# REFERENTIAL GROUNDING TOWARDS MEDIATING SHARED PERCEPTUAL BASIS IN SITUATED DIALOGUE

By

Changsong Liu

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science - Doctor of Philosophy

2015

# ABSTRACT

## REFERENTIAL GROUNDING TOWARDS MEDIATING SHARED PERCEPTUAL BASIS IN SITUATED DIALOGUE

**By**

**Changsong Liu**

In situated dialogue, although an artificial agent (e.g., robot) and its human partner are co-present in a shared environment, they have significantly mismatched perceptual capabilities (e.g., recognizing objects in the surroundings). When a shared perceptual basis is missing, it becomes difficult for the agent to identify referents in the physical world that are referred to by the human (i.e., a problem of *referential grounding*). The work presented in this dissertation focuses on computational approaches to enable robust and adaptive referential grounding in situated settings.

First, graph-based representations are employed to capture a human speaker's linguistic discourse and an agent's visual perception. Referential grounding is then formulated as a graph-matching problem, and a state-space search algorithm is applied to ground linguistic references onto perceived objects. Furthermore, hypergraph representations are used to account for group-based descriptions, and one prevalent pattern of collaborative communication observed from a human-human dialogue dataset is incorporated into the search algorithm. This graph-matching based approach thus provides a principled way to model and utilize spatial relations, group-based descriptions, and collaborative referring discourse in situated dialogue. Evaluation results demonstrate that, when the agent's visual perception is unreliable due to computer vision errors, the graph-based approach significantly improves referential grounding accuracy over a baseline which only relies on object-properties.

Second, an optimization based approach is proposed to mediate the perceptual differences

between an agent and a human. Through online interaction with the human, the agent can learn a set of weights which indicate how reliably/unreliably each dimension (object type, object color, etc.) of its perception of the environment maps to the human's linguistic descriptors. Then the agent can adapt to the situation by applying the learned weights to the grounding process and/or adjusting its word grounding models accordingly. Empirical evaluation shows this weight-learning approach can successfully adjust the weights to reflect the agent's perceptual insufficiencies. The learned weights, together with updated word grounding models, can lead to a significant improvement for referential grounding in subsequent dialogues.

Third, a probabilistic labeling algorithm is introduced to handle uncertainties from visual perception and language processing, and to potentially support generation of collaborative responses in the future. The probabilistic labeling algorithm is formulated under the Bayesian reasoning framework. It provides a unified probabilistic scheme to integrate different types of evidence from the collaborative referring discourse, and to generate ranked multiple grounding hypotheses for follow-up processes. Evaluated on the same dataset, probabilistic labeling significantly outperforms state-space search in both accuracy and efficiency.

All these approaches contribute to the ultimate goal of building collaborative dialogue agents for situated interaction, so that the next generation of intelligent machines/devices can better serve human users in daily work and life.

To my family.

## ACKNOWLEDGMENTS

First of all, I want to express my deepest gratitude to my advisor, Dr. Joyce Chai, for her support, guidance, encouragement, and trust during the past several years. I am grateful for the trust and freedom she offers, with which I can pursue various research directions that interest me. Her insightful advice on tackling difficult problems, writing convincing papers, giving strong presentations, and many other research related aspects has been great help for me to become a qualified researcher. Her passion in research and diligence in work always motivate me to immerse myself in study and research, and love it. I also appreciate her a lot for her patience and encouragement that helped me to learn from failures and overcome difficulties. I am very fortunate to have had Dr. Chai as my PhD advisor, and I will follow the great example she sets to us in my career life.

I am also grateful to my PhD Committee members, Dr. Erik Altmann, Dr. Rong Jin, and Dr. Yiying Tong, for their insightful criticism, inspiring advice, and kind patience. I enjoyed sitting in Dr. Altmann's cognitive psychology class, from which I learned many interesting and inspiring cases on experiment-based studies. His deep insight on the strength and weakness of these discussed studies helped me to better design and conduct my own data collection experiments. It was also very pleasant learning experience to take both Dr. Jin's and Dr. Tong's classes in the filed of computer science. I deeply appreciated the beauty and power of computational methods in these classes, and learned to use mathematics as a tool to tackle hard research problems including those contribute to this dissertation.

I am also fortunate enough to have two summers of internship at SRI International, where I could truly experience how multidisciplinary research led to many world-changing innovations such as SIRI. I especially want to express my gratitude to Dr. Edgar Kalns, who

provided me such a great opportunity. I also would like to thank Mr. Girish Acharya and many other SRI colleagues, for their collaboration and support to make my two summers productive and fruitful.

In the past several years, I greatly enjoyed the collaboration and friendship with my colleagues at the LAIR lab: Rui Fang, Lanbo She, Shaohua Yang, Malcolm Doering, Tyler Baldwin, Matthew Gerber, Qiaozi Gao, and many others. It has been very rewarding to work with them, and joyful to play with them. I wish all the best in their future study, career, and life. I also appreciate the help from the staff of the CSE department, especially Linda Moore and Norma Teague. I wish Linda and Norma a very happy retirement life.

Last but not least, this journey would not be possible without the love and support of my family. I especially want to thank my best friend, soul mate, and other half, Ou Ni. I am very grateful for having her in my life, who really loves me, cares for me, and goes through all the ups and downs hand in hand with me. Even when I sometimes feel lost and don't have faith in myself, she always has faith in me and brings me the belief in a bright future. Another special thank is for my dear daughter Nini and son Mike, my bundles of joys. Life feels so sweet whenever I am with them, or thinking about them. Every moment becomes a cherishable memory because of them. To both my parents and parents-in-law, I owe them a debt of gratitude which I shall never be able to repay, but I will do my best to deserve and return their unwavering and unconditional love and support. The eternal love of my family is the truest reason and motivation for me to become a better man, to contribute more to our world, and to live a more worthful life.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# Chapter 1

# Introduction

As a new generation of intelligent machines/devices starts to emerge into our daily life, techniques that enable efficient and effective human-machine interaction have become increasingly important. Particularly, using natural language to dialogue with these intelligent "agents" has become an important direction to pursue [1, 2]. Different from traditional telephone-based dialogue systems (e.g., [3, 4]) and conversational interfaces (e.g., [5, 6]), humans and artificial agents (e.g., robots) are now co-present in a shared physical environment to achieve joint tasks. The meaning of language thus depends on the physical environment and the goals of communication partners. One cannot understand the dialogue without knowledge of the immediate physical and intentional context [7, 8]. This kind of dialogue is called "*situated dialogue*" [9].

In situated dialogue, although an artificial agent and its human partner are co-present in a shared environment, they have different capabilities in perceiving and reasoning about the physical world. A shared perceptual basis, which plays an important role in supporting situated dialogue between the human and the agent [10], thus is missing. Dialogue between them then becomes difficult, and both the agent and the human will need to make extra efforts to mediate a joint perceptual basis and reach a mutual understanding [11]. To address this challenging problem, our ultimate goal is to develop situated dialogue systems that are robust and adaptive to handle inaccuracies and uncertainties of perceiving the physical environment, and can interact with humans collaboratively to mediate a joint perceptual

basis and facilitate the communication.

## 1.1 Situated Referential Communication

The work presented in this dissertation focuses on *situated referential communication* between human users and artificial agents (e.g., robots). A situated dialogue often involves objects and their identities in the shared environment. One critical task thus is referential communication - a process to establish mutual understanding between conversation partners about the intended referents [12]. The agent needs to identify referents in the environment that are specified by its human partner, and the human needs to recognize that the intended referents are correctly understood. It is critical for the agent and its human partner to quickly and reliably reach the mutual acceptance of the referents before the dialogue can move forward [11].

Although referential communication between human partners often can be easily done, it becomes more difficult between a human and an artificial agent because of the "mismatched" perceptual capabilities. For example, let's consider the situation illustrated in Figure 1.1. Suppose the human wants to refer to the toy panda by issuing an utterance as

"*Give me the black panda to the right of the blue cup*".

