by averaging the segmented points. Finally, OG computes the motion planning [Coleman et al., 2014] and performs object grasping.

4.4 Experimental Setup

4.4.1 Dataset

We evaluate our proposed method on the IM-Dial dataset, collected by the chatting between two players about images. The IM-Dial dataset consists of 800 images and 500 human-to-human dialogue data that cover 86 categories of everyday objects, as shown in Figure 4.3. We divide the IM-Dial dataset into five splits: train, validation, test-seen, test-unseen, and test-cluttered. The train split contains 400 images and corresponding dialogue data for training. The validation split has 100 image and dialogue pairs. The test-seen, test-unseen, and test-cluttered data contain 100 pairs of images and intention-oriented instructions each. Note that these test splits do not have question-and-answer data, so the robotic system should identify the target object interacting with a human user. The test-unseen split includes never-before-seen objects not observed in the training procedure. The goal of the test-unseen split is to evaluate the gener-

Figure 4.3: The 86 categories of everyday objects used in the experiments.

alization ability of the robotic system. Furthermore, we define the test-cluttered as a cluttered version of the test-seen split where objects are arbitrarily placed (e.g., (a) in Figure 4.7).

4.4.2 Robotic Platform

We conduct online experiments using a physical robot arm, the 6-DoF Kinova Gen3 lite, with a two-fingered gripper. Our system utilizes Intel Realsense Depth Camera D435 to get an RGB-D image. The remote server processes our proposed algorithm and communicates with the robotic platform, which locally computes motion trajectory planning.

4.4.3 Compared Methods

We compare PROGrasp with four methods:

- **Zero-Shot**: The zero-shot approach is a visual grounding model not trained on the IM-Dial dataset. We implement it by finetuning the pre-trained OFA model on the visual grounding dataset, RefCOCO [Yu et al., 2016]. This approach predicts the object region given an input image and intention-oriented instruction (*i.e.*, $\hat{r} \leftarrow \underset{r \in \mathcal{R}}{\operatorname{argmax}} P_{\mathcal{V}_\theta}(r|\mathcal{I}, \mathcal{D} = \ell)$).
- **SilentGrasp**: SilentGrasp is an ablative model of PROGrasp that does not have the question generation (Q-gen) and answer interpretation (A-

int) modules. It predicts the object region in the same way as Zero-Shot, but the visual grounding module (VG) is trained on the IM-Dial dataset.

- **LiteralGrasp**: LiteralGrasp is another ablative method of PROGrasp that does not have A-int. It is equivalent to PROGrasp without pragmatic inference (*i.e.,* $\lambda = 0$).

- **A-int-only**: This method is equivalent to PROGrasp that does not utilize VG in pragmatic inference (*i.e.,* $\lambda = 1$).

4.5 Results and Discussions

4.5.1 Results on Offline Experiments

Evaluation Protocol. The offline experiment aims to verify how well the robotic system discovers the target object through human-robot natural language interaction. We measure the Intersection over Union (IoU) between the target object region r^* and the predicted region after the interaction $\hat{r}$. The IoU is defined as the overlapping region between the two divided by their union region. The percentage of examples with an IoU value greater than 0.5 is typically reported as Acc@0.5. However, the threshold value of 0.5 may not be a reliable indicator to estimate the success of object grasping since object grasping requires accurate prediction of the target coordinates. We thus additionally report Acc@0.9, which requires more tight alignment between r^* and $\hat{r}$.

Results on the Validation Split. We first evaluate PROGrasp and the compared methods on the validation split. As shown in Table 4.1, PROGrasp outperforms all compared methods on all evaluation metrics. It indicates our proposed components (*i.e.,* Q-gen, A-int, and pragmatic inference) play a crucial role in boosting performance. Moreover, comparing Zero-Shot and Silent-

Table 4.1: Results on the offline experiments. Underlined scores indicate the performance of the runner-up method.

		Validation		Test-Seen		Test-Unseen		Test-Cluttered	
Method	GDH	Acc@0.5	Acc@0.9	Acc@0.5	Acc@0.9	Acc@0.5	Acc@0.9	Acc@0.5	Acc@0.9
Zero-Shot [Wang et al., 2022]		14%	4%	14%	7%	3%	2%	6%	4%
SilentGrasp		50%	44%	54%	45%	45%	31%	41%	22%
A-int-only		82%	75%	81%	66%	78%	57%	83%	40%
LiteralGrasp		84%	75%	85%	74%	73%	54%	84%	41%
Zero-Shot [Wang et al., 2022]	✓	51%	16%	-	-	-	-	-	-
SilentGrasp	✓	83%	72%	-	-	-	-	-	-
PROGrasp (ours)		**87%**	**79%**	**90%**	**75%**	**83%**	**61%**	**88%**	**42%**

Grasp, even the strong pre-trained model performs poorly without adapting to Pragmatic-IOG.

We further study a new setting for Zero-Shot and SilentGrasp, called Grounding from Dialog History (GDH). We naturally assume Zero-Shot and SilentGrasp can only access the intention-oriented instruction (*i.e.,* ℓ) in the inference phase since neither approach has a module for question generation. However, the initial instruction is insufficient to pinpoint the target object. In GDH, we assume that Zero-Shot and SilentGrasp can access the ground-truth human-to-human dialogue history, so we feed the entire dialogue history (*i.e.,* intention-oriented instruction and a set of question and answer pairs) into the models. As shown in Table 4.1, GDH significantly boosts Acc@0.9 of Zero-Shot (4%→16%) and SilentGrasp (44%→72%). The results indicate that additional question and answer pairs contain detailed information to identify the target object. Remarkably, PROGrasp outperforms SilentGrasp with GDH, although it does not require the ground-truth dialogue history. The results illustrate that PROGrasp works effectively, even in a more realistic scenario.

Results on the Test Splits. We compare PROGrasp with the compared methods on the test-seen, test-unseen, and test-cluttered splits. In Table 4.1, PROGrasp consistently improves on all test splits compared with Zero-Shot, SilentGrasp, A-int-only, and LiteralGrasp. Specifically, compared with LiteralGrasp, PROGrasp improves Acc@0.5 on ten points (73%→83%) and Acc@0.9 on seven points (54%→61%) in the test-unseen split. We could not investigate the results of GDH on the test splits since the test splits do not have ground-truth dialogue data. Surprisingly, PROGrasp shows decent Acc@0.5 scores even in the test-cluttered and test-unseen splits, but relatively lower scores are observed on Acc@0.9. It illustrates that (1) accurately identifying partially oc-

Table 4.2: The effect of pragmatic inference (PI).

w/o PI \ PI	Test-Seen (N=100)	Correct	Incorrect
	Correct	83%	**2%**
	Incorrect	**7%**	8%

w/o PI \ PI	Test-Unseen (N=100)	Correct	Incorrect
	Correct	72%	**1%**
	Incorrect	**11%**	16%

Table 4.3: Comparison with the multimodal foundation model.

Method	Acc@0.1	Acc@0.5	Acc@0.9
GPT-4V [OpenAI, 2023b]	29%	9%	1%
GPT-4 [OpenAI, 2023a] + VG	69%	69%	63%
GPT-4V [OpenAI, 2023b] + VG	82%	82%	68%
PROGrasp (ours)	**87%**	**87%**	**79%**

cluded target objects and (2) generalizing a robotic system to previously unseen objects are challenging aspects of this task. We will discuss more details in the qualitative analysis.

Analysis on Pragmatic Inference. We identify how predictions of PROGrasp change due to the use of pragmatic inference (PI). Thus, we employ two models: PROGrasp with and without PI, and PROGrasp without PI is equivalent to LiteralGrasp. As shown in Table 4.2, pragmatic inference changes incorrect predictions (7% and 11%) to correct predictions in test-seen and test-unseen splits, respectively. Only a small percentage of correct predictions (2% and 1%) were changed to incorrect predictions. Furthermore, pragmatic inference recovers 46.67% (7/15) and 40.74% (11/27) incorrect predictions from LiteralGrasp by employing the predictions of the answer interpretation module. It indicates that pragmatic inference effectively discovers target objects.

Comparison with Foundation Models. We further identify the performance of the state-of-the-art foundation models, GPT-4 [OpenAI, 2023a] and GPT-

4V(ision) [OpenAI, 2023b] on the validation split. Both models are provided detailed text prompts (see Figure 4.4). Table 4.3 shows the results. We observe that GPT-4V infers the target object well, but it poorly specifies the region coordinates of the target object. We thus study a hybrid approach: (1) GPT-4 and GPT-4V interacts with users and generates a distinctive caption of the best-fit object and (2) our VG model then predicts the coordinates $\hat{r}$ based on the caption. In Table 4.3, we draw two observations. First, PROGrasp outperforms existing foundation models on all evaluation metrics. It demonstrates the significance of pragmatic inference. Second, the multimodal foundation model (*i.e.,* GPT-4V) shows improved performance compared with the language-only model (*i.e.,* GPT-4). This indicates that the Pragmatic-IOG task requires understanding of visual inputs, including spatial relationships between objects or visual attributes of objects.

Hyperparameter Study. We study how the hyperparameters in PROGrasp (*i.e.,* λ and T) affect performance. Note that λ indicates the importance of the evaluation from A-int in pragmatic inference, and T denotes the number of interactions between PROGrasp and the human in Algorithm 1. As shown in Figure 4.5, we visualize the Acc@0.9 performance on the validation split. The blue, red, and green lines denote the results when $T = 1$, $T = 2$, and $T = 3$, respectively. We observe a huge performance gap between $T = 1$ and $T = 2$. It implies that many incorrect guesses in the first round of dialogue are corrected in the second round. Moreover, comparing $T = 2$ with $T = 3$, improvements seem saturated.

We also identify that performance varies depending on the value of the rationality parameter λ. LiteralGrasp is equivalent to $\lambda = 0$, and A-int-only corresponds to $\lambda = 1$. The best results are observed in $\lambda = 0.9$ across all T

You are a generalist agent following natural language instructions through dialogue with humans. Please return the desired output by referring to the following explanation.

TASK DESCRIPTION:
The task you should perform requires pragmatic reasoning capabilities. Specifically, your goal is to specify a target object in the given image based on natural language instructions, but the instructions do not explicitly contain the category of the target object. The instructions are intention-oriented, for example, "I am thirsty" or "My device runs out of battery." So, you need to infer the intent of the instructions based on context (i.e., an input image) and specify one target object that best matches the given instruction.

Furthermore, the instructions are often ambiguous. For example, you can see more than two drinkable objects in the image when the initial instruction is "I am thirsty." In this case, you can ask questions to disambiguate the target object, such as "Do you need the can of Coke?". The human users will answer your questions. You can ask up to three questions for each sample. If you think the target object is clearly specified, you should finally generate a distinctive description of the target object.

OUTPUT FORMAT:
Your output should be either the question for disambiguation or the final description of the target object. Both should be generated in one sentence.

EXAMPLE:
The instruction is "I am hungry". You detect kiwi, strawberry, tomato, and banana in the image. So, you can ask, "Do you need the banana?". The human user answers "No." So, you ask the follow-up question, "Do you want to eat the strawberry?" Then, the human says, "No, the kiwi seems better for me." You should describe the target object, such as "The kiwi."

Based on the task description and example above, please describe the target object through dialogue with humans. The current instruction is {Instruction}.

Figure 4.4: A text prompt for foundation models.

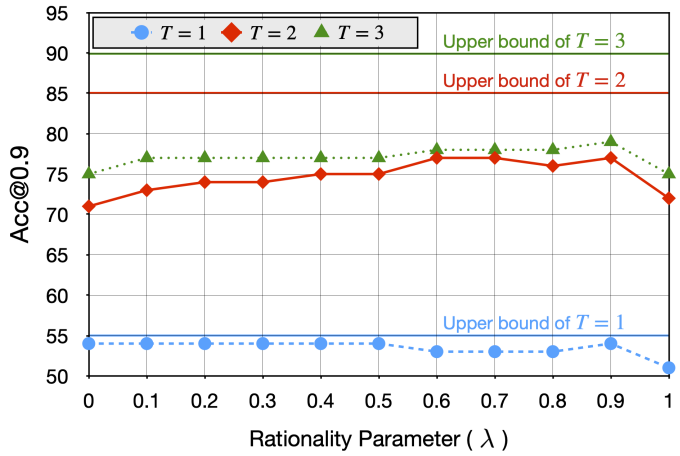


Figure 4.5: Validation scores adjusting the hyperparameters, λ and T.

values. However, we do not see any merit in PROGrasp compared with LiteralGrasp when $T = 1$. To delve into this phenomenon, we visualize the upper bound performance for each T value. The upper bound is defined as the performance when the system perfectly selects the object region in a set of object region candidates $\mathcal{R}$. The upper bound performance of $T = 1$ is 55%, and PROGrasp and LiteralGrasp both show 54%. It illustrates that there is little room for improvement in $T = 1$. In contrast, the upper bound of $T = 2$ is 85%, and LiteralGrasp shows 71% in Figure 4.5. PROGrasp boosts 6% compared with LiteralGrasp (71%→77%). Likewise, we observe 4% gains (75%→79%) when $T = 3$. Unless stated otherwise, λ is 0.9, and T is 3.

Communicative Efficiency. Beyond task success, communicative efficiency is also an important criterion for pragmatic systems [Fried et al., 2022]. Accordingly, we measure the average number of interactions (*i.e.,* question answering) required for the system to identify the target object. In this study, we assume that the dialogue immediately ends when the Intersection over Union (IoU) between the predicted and target regions is greater than 0.5. The efficiency can range from 1.0 to $T = 3$. We compare PROGrasp with three baselines: Random, A-int-only, and LiteralGrasp. The Random randomly selects the predicted object in the set of candidates (*i.e.,* $\mathcal{R}$ in Algorithm 1). In Table 4.4, PROGrasp consistently improves the efficiency across all test splits. It indicates that our system efficiently identifies the target object through fewer interactions.

4.5.2 Results on Online Experiments

Evaluation Protocol. We reproduce 100 images in the test-seen split and conduct online experiments to study how well the system picks the desired object up through human-robot dialogue. We compare PROGrasp with SilentGrasp

Table 4.4: Analysis of communicative efficiency. ↓ indicates lower is better.

	Avg. # of Interactions ↓		
Method	Test-Seen	Test-Unseen	Test-Cluttered
Random	1.76	2.00	1.98
A-int	1.60	1.78	1.76
LiteralGrasp	1.55	1.78	1.71
PROGrasp (ours)	**1.53**	**1.72**	**1.69**

Table 4.5: Results on the online experiments.

	Ambiguous	Non-Ambiguous	Total
Method	Object Discovery / Success Rate		
SilentGrasp	42 / 30	78 / 38	56 / 33
LiteralGrasp	77 / 47	**90** / **45**	82 / 46
PROGrasp (ours)	**80** / **53**	**90** / **45**	**84** / **50**

and LiteralGrasp. The human rater evaluates object discovery and success rate. Object discovery measures whether the system correctly localizes the target.

Results. We categorize 100 samples into two groups: Ambiguous (60) and Non-Ambiguous (40). The Ambiguous group is a set of samples that require distinguishing between two objects given an initial instruction. The Non-Ambiguous group corresponds to the remaining samples. In Table 4.5, PROGrasp achieves a total execution success rate of 50%, outperforming all baselines. It demonstrates that the superiority of PROGrasp's target object discovery is successfully transferred to the success rate. Not surprisingly, PROGrasp is effective in the Ambiguous, boosting success rate of 23% compared with SilentGrasp. However, we observe many failure cases, although the target object is correctly localized. The system often drops the objects during lifting or fails to grasp them since objects are highly unstructured (see Figure 4.3). More precise object grasping will mitigate this issue. We leave it as a future work.

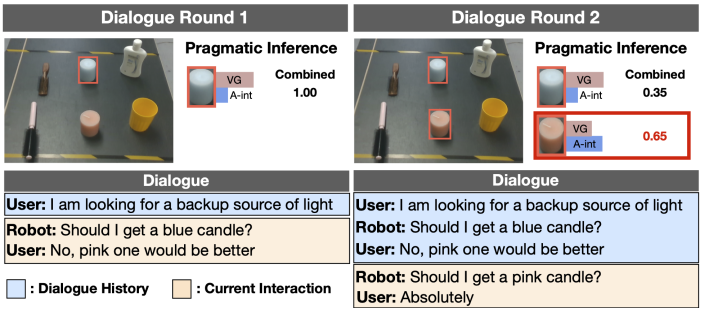


Figure 4.6: Visualization of PROGrasp's target object recovery.

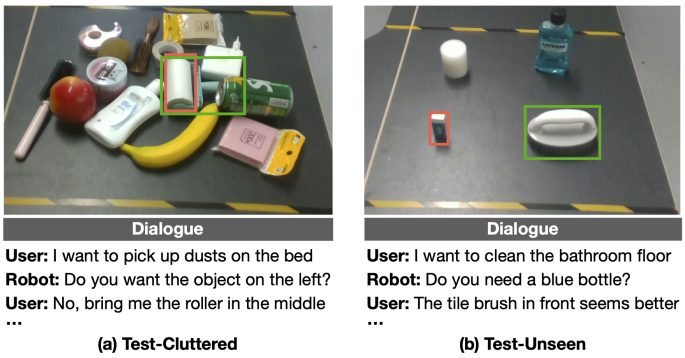


Figure 4.7: Visualization of the failure cases.

4.5.3 Qualitative Analysis

Results. In Figure 4.6, we visualize the inferred results from PROGrasp when $T = 2$. PROGrasp fails to find the target object (*i.e.*, a pink candle) in the first round of the dialogue. Still, it corrects the target in the second round by utilizing the question-and-answer pair from the first round as additional context and includes the target object as a candidate. Pragmatic inference finally selects the desired object region. This example clearly explains the performance gap between $T = 1$ and $T = 2$ in Figure 4.5. We also visualize two failure examples from the test-cluttered and test-unseen splits in Figure 4.7. The red and green boxes indicate the predicted and ground-truth regions, respectively. As

in (a) at Figure 4.7, our system fails to identify the target region accurately due to the occlusion, which explains low scores in Acc@0.9 of the test-cluttered. Furthermore, PROGrasp sometimes makes an incorrect guess (*i.e.,* (b) in Figure 4.7) when observing never-seen-before objects, highlighting the need for further generalization in future work.

4.6 Conclusion

We propose a new task scenario for interactive object grasping called Pragmatic-IOG to study the pragmatic reasoning behavior of embodied language agents. Furthermore, we introduce a modular approach for Pragmatic-IOG called pragmatic object grasping (PROGrasp), consisting of five components: visual grounding, question generation, answer interpretation, object grasping, and pragmatic inference. Experiments demonstrate that PROGrasp effectively reasons about underspecified instructions and disambiguates the target object through dialogue with humans. Moreover, PROGrasp outperforms several compared methods in online experiments (*i.e.,* IOG with a physical robot arm). We believe our proposed task setup and the robotic system to open the door to developing intelligent agents with pragmatic reasoning abilities.

Chapter 5

Learning Robotic Skills from Natural Language

5.1 Introduction

Advances in artificial intelligence and robotics accelerate the development of robots that can follow natural language instructions. Studies on robotic manipulation have developed such robots by training language-conditioned policies through imitation learning [Stepputtis et al., 2020, Jang et al., 2022, Mees et al., 2022, Brohan et al., 2022, 2023, Padalkar et al., 2023]. However, imitation learning involves robot demonstration data, and collecting demonstrations often requires expertise in robot control or access to specialized devices, such as teleoperation or virtual reality systems [Xiao et al., 2023, Fu et al., 2024, Seo et al., 2023]. This barrier severely limits the accessibility and scalability of robot data collection, reducing the potential diversity of data and the ability to collect demonstrations at scale. We thus ask: *how can non-experts train robotic policies without relying on specialized expertise or devices for collecting data?*

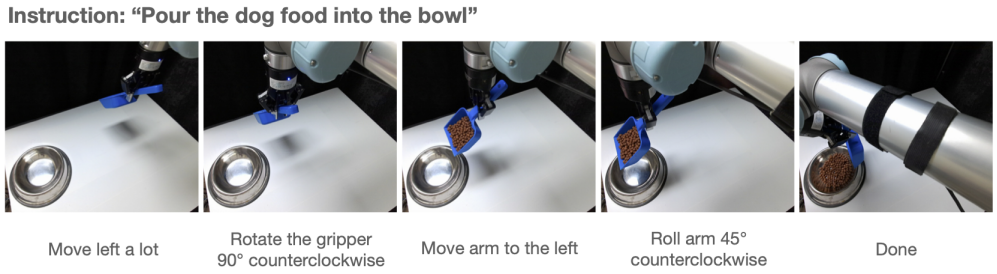


Figure 5.1: Overview of language-based teleoperation.

We explore a method for training robotic manipulation skills solely through natural language, leveraging it as an intuitive and accessible interface for robot learning. To this end, we first propose a data collection framework that allows non-experts to collect in-domain robot data from natural language supervision. The framework consists of two components: language-based teleoperation and stochastic trajectory diversification (STD). Figure 5.1 illustrates language-based teleoperation in which a human collects data for a skill described in the instruction (*e.g.,* “pour the dog food into the bowl”). The human provides natural language supervision (*e.g.,* “move left a lot”) in each state. Our framework employs large language models (LLMs) [OpenAI, 2023a] to translate this supervision into appropriate robotic actions, which are then executed by the robot. By repeating this process, robot demonstrations paired with instructions become available. However, this process requires the user’s language supervision, making it costly to collect large-scale robotic data. Thus, our framework automatically augments existing demonstration data using STD. Specifically, STD treats demonstration trajectories collected by humans as near-optimal and iteratively samples alternative trajectories. During sampling, it also makes the robot deviate from the points within these trajectories. A simple heuristic algorithm labels how robots should behave at each point. Consequently, STD scales

human-collected robotic data by an order of magnitude, allowing cost-effective training with as few as 10 episodes of human supervision per task.

We introduce a model that learns language-conditioned policies from natural language supervision, which we call CLIP-RT (CLIP-based Robotics Transformer). Our model extends CLIP [Radford et al., 2021], which uses language as a training signal for visual representations, to robot learning for generalist manipulation policies. A key idea is to introduce natural language itself as a supervision to train robotic policies. CLIP-RT employs CLIP models trained on Internet-scale data [Schuhmann et al., 2022, Fang et al., 2023] and adapts them to predict actions specified in a language based on contrastive imitation learning. Specifically, our model learns to measure the pairwise similarity between language supervision and contextual information (*i.e.,* current scene and language instruction) for language-conditioned policies. It is in stark contrast to existing approaches [Brohan et al., 2023, Padalkar et al., 2023, Belkhale et al., 2024, Kim et al., 2024] in two aspects: (1) CLIP-RT does not need to specify tokens to encode actions since actions are represented in natural language and (2) our model is *discriminative* rather than generative when predicting robotic actions. We train CLIP-RT via a two-step process: robot action pretraining and in-domain skill acquisition. In the pretraining stage, we train our model on the large-scale robot learning dataset (*i.e.,* Open X-Embodiment [Padalkar et al., 2023]) to improve generalization capabilities. Since the dataset does not contain natural language supervision, we transform existing low-level actions into natural language supervisions to train CLIP-RT. Next, CLIP-RT learns the desired skills using our collected in-domain data.

Our contributions are fourfold. First, we propose CLIP-RT, a vision-language-action (VLA) model that learns language-conditioned policies from natural language supervision. Second, we propose a data collection framework that enables

non-experts to collect robot data only through natural language and expand the size of the human-collected data through the automatic data collection method. Third, we demonstrate that CLIP-RT outperforms the state-of-the-art model, OpenVLA [Kim et al., 2024], by 17% in average success rates in 10 novel manipulation tasks. Fourth, our ablation studies reveal two key findings: (1) the CLIP model favors natural language supervision over existing action encoding strategies, significantly boosting task success rates, and (2) STD proves effective when human-collected robot demonstration data are scarce.

5.2 Related Work

5.2.1 Collection of Real-World Robot Data

Data collection has become an increasingly important challenge in robot learning. Previous works have collected real-world robot demonstrations through various interfaces, such as teleoperation devices [Fu et al., 2024, Abbeel et al., 2010, Hristov and Ramamoorthy, 2021], virtual reality (VR) [Zhang et al., 2018, Seo et al., 2023], and kinesthetic teaching [Billard et al., 2006, Maeda et al., 2017, Eteke et al., 2020, Yang et al., 2023]. Some studies introduce natural language interfaces [Liu et al., 2023, Belkhale et al., 2024] for data collection, but they are often used in limited scenarios. RT-H [Belkhale et al., 2024] and OLAF [Liu et al., 2023] first train robotic policies using data collected from other interfaces (*e.g.,* VR). During deployment, humans provide language feedback to correct robotic behaviors and policies are updated based on this feedback. In other words, these approaches focus on refining learned policies on *existing* skills. In contrast, our focus is on using language as an interface to obtain complete demonstration trajectories for learning *any desired* skills. To achieve this, our framework leverages the in-context learning capabilities of large language models (LLMs) [He et al., 2024] for translation from language to action.

5.2.2 Language-Conditioned Robotic Policies

The research community has made extensive efforts to develop robotic systems that can follow language instructions [Kollar et al., 2010, Chen and Mooney, 2011, Thomason et al., 2020, Kim et al., 2023, Kang et al., 2024a], often training language-conditioned policies [Stepputtis et al., 2020, Lynch and Sermanet, 2020, Shridhar et al., 2022, Jang et al., 2022, Mees et al., 2022, Brohan et al., 2022, 2023, Kim et al., 2024, Belkhale et al., 2024]. We train language-conditioned policies through imitation learning similar to existing studies. Unlike existing studies, we train language-conditioned policies with contrastive imitation learning, which combines the ideas of contrastive learning [Radford et al., 2021] with imitation learning [Pomerleau, 1988] for more discriminative representations of robotic behaviors.

5.2.3 Vision-Language-Action (VLA) Models

Vision-language models (VLM) trained on Internet-scale data have been extensively studied for robotics, such as high-level planning [Driess et al., 2023, Hu et al., 2023], success detection [Du et al., 2023], physical reasoning [Gao et al., 2023], and robotic control [Shridhar et al., 2022]. In particular, a line of research [Brohan et al., 2023, Padalkar et al., 2023, Belkhale et al., 2024, Kim et al., 2024] has directly fine-tuned VLMs to predict robotic actions without any new parameters. This category of models is called vision-language-action (VLA) models. CLIP-RT falls into the category of VLA models. Current VLA models consider robotic action as a foreign language and encode it in the same way as a natural language. Specifically, each action is discretized and mapped to *existing* tokens that are the least used in the vocabulary. Thus, VLA models do not require additional parameters to encode actions. However, CLIP-RT learns to predict actions specified in natural language (*e.g.,* "move left"), so it

does not overwrite existing text tokens to represent robotic actions. We discuss the effect of natural language-based action encoding in experiments.