If the hearer is another human, she/he would have no problem to identify the intended object. However, it becomes problematic for the robot to correctly identify the referent due to the robot's mismatched perceptual capabilities. As illustrated in Figure 1.1, the robot has limited object recognition and color perception: the blue cup is mis-recognized as a blue can; the toy panda and the computer mouse are not recognizable; the color of the

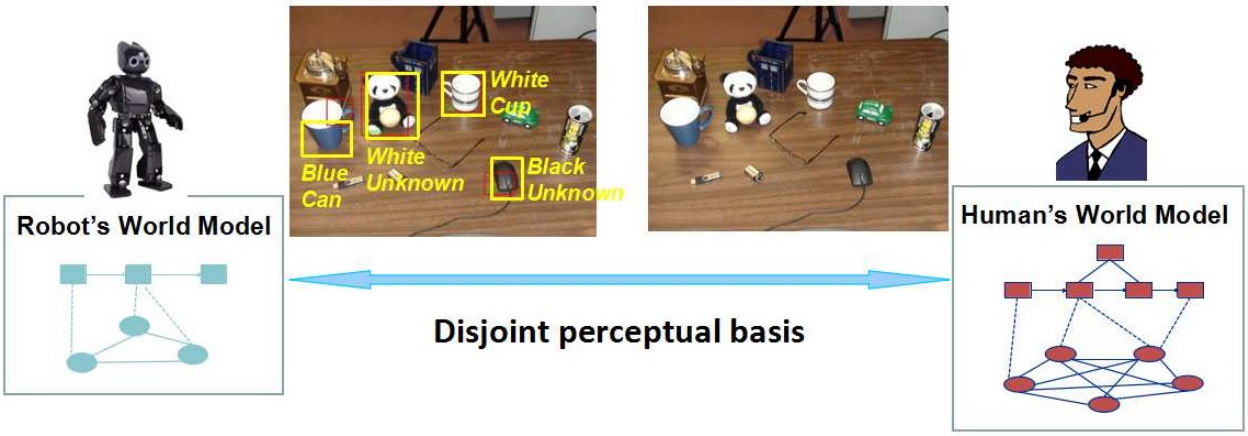



Figure 1.1 An illustration of the mismatched perceptions of the shared environment between a human and an agent.

toy panda is perceived as white but not black. With such mismatched perceptions, relying on object-specific properties (e.g., object-class and color) becomes insufficient for referential communication, and a single exchange is often not adequate. Therefore, the agent and the human need to make extra efforts to establish a joint perceptual basis for successful referential communication.

To mediate a joint perceptual basis between an agent and a human in situated dialogue, previous studies have suggested two important strategies. The first one is to rely more on spatial language (e.g., [13, 14, 15]). This is not only because modern robotic/intelligent agents are often equipped with advanced sensors for acquiring accurate spatial information, but also because spatial language usually can uniquely identify the referents if properly stated. The second strategy is to utilize extended and collaborative dialogue (e.g., [12, 16, 17, 18]). For example, the speaker can refer to the intended object iteratively: the speaker first issues an initial *installment*, and then continues with further *refashionment* (e.g., *expansion*, *replacement*, or *repair*) based on the hearer's immediate feedback. The hearer, on the other hand, can provide useful feedback based on which further refashionment can be made, or

directly refashion what the speaker just said in a *relevant next turn.*[1]

Through utilizing spatial language and collaborative dialogue, human partners can often succeed in referential communication even under mismatched perceptions (see the next section for the experiment we conducted to investigate human partners' behaviors under simulated perceptual differences). Therefore, it is important that a situated dialogue system can also capture and utilize spatial language and dialogue dynamics, and engage in communication collaboratively. It is the goal of this thesis to develop computational approaches that enable such situated dialogue systems.

## 1.2 A Human-human Dialogue Experiment

Although previous studies had done some experiments on human-human (e.g., [12]) and human-agent (e.g., [19]) referential communication in situated settings, the problem of mismatched perceptual capabilities between the conversation partners had not been taken into consideration in these experiments. To address this problem, we conducted an empirical study to investigate how two human partners collaborate and mediate a joint basis when they have simulated perceptual differences.

Our experiment was inspired by the "ablated human capability" studies that was demonstrated useful in investigating the problem-solving strategies humans would resort to when they encountered the machine-like speech recognition errors [20, 21]. In the ablated capability studies, a human subject's capacities were incrementally restricted in the direction of a real system, thus the system could learn better error-handling strategies from the way the human handled the same errors. To implement this idea in our experiment, we ablated one

[1] See [12] and [16] for a formal account on these collaborative actions in referential communication.

human participant's perceptual capability by showing her/him an impoverished image. The impoverished image simulated what a computer-vision based agent could perceive from the environment (i.e., the original image), thus we could observe how human participants strive to overcome the mismatched perceptions in situated referential communication.

### 1.2.1 Experiment Design

The setup of our experimental system is shown in Figure 1.2. Two partners (a *director* and a *matcher*) collaborate on an object naming task. They both face the same scene that is composed by some daily-life items (office supplies, friuts, etc.). However, what they actually see are different: the director is shown the original image of the scene, whereas the matcher is shown an impoverished version of the original image. An example of the two different images is illustrated at the top of Figure 1.2.

To faithfully simulate the perceptual capability of an artificial agent, we applied standard computer vision algorithms to process the original image and generate the impoverished representation of the same scene. This procedure is illustrated in Figure 1.3. To create the impoverished image, we first used the OTSU algorithm to separate foreground objects from the background [22]. Then each segmented object was fed into a feature extraction routine that computed a set of region-based and contour-based shape features of the object [23]. The feature vector of the object was then compared with all the "known" objects in a knowledge base, and the object was recognized as the class of its nearest neighbor in the knowledge base.

After this $segmentation \rightarrow feature\ extraction \rightarrow recognition$ pipeline, the final outcome was then displayed as an abstract illustration in the impoverished image. For instance, if an object was recognized as a pear, an abstract illustration of pear was displayed in the

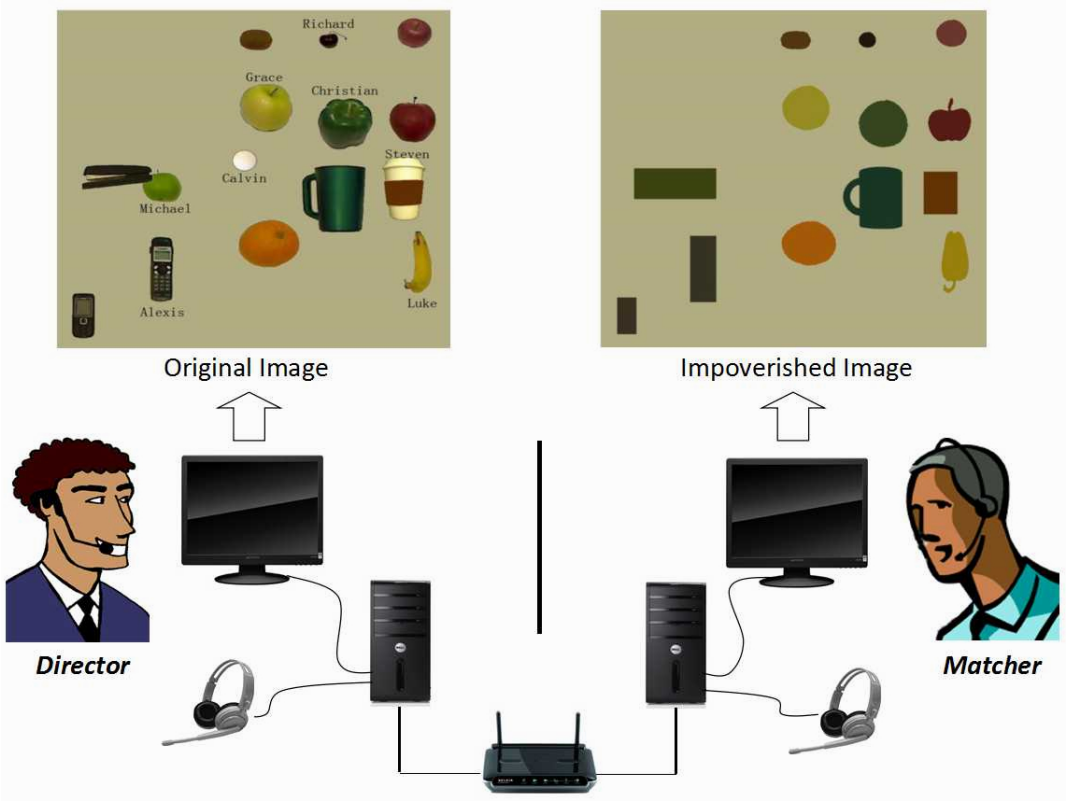



Figure 1.2 Our experimental system. Two partners collaborate on an object naming task using this system. The *director* on the left side is shown an original image, while the *matcher* on the right side is shown an impoverished version of the original image.

impoverished image at the position of the original object. The color of the illustration was set to the average color of the pixels of the original object, and the height and width were set according to the segmented bounding box of the original object.

In the object naming task, the director's goal was to communicate the "secret names" of some randomly selected objects (i.e., target objects) in the original image to the matcher, so that the matcher would know which object had what name. As shown in the example image in Figure 1.2, those secret names were displayed only on the director's side but not the matcher's. Once the matcher believed that the name of an target object was correctly communicated, s/he recorded the name by clicking on the target and repeating the name. A task was considered complete when the matcher had recorded the names of all the target

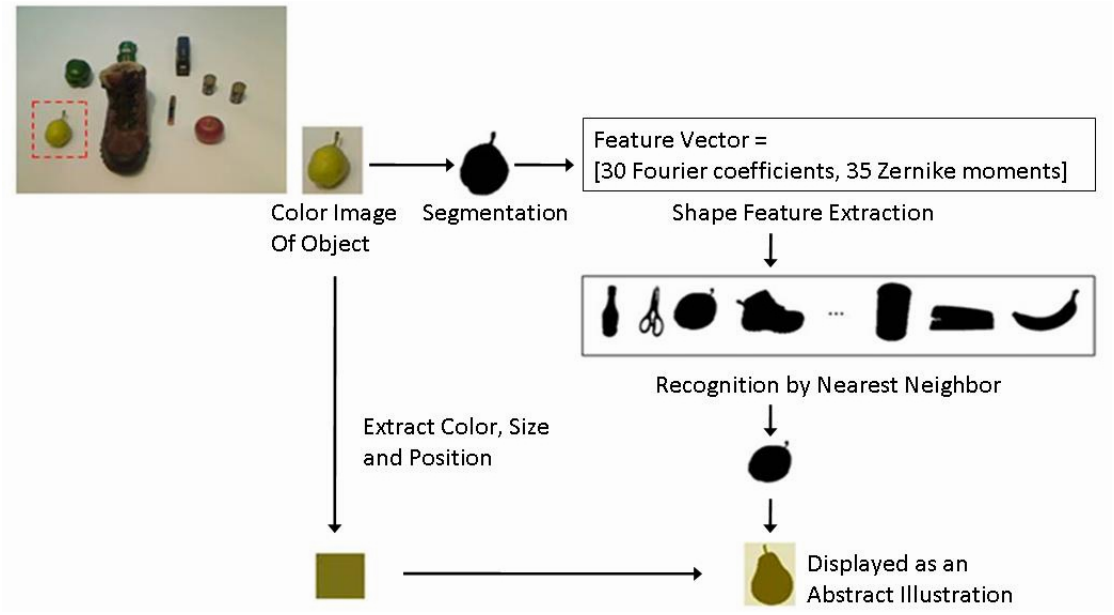



Figure 1.3 The procedure of generating the impoverished image from the original image.

objects.

### 1.2.2 Observations

Using this experimental setup, we collected a set of human-human[2] dialogues on the object-naming task (namely, a referential communication task). Consistent with previous studies (e.g., [12, 14]), our collected data have demonstrated overwhelming use of spatial relations and collaborative referring actions to overcome the mismatched perceptions. Here are two example excerpts from the collected data.

Example 1:

$D$[3]: what I am seeing on my screen are three apples (1)

$D$: and there is an apple that is directly below, slightly to the right of the battery (2)

[2]All the participants were undergraduate/graduate students recruited from the campus of Michigan State University.

[3]$D$ stands for the *Director*.

$M$[4]: ok (3)

$D$: and then there is an apple to the right of that (4)

$D$: and there is an apple below that last apple (5)

$M$: ok (6)

$D$: so the apple directly below the battery is called Alexis (7)

$M$: ok, this is Alexis (8)

$D$: and then to the right of Alexis is an apple (9)

$D$: and then below that apple is (10)

$D$: I am sorry, actually that is a red pear (11)

$D$: but it looks like an apple (12)

... ...

Example 2:

$D$: there is basically a cluster of four objects in the upper left, do you see that (1)

$M$: yes (2)

$D$: ok, so the one in the corner is a blue cup (3)

$M$: I see there is a square, but fine, it is blue (4)

$D$: alright, I will just go with that, so and then right under that is a yellow pepper (5)

$M$: ok, I see apple but orangish yellow (6)

$D$: ok, so that yellow pepper is named Brittany (7)

$M$: uh, the bottom left of those four? Because I do see a yellow pepper in the upper right (8)

$D$: the upper right of the four of them? (9)

$M$: yes (10)

$D$: ok, so that is basically the one to the right of the blue cup (11)

[4] $M$ stands for the *Matcher*.

$M$: yeah (12)

$D$: that is actually an apple (13)

$D$: that is a green apple and it is named Ashley (14)

... ...

As we can see from the two examples, the participants relied on both object-specific properties and spatial relations for communicating the identities of intended objects. The most commonly used object properties included *object-class*, *color*, and *spatial location*. Other properties such as *size*, *length*, and *shape* were also used. For spatial relations, the most commonly used were the projective relations [24], such as *right*, *left*, *above*, and *below*. Besides, as demonstrated in Example 2 (utterance (1), (8), (9)), descriptions based on grouping of multiple objects were also very useful [25].

Furthermore, as illustrated by these two examples, referential communication under mismatched perceptions was often a highly incremental and collaborative process. In example 1, after the director initiated "three apples" in utterance (1), he kept elaborating on them till they were accepted by the matcher. Each of the director's following utterances can be viewed as either an *expansion* (e.g., utterance (4), (5), (7), (9)) or a *replacement* (e.g., utterance (11)) to a previous utterance. In example 2, the matcher played a more active role in the dialogue. He proactively described what he perceived (utterance (4), (6)), and asked for clarification when he could not uniquely resolve the reference (utterance (8)). The director, on the other hand, could either accept the matcher's presentation and further expand it (utterance (5)), or refashion it with a replacement (utterance (13)).

All these observed dynamics conform to the *collaborative referring model* in the literature [12, 16]. We thus applied this model to categorize all the prevalent patterns of these collaborative dialogues in our data (see the next section). These categories have provided

important guidances for developing our computational approaches to support collaborative referential communication.

### 1.2.3 Patterns of Collaborative Referential Communication

In our data, referential communication between the director and the matcher generally falls into two phases: a *presentation* phase and an *acceptance* phase. A presentation phase can be in one of the following forms:

- A complete description: the speaker issues a complete description in a single turn. For example, "there is a red apple on the top right".
- An *installment*: a description is divided into several parts/installments, each of which needs to be confirmed before continuing to the rest. For example,

  A: under the big green cup we just talked about,

  B: yes

  A: there are two apples,

  B: OK

  A: one is red and one is yellow.

  .....
- An *expansion*: a description that adds more information to the previous presentation, such as speaker A's second and third utterances in last example.
- A *replacement*: a description that replaces some information in the previous presentation. For example, "that is actually an apple (not a pepper)".

- A *trial*: a description with a try marker. For example, "Is there a red apple on the top right?"

In an acceptance phase, the other interlocutor (i.e., the hearer) can either accept or reject the current presentation, or postpone the decision. Two major forms of accepting a presentation are observed in our data:

- An *acknowledgement*: the hearer explicitly shows his/her understanding, using assertions (e.g., "Yes","Right", "I see") or affirmative continuers (e.g., "uh huh", "OK").

- A *relevant next turn*: the hearer proceeds to the next turn that is relevant to the current presentation. For example: A says "I see a red apple" and directly following that B says "there is also a green apple to the right of that red one".

Furthermore, there are also two forms of rejecting a presentation:

- A *rejection*: the hearer explicitly rejects the current presentation, for example, "I don't see any apple".

- An *alternative description*: the hearer instead presents an alternative description. For example, A says "there is a red apple on the top left," and immediately following that B says "I only see a red apple on the right".

Besides explicitly accepting or rejecting a presentation, the hearer can also postpone the decision and wait/request for more information. For example:

- A *clarification question/request*: the hearer asks a question to let the speaker clarify, such as "uh, the bottom left of those four?", "I see a blue object there, is that what you are talking about?"

Actually, the acceptance to a previous presentation often represents a new presentation itself, which then triggers further acceptance. For example, a *relevant next turn* as an acceptance can also be viewed as a new presentation itself. An *alternative description* is also a *replacement*. And a *clarification question* often presents a *trial* that awaits for further response. In general, referential communication in our data emerges as a hierarchical structure of recursive presentation-acceptance phases. It is a highly collaborative process that cannot be moved forward without the joint efforts from both of the two dialogue participants.

## 1.3 Contributions

The observations from our empirical study have indicated that, to enable effective and efficient situated referential communication between human users and artificial agents, computational approaches should take the following issues into consideration:

- It needs to model different types of referring expressions, i.e., referring expressions based on object-specific properties and spatial relations. Especially, spatial relations can provide very useful information for identifying intended objects, when object-specific properties alone is not sufficient due to mismatched perceptual capabilities.

- The model of the linguistic contents also needs to capture the rich dynamics of collaborative dialogue. It should identify and capture various relationships between interrelated utterances in the discourse, for instance, those patterns of collaborative referential communication we discussed in the previous section.

- Besides modeling the linguistic discourse, it needs to represent the perceived visual features and spatial relations of the physical objects in the environment. Then a com-

putational approach should use the semantic representation of the linguistic discourse as constraints to search for feasible mappings from the model of the linguistic discourse to the model of the perceived environment. We call such a process as "*situated referential grounding*".

- Because of the mismatched perceptual capabilities between a human and an agent, situated referential grounding needs some level of approximation without enforcing complete satisfaction of the constraints. Furthermore, the dynamics of collaborative dialogue should be utilized to overcome mismatched perceptions and facilitate referential grounding.