5.3 Approach

5.3.1 Preliminaries

Language-Conditioned Imitation Learning. A dataset $\mathcal{D} = \{(\tau_n, \ell_n)\}_{n=1}^{N}$ consists of demonstrations τ paired with language instructions ℓ. Each demonstration contains a sequence of visual observations and expert actions $\tau_n = \{(v_1, a_1), \ldots, (v_{|\tau_n|}, a_{|\tau_n|})\}$. The goal of language-conditioned imitation learning is minimizing the negative log-likelihood of the expert action a_t given the observation history $v_{1:t} = (v_1, \ldots, v_t)$ and language instruction ℓ:

$$\mathcal{L}_{\text{IL}} = -\mathbb{E}_{(\tau,\ell)\sim\mathcal{D}} \left[\sum_{t=1}^{|\tau|} \log \pi_\theta(a_t | v_{1:t}, \ell) \right] \tag{5.1}$$

where π_θ denotes the policy model with model parameters θ. For vision-language action (VLA) models, θ is initialized from the parameters of vision-language models (VLMs). To maintain consistency with the pre-training setup of VLMs, existing VLA models [Brohan et al., 2023, Kim et al., 2024, Belkhale et al., 2024] typically use single-image observations v_t rather than utilizing the full observation history $v_{1:t}$. CLIP-RT follows this same approach. At test time, the policy model performs closed-loop robot control until it completes instructions.

End-Effector Actions. The seven degrees of freedom (7-DoF) end-effector action, which enables a robotic arm to control the movement and orientation of its end-effector within the Cartesian coordinate system, is commonly used in robotic manipulation tasks. The action is defined as:

$$a_t \triangleq (\Delta\texttt{pos}_x, \Delta\texttt{pos}_y, \Delta\texttt{pos}_z, \Delta\texttt{rot}_x, \Delta\texttt{rot}_y, \Delta\texttt{rot}_z, \texttt{gripper}) \in \mathbb{R}^7$$

where all positional (`pos`) and rotational (`rot`) displacements along the x, y, and z axes are represented as delta values. The `gripper` controls the opening of the gripper, ranging from 0 (closed) to 1 (open). Policy models typically predict the seven-dimensional action at each state. In contrast, CLIP-RT learns to predict language actions, which can then be translated into end-effector actions. End-effector actions are finally converted into joint angles through inverse kinematics (IK), which maps a desired position and orientation in Cartesian space back to the joint configuration needed to achieve that position.

Contrastive Language-Image Pretraining (CLIP). CLIP [Radford et al., 2021] is a method for learning visual representations from natural language supervision at scale. By employing the contrastive objective, CLIP trains an image encoder $f(\cdot)$ and a text encoder $g(\cdot)$ on 400M image-text pairs. Given a mini-batch of M image-text pairs $\{(I_i, T_i)\}_{i=1}^{M}$, the two encoders are jointly optimized to maximize the similarity between correct pairs of image and text (I_i, T_i) while minimizing the similarity for incorrect pairs $(I_i, T_{j\neq i})$:

$$\mathcal{L}_{\text{CL}} = -\frac{1}{2M}\sum_{i=1}^{M}\left[\log \underbrace{\frac{\exp(\mathbf{x}_i \cdot \mathbf{y}_i)}{\sum_{j=1}^{M}\exp(\mathbf{x}_i \cdot \mathbf{y}_j)}}_{\text{image}\to\text{text softmax}} + \log \underbrace{\frac{\exp(\mathbf{x}_i \cdot \mathbf{y}_i)}{\sum_{j=1}^{M}\exp(\mathbf{x}_j \cdot \mathbf{y}_i)}}_{\text{text}\to\text{image softmax}}\right] \tag{5.2}$$

where $\mathbf{x}_i = \frac{f(I_i)}{\|f(I_i)\|_2}$ and $\mathbf{y}_i = \frac{g(T_i)}{\|g(T_i)\|_2}$ are normalized vector embeddings for image and text, respectively. The pairwise similarity is defined as dot product of these two embeddings, equivalent to the cosine similarity. Note that each pairwise similarity is independently normalized by two softmax functions: one over all texts for each image (*i.e.,* image $\to$ text softmax) and vice versa (*i.e.,* text $\to$ image softmax). As we describe later, we modify the contrastive loss to make CLIP-RT compatible with language-conditioned imitation learning.

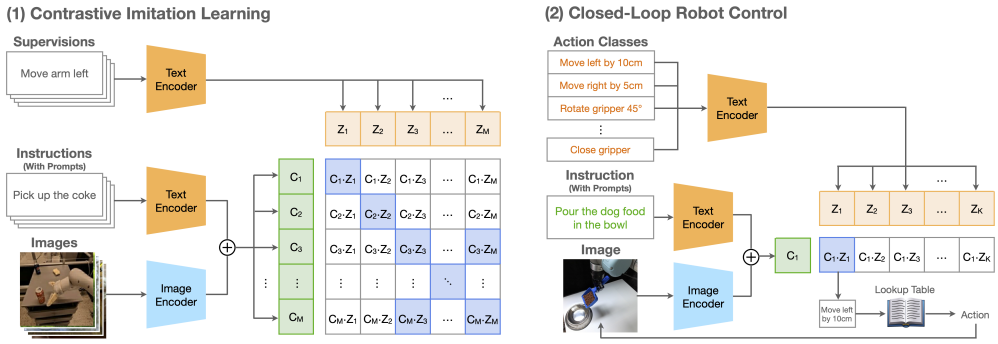


Figure 5.2: Overview of CLIP-RT. In practice, we add a simple text prompt to language instructions: *What motion should the robot arm perform to complete the instruction {instruction}?*

5.3.2 CLIP-based Robotics Transformer (CLIP-RT)

In this subsection, we introduce CLIP-RT, a new vision-language-action model that learns language-conditioned robotic policies from natural language supervision. We first provide an overview of natural language supervision and then describe how CLIP-RT learns robotic policies from this supervision. Finally, we describe our model's closed-loop robot control at test time.

Natural Language Supervision. Inspired by CLIP [Radford et al., 2021] that appreciates natural language as a training signal, we build an analogous model to learn language-conditioned policies from natural language supervision. We define natural language supervision as language-based direction that guides robots on how to behave in specific states to complete given instructions. This typically involves altering the robot's position, rotation, or gripper state. As we discuss later, each language supervision is associated with a specific low-level expert action, and this information is used to identify language supervisions that are semantically interchangeable. Robot learning from natural language supervision has several merits. It establishes a clear hierarchy between initial

instruction and language supervision, enabling models to learn *shared structures* across diverse tasks [Belkhale et al., 2024]. Moreover, since language supervision does not rely on specific action formats (*e.g.,* 7D end-effector commands), the same supervision can be used across different robot embodiments with varying hardware and control systems.

VLM Backbone. CLIP-RT maintains the original CLIP model architecture without any new parameters. We employ an open-source CLIP model[1] [Fang et al., 2023, Ilharco et al., 2021] of 1B parameters that achieves state-of-the-art performance in zero-shot image classification [Russakovsky et al., 2015]. It consists of an image encoder [Dosovitskiy et al., 2021] and a text encoder [Radford et al., 2019], both built on a Transformer architecture [Vaswani et al., 2017].

Contrastive Imitation Learning (CIL). We describe contrastive imitation learning in Figure 2. CLIP-RT takes a mini-batch of M triplets $\{(v_i, \ell_i, u_i)\}_{i=1}^{M}$, where v, ℓ, and u denote image observation, instruction, and language supervision. Contrastive imitation learning aims to optimize the pairwise similarities in the set $\{((v_i, \ell_i), u_j) | i, j \in \{1, \ldots, M\}\}$. Specifically, CLIP-RT first extracts vector embeddings of v_i, ℓ_i and u_j using the CLIP model's image encoder $f(\cdot)$ and the text encoder $g(\cdot)$, and subsequently combines the image and instruction embeddings:

$$\mathbf{c}_i = f(v_i) + g(\ell_i), \quad \mathbf{z}_j = g(u_j) \tag{5.3}$$

where $\mathbf{c}_i$ represents the context that encapsulates the robot's current visual state and its explicit goal. $\mathbf{z}_j$ represents the immediate action that should be

[1] `https://github.com/mlfoundations/open_clip`

taken given the context. We design the loss function as:

$$\mathcal{L}_{\text{CIL}} = -\frac{1}{M^2}\sum_{i=1}^{M}\sum_{j=1}^{M}\Big[y_{ij}\log\sigma(\hat{\mathbf{c}}_i\cdot\hat{\mathbf{z}}_j) + (1-y_{ij})\log(1-\sigma(\hat{\mathbf{c}}_i\cdot\hat{\mathbf{z}}_j))\Big] \quad (5.4)$$

where $\hat{\mathbf{c}}_i = \frac{\mathbf{c}_i}{\|\mathbf{c}_i\|_2}$ and $\hat{\mathbf{z}}_j = \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|_2}$ are normalized vector embeddings of $\mathbf{c}_i$ and $\mathbf{z}_j$. $\sigma(\cdot)$ is a sigmoid activation function, and $y_{ij} \in \{0, 1\}$ denotes a label for the pairwise similarity. The loss function maximizes the cosine similarity between the context and language supervision for positive pairs, while minimizing it for negative pairs. The label y_{ij} is basically one if $i = j$; otherwise, it is zero. In other words, $((v_i, \ell_i), u_i)$ are positive pairs, and $((v_i, \ell_i), u_{j\neq i})$ are negative pairs. However, due to the limited number of language supervisions, the mini-batch often contains semantically interchangeable supervisions, such as "move upwards" and "raise the arm". Thus, CIL consults low-level actions a_i associated with language supervision u_i and treats the pair $((v_i, \ell_i), u_{j\neq i})$ as positive if two supervisions share the same low-level action. As a result, y_{ij} is one if $i = j$ or $a_i = a_j$ (see the blue boxes in Figure 2); otherwise, it is zero. Consequently, CLIP-RT learns to measure the likelihood of each motion described in language, given visual observation and language instruction.

Robot-Action Pretraining. We train CLIP-RT on the Open X-Embodiment dataset [Padalkar et al., 2023], an open large-scale dataset for robot learning. The data set includes 2.4M robotic trajectories from 70 individual data sets. Recent work [Kim et al., 2024] has made a significant effort in data curation, so we use the curated data for training. However, the data do not contain natural language supervision. We thus synthesize it from end-effector actions. As described in Section 5.3.1, the end-effector action is represented as a seven-dimensional tuple consisting of the delta positions, the delta rotations, and the gripper action to open or close. We identify the entry with the maximum value

and its corresponding axis for each action. This information is mapped to one of 50 predefined natural language supervisions (see Figure 5.3). To increase the diversity of natural language supervision, we use large language models (LLMs) to paraphrase the original supervisions. As a result, we train CLIP-RT on approximately 18.1M instances with 899 different natural language supervisions through contrastive imitation learning. It requires four H100 GPUs for one day with a batch size of 128.

1. move arm back by 20cm
2. move arm back by 10cm
3. move arm back by 5cm
4. move arm back by 1cm
5. move arm forward by 1cm
6. move arm forward by 5cm
7. move arm forward by 10cm
8. move arm forward by 20cm
9. move arm to the right by 20cm
10. move arm to the right by 10cm
11. move arm to the right by 5cm
12. move arm to the right by 1cm
13. move arm to the left by 1cm
14. move arm to the left by 5cm
15. move arm to the left by 10cm
16. move arm to the left by 20cm
17. lower arm by 20cm
18. lower arm by 10cm
19. lower arm by 5cm
20. lower arm by 1cm
21. raise arm up by 1cm
22. raise arm up by 5cm
23. raise arm up by 10cm
24. raise arm up by 20cm
25. roll arm 90 degrees counterclockwise
26. roll arm 45 degrees counterclockwise
27. roll arm 15 degrees counterclockwise
28. roll arm 5 degrees counterclockwise
29. roll arm 5 degrees clockwise
30. roll arm 15 degrees clockwise
31. roll arm 45 degrees clockwise
32. roll arm 90 degrees clockwise
33. tilt arm up 90 degrees

34. tilt arm up 45 degrees
35. tilt arm up 15 degrees
36. tilt arm up 5 degrees
37. tilt arm down 5 degrees
38. tilt arm down 15 degrees
39. tilt arm down 45 degrees
40. tilt arm down 90 degrees
41. yaw arm 90 degrees counterclockwise
42. yaw arm 45 degrees counterclockwise
43. yaw arm 15 degrees counterclockwise
44. yaw arm 5 degrees counterclockwise
45. yaw arm 5 degrees clockwise
46. yaw arm 15 degrees clockwise
47. yaw arm 45 degrees clockwise
48. yaw arm 90 degrees clockwise
49. close gripper
50. open gripper

Figure 5.3: A list of predefined natural language supervisions.

Closed-Loop Robot Control. An overview of closed-loop robot control is shown in Figure 5.2. At each time step, CLIP-RT infers pairwise similarities between context information and K action classes that describe robotic behaviors in the language. Our model selects the language action with the maximum probability. Finally, the language action is translated into the lower-level end-effector commands based on a pre-defined lookup table. Unlike other Transformer-based policy models [Brohan et al., 2023, 2022, Padalkar et al., 2023, Belkhale et al., 2024, Kim et al., 2024] relying on autoregressive decoding, CLIP-RT can predict action in a *single* forward pass since it is a discriminative model. CLIP-RT requires 7GB of GPU memory and runs at 16Hz (one H100 GPU using float32 precision) and 8Hz (one NVIDIA RTX 3090 GPU using float32 precision) without applying any speed-up tricks, such as model quantization and compilation.

5.3.3 In-Domain Robot Data Collection

In this subsection, we describe how we can collect robotic data solely through natural language. We introduce our data collection framework consisting of two steps: (1) language-based teleoperation and (2) stochastic trajectory diversification (STD) which augments demonstrations collected from the first step.

Language-based Teleoperation. This step aims to collect a set of tuples containing visual observation, initial instruction, natural language supervision, and low-level action. To this end, we design a scenario in which users collect such data by interacting with large language models (LLMs) [OpenAI, 2023a]. Specifically, users first provide an initial language instruction for a skill. Then, they provide natural language supervisions in specific states to complete the instruction. LLMs finally translate the language supervision into the low-level end-effector command based on a detailed text prompt (see Figure 5.4). The prompt outlines information about (1) input and output space, (2) the 3D Cartesian coordinate system of the environment, and (3) input-output samples for in-context learning. We collect 10 episodes for each skill through this process.