These issues comprise the basic requirements that a computational approach to situated referential grounding needs to fulfill. Besides, some other features/functionalities are also desirable to support robust and adaptive referential grounding, which include but not limited to:

- It is desirable that the computational approach is based on a probabilistic framework, within which the uncertainties from different sources (e.g., visual perception, speech and language processing) can be handled under a unified probabilistic scheme. Using such a probabilistic approach, the referential grounding component should efficiently generate ranked grounding hypotheses to facilitate the dialogue manager's decision making.

- Since an agent's perceptual and grounding abilities can be affected by many situational factors, such as noisy environment, faulty sensors, and human speaker's individual differences, even previously well-performed system can become unreliable and need to

be adjusted under a new situation. It is thus important to develop automatic learning mechanisms to allow the system efficiently adapt to changed situations.

Towards the ultimate goal of building robust and adaptive dialogue systems for situated referential communication, this dissertation has made the following contributions to address all these issues.

First, graph-based representations are employed to capture a human speaker's linguistic discourse and an agent's visual perception. Referential grounding is then formulated as a graph-matching problem, and a state-space search algorithm is applied to ground linguistic references onto perceived objects. Furthermore, hypergraph representations are used to account for group-based descriptions, and one prevalent pattern of collaborative communication observed from a human-human dialogue dataset is incorporated into the search algorithm. This graph-matching based approach thus provides a principled way to model and utilize spatial relations, group-based descriptions, and collaborative referring discourse in situated dialogue. Evaluation results demonstrate that, when the agent's visual perception is unreliable due to computer vision errors, the graph-based approach significantly improves referential grounding accuracy over a baseline which only relies on object-properties.

Second, an optimization based approach is proposed to mediate the perceptual differences between an agent and a human. Through online interaction with the human, the agent can learn a set of weights which indicate how reliably/unreliably each dimension (object type, object color, etc.) of its perception of the environment maps to the human's linguistic descriptors. Then the agent can adapt to the situation by applying the learned weights to the grounding process and/or adjusting its word grounding models accordingly. Empirical evaluation shows this weight-learning approach can successfully adjust the weights to reflect the

agent's perceptual insufficiencies. The learned weights, together with updated word grounding models, can lead to a significant improvement for referential grounding in subsequent dialogues.

Third, a probabilistic labeling algorithm is introduced to handle uncertainties from visual perception and language processing, and to potentially support generation of collaborative responses in the future. The probabilistic labeling algorithm is formulated under the Bayesian reasoning framework. It provides a unified probabilistic scheme to integrate different types of evidence from the collaborative referring discourse, and to generate ranked multiple grounding hypotheses for follow-up processes. Evaluated on the same dataset, probabilistic labeling significantly outperforms state-space search in both accuracy and efficiency.

All these approaches contribute to the ultimate goal of building collaborative dialogue agents for situated interaction, so that the next generation of intelligent machines/devices can better serve human users in daily work and life.

## 1.4 Organization of the Dissertation

The remainder of this dissertation is organized as follows. Chapter 2 first discusses some related linguistic studies on referential communication and collaborative dialogue, and then reviews related work on computational approaches to the language grounding problem. Chapter 3 presents the graph-matching based approach to situated referential grounding. The contents in Chapter 3 come from our papers appeared in the 13th (2012) and 14th (2013) annual meetings of the Special Interest Group on Discourse and Dialogue (SIGDIAL) [26, 27]. Chapter 4 presents the weight learning approach to mediate the perceptual differences between humans and agents. This is based on the work published in the 29th con-

ference of the Association for the Advancement of Artificial Intelligence (AAAI 2015) [28]. Chapter 5 introduces the probabilistic labeling based framework. The material follows the paper published in the 52nd annual meeting of the Association for Computational Linguistics (ACL 2014) [29]. Lastly, Chapter 6 concludes the thesis and discusses some future directions to be explored.

# Chapter 2

# Related Work

SHRDLU [30], the seminal work on situated language understanding by Terry Winograd in the 1970's, still remains to be one of the most sophisticated systems today. With a relatively static and symbolic representation of the "physical" situation (a virtual block world), it used a systematic grammar to analyze the user's language input and identify the key units (e.g., noun, verb, or preposition phrases). The connection between the language units and the physical world were implicitly encoded in the form of computational procedures attached to the grammar, and a higher-level parsing procedure was used to combine the sentence structural analysis with the semantics (i.e., the meaning interpreting procedures attached to grammar units) to select the referent.

Many nowadays situated language understanding systems have followed the similar idea established by SHRDLU (e.g., [31, 32, 33, 34, 35, 36]). These systems often consist of two key components: The first component addresses formalisms and methods that connect linguistic terms to the agent's representations of the shared environment and task. Given these connection models, the second component analyzes the syntactic structure over the linguistic units in a sentence, and combines their models accordingly to generate situational context based interpretation. Comparing to those pioneer systems like SHRDLU, the possibly two most important advancements that modern situated language understanding systems have made are: (1) relying more on statistical models and machine-learning approaches to build more robust and adaptive systems, whereas the earlier systems were mostly based on symbolic and

hand-crafted models and procedures; (2) paying more attention to human behavior studies and incorporating more linguistic models/theories into the computational approaches.

In the rest of this chapter, we will first review some linguistic studies that provide insights on how humans behave in referential communication tasks and shed light on what a situated dialogue system should take into account when it interacts with human. Then we will review previous work on computational approaches to situated language grounding. Some of them focus on the first component, some on the second, and some on both.

## 2.1 Linguistic Studies on Referential Communication

Referential communication is a conversation focusing on objects and their identities [11]. It arises very often in our daily life. For example, when two people build something together, or one teaches the other how to build, the two of them need to refer to the construction pieces again and again. Referential communication also arises in tasks in which people need to arrange objects, transport objects, or do other things with objects in the surrounding environment. In the linguistic community, referential communication has been studied for a long time, tracing back to the 1960's [37, 38].

In situated settings, the typical way of establishing a reference is to compare the properties of the referent with the properties of other surrounding items, and the expression used conforms to the Gricean maxims [39] in general [40]. For example, in the study of [41], they found that people often use object-specific properties such as object-class, color, shape, and size, to construct their referring expressions. Spatial relations between objects is another way for referring to objects, but they tend to be less-preferred because of the more cognitive efforts it imposes to the speaker.

However, the patterns in a monologue setting and in a two-person dialogue setting are profoundly different [38, 10]. Conversation between two participants is a joint activity [10]. In conversation, participants coordinate their mental states based on their mutual understanding of their intention, goals, and current tasks [42]. An important notion, also a key to the success of communication is *grounding*, a process to establish mutual understanding between conversation participants. Specifically for the referential communication in conversation, Clark and Wilkes-Gibbs developed *the collaborative model of referring* to explain the referential behaviors of participants [43]. This work states that grounding references is a collaborative process following *the principle of least collaborative effort* [43]:

> ... speakers and addressees try to minimize collaborative effort, the work both speakers and addressees do from the initiation of the referential process to its completion.

The collaborative model indicates that speakers tend to use different types of noun phrases other than elementary noun phrases during communication. The addressee attends to what has been said almost at the same time that the utterance is produced by the speaker. The speaker often adapts language production in the middle of the planning based on the feedback from the addressee. Similarly, addressees make efforts to accept or reject references using alternative descriptions and indicative gestures (e.g., pointing, looking, or touching) [44]. Different types of evidence can be used to indicate grounding of references such as back-channel responses, relevant next turn, and continued attention [44]. When an initial noun phrase is not acceptable, it must be refashioned. The collaborative model also identified three types of refashioning: repair, self-expansion, and replacement. Through these mechanisms, the speaker and the addressee strive to minimize the collaborative effort in grounding the reference [43].

The collaborative model and the concept of grounding have motivated previous work on spoken dialogue systems [45], embodied conversational agents [46], and recent work on human robot interaction [47, 48, 49]. However, the implications of the collaborative model is not clear in situated dialogue where conversation partners have significantly mismatched capabilities in perceiving the environment (e.g., speech recognition and object recognition). It is not clear in this setting how participants strive to collaborate and minimize the collaborative effort in grounding references. Understanding these implications is important to enable the collaborative referential process between a human and an artificial agent such as a robot. The experiment presented in Section 1.2 is our first step towards understanding how participants with mismatched capabilities coordinate the collaborative referring process through dialogue.

## 2.2 Computational Approaches to Situated Language Grounding

Similar to SHDRLU [30], some early situated language understanding work such as SAM [50] and Ubiquitous Talker [51] were also based on the syntactic-analysis driven framework. These systems used a phrase parser to analyze the syntactic structure of language inputs, based on which sequences of operations (i.e., plans) were constructed. These plans operated on a symbolic knowledge base that was constructed by a vision system to represent the perceived environment such as the properties of perceived objects. The key limitation of these systems is plans can only operate on symbolic representations of visual scenes, which cannot represent subtle visual features that is always available to human speakers.