Stochastic Trajectory Diversification (STD) aims to augment demonstration data collected from language-based teleoperation. Before delving into the details, we first define a *waypoint* as a special state in demonstrations that satisfies either of the following conditions: (1) the gripper state changes (*i.e.,* open $\rightarrow$ close and close $\rightarrow$ open) or (2) the cumulative progress of delta positions or orientations (see Section 5.3.1) along any axis reverses. For example, w_1 in Figure 5.5-(a) is a waypoint since the cumulative progress on a horizontal axis starts to reverse at w_1.

You are a generalist agent who can control a physical robot arm with a two-finger gripper, given natural language supervision from humans. Please return the desired output by referring to the following explanation.

TASK DESCRIPTION:
The task you should perform is to translate the natural language supervision from humans into the corresponding robot end effector command. Language supervision entails diverse actions: (1) displacing the end effector's 3D Cartesian coordinates or poses and (2) directly rotating the gripper.

OUTPUT FORMAT:
You should output the end effector command, which is a list of length seven. The first six elements correspond to the standard end effector commands. Specifically, the first three elements of the command represent delta Cartesian coordinates of the end effector (i.e., modified x, y, and z coordinates), and the next three correspond to the orientation of the end effector (i.e., modified roll, pitch, and yaw). In addition to the list of length six, we define the last element of the end effector command as the robotic arm's last joint angles to directly rotate gripper. Please note that you should output the list of length seven without detailed explanation.

ENVIRONMENT SETUP:
The physical robot arm is standing on the table, and the gripper is mounted at the end of the robotic arm. The 3D Cartesian coordinate system of the environment is as follows:

1. The x-axis is in the depth direction, increasing away from you.
2. The y-axis is in the horizontal direction, increasing to the left.
3. The z-axis is in the vertical direction, increasing upwards.

RULES:
Please note that the following rules when predicting the end effector command:

1. The units for the Cartesian coordinate system are meters.
2. The units for the roll, pitch, and yaw are degrees, from -90 to 90 degrees.
3. The joint angles of the gripper also ranges from -90 to 90 degrees.
4. Positive rotation values represent clockwise rotation, and negative rotation values represent counterclockwise rotation.
5. The end effector gripper has two fingers, and the fingers are opened and pointing downward in the initial state (i.e., parallel to the z-axis). You should predict the delta roll, pitch, and yaw based on the initial orientation.
6. If the natural language supervision does not seem relevant to the end effector commands, you should output a list of zero values.

EXAMPLE:
Here are a few examples for natural language supervision and the end effector command:

1. move to the right: [0.0, -0.1, 0.0, 0.0, 0.0, 0.0, 0.0]
2. move forward a bit: [0.05, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
3. lower arm a tiny bit: [0.0, 0.0, -0.01, 0.0, 0.0, 0.0, 0.0]
4. raise arm up a lot: [0.0, 0.0, 0.2, 0.0, 0.0, 0.0, 0.0]
5. roll arm to the left a bit: [0.0, 0.0, 0.0, 15.0, 0.0, 0.0, 0.0]
6. tilt end effector up a lot: [0.0, 0.0, 0.0, 0.0, -90.0, 0.0, 0.0]
7. yaw arm to the left a tiny bit: [0.0, 0.0, 0.0, 0.0, 0.0, -5.0, 0.0]
8. rotate gripper 45 degrees clockwise: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 45.0]
9. close the gripper : [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0].

Based on the description above, please infer the end effector command for the natural language supervision, {supervision}.

Figure 5.4: A text prompt for language-based teleoperation.

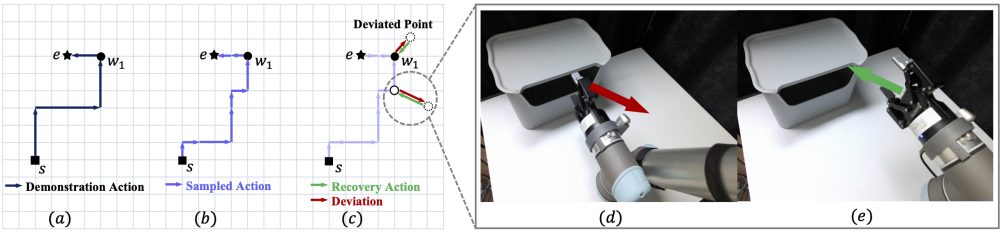


Figure 5.5: A simplified 2D example of stochastic trajectory diversification. (a): a demonstration trajectory from the start s to the endpoint e, passing through a waypoint w_1. (b): a sampled trajectory generated by the diversification phase. (c)-(e): a visualization of the recovery phase.

STD consists of two phases the *diversification phase* and the *recovery phase*. The diversification phase first builds alternative trajectories toward each waypoint (see Figure 5.5-(b)) by composing a new action sequence from the action list shown in Figure 5.3. The robot then performs each action in the sequence, recording an image at each visited state. This process repeats until the robot completes the task. As a result, the diversification enriches the robot's understanding of how to act in various states leading toward the waypoint.

In the recovery phase, the robot intentionally visits states deviating from the planned trajectory (see Figure 5.5-(d)) and then executes a recovery action, a simple reversal of the deviation to return to the trajectory (see Figure 5.5-(e)). This deviate-then-recover process occurs within $K = 3$ time steps to reach the waypoint, allowing the robot to handle errors near the waypoint. Note that the robot records only the recovery actions and images at the deviated states, not the deviation data itself. As a consequence, CLIP-RT can learn various alternative actions and how to behave at the deviated points.

5.4 Experimental Setup

We train and evaluate our models across 19 robotic manipulation tasks which are categorized into two groups: *Common* and *Novel*. **Common tasks** consist of nine tasks that are closely aligned with those found in the Open X-Embodiment dataset [Padalkar et al., 2023], such as "*pick the banana*". In contrast, **Novel tasks** consist of ten tasks barely observed during pretraining, serving as a testbed to evaluate the models' ability to acquire new skills based solely on in-domain data. In the following, we describe each robotic task.

5.4.1 Common Tasks

1. **Point**: The robot is expected to move its arm close to the object (*e.g.*, cups with different colors, dice).

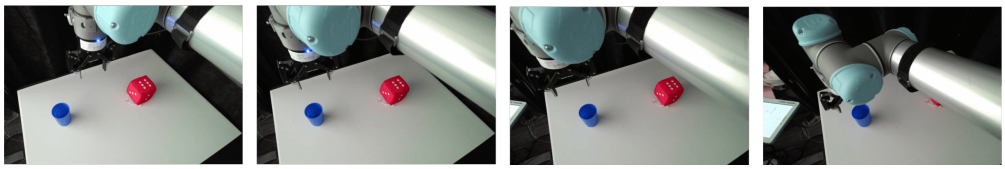


Figure 5.6: An example of a task with "point to the blue cup".

2. **Pull**: The robot is required to pull out the tissue from the tissue box.

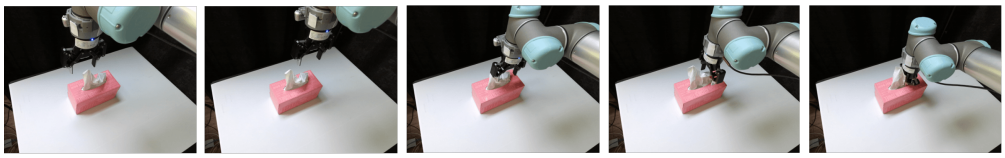


Figure 5.7: An example of a task with "pull out the tissue".

3. **Place**: The robot is required to place an object at a designated location (*e.g.*, in colored boxes or on a shape such as a star and circle). The task begins while the robot grasping the object.

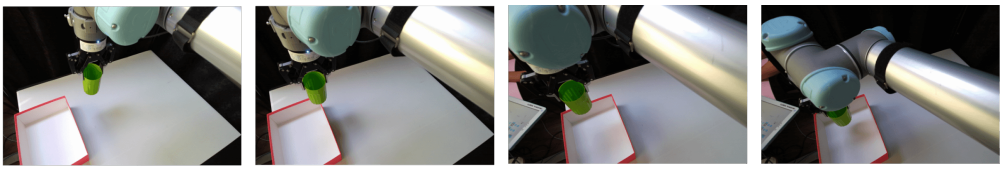


Figure 5.8: An example of a task with "place the green cup on the red box".

4. **Pick**: The robot is expected to find and grasp the object. The target objects include cups, dice, and stamps, as well as bananas.

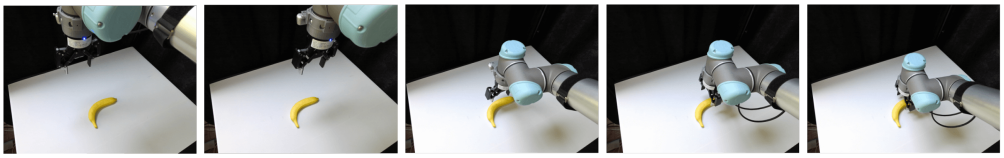


Figure 5.9: An example of a task with "pick up the banana".

5. **Push**: The robot is required to move a arm in front of the `<obj1>` (*e.g.*, a dice, box, or cup) and push the object to move it toward another `<obj2>`.

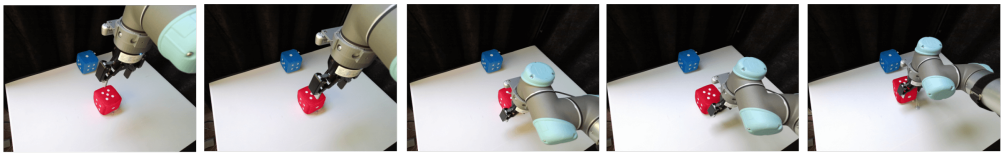


Figure 5.10: An example of a task with "push the red dice to the blue dice".

6. **Flip**: The robot is required to locate and grasp the object (*e.g.*, cups or plates), lift it, and flip it over by rotating the gripper.

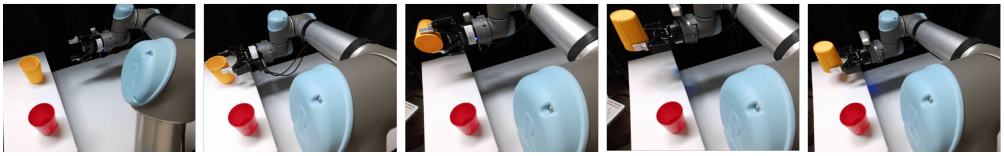


Figure 5.11: An example of a task with "flip the yellow cup".

7. **Knock Over**: The robot is required to locate and grasp the object (*e.g.*, cups), tilt the object and open the gripper to knock the object over.

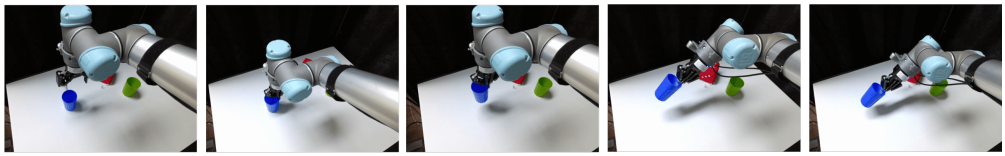


Figure 5.12: An example of a task with "knock over the blue cup".

8. **Slide**: The robot is expected to grasp the object (*e.g.*, dice, toy car), and slide it towards another object (*e.g.*, Pooh, Piglet, or a board eraser).

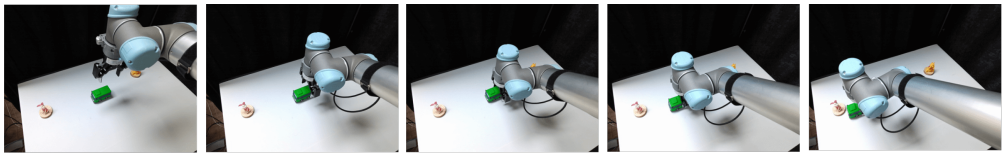


Figure 5.13: An example of a task with "slide the green car to the Piglet".

9. **Move**: The robot is required to pick up the object (*e.g.*, banana, cups, plate) and place it to the desired location (*e.g.* near or on another object).

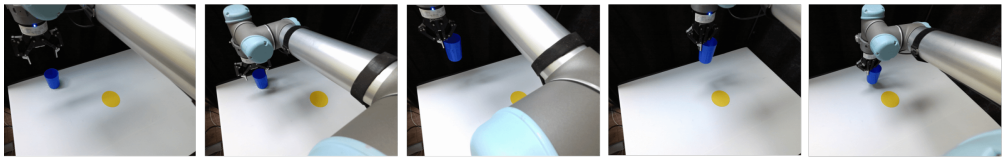


Figure 5.14: An example of a task with "move the blue cup on the yellow circle".

5.4.2 Novel Tasks

1. **Pour the Dog Food**: The robot starts by holding a blue shovel filled with dog food. The robot has to locate the silver bowl, and tilt its arm to pour the dog food into the bowl.

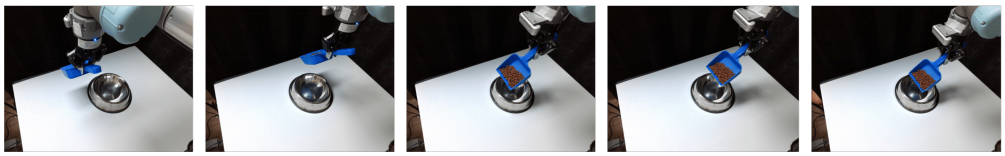


Figure 5.15: An example of a task with "pour the dog food in the bowl".