To retain the subtle visual information sensed by vision systems and achieve robust lan-

guage grounding performance, later work often relies on sophisticated "language grounding models" that associate rich visual representations (i.e., with low-level visual features and complex internal structures) to linguistic terms. These models are often based on psychology/linguistics studies or machine-learning approaches.

For example, Mojsilovic developed computational models for color categorization and composition [52]. Through subjective experiments, a vocabulary and syntax for describing colors was collected first. Then a perceptually based color-naming metric, which followed neurophysiological findings on human color categorization, was designed to measure the compatibility between a color name from the vocabulary and an arbitrary numeric color values. Based on this metric, their algorithm could accurately identify regions of known colors in different color spaces ($RGB$ and $HSV$), and assign names to arbitrary colors that were compatible with human judgments.

Regier and Carlson's work focused on building grounding models to map spatial language onto aspects of visual perception [53]. The grounded meaning of projective spatial terms, such as *above*, *below*, *left*, or *right*, was computationally modeled by the combination of an attention process (i.e., finding a focused location and direction) and a vector-sum coding of overall direction (i.e., a set of constituent directions). Their model is thus called the Attentional Vector-Sum (AVS) model for projective spatial terms. In a series of evaluation experiments, their computational model accurately predicted linguistic acceptability judgments for spatial terms comparing to some other relatively simple models. The AVS model thus has provided a good formalism to connect spatial language to the perceive environment, that can be especially useful in situated referential grounding.

The work by Moratz and Tenbrink also focused on developing grounding models of projective spatial terms [24]. They developed an empirically supported model of projective spatial

relations through iterative experiments involving uninformed users. Their grounding models were mapping functions between spatial projective terms and geometrical configurations of objects in the environment, which was represented by a set of characteristic points on a 2D plane. To allow the grounding models capable of interpreting projective relations under different reference systems, they iteratively developed and tuned their models based on formal linguistic analysis of a series of experiment data, which provided them a systematic categorization of the variability in speakers' spontaneous strategies. Their work demonstrated a practical methodology to develop flexible language grounding models through an iterative development process.

Dhande's thesis work presented a computational model that connects gestalt visual perception and language [54]. This work focused on a particular class of words or phrases called "aggregates", such as *pair* and *group*, which inherently refer to groups of objects. The model grounded the meaning of these words and phrases onto the perceptual properties of visually salient groups, and provided an explanation for how the semantics of these terms interact with gestalt processes. The work by Funakoshi et al. also focused on understanding referring expressions involving perceptual grouping [55]. Similar to the iterative and empirical approach used in [24], their conducted experiments with spontaneous speakers to collect group-based referring expressions, and developed a set of models based on analyzing the collect data. Their models showed a good coverage on identifying referents by linking group-based descriptions with perceptual grouping features extracted from the visual scenes.

Since advanced sensing technologies can provide very accurate spatial representation of the physical environment [56], a body of research has focused on spatial language understanding to interpret referents, especially for situated human-robot dialogue [57, 13, 15]. Spatial information can be very helpful to overcome the limitation of object recognition in

robotic systems, however, spatial language understanding has a challenging problem in itself. The use of spatial expressions presupposes underlying conceptual reference systems, or the so-called *frame-of-reference* [58], and different frames-of-reference can lead to different interpretations of the same expression [59]. As shown in studies of spatial language using in situated interactions [60, 24, 61], automated prediction of frames-of-reference can be difficult because "speakers vary considerably in their individual solutions to the problem at hand, but they often do not account for potential underlying ambiguities (of frames-of-reference) sufficiently" [61]. A potential solution to this problem perhaps should be based on the interaction aspect, e.g., develop more interactive agent that can resolve such ambiguities through collaborative dialogue.

Ripley, a conversational robotic system developed by Roy's group [62, 63], provided a computational framework for understanding situated interaction and addressing the frame-of-reference problem as well. To ground language to the physical world, a key component of their system was a "grounded lexicon", which defined the meaning of words in terms of structured representations of the robot's rich sensorimotor information. To resolve the referent of natural language expressions, it applied a linguistic parsing procedure that is compatible with the sensor-grounded lexical meanings [31]. Furthermore, to cope with the ambiguous frames-of-reference problem, they developed a perceptually-coupled physical simulator to maintain a "mental model" of the physical environment. Using the simulator, the system can synthesize imagined views of the environment from arbitrary perspectives and generate different interpretations based on the synthesized views. This mental imaginary mechanism provides a nice basis for spatial language understanding and also generation in situated dialogue.

While it is only in a virtual world, Gorniak and Roy's work [31] on building a visually-

grounded language understanding system has been an influential piece of work in recent years. To build their system, they also first conducted an experiment to collect language data and investigate how people verbally describe objects in synthesized virtual scenes. Based on the collected data, they built a visually-grounded language understanding system that can handle the most commonly used descriptive strategies in their experiments. Their system again consisted of some "standard" components that commonly exist in the later and the earlier referential grounding systems: a set of visual feature extraction algorithms, a lexicon that is grounded in terms of these visual features, a robust parser to analyze the syntactic structure of spoken utterances, and a compositional engine driven by the parser that combines visual groundings of lexical units to select the most proper referent(s). For a large portion of the test cases, their implemented system demonstrated impressive performances on selecting the correct referents in response to natural language expressions.

In a later work [8], Gorniak and Roy further extended their computational approach to support a theory of situate language understanding, which depended on not only the physical environment, but also the goals and plans of communication partners to interpret the meaning of language. Their work was based on the concept of "perceived affordances", which were structured representations used to encode not only the perception of the world, but also the past happened and future possible interactions between the perceiver and the world. Given such representations, they used probabilistic hierarchical plan recognition mechanism to predict possible interactions between the speaker and the environment, and treated situated language understanding as filtering/ranking the predicted interactions based on the parsing outcome of the utterance. This work demonstrated an important aspect in the evaluation of their affordance-based approach, that is to take the communication partner's intention into consideration for situated language understanding.

Although not in a situated setting, the probabilistic approach to reference resolution in multimodal user interfaces by Chai et al. [6, 64] also inspired our work. In their approach, a graph-based representation was used to capture a rich set of constraints from different perspectives, including temporal alignments between speech and pointing gesture, coherence of ongoing conversation, and domain information. It then applied a probabilistic graph-matching algorithm [65] to identify the most probable referents by optimizing the satisfaction of all the constraints captured in the graphs simultaneously. As demonstrated by their evaluation results, this graph-matching based approach was very effective, especially for complex user inputs that involve multiple referring expressions and multiple gestures at the same time. Inspired by this approach, our work in situated referential grounding also employs a similar graph representation to capture various information from the linguistic discourse and the visual perception. Referential grounding is then formulated as a graph-matching problem and different algorithms have been developed to allow robust and adaptive matching.

These work we have discussed so far all aimed to build hand-crafted models and/or systems to address the language grounding problem. They completely rely on the developers' manual efforts to analyze the patterns in data, design models and tune parameters, write grammars, or implement grounding procedures. Another line of work tackles the problem using machine-learning based approaches, in which various machine-learning techniques have been used to automatically build some or all the components of a situated language understanding system.

For example, to learn the perceptually grounded semantics of spatial terms (such as *above*, *below*, *left*, *right*, *in* and *on*), Regier used artificial neural networks to learn a set of classification models from line drawing animations labeled by spatial terms [66]. Based on a

simple set of features, such as the shapes, sizes, and relative positions of objects, the learned model could reliably classify a variety of spatial configurations and events in accordance with spatial terms across several languages. This work gives an excellent example about how perceptually grounded semantics of natural language can be automatically acquired using machine learning technique.

Another good example for learning perceptually grounded meanings of linguistic terms is the computational model of word acquisition developed by Roy and Pentland [67]. They proposed an information theoretic framework which could learn models for speech segmentation, word discovery and visual categorization simultaneously from spontaneous infant-directed speech data paired with video images of objects. The learning of sensor-grounded word semantics in this work focused on only the shape of objects, but other attributes such as color, size, and so forth could be handled in the same way. The visual inputs to their learning algorithms were histograms of local shape features of objects, and such raw inputs were categorized into prototypes using statistical learning mechanisms. To associate the identified spoken words, they used mutual information to evaluate the strength of the association between word and shape representations based on their occurrence from multiple observations. This work demonstrated the possibility of learning grounded word-to-meaning models from raw sensor data without the need for human transcription and labeling.

While the just mentioned two examples only focused on learning the grounded semantics of one particular category of natural language descriptors (i.e., spatial terms and shape terms, respectively), Roy's another work developed a unified scheme for learning perceptually grounded word semantics on different object-attributes including shape, color, size, and spatial terms as well [68]. The grounded meanings of the concepts of different categories were all modeled as multivariate Gaussian distributions in relevant feature spaces. A set of

learning algorithms were developed to extract the syntactic structures, cluster words into word classes, select relevant visual features, and establish the grounding models of individual words. This approach successfully acquired the visual semantics of individual words, word classes, and phrase structures in terms of probabilistic models over the visual feature spaces. All the models were learned from a dataset of visual scenes paired with transcribed natural language descriptions (collected in a "show-and-tell" experiment), without any manual annotation.

Yu and Ballard developed a multimodal learning interface to learn to associate spoken language with perceptual features in an unsupervised mode [69]. To facilitate learning, their interface collected a rich set of multimodal sensory information simultaneously with users' speech inputs, while they were performing everyday tasks and providing natural language descriptions of their behaviors at the same time. The collected multimodal sensory information included users' perspective videos, hand movements, gaze positions, and head directions. These multisensory information were used to extract visual feature representations of objects and actions, and estimate users' attentions. Expectation-Maximization (EM) algorithm was then applied to find the reliable associations of spoken words and temporally co-occurring attentional objects and actions. As demonstrated by their work, utilizing the user-centric multisensory information can have a key advantage of reducing large amount of irrelevant computation that a solely co-occurrence based statistical learning can hardly avoid.

Tellex et al. proposed a probabilistic graphical model based approach for learning and understanding natural language commands in terms of a robot's sensorimotor representations [70]. Given a natural language command, they first built a hierarchical representation of its semantic structure based on linguistic parsing. Then a Conditional Random Fields (CRF) [71] model was constructed from the semantic structure, which captured the interde-

pendencies between the language phrases in the command and their groundings. With such graphical models, they further developed efficient inference and learning algorithms to find proper mappings from natural language to the robot's internal representations of its action and the perceived environment. Their model was very successful in learning and inferring the grounded meaning of navigation and manipulation verbs and spatial phrases. One limitation of this approach is it requires the annotated grounding of each individual phrase in a command in order to train the model. To address this limitation, their later work proposed a weakly supervised learning algorithm to train the model with unaligned parallel data [72]. This new learning algorithm only needs to observe the top-level action corresponding to each command, but no detailed annotations are required at training time.

These work we have just reviewed demonstrate how meaning of language that is mapped to perceptual features of the external world can be acquired using machine learning techniques. Besides, work on situated language understanding often leverages linguistic parsing to map from natural language to the perceptual context. The parsing component can also be constructed automatically through machine learning.

For example, Kate and Mooney developed a weakly-supervised method for learning a semantic parser from ambiguous training data [73]. In their method, semantic parsers were automatically learned using Support-Vector Machines (SVMs). To handle ambiguous training data, in which each natural language sentence was paired with a set of potential meaning representations but only one was correct, their method employed an EM-like procedure to iteratively score the potential sentence-meaning pairs and retrain the parser. This method was used in a later work by Chen and Mooney [33] to build a system that could learn to transform natural language instructions into executable navigation plans in complex virtual environments. The system was able to automatically learn a grounded lexicon and a seman-

tic parser by simply observing how humans follow navigation instructions, without any prior knowledge or manual annotation.

Matuszek et al. presented an approach to jointly learn language grounding and parsing models [34] from weakly-supervised data. In their work, the grounded meaning of a word was associated with a set of learned visual attribute classifiers, and semantic parsing was based on a learned Combinatory Categorial Grammar (CCG) lexicon [74]. Their training data was collected using Amazon Mechanical Turk, and contained spontaneous referring expressions paired with raw visual perceptions of scenes of multiple objects. To jointly train the parameters of different models, they developed an online and EM-like learning algorithm to optimize the log-likelihood of training data. Their approach was able to learn accurate language and perception models, given only the annotations of target objects but no explicit labelings of linguistic parsing or visual attribute classifier outputs. Based on a very close idea, Krishnamurthy and Kollar also developed a grounded language acquisition model for jointly learning to parse and perceive [35]. The advancements of their work were being able to learn models for not only one-argument categories (object attributes) but also two-argument relations, and allowing for entirely weakly supervised training without a bootstrapping phase.

All these related works we have discussed in this section provide valuable insights on how to manually or automatically build the key components (e.g., linguistic parsing, grounding models to connect words to visual perceptions, and computational procedures to combine visual groundings of lexical units) for a situated referential grounding system. However, most of them have only dealt with the interpretation of single referring expressions/utterances, but not interrelated utterances as in a dialogue context.

Some earlier works [17, 18, 75] developed symbolic reasoning (e.g., planning) based approach to incorporate dialogue dynamics into interpreting referring expressions. These works

provided good examples for computationally modeling the *collaborative referring* theory [12]. However, they have not addressed situated referential grounding, for which pure symbolic reasoning based approach may not be sufficient and new approaches that are more robust against uncertainties need to be pursued. Some recent works have tackled this problem using "hybrid" models. For example, DeVault and Stone proposed an approach that combined symbolic reasoning and machine learning for interpreting situated referential grounding dialogue [19]. Kennington and Schlangen used Markov Logic Networks (MLN) [76] to incorporate the discourse and situational context for situated language understanding in an incremental manner [77]. But their "environments" were just simplistic block worlds and the issue of mismatched perceptions between humans and agents was not addressed.

# Chapter 3

# A Graph-matching Approach

In this chapter, we present a graph-matching based approach for situated referential grounding.[1] This approach uses a graph-based representation to model the referential communication dialogue. The graph representation models different types of referring expressions, including object-specific properties, spatial relations, and group-based descriptions. It also captures the discourse relations between different referring expressions. The environment perceived via computer vision is also represented using the same graph model. Referential grounding is then formulated as a graph-matching problem to find feasible matchings between the graph representation of the linguistic discourse and the graph representation of the visual perception.

This chapter begins with an overview of our situated referential communication system. Then it presents the graph representation and the graph-matching algorithm in detail, followed by the evaluation of this approach using the data collected from the experiment as described in Section 1.2. Lastly, we conclude this chapter with some discussions.

[1] This chapter is based on the following publications:
C. Liu, R. Fang, and J. Chai, "Towards mediating shared perceptual basis in situated dialogue," in *Proceedings of the 13th SIGDIAL Conference*, pp. 140–149, July 2012.
C. Liu, R. Fang, L. She, and J. Chai, "Modeling collaborative referring for situated referential grounding," in *Proceedings of the 14th SIGDIAL Conference*, pp. 78–86, August 2013.

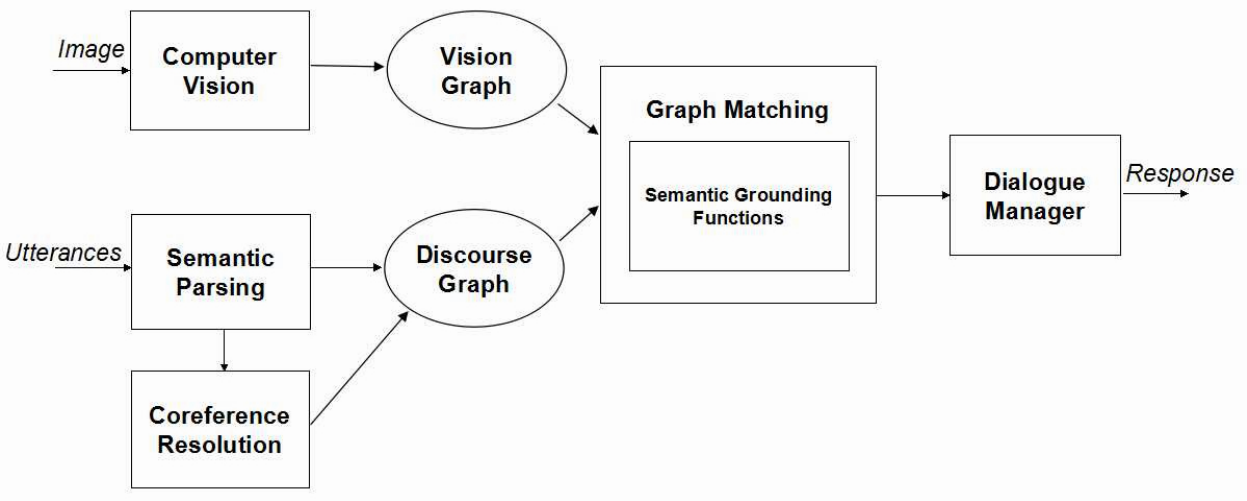



Figure 3.1 An overview of our graph-matching based referential grounding system.

## 3.1 System Overview

Figure 3.1 illustrates the key elements and the process of our graph-based method. The key elements of our method are two graph representations, one of which is called the *discourse graph* and the other called the *vision graph*.

The discourse graph captures the information extracted from the linguistic discourse. To create the discourse graph, the linguistic discourse first needs to be processed by Natural Language Processing (NLP) components, such as the semantic parsing and coreference resolution components. The output of the NLP components are usually in the form of some formal semantic representations, e.g., in the form of First-Order Logic (FOL) representations. The discourse graph is then created based on the formal semantics and discourse relations between different utterances. The vision graph, on the other hand, is a representation of the visual features extracted from the perceived environment. It is built based on the information output by the Computer Vision (CV) component. Given the discourse graph and the vision graph, then we can formulate referential grounding as constructing a

node-to-node mapping from the discourse graph to the vision graph, or in other words, a *matching* between the two graphs.