2. **Draw a Line**: The task starts with grasping a board marker. The robot should draw a line that meets the specified condition, such as drawing the line vertically or horizontally, from one object to another.

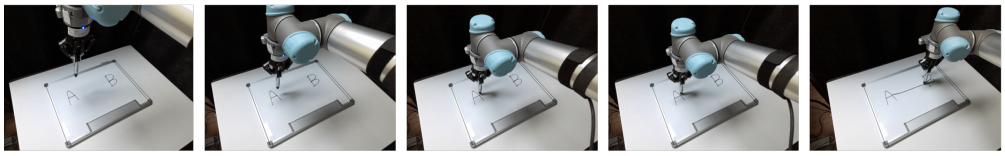


Figure 5.16: An example of a task with "draw a line from A to B".

3. **Open the Cabinet**: The robot is required to pick up the lid of the cabinet by adjusting the pose of the arm.

Figure 5.17: An example of a task with "open the cabinet".

4. **Play with the Car**: The toy car is a pull-back car, so the robot is required to grasp the toy car, move it backward, and open the gripper.

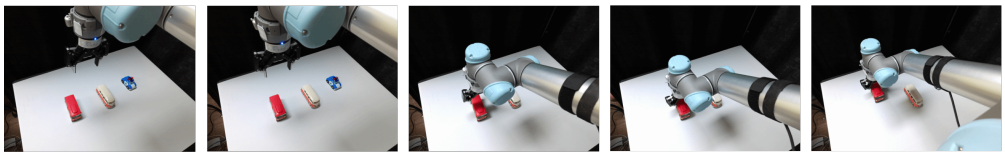


Figure 5.18: An example of a task with "play with the car".

5. **Erase the Whiteboard**: The robot is required to grasp the eraser and scrub it over the doodles.

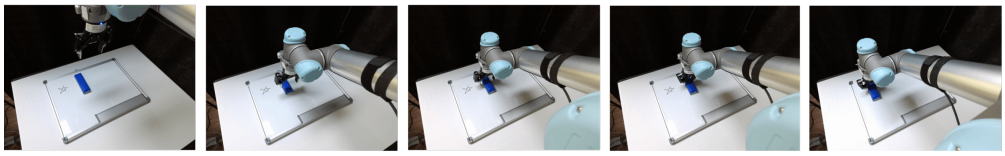


Figure 5.19: An example of a task with "erase the whiteboard".

6. **Close the Laptop**: The robot is expected to close the laptop.

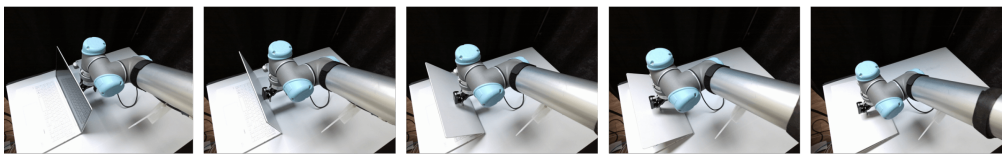


Figure 5.20: An example of a task with "close the laptop".

7. **Open the Trashcan**: The robot is required to approach the trashcan, precisely press the lid downward, and lift its arm to open it.

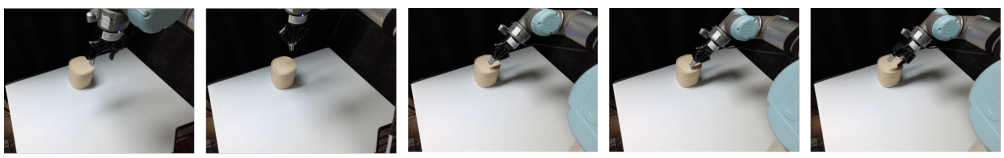


Figure 5.21: An example of a task with "open the trashcan".

8. **Stamp**: The robot is required to grasp the stamp, and precisely apply it at

the specified position (*e.g.*, next to `<obj>`, on `<obj>`, or between `<obj1>` and `<obj2>`).

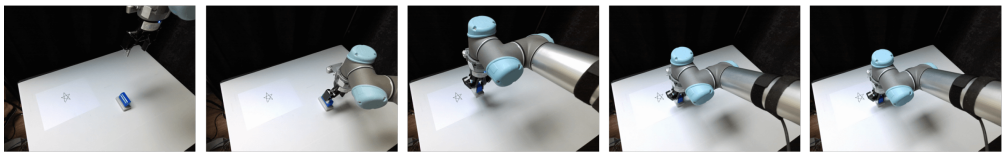


Figure 5.22: An example of a task with "stamp next to the star".

9. **Hide**: The robot should grasp the cup, flip it upside-down, and place it over the object, ensuring the cup is positioned correctly to hide the object. The cup may start upside-down from the beginning, and the object to hide could be a toy like Piglet, Pooh, or a small block.

Figure 5.23: An example of a task with "hide the Pooh with the green cup".

10. **Hang the Cup**: The robot is expected to grasp the cup with a handle and precisely hang it on the hanger.

Figure 5.24: An example of a task with "hang the cup".

5.4.3 Data

For each task — both *Common* and *Novel* — humans collect 10 episodes per task through interacting with LLMs. This results in 911 instances for Common

tasks and 1276 instances for Novel tasks. Leveraging stochastic trajectory diversification (STD), we augment 3 additional trajectories for each episode across all tasks. We get 9,841 instances for Common tasks and 11,578 for Novel tasks. In total, we use 23k in-domain data to train models, and all compared models are trained on the data unless stated otherwise.

5.4.4 Compared Methods

We compare CLIP-RT with several baselines, including ablated versions of our model to evaluate the impact of each component. **CLIP-RT** is our proposed model pretrained on the Open X-Embodiment dataset [Padalkar et al., 2023] and further fine-tuned using data collected via Section 5.3.3. **OpenVLA** [Kim et al., 2024] is an open-source, state-of-the-art Vision-Language-Action (VLA) model; it integrates the Llama2 language model [Touvron et al., 2023] with visual features from DINOv2 [Oquab et al., 2023] and SigLIP [Zhai et al., 2023], using an end-to-end approach where actions are treated as language tokens within the vision-language model [Karamcheti et al., 2024]. Note that we also finetune OpenVLA on the same in-domain data as CLIP-RT through low-rank adaptation [Hu et al., 2021]. **CLIP-RT-Action** is a variant of CLIP-RT where actions are mapped to specific text tokens similar to existing VLA models [Kim et al., 2024, Brohan et al., 2023, 2022, Padalkar et al., 2023]. We can identify the effect of actions represented in natural language. **CLIP-RT-Passive** is another ablated version of CLIP-RT without stochastic trajectory diversification (STD), relying solely on human-collected demonstrations to investigate the effect of STD. Finally, **CLIP-RT-Zero** is also an ablation model trained only on the Open X-Embodiment dataset [Padalkar et al., 2023], without accessing any in-domain data.

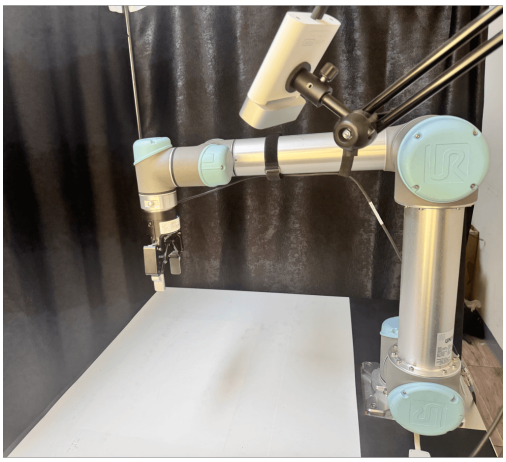


Figure 5.25: A robotic platform used in the experiments.

5.4.5 Robotic Platform

We perform experiments using a physical robot arm, 6-DoF Universal Robots (UR5) with a two-finger gripper. All episodes begin from a standardized home pose, as shown in Figure 5.25, and objects are placed within the white area, ensuring they are within the robot's reachable workspace. For visual input, we utilize the Azure Kinect DK, which provides an RGB image of the scene. The camera position remains fixed throughout all experiments, positioned to the left and slightly behind the robot arm to ensure consistent visual perspectives across tasks.

5.5 Results and Discussions

We evaluated the performance of CLIP-RT and compared it with OpenVLA and several ablated variants of CLIP-RT. The results are presented for Common Tasks (Figure 5.26 top) and Novel Tasks (Figure 5.26 bottom). The experimental results demonstrate that CLIP-RT outperforms OpenVLA in both Common and Novel tasks, with a particularly significant improvement in Novel

tasks, where CLIP-RT achieves an average success rate of 40% compared to OpenVLA's 23%. In Common tasks, CLIP-RT attains the highest average success rate of 54%, slightly surpassing OpenVLA's 51%, while ablated models exhibit lower performance.

5.5.1 Comparison with State-of-the-Art

Performance on Common vs. Novel Tasks. CLIP-RT demonstrates robust performance on both Common and Novel tasks, while OpenVLA exhibits a more pronounced decline in Novel tasks. Specifically, CLIP-RT maintains a high success rate in Common tasks and shows superior generalization capabilities in Novel tasks, outperforming OpenVLA by 17% on average in the latter. This suggests that CLIP-RT's architecture and training method take advantage of handling tasks previously unseen in the pre-training stage. We observe that the enhanced generalization of CLIP-RT can be attributed to its use of natural language supervision for learning actions. By representing actions in natural language, CLIP-RT leverages the abstract and high-level reasoning encoded in language-based representations. This facilitates better comprehension and execution of complex tasks, particularly those that are not encountered during pre-training. For further details, please see the ablation studies presented below.

Task-wise Performance Analysis. A detailed comparison of individual Common tasks reveals differences between CLIP-RT and OpenVLA. In relatively simpler tasks, such as *Point*, *Pull*, *Place*, *Pick*, and *Push*, both models perform comparably, with OpenVLA slightly outperforming CLIP-RT. This suggests that for tasks requiring straightforward action mappings, OpenVLA is effective. However, in more challenging tasks such as *Flip*, *Knock Over*, and *Slide*, CLIP-RT significantly outperforms OpenVLA. These tasks require a higher

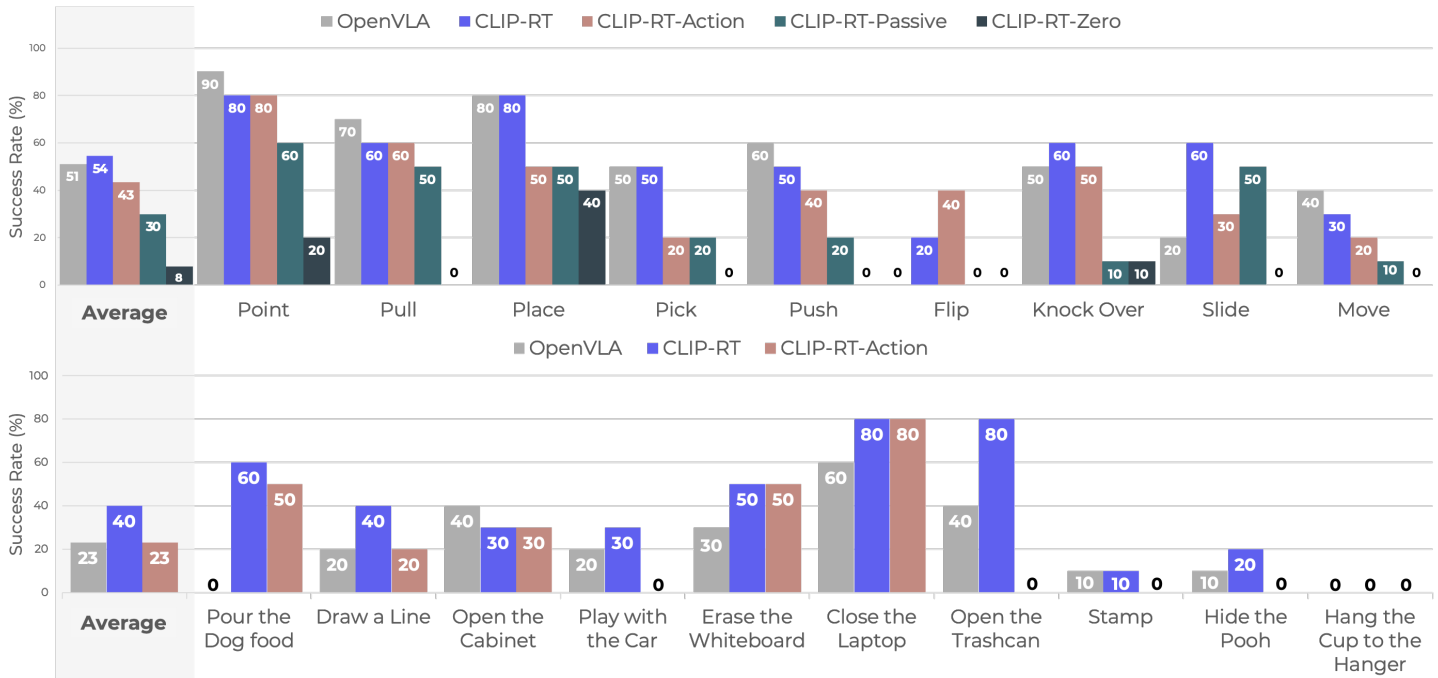


Figure 5.26: Success rates on nine Common Tasks (top) and ten Novel Tasks (bottom). We conduct experiments using all compared methods on Common tasks and three models (OpenVLA, CLIP-RT, and CLIP-RT-Action) on Novel Tasks. The success rate for each task is measured by averaging the results of ten trials. Average success rates of all tasks are shown on the left for both Common and Novel task sets. Tasks are arranged from left to right based on the average number of steps per episode. The task on the right indicates that it requires more steps on average compared with the task on the left.

level of reasoning, such as determining whether an object is upside down (Flip) or whether the cup is sufficiently tilted (Knock Over). We conjecture that CLIP-RT excels in these scenarios due to its discriminative approach, which involves selecting actions by directly matching the natural language representation of the desired action with the current context. This method allows CLIP-RT to leverage rich semantic information from both language and visual input, enabling nuanced decision-making in complex tasks. Conversely, OpenVLA's generative approach to action prediction may introduce errors in complex tasks, as generating precise actions becomes more challenging with increased task complexity.