Note that, the matching between the discourse graph and the vision graph we encounter here is different from the original graph matching problem that is often used in the CV field [78, 79]. The original problem only considers matching between two graphs that have the same type of values for each attribute. But in the case of situated referential grounding, all the attributes in the discourse graph possess symbolic values since they come from formal semantic representations, whereas the attributes in the vision graph are often numeric values produced by CV algorithms. Our solution is to introduce a set of *semantic grounding functions*, which bridges the heterogeneous attributes of the two graphs and makes general graph matching algorithms applicable to referential grounding. The details will be presented later in this chapter.

## 3.2 Graph-matching for Situated Referential Grounding

In the field of Computer Vision and Pattern Recognition, *Attributed Relational Graph* (ARG) is a very useful data structure to represent an image [80, 81]. In an ARG, the underlying unlabeled graph represents the topological structure of the scene. Then each node and edge are labeled with a set of attributes that represents local features of a single node or the topological features between two nodes. Based on the ARG representations, an inexact graph matching is to find a graph or a subgraph whose *error-transformation cost* with the already given graph is minimum [82, 83].

Motivated by the representation power of ARG and the error-correcting capability of

inexact graph matching, we developed a graph-based approach to address the referential grounding problem. In the following we first demonstrate how the ARG representations are created, and then present the formal formulation of our graph-based method for referential grounding.

### 3.2.1 Graph Representation

Figure 3.2 illustrates the graph representations in our method. The discourse graph is created based on the formal semantic representation (e.g., FOL representations) from parsing and the coreference resolution results. For instance, each linguistic entity is represented by a node in the graph, and one-arity predicates are translated to the node attributes. Two-arity predicates that correspond to the mentioned relations between entities are represented by the edges and their attributes in the graph. Furthermore, multiple nodes can be merged together based on coreference, since coreferential discourse entities should refer to the same object in the environment (i.e., correspond to the same node in the vision graph).

The vision graph, on the other hand, is a representation of the visual features extracted from the scene. It is built based on the information output by the CV component. For instance, each object detected by CV is represented as a node in the vision graph, and the attributes of the node correspond to visual features, such as the color, size and position of the object. The edges between nodes represent their relations in the physical space.

Based on the graph representations of the linguistic discourse and the visual perception, we formulate referential grounding as a graph matching problem, which has extended the original graph matching approach used in the CV and PR filed.

Formally, an *attributed relational graph* (ARG) is a labeled graph

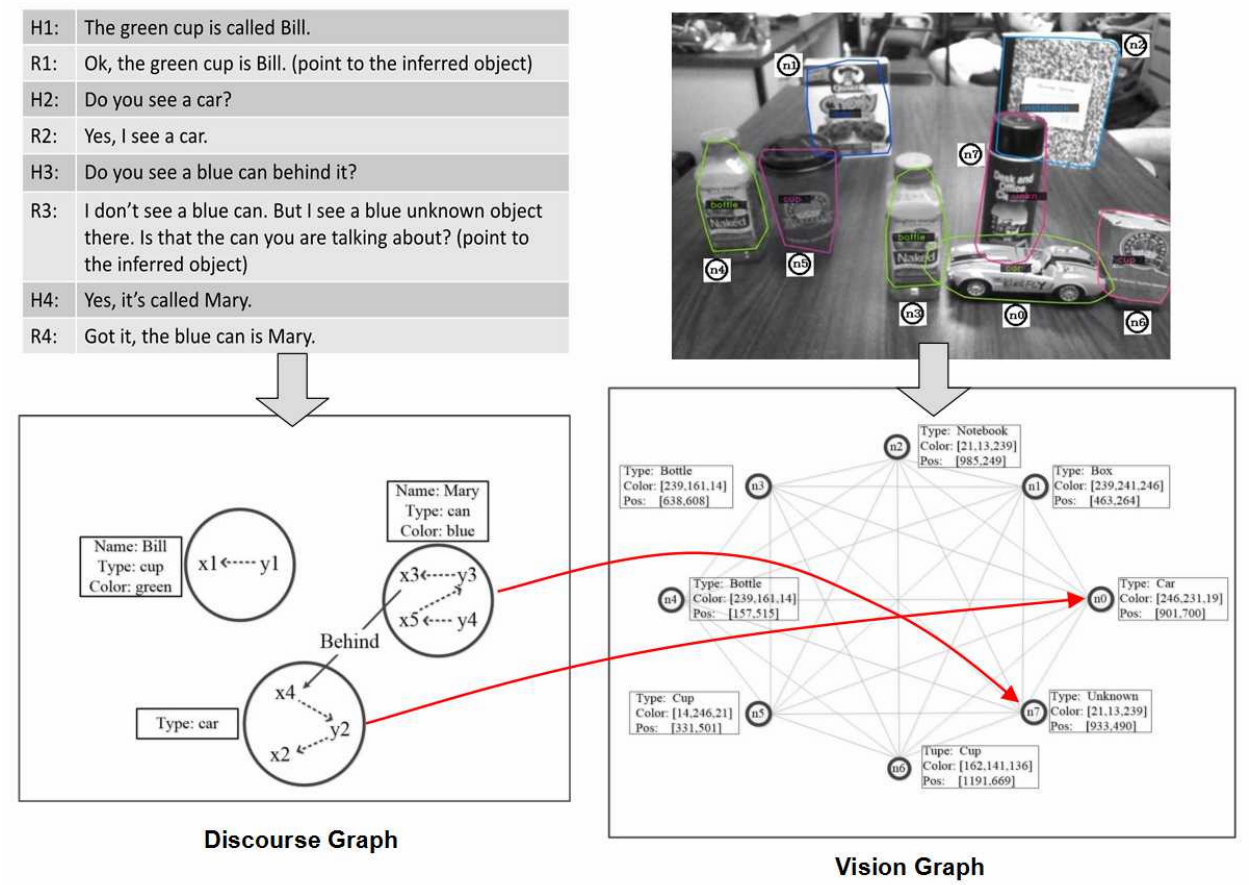



Figure 3.2 An illustration of the graph representations in our method. The discourse graph is created from formal semantic representations of the linguistic discourse; The vision graph is created by applying CV algorithms on the visual scene. Given the two graphs, referential grounding is to find a proper node-to-node mapping from the discourse graph to the vision graph.

$$
\begin{aligned}
&\mathrm{G} = (\mathrm{X}, \mathrm{E}) \\
&\mathrm{X} = \{x_m \mid m = 1, \ldots, M\} \\
&\mathrm{E} = \{e_i = (e_{i1}, e_{i2}) \mid i = 1, \ldots, I; e_{i1} \in \mathrm{X}, e_{i2} \in \mathrm{X}\}
\end{aligned}
\tag{3.1}
$$

So it is a directed graph with $M$ nodes and $I$ (directed) edges. Furthermore, each node $x_m$ is assigned a set of $A$ attributes $\mathrm{U}_m$ and each edge $e_i$ is assigned a set of $B$ attributes $\mathrm{V}_i$, i.e.,

$$\begin{aligned} \mathrm{U}_m &= \{u_{ma} \mid a = 1, \ldots, A\} \\ \mathrm{V}_i &= \{v_{ib} \mid b = 1, \ldots, B\} \end{aligned} \tag{3.2}$$

For example, a node can be assigned a set of attributes as

$$\{v_1 = Apple, v_2 = Red, v_3 = Small, v_4 = Front\}$$

which specify the type, color, size and location of an object represented by the node.

The vision graph is defined in the same way and denoted as $\mathrm{G}'$, i.e.,

$$\begin{aligned} \mathrm{G}' &= \left(\mathrm{X}', \mathrm{E}'\right) \\ \mathrm{X}' &= \left\{x'_n \mid n = 1, \ldots, N\right\} \\ \mathrm{E}' &= \left\{e'_j = \left(e'_{j1}, e'_{j2}\right) \mid j = 1, \ldots, J; e'_{j1} \in \mathrm{X}', e'_{j2} \in \mathrm{X}'\right\} \end{aligned} \tag{3.3}$$

In an ARG, the value of a node/edge attribute can be symbolic, numeric, or as a vector of numeric values. The attributes in a discourse graph always contain symbolic values since they are from linguistic inputs. On the other hand, the attribute values in a vision graph are mostly numeric because they are from computer vision algorithms' outputs. For example, if in the vision graph the attribute $v_1$ is used to represent the color feature of an object, then a possible assignment could be $v_1 = (255, 0, 0)$, which is the $rgb$ color vector. Some commonly used attributes and their typical values from our data are shown in Table 3.1.