5.5.2 Ablation Studies

We conducted an ablation study to assess the contributions of different components of CLIP-RT by comparing it with its variants: CLIP-RT-Action, CLIP-RT-Passive, and CLIP-RT-Zero.

Effect of Language-based Action Prediction. We compare CLIP-RT with CLIP-RT-Action to identify the effect of learning actions represented in natural language. As shown in Figure 5.26, CLIP-RT outperforms CLIP-RT-Action on both Common and Novel tasks by a significant margin. This indicates that representing actions in natural language, as in CLIP-RT, yields better results by leveraging the language priors in pre-trained vision-language models (*i.e.,* CLIP). Furthermore, it is worth noticing that the performance gap between the two models in Novel tasks (17%) is larger than that of Common tasks (11%). This result shows that learning language actions enhances the model's generalization capabilities.

Impact of Stochastic Trajectory Diversification. CLIP-RT-Passive, which

omits stochastic trajectory diversification (STD), struggles in most tasks, highlighting the critical role of STD in model performance. STD enables the model to learn from diverse possible trajectories, enhancing its robustness and generalization capabilities. By exploring deviations from optimal trajectories, the model becomes adept at recovering from nonoptimal states and adapting to unexpected variations in the robot's position. The absence of STD in CLIP-RT-Passive results in depending on a fewer number of training data, reducing its adaptability and effectiveness.

5.6 Conclusion

This paper presents CLIP-RT, enabling non-experts to teach robots new manipulation skills using human language. By learning actions through the user's natural language supervisions and stochastic trajectory diversification, CLIP-RT leverages pretrained vision-language models to effectively generalize to novel tasks. Experiments show that CLIP-RT outperforms the state-of-the-art OpenVLA by 17%, demonstrating the potential of natural language supervision for accessible and versatile robot learning. We believe that our work represents a valuable step towards making robot learning more accessible and scalable, allowing everyday users to teach robots in diverse environments.

Chapter 6

Concluding Remarks

6.1 Summary of Methods and Contributions

This dissertation discussed three key capabilities for embodied language agents to perform real-world robotic tasks. First, we presented generative self-training (GST) for visually-grounded communication to effectively leverage unlabeled Web images. GST first retrieves in-domain images through out-of-distribution detection and generates synthetic dialogs regarding the images via multimodal conditional text generation. GST then trains a dialog agent on the synthetic and the original visual dialog data. For robust training of the synthetic dialogs, we also propose perplexity-based data selection and multimodal consistency regularization. Evaluation on Visual Dialog v1.0 and v0.9 datasets shows that GST achieves new state-of-the-art results on both datasets. We further observe the robustness of GST against both visual and textual adversarial attacks. Finally, GST yields strong performance gains in the low-data regime.

Second, we explored the capability of reasoning about underspecified in-

structions for the robotic task. Inspired by pragmatics, where humans often convey their intentions by relying on context to achieve goals, we first presented a new task setup for interactive object grasping (IOG), which we call Pragmatic-IOG and the corresponding dataset. Then, we presented the modular approach for Pragmatic-IOG, the pragmatic object grasping (PROGrasp) system. PROGrasp performs Pragmatic-IOG by incorporating modules for visual grounding, question asking, object grasping, and most importantly, answer interpretation for pragmatic inference. Experimental results show that PROGrasp is effective in offline (i.e., target object discovery) and online (i.e., IOG with a physical robot arm) settings.

Third, we presented CLIP-based robotics transformer (CLIP-RT) to learn robotic skills directly from natural language supervision. CLIP-RT seamlessly extends the CLIP model [Radford et al., 2021] and learns to predict actions represented in language via contrastive imitation learning. Furthermore, we proposed a data collection framework that collects robot demonstrations based on natural language supervision (e.g., "move the arm forward") and further augments these demonstrations. We first train CLIP-RT on large-scale robotic data and then enable it to learn desired skills using data collected from our framework. CLIP-RT shows strong capabilities in acquiring novel manipulation skills, outperforming the state-of-the-art model, OpenVLA (7B parameters), by 17% in average success rates, while using 7x fewer parameters (1B).

This dissertation bridges the gap between interaction, reasoning, and control in embodied language agents. While existing research at the intersection of natural language processing (NLP) and robotics (see Chapter 2) has advanced individual capabilities in communication, reasoning, and control, these dimensions are often addressed in isolation. This dissertation repositions interaction not as a standalone capability but as a mechanism to facilitate deeper reason-

ing and more effective action learning, addressing the complexities of real-world human-robot collaboration.

The dissertation introduces a novel perspective by framing the challenge through the dual lens of "learning for interaction" and "interaction for learning." "Learning for interaction" focuses on how robots can acquire the desired perceptual and reasoning capabilities to engage in effective and robust communication with humans, as in Chapter 3, which focuses on visually-grounded communication, and Chapter 4, which addresses reasoning about underspecified instructions. Conversely, "interaction for learning" explores how robots leverage interaction — in the form of natural language instructions, feedback, or demonstrations — to acquire new skills and improve their ability to perform real-world robotic tasks, as demonstrated in Chapter 5. By combining these perspectives, this dissertation lays the groundwork for a unified framework where interaction and learning mutually reinforce each other, enabling robots to adapt seamlessly to complex and dynamic real-world tasks.

6.2 Suggestions for Future Research

6.2.1 Lifelong Learning

While this dissertation explored the *communication, reasoning, and learning* capabilities of embodied language agents, these capabilities were addressed independently rather than interwoven into a cohesive framework. For example, while reasoning about underspecified instructions (Chapter 4) interplays between visually-grounded communication and pragmatic reasoning, it did not directly feed into the learning process to improve task performance over time. Furthermore, learning robotic skills from natural language (Chapter 5) is at the intersection of communication and action learning, but it did not account for the interpretation of ambiguous or high-level commands (*e.g.,* "move forward a

bit"). In other words, this process lacks the integration of contextual or pragmatic reasoning, limiting the robot's ability to infer intent or adapt instructions based on situational nuances. As a result, the system remains reliant on precise, low-level commands (*e.g.,* "move forward 10cm") that fail to capture the complexity of real-world interactions.

Inspired by humans who continuously refine their understanding of the world through experience, a promising direction for future research is the development of lifelong learning systems that seamlessly integrate interaction and learning. This would involve advances in continuous learning methods, such as modular or memory-based approaches [Guo et al., 2020, Suhr and Artzi, 2023], to maintain a balance between stability (retaining past knowledge) and adaptability (acquiring new knowledge without catastrophic forgetting [Kirkpatrick et al., 2017, Lee et al., 2017]). By achieving this integration, embodied language agents could dynamically evolve their interaction and learning capabilities to address a wider array of tasks in diverse and changing environments, such as adapting to unseen objects in robotic manipulation tasks or learning new forms of collaborative behaviors.

6.2.2 Long-Horizon Task Execution

The robotic tasks addressed in this dissertation are relatively *short-horizon* compared with the complexity and duration of everyday tasks, such as folding laundry [Black et al., 2024] or delivering objects in indoor environments [LABS, 2023]. While CLIP-RT in Chapter 5 successfully demonstrates diverse manipulation skills — such as opening the trash can and closing the laptop — extending these capabilities to long-horizon tasks requires novel approaches that can handle increased task complexity.

One promising strategy for long-horizon task execution involves developing

a high-level task planner [Huang et al., 2022a, Song et al., 2023, Shin et al., 2024] that decomposes complex tasks into sequences of primitive skills. For example, a task planner could break down "set the dinner table" into subtasks like "retrieve plates," "place utensils," and "arrange napkins." Integrating such planners with CLIP-RT's manipulation skills could enable embodied language agents to execute structured, multi-step tasks. However, such a *top-down approach* might struggle with unstructured tasks, such as cleaning a cluttered room, where objects and goals are not predefined, requiring adaptability and on-the-fly decision-making.

A complementary strategy is *incremental learning* [Wu et al., 2019], where agents progressively build a task hierarchy by combining previously learned skills. For instance, an agent might learn how to "fold a towel" by first mastering simpler actions like "grab towel edges" and "apply folding motion." This *bottom-up approach* leverages learned behaviors as building blocks for more complex tasks, enabling adaptation to scenarios with poorly specified goals or dynamic conditions, such as sorting laundry with unknown item categories. However, this strategy may face challenges in maintaining global task coherence, such as ensuring all folded towels are eventually placed in the laundry basket. This highlights the need for a structured framework to guide overall task execution.

A hybrid strategy that combines the strengths of top-down planning and bottom-up incremental learning offers a robust solution for long-horizon task execution. By integrating a high-level task planner to provide structure and global guidance while allowing incremental learning to refine and expand skills locally, agents can dynamically adapt to diverse and evolving scenarios. In other words, the task planner could outline a broad sequence of objectives, while incremental learning enables fine-tuning and skill discovery within each objective. This hybrid approach balances the stability and foresight of top-down methods

with the adaptability of bottom-up strategies, creating a robust and scalable framework for addressing complex, long-horizon robotic tasks.

Bibliography

English wikipedia. `https://www.wikipedia.org`, 2019.

Pieter Abbeel, Adam Coates, and Andrew Y Ng. Autonomous helicopter aerobatics through apprenticeship learning. *The International Journal of Robotics Research*, 29(13):1608–1639, 2010.

Shubham Agarwal, Trung Bui, Joon-Young Lee, Ioannis Konstas, and Verena Rieser. History for visual dialog: Do we really need it? In *Annual Meeting of the Association for Computational Linguistics*, 2020.

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

Huda Alamri, Vincent Cartillier, Abhishek Das, Jue Wang, Anoop Cherian, Irfan Essa, Dhruv Batra, Tim K Marks, Chiori Hori, Peter Anderson, et al. Audio visual scene-aware dialog. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for im-

age captioning and visual question answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018a.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3674–3683, 2018b.

Peter Anderson, Ayush Shrivastava, Joanne Truong, Arjun Majumdar, Devi Parikh, Dhruv Batra, and Stefan Lee. Sim-to-real transfer for vision-and-language navigation. In *Conference on Robot Learning*, pages 671–681. PMLR, 2021.

Jacob Andreas and Dan Klein. Reasoning about pragmatics with neural listeners and speakers. In *2016 Conference on Empirical Methods in Natural Language Processing*, pages 1173–1182. Association for Computational Linguistics, 2016.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *International Conference on Computer Vision*, 2015.

Yoav Artzi and Luke Zettlemoyer. Weakly supervised learning of semantic parsers for mapping instructions to actions. *Transactions of the association for computational linguistics*, 1:49–62, 2013.

Suneel Belkhale, Tianli Ding, Ted Xiao, Pierre Sermanet, Quon Vuong, Jonathan Tompson, Yevgen Chebotar, Debidatta Dwibedi, and Dorsa Sadigh. Rt-h: Action hierarchies using language. *arXiv preprint arXiv:2403.01823*, 2024.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, 2019.

Aude G Billard, Sylvain Calinon, and Florent Guenter. Discriminative and adaptive imitation in uni-manual and bi-manual tasks. *Robotics and Autonomous Systems*, 54(5):370–384, 2006.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

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. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Joyce Y Chai, Lanbo She, Rui Fang, Spencer Ottarson, Cody Littley, Changsong Liu, and Kenneth Hanson. Collaborative effort towards common ground

in situated human-robot dialogue. In *2014 ACM/IEEE international conference on Human-robot interaction*, pages 33–40, 2014.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

Cheng Chen, Yudong Zhu, Zhenshan Tan, Qingrong Cheng, Xin Jiang, Qun Liu, and Xiaodong Gu. Utc: A unified transformer with inter-task contrastive learning for visual dialog. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

David Chen and Raymond Mooney. Learning to interpret natural language navigation instructions from observations. In *AAAI Conference on Artificial Intelligence*, pages 859–865, 2011.

Feilong Chen, Fandong Meng, Jiaming Xu, Peng Li, Bo Xu, and Jie Zhou. Dmrm: A dual-channel multi-hop reasoning model for visual dialog. In *AAAI Conference on Artificial Intelligence*, 2020a.

Feilong Chen, Fandong Meng, Xiuyi Chen, Peng Li, and Jie Zhou. Multimodal incremental transformer with visual grounding for visual dialogue generation. In *Annual Meeting of the Association for Computational Linguistics*, 2021.

Haonan Chen, Hao Tan, Alan Kuntz, Mohit Bansal, and Ron Alterovitz. Enabling robots to understand incomplete natural language instructions using commonsense reasoning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1963–1969. IEEE, 2020b.

Ching-An Cheng, Andrey Kolobov, Dipendra Misra, Allen Nie, and Adith Swaminathan. Llf-bench: Benchmark for interactive learning from language feedback. *arXiv preprint arXiv:2312.06853*, 2023.

Kevin Clark and Christopher D Manning. Deep reinforcement learning for mention-ranking coreference models. In *Annual Meeting of the Association for Computational Linguistics*, 2016.

Reuben Cohn-Gordon, Noah Goodman, and Christopher Potts. Pragmatically informative image captioning with character-level inference. In *2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 439–443. Association for Computational Linguistics, 2018.

David Coleman, Ioan Sucan, Sachin Chitta, and Nikolaus Correll. Reducing the barrier to entry of complex robotic software: a moveit! case study. *arXiv preprint arXiv:1404.3785*, 2014.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*, 2017.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

Harm De Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron Courville. Guesswhat?! visual object discovery through multi-modal dialogue. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. In *arXiv preprint arXiv:2303.03378*, 2023.

Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Ves Stoyanov, and Alexis Conneau. Self-training improves pre-training for natural language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2021.

Yuqing Du, Ksenia Konyushkova, Misha Denil, Akhil Raju, Jessica Landon,

Felix Hill, Nando de Freitas, and Serkan Cabi. Vision-language models as success detectors. *arXiv preprint arXiv:2303.07280*, 2023.

Juraj Dzifcak, Matthias Scheutz, Chitta Baral, and Paul Schermerhorn. What to do and how to do it: Translating natural language directives into temporal and dynamic logic representation for goal management and action execution. In *2009 IEEE International Conference on Robotics and Automation*, pages 4163–4168. IEEE, 2009.

Cem Eteke, Doğancan Kebüde, and Barış Akgün. Reward learning from very few demonstrations. *IEEE Transactions on Robotics*, 37(3):893–904, 2020.

Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. In *Annual Meeting of the Association for Computational Linguistics*, 2018.

Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data filtering networks. *arXiv preprint arXiv:2309.17425*, 2023.

Michael C Frank and Noah D Goodman. Predicting pragmatic reasoning in language games. *Science*, 336(6084):998–998, 2012.

Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 31, 2018.

Daniel Fried, Nicholas Tomlin, Jennifer Hu, Roma Patel, and Aida Nematzadeh. Pragmatics in grounded language learning: Phenomena, tasks, and modeling approaches. *arXiv preprint arXiv:2211.08371*, 2022.

Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*, 2024.

Zhe Gan, Yu Cheng, Ahmed EI Kholy, Linjie Li, Jingjing Liu, and Jianfeng Gao. Multi-step reasoning via recurrent dual attention for visual dialog. In *Annual Meeting of the Association for Computational Linguistics*, 2019.

Jensen Gao, Bidipta Sarkar, Fei Xia, Ted Xiao, Jiajun Wu, Brian Ichter, Anirudha Majumdar, and Dorsa Sadigh. Physically grounded vision-language models for robotic manipulation. *arXiv preprint arXiv:2309.02561*, 2023.

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

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Noah D Goodman and Michael C Frank. Pragmatic language interpretation as probabilistic inference. *Trends in cognitive sciences*, 20(11):818–829, 2016.

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

Herbert P Grice. Logic and conversation. In *Speech acts*, pages 41–58. Brill, 1975.

Dalu Guo, Chang Xu, and Dacheng Tao. Image-question-answer synergistic network for visual dialog. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Yunhui Guo, Mingrui Liu, Tianbao Yang, and Tajana Rosing. Improved schemes for episodic memory-based lifelong learning. *Advances in Neural Information Processing Systems*, 33:1023–1035, 2020.

Muzhi Han, Yifeng Zhu, Song-Chun Zhu, Ying Nian Wu, and Yuke Zhu. Interpret: Interactive predicate learning from language feedback for generalizable task planning. *arXiv preprint arXiv:2405.19758*, 2024.

Jun Hatori, Yuta Kikuchi, Sosuke Kobayashi, Kuniyuki Takahashi, Yuta Tsuboi, Yuya Unno, Wilson Ko, and Jethro Tan. Interactively picking real-world objects with unconstrained spoken language instructions. In *2018 IEEE International Conference on Robotics and Automation*, pages 3774–3781. IEEE, 2018.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. Revisiting self-training for neural sequence generation. In *International Conference on Learning Representations*, 2020.

Xingwei He, Zhenghao Lin, Yeyun Gong, Alex Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, Weizhu Chen, et al. Annollm: Making large language models to be better crowdsourced annotators. In *2024 Conference of North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, 2024.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. Learning to write with cooperative discriminators. In *Annual Meeting of the Association for Computational Linguistics*, 2018.

Yordan Hristov and Subramanian Ramamoorthy. Learning from demonstration with weakly supervised disentanglement. In *International Conference on Learning Representations*, 2021.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Yingdong Hu, Fanqi Lin, Tong Zhang, Li Yi, and Yang Gao. Look before you leap: Unveiling the power of gpt-4v in robotic vision-language planning. *arXiv preprint arXiv:2311.17842*, 2023.

Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International conference on machine learning*, pages 9118–9147. PMLR, 2022a.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner monologue: Embodied reasoning through planning with language models. *arXiv preprint arXiv:2207.05608*, 2022b.

Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. *arXiv preprint arXiv:2409.01652*, 2024.

Xinting Huang, Jianzhong Qi, Yu Sun, and Rui Zhang. Mala: Cross-domain dialogue generation with action learning. In *AAAI Conference on Artificial Intelligence*, 2020.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas

Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL `https://doi.org/10.5281/zenodo.5143773`.

Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pages 991–1002. PMLR, 2022.

Xiaoze Jiang, Siyi Du, Zengchang Qin, Yajing Sun, and Jing Yu. Kbgn: Knowledge-bridge graph network for adaptive vision-text reasoning in visual dialogue. In *28th ACM International Conference on Multimedia*, pages 1265–1273, 2020a.

Xiaoze Jiang, Jing Yu, Yajing Sun, Zengchang Qin, Zihao Zhu, Yue Hu, and Qi Wu. Dam: Deliberation, abandon and memory networks for generating detailed and non-repetitive responses in visual dialogue. In *International Joint Conference on Artificial Intelligence*, 2020b.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *AAAI Conference on Artificial Intelligence*, 2020.

Gi-Cheon Kang, Jaeseo Lim, and Byoung-Tak Zhang. Dual attention networks for visual reference resolution in visual dialog. In *Conference on Empirical Methods in Natural Language Processing*, 2019.

Gi-Cheon Kang, Junseok Park, Hwaran Lee, Byoung-Tak Zhang, and Jin-Hwa Kim. Reasoning visual dialog with sparse graph learning and knowledge transfer. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

Gi-Cheon Kang, Sungdong Kim, Jin-Hwa Kim, Donghyun Kwak, and Byoung-Tak Zhang. The dialog must go on: Improving visual dialog via generative self-training. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6746–6756, 2023.

Gi-Cheon Kang, Junghyun Kim, Jaein Kim, and Byoung-Tak Zhang. Prograsp: Pragmatic human-robot communication for object grasping. In *2024 IEEE International Conference on Robotics and Automation*, pages 3304–3310. IEEE, 2024a.

Gi-Cheon Kang, Junghyun Kim, Kyuhwan Shim, Jun Ki Lee, and Byoung-Tak Zhang. Clip-rt: Learning language-conditioned robotic policies from natural language supervision. *arXiv preprint arXiv:2411.00508*, 2024b.

Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic vlms: Investigating the design space of visually-conditioned language models. *arXiv preprint arXiv:2402.07865*, 2024.

Eun-Sol Kim, Woo Young Kang, Kyoung-Woon On, Yu-Jung Heo, and Byoung-Tak Zhang. Hypergraph attention networks for multimodal learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14581–14590, 2020.

Jin-Hwa Kim, Sang-Woo Lee, Donghyun Kwak, Min-Oh Heo, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Multimodal residual learning for visual qa. In *Advances in neural information processing systems*, volume 29, 2016.

Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha,

and Byoung-Tak Zhang. Hadamard product for low-rank bilinear pooling. In *International Conference on Learning Representations*, 2017.

Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Jin-Hwa Kim, Nikita Kitaev, Xinlei Chen, Marcus Rohrbach, Byoung-Tak Zhang, Yuandong Tian, Dhruv Batra, and Devi Parikh. Codraw: Collaborative drawing as a testbed for grounded goal-driven communication. In *Annual Meeting of the Association for Computational Linguistics*, 2019.

Junghyun Kim, Gi-Cheon Kang, Jaein Kim, Suyeon Shin, and Byoung-Tak Zhang. Gvcci: Lifelong learning of visual grounding for language-guided robotic manipulation. *arXiv preprint arXiv:2307.05963*, 2023.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2014.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13): 3521–3526, 2017.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. Opennmt: Open-source toolkit for neural machine translation. In *Annual Meeting of the Association for Computational Linguistics*, 2017.

Thomas Kollar, Stefanie Tellex, Deb Roy, and Nicholas Roy. Toward understanding natural language directions. In *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 259–266. IEEE, 2010.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Satwik Kottur, José MF Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. Visual coreference resolution in visual dialog using neural module networks. In *European Conference on Computer Vision*, 2018.

Jacob Krantz and Stefan Lee. Sim-2-sim transfer for vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, pages 588–603. Springer, 2022.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 104–120. Springer, 2020.

Jacob Krantz, Aaron Gokaslan, Dhruv Batra, Stefan Lee, and Oleksandr Maksymets. Waypoint models for instruction-guided navigation in continuous environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15162–15171, 2021.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua

Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. In *International Conference on Computer Vision*, 2017.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. *arXiv preprint arXiv:2010.07954*, 2020.

Teyun Kwon, Norman Di Palo, and Edward Johns. Language models as zero-shot trajectory generators. *arXiv preprint arXiv:2310.11604*, 2023.

NAVER LABS. Ambidex. 2023. https://www.naverlabs.com/ambidex.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521 (7553):436–444, 2015.

Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop on challenges in representation learning*, 2013.

Sang-Woo Lee, Jin-Hwa Kim, Jaehyun Jun, Jung-Woo Ha, and Byoung-Tak Zhang. Overcoming catastrophic forgetting by incremental moment matching. *Advances in neural information processing systems*, 30, 2017.

Sang-Woo Lee, Yu-Jung Heo, and Byoung-Tak Zhang. Answerer in questioner's mind: Information theoretic approach to goal-oriented visual dialog. *Advances in neural information processing systems*, 31, 2018.

Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. In *Conference on Empirical Methods in Natural Language Processing*, 2016.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. Adversarial learning for neural dialogue generation. In *Conference on Empirical Methods in Natural Language Processing*, 2017a.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. Bert-attack: Adversarial attack against bert using bert. In *Conference on Empirical Methods in Natural Language Processing*, 2020.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. Dailydialog: A manually labelled multi-turn dialogue dataset. In *IJCNLP*, 2017b.

Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *2023 IEEE International Conference on Robotics and Automation*, pages 9493–9500. IEEE, 2023.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014.

Huihan Liu, Alice Chen, Yuke Zhu, Adith Swaminathan, Andrey Kolobov, and Ching-An Cheng. Interactive robot learning from verbal correction. *arXiv preprint arXiv:2310.17555*, 2023.

Luís Seabra Lopes and António Teixeira. Human-robot interaction through spoken language dialogue. In *Proceedings. 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)(Cat. No. 00CH37113)*, volume 1, pages 528–534. IEEE, 2000.

Jiasen Lu, Anitha Kannan, Jianwei Yang, Devi Parikh, and Dhruv Batra. Best of both worlds: Transferring knowledge from discriminative learning to a

generative visual dialog model. In *Advances in Neural Information Processing Systems*, 2017.

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

Corey Lynch and Pierre Sermanet. Language conditioned imitation learning over unstructured data. *arXiv preprint arXiv:2005.07648*, 2020.

Guilherme J Maeda, Gerhard Neumann, Marco Ewerton, Rudolf Lioutikov, Oliver Kroemer, and Jan Peters. Probabilistic movement primitives for coordination of multiple human–robot collaborative tasks. *Autonomous Robots*, 41:593–612, 2017.

Zhao Mandi, Shreeya Jain, and Shuran Song. Roco: Dialectic multi-robot collaboration with large language models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 286–299. IEEE, 2024.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

Shikib Mehri and Maxine Eskenazi. Unsupervised evaluation of interactive dialog with dialogpt. In *SIGDIAL*, 2020.

Linghui Meng, Muning Wen, Chenyang Le, Xiyun Li, Dengpeng Xing, Weinan Zhang, Ying Wen, Haifeng Zhang, Jun Wang, Yaodong Yang, et al. Offline pre-trained multi-agent decision transformer. *Machine Intelligence Research*, 20(2):233–248, 2023.

Yuchen Mo, Hanbo Zhang, and Tao Kong. Towards open-world interactive disambiguation for robotic grasping. In *CoRL Workshop on Learning, Perception, and Abstraction for Long-Horizon Planning*, 2022.

Will Monroe, Robert XD Hawkins, Noah D Goodman, and Christopher Potts. Colors in context: A pragmatic neural model for grounded language understanding. *Transactions of the Association for Computational Linguistics*, 5: 325–338, 2017.

Nikola Mrkšić, Diarmuid O Séaghdha, Blaise Thomson, Milica Gašić, Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. Counter-fitting word vectors to linguistic constraints. In *North American Chapter of the Association for Computational Linguistics*, 2016.

Vishvak Murahari, Dhruv Batra, Devi Parikh, and Abhishek Das. Large-scale pretraining for visual dialog: A simple state-of-the-art baseline. In *European Conference on Computer Vision*, 2020.

Thao Nguyen, Nakul Gopalan, Roma Patel, Matthew Corsaro, Ellie Pavlick, and Stefanie Tellex. Robot Object Retrieval with Contextual Natural Language Queries. In *Proceedings of Robotics: Science and Systems*, 2020a.

Van-Quang Nguyen, Masanori Suganuma, and Takayuki Okatani. Efficient attention mechanism for visual dialog that can handle all the interactions between multiple inputs. In *European Conference on Computer Vision*, 2020b.

Nils J Nilsson et al. *Shakey the robot*, volume 323. Sri International Menlo Park, California, 1984.

Yulei Niu, Hanwang Zhang, Manli Zhang, Jianhong Zhang, Zhiwu Lu, and

Ji-Rong Wen. Recursive visual attention in visual dialog. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023a.

OpenAI. Gpt-4v(ision) system card. 2023b. `https://cdn.openai.com/papers/GPTV_System_Card.pdf`.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

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. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Abhishek Padalkar, Acorn Pooley, Ajinkya Jain, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anikait Singh, Anthony Brohan, et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023.