An edge in the graph represents a 2-tuple of nodes, i.e., $e_i = (e_{i1}, e_{i2})$ in which $e_{i1} \in \mathrm{X}$ and $e_{i2} \in \mathrm{X}$. So we use a labeled edge to represent a binary relation between one object and another, such as "the apple to the right of the cup". Besides binary relations, group-based relations can also be used to distinguish an object or set of objects from others [25, 55]. A

| | *Discourse graph* | *Vision graph* |
|---|---|---|
| *type* | "apple" | "ball" |
| *color* | "red" | $(r:210, g:12, b:90)$ |
| *size* | "large" | $(w:45, h:45)$ |
| *spatial relation* | "right of" | $(x1:700, y1:450)$, $(x2:300, y2:600)$ |
| ... | ... | ... |

Table 3.1 Commonly used attributes and relations and their typical values in our collected data.

group-based relation involves more than two objects, for example, an *intra-group* relation (such as "the middle one of the three balls"), an *inter-group* relation (such as "the two objects in front of those three balls"), and a *geometric formation relation* (such as "a row of three balls")[2].

To account for the group-based relations, we can further extend the regular graph representation to *hypergraph* [84]. Hypergraph is a more general representation than regular graph. It can represent not only binary relations between two nodes, but also group-based relations among multiple nodes. Formally, a hypergraph is defined as

$$\begin{aligned} &\mathrm{G} = (\mathrm{X}, \mathrm{E}) \\ &\mathrm{X} = \{x_m \mid m = 1, \ldots, M\} \\ &\mathrm{E} = \{e_i = (e_{i1}, e_{i2}) \mid i = 1, \ldots, I; e_{i1} \subseteq \mathrm{X}, e_{i2} \subseteq \mathrm{X}\} \end{aligned} \tag{3.4}$$

The difference between a hypergraph and a regular graph is that E now is a set of "hyperedges". Just the same as an edge in a regular graph, each hyperedge $e_i$ in a hypergraph also consists of two "ends", i.e., a tail ($e_{i1}$) and a head ($e_{i2}$). However, the tail and head of a hyperedge are both subsets of X, thus they can contain any number of nodes in X (i.e., a "group").

[2]See [55] for definitions of these different types of group-based relations.

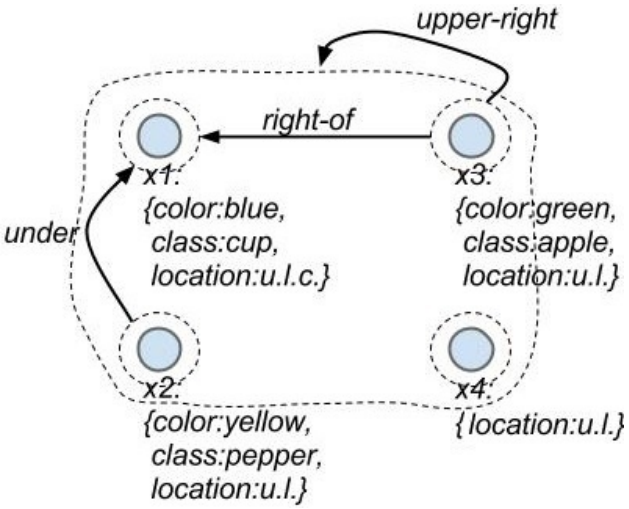



Figure 3.3 An illustration of the hypergraph representation for modeling group-based descriptions.

For example, suppose the discourse includes the following utterances:

(1) There is a cluster of four objects in the upper left.

(2) The one in the corner is a blue cup.

(3) Under the blue cup is a yellow pepper.

(4) To the right of the blue cup, which is also in the upper right of the four objects, is a green apple.

The corresponding dialogue graph G = (X, E) is shown in Figure 3.3, where X = $\{x_1, x_2, x_3, x_4\}$ and E = $\{e_1, e_2, e_3\}$. In E, for example, $e_1 = (\{x_1\}, \{x_3\})$ represents the relation "right of" between the tail $\{x_3\}$ and the head $\{x_1\}$, and $e_3 = (\{x_3\}, \{x_1, x_2, x_3, x_4\})$ represents the group-based relation "upper right" between one node and a group of nodes.

### 3.2.2 State-space Search Based Graph-matching

Given a discourse graph G and a vision graph G$'$, a *matching* between G and G$'$ is a set of node-to-node mappings between the nodes in G and the nodes in G$'$. Or in other words, a matching (denoted as $\Theta$) between G and G$'$ is to assign each node $x_m$ in X a "label" $\theta_m$ to indicate which node in X$'$ that $x_m$ maps to. Formally defined as:

$$\Theta = \left\{\theta_m = x'_n \mid m = 1, \ldots, M; x'_n \in \mathrm{X}'\right\} \tag{3.5}$$

We then define the *compatibility function* of a matching $\Theta$ as

$$f(\Theta) \;=\; \sum_{x_m \in \mathrm{X}} g\left(x_m,\, x'_n\right) \;+\; \sum_{e_i \in \mathrm{E}} h\left(e_i,\, e'_j\right) \tag{3.6}$$

where $x'_n$ and $e'_j$ are the corresponding node and edge of $x_m$ and $e_i$ according to $\Theta$, respectively[3].

To compute $f(\Theta)$, we need to further define $g\left(x_m,\, x'_n\right)$ and $h\left(e_i,\, e'_j\right)$, i.e., the compatibility function for a matched pair of nodes and edges, respectively. This is based on the attributes assigned to the nodes/edges (note that subscripts $m$, $n$, $i$ and $j$ are dropped since the definition is general for any pair of nodes/edges):

$$\begin{aligned} g\left(x,\, x'\right) &= \tfrac{1}{A} \sum_{a=1}^{A} g_a\left(u_a,\, u'_a\right) \\ h\left(e,\, e'\right) &= \tfrac{1}{B} \sum_{b=1}^{B} h_b\left(v_b,\, v'_b\right) \end{aligned} \tag{3.7}$$

We call $g_a\left(u_a,\, u'_a\right)$ and $h_b\left(v_b,\, v'_b\right)$ the *semantic grounding function* for the $a$-th node attribute and $b$-th edge attribute, respectively. Namely, a semantic grounding function for

[3]Note that, we treat the vision graph G$'$ as a *complete* graph, in which there exists an edge between each pair of nodes. In the case of a hypergraph, it means there exists a hyperedge between any two subsets of nodes.

the $a$-th attribute takes two input arguments $u_a$ and $u'_a$, which are the values of the $a$-th attribute from node $x$ and $x'$ respectively. The output of the function is a real number $p$ in the range of $[0, 1]$, which can be interpreted as a measurement of the compatibility between a symbolic value (or word) $u_a$ and a visual feature value $u'_a$. $p$ can also be interpreted as the likelihood of observing the value $u'_a$ given the symbol $u_a$ provided by the linguistic input. In essence, what we have defined as symbol grounding functions here are very similar to the "visual word semantics" in previous work [68, 85, 32].

Let $\mathrm{Y} = \left\{y_1, y_2, \cdots, y_K\right\}$ be the set of all possible symbolic values of $u_a$ (i.e., Y is the *lexicon* of the $a$-th node attribute in the dialogue graph), then $g_a\left(u_a, u'_a\right)$ can be further decomposed as:

$$g_a\left(u_a, u'_a\right) = \begin{cases} g_{a1}\left(u'_a\right) & \text{if } u_a = y_1; \\ g_{a2}\left(u'_a\right) & \text{if } u_a = y_2; \\ \vdots & \vdots \\ g_{aK}\left(u'_a\right) & \text{if } u_a = y_K \end{cases} \tag{3.8}$$

So for each value (e.g., $y_k$) of $u_a$, there is a stand-alone grounding function/model $g_{ak}\left(u'_a\right)$. Its output measures how well $u'_a$ (e.g., a numeric value from the vision graph) matches with $u_a$ (e.g., the word that human speaker used to describe this attribute of the object). In fact, it can also be interpreted as the (conditional) probability distribution of observing $u'_a$ given $u_a = y_k$, i.e., $g_{ak}\left(u'_a\right) = p\left(u'_a | u_a = y_k\right)$.

Similarly, the grounding function for edge attributes is in the same form as for the node attributes. Let $\mathrm{Z} = \left\{z_1, z_2, \cdots, z_L\right\}$ be the lexicon of the $b$-th edge attribute, then we have:

$$h_b\left(v_b, v_b'\right) = \begin{cases} h_{b1}\left(v_b'\right) & \text{if } v_b = z_1; \\ h_{b2}\left(v_b'\right) & \text{if } v_b = z_2; \\ \quad\vdots & \quad\vdots \\ h_{bL}\left(v_b'\right) & \text{if } v_b = z_L \end{cases} \tag{3.9}$$

where each $h_{bl}\left(v_b'\right)$ is a stand-alone grounding function for the value $z_l$ of $v_b$.

These semantic