Rohan Paul, Jacob Arkin, Nicholas Roy, and Thomas M. Howard. Grounding abstract spatial concepts for language interaction with robots. In *Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4929–4933, 2017.

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model

for abstractive summarization. In *International Conference on Learning Representations*, 2018.

Steven T Piantadosi, Harry Tily, and Edward Gibson. The communicative function of ambiguity in language. *Cognition*, 122(3):280–291, 2012.

Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1, 1988.

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.

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

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Annual Meeting of the Association for Computational Linguistics*, 2019.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. Leveraging pre-trained checkpoints for sequence generation tasks. In *Transactions of the Association for Computational Linguistics*, 2020.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Pearson, 2016.

Abdelrhman Saleh, Natasha Jaques, Asma Ghandeharioun, Judy Shen, and Rosalind Picard. Hierarchical reinforcement learning for open-domain dialog. In *AAAI Conference on Artificial Intelligence*, 2020.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. In *IEEE Transactions on Neural Networks*. IEEE, 2008.

Ruwen Schnabel, Roland Wahl, and Reinhard Klein. Efficient ransac for point-cloud shape detection. In *Computer graphics forum*, volume 26, pages 214–226. Wiley Online Library, 2007.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jit-

sev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL `https://openreview.net/forum?id=M3Y74vmsMcY`.

Idan Schwartz, Seunghak Yu, Tamir Hazan, and Alexander G Schwing. Factor graph attention. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

John R Searle and John Rogers Searle. *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge university press, 1969.

Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. What makes a good conversation? how controllable attributes affect human judgments. In *North American Chapter of the Association for Computational Linguistics*, 2019.

Mingyo Seo, Steve Han, Kyutae Sim, Seung Hyeon Bang, Carlos Gonzalez, Luis Sentis, and Yuke Zhu. Deep imitation learning for humanoid loco-manipulation through human teleoperation. In *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*, pages 1–8. IEEE, 2023.

Paul Hongsuck Seo, Andreas Lehrmann, Bohyung Han, and Leonid Sigal. Visual reference resolution using attention memory for visual dialog. In *Advances in Neural Information Processing Systems*, 2017.

Iulian Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI Conference on Artificial Intelligence*, 2017.

Siamak Shakeri, Cicero Nogueira dos Santos, Henry Zhu, Patrick Ng, Feng Nan, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. End-to-end synthetic data generation for domain adaptation of question answering systems. In *Conference on Empirical Methods in Natural Language Processing*, 2020.

Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. In *Annual Meeting of the Association for Computational Linguistics*, 2015.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Annual Meeting of the Association for Computational Linguistics*, 2018.

Sheng Shen, Daniel Fried, Jacob Andreas, and Dan Klein. Pragmatically informative text generation. In *2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4060–4067. Association for Computational Linguistics, 2019.

Suyeon Shin, Junghyun Kim, Gi-Cheon Kang, Byoung-Tak Zhang, et al. Socratic planner: Inquiry-based zero-shot planning for embodied instruction following. *arXiv preprint arXiv:2404.15190*, 2024.

Mohit Shridhar and David Hsu. Grounding spatio-semantic referring expressions for human-robot interaction. *arXiv preprint arXiv:1707.05720*, 2017.

Mohit Shridhar and David Hsu. Interactive visual grounding of referring expressions for human-robot interaction. *arXiv preprint arXiv:1806.03831*, 2018.

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 *IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749, 2020.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on Robot Learning*, pages 894–906. PMLR, 2022.

Nathaniel J Smith, Noah Goodman, and Michael Frank. Learning and using language via recursive pragmatic reasoning about other agents. *Advances in neural information processing systems*, 26, 2013.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems*, 2020.

Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M Sadler, Wei-Lun Chao, and Yu Su. Llm-planner: Few-shot grounded planning for embodied agents with large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2998–3009, 2023.

Mark Steedman. Surface structure and interpretation, 1996.

Simon Stepputtis, Joseph Campbell, Mariano Phielipp, Stefan Lee, Chitta Baral, and Heni Ben Amor. Language-conditioned imitation learning for robot manipulation tasks. *Advances in Neural Information Processing Systems*, 33:13139–13150, 2020.

Yu Su. Language agents: a critical evolutionary step of artificial intelligence. *yusu.substack.com*, Sep 2023. URL `https://yusu.substack.com/p/language-agents`.

Alane Suhr and Yoav Artzi. Continual learning for instruction following from realtime feedback. *Advances in Neural Information Processing Systems*, 2023.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

Hao Tan and Mohit Bansal. LXMERT: Learning cross-modality encoder representations from transformers. In *2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111. Association for Computational Linguistics, 2019.

Stefanie Tellex, Nakul Gopalan, Hadas Kress-Gazit, and Cynthia Matuszek. Robots that use language. *Annual Review of Control, Robotics, and Autonomous Systems*, 3(1):25–55, 2020.

Jesse Thomason, Aishwarya Padmakumar, Jivko Sinapov, Nick Walker, Yuqian Jiang, Harel Yedidsion, Justin Hart, Peter Stone, and Raymond J Mooney. Improving grounded natural language understanding through human-robot dialog. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6934–6941. IEEE, 2019.

Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. Vision-and-dialog navigation. In *Conference on Robot Learning*, 2020.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi,

Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

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*, 2017.

Sagar Gubbi Venkatesh, Anirban Biswas, Raviteja Upadrashta, Vikram Srinivasan, Partha Talukdar, and Bharadwaj Amrutur. Spatial reasoning from natural language instructions for robot manipulation. In *2021 IEEE International Conference on Robotics and Automation*, pages 11196–11202. IEEE, 2021.

Kai Wang, Junfeng Tian, Rui Wang, Xiaojun Quan, and Jianxing Yu. Multi-domain dialogue acts and response co-generation. In *Annual Meeting of the Association for Computational Linguistics*, 2020a.

Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022.

Yue Wang, Shafiq Joty, Michael R Lyu, Irwin King, Caiming Xiong, and Steven CH Hoi. Vd-bert: A unified vision and dialog transformer with bert. In *Conference on Empirical Methods in Natural Language Processing*, 2020b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning

in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

David Whitney, Eric Rosen, James MacGlashan, Lawson LS Wong, and Stefanie Tellex. Reducing errors in object-fetching interactions through social feedback. In *2017 IEEE International Conference on Robotics and Automation*, pages 1006–1013. IEEE, 2017.

Terry Winograd. Understanding natural language. *Cognitive psychology*, 3(1): 1–191, 1972.

Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. Tidybot: Personalized robot assistance with large language models. *Autonomous Robots*, 47(8):1087–1102, 2023.

Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton Van Den Hengel. Are you talking to me? reasoned visual dialog generation through adversarial learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 374–382, 2019.

Ted Xiao, Harris Chan, Pierre Sermanet, Ayzaan Wahid, Anthony Brohan, Karol Hausman, Sergey Levine, and Jonathan Tompson. Robotic skill acquistion via instruction augmentation with vision-language models. In *Proceedings of Robotics: Science and Systems*, 2023.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems*, 2020a.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020b.

Taozheng Yang, Ya Jing, Hongtao Wu, Jiafeng Xu, Kuankuan Sima, Guangzeng Chen, Qie Sima, and Tao Kong. Moma-force: Visual-force imitation for real-world mobile manipulation. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6847–6852. IEEE, 2023.

Yang Yang, Xibai Lou, and Changhyun Choi. Interactive robotic grasping with attribute-guided disambiguation. In *2022 International Conference on Robotics and Automation*, pages 8914–8920. IEEE, 2022.

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

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 *Advances in Neural Information Processing Systems*, volume 36, 2024.

Kang Min Yoo, Jaegeun Han, Sookyo In, Heewon Jeon, Jisu Jeong, Jaewook Kang, Hyunwook Kim, Kyung-Min Kim, Munhyong Kim, Sungju Kim, et al. Hyperclova x technical report. *arXiv preprint arXiv:2404.01954*, 2024.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L

Berg. Modeling context in referring expressions. In *European Conference on Computer Vision*, pages 69–85. Springer, 2016.

Wenhao Yu, Nimrod Gileadi, Chuyuan Fu, Sean Kirmani, Kuang-Huei Lee, Montse Gonzalez Arenas, Hao-Tien Lewis Chiang, Tom Erez, Leonard Hasenclever, Jan Humplik, et al. Language to rewards for robotic skill synthesis. *arXiv preprint arXiv:2306.08647*, 2023.

Zhou Yu, Jun Yu, Chenchao Xiang, Jianping Fan, and Dacheng Tao. Beyond bilinear: Generalized multimodal factorized high-order pooling for visual question answering. *IEEE transactions on neural networks and learning systems*, 29(12):5947–5959, 2018.

Andy Zeng, Maria Attarian, Brian Ichter, Krzysztof Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*, 2022.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023.

Hanbo Zhang, Yunfan Lu, Cunjun Yu, David Hsu, Xuguang La, and Nanning Zheng. Invigorate: Interactive visual grounding and grasping in clutter. *arXiv preprint arXiv:2108.11092*, 2021.

Tianhao Zhang, Zoe McCarthy, Owen Jow, Dennis Lee, Xi Chen, Ken Goldberg, and Pieter Abbeel. Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In *2018 IEEE international conference on robotics and automation*, pages 5628–5635. IEEE, 2018.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. Dialogpt: Large-scale generative pre-training for conversational response generation. In *Annual Meeting of the Association for Computational Linguistics*, 2020.

Zilong Zheng, Wenguan Wang, Siyuan Qi, and Song-Chun Zhu. Reasoning visual dialogs with structural and partial observations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *International Conference on Computer Vision*, 2015.

Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. Rethinking pre-training and self-training. In *Advances in Neural Information Processing Systems*, 2020.

요약

인간의 자연어 지시를 이해하여 다양한 실세계 작업을 수행할 수 있는 지능형 에이전트의 개발은 인공지능의 오랜 목표이다. 기계 학습과 자연어 처리 기술의 발달로 이러한 에이전트의 역량이 크게 향상되어 언어를 의사소통, 추론 및 학습의 수단으로 사용할 수 있게 되었다. 해당 역량을 보유한 에이전트를 흔히 언어 에이전트라고 한다. 이 논문은 물리적 환경에서 인간의 명령을 해석하고 로봇 작업을 실행하는 체화된 언어 에이전트를 탐구한다. 이전 연구들 [Kang et al., 2023, 2024a,b]을 바탕으로 체화된 언어 에이전트의 세 가지 과제가 논의되며, 각 과제는 언어 에이전트의 핵심 역량을 기반으로 하며 다음과 같다: 1) 시각 기반 의사소통, 2) 구체화되지 않은 명령에 대한 추론, 3) 자연어를 통한 로봇 기술 학습.

시각 기반 의사소통은 시지각에 대해 인간과 효과적으로 의사소통할 수 있도록 해주므로 체화된 언어 에이전트의 핵심 역량에 해당한다. 우리는 체화된 언어 에이전트가 시각 기반 의사소통의 강건성과 일반화 기능을 향상시킬 수 있는 방법을 탐구한다. 이를 위해, 생성형 자가 학습이라는 준지도 학습 접근 방식을 소개한다. 생성형 자가 학습의 핵심 사상은 웹에 있는 방대한 양의 라벨이 없는 사진 데이터를 기반으로 인공 시각 대화 데이터를 생성하고 이를 학습에 사용하는 것이다. 생성형 자가 학습은 적대적 강건성과 학습 시 관측하지 않은 데이터에 대한 일반화 성능을 향상시켰다.

자연어는 본질적으로 모호하며 그 의미가 상황에 의존적인 특성을 가지고 있다. 이에, 우리는 체화된 언어 에이전트가 “내 장치의 배터리는 부족하다”와 같은 구체화되지 않은 지시를 추론하고 인간-로봇 상호작용을 통해 원하는 객체 (예: 충전기)를 파지할 수 있는 방법을 논한다. 구체적으로, 제안 시스템은 시각적 장면 속의 목표 객체를 명확하게 하기 위해 인간에게 질문을 하며 인간의 답변을 기반으로 여러 객체 후보에 대한 믿음을 계속적으로 갱신한다. 우리는 각 객체 후보가

현재의 시각적 및 대화 맥락을 얼마나 잘 설명하는지 평가하는 과정을 화용론적 추론이라고 부른다. 실험 결과는 화용론적 추론이 구체화되지 않은 명령이 주어졌을 때 목표 객체 발견 및 객체 파지 작업 성공률을 향상시켰다.

세 번째 주제에서는 언어가 로봇 학습을 위한 인터페이스로 어떻게 활용될 수 있는지 논의한다. 목표는 언어 감독을 통해서만 언어 조건부 로봇 정책을 학습하는 것이다. 먼저 로봇 데이터 수집을 위한 언어 기반 원격 조작 시스템을 소개한다. 그다음 자연어 감독에서 직접 언어 조건부 정책들을 학습하는 시각-언어-행동 모델을 소개하며, 이를 클립 기반 로봇 트랜스포머라고 부른다. 자연어를 학습 신호로 사용하는 클립에서 영감을 얻은 클립 기반 로봇 트랜스포머는 이 사상을 로봇 학습으로 확장한다. 구체적으로 클립 기반 로봇 트랜스포머는 인간의 언어 감독 (예: “팔을 앞으로 움직여라”)을 로봇 정책에 대한 감독 정보로 취급하고 대조 학습을 통해 언어 감독과 로봇의 현재 상태 간의 벡터 유사성을 최적화하는 방법을 학습한다. 클립 기반 로봇 트랜스포머는 새로운 로봇 기술을 학습하는 강력한 능력을 보여주며 이전 기술의 성능을 능가한다.

주요어: 깊은 학습, 로봇 학습, 인간-로봇 인터랙션, 시각 기반 대화, 화용론적 추론, 언어 기반 정책 학습
학번: 2020-36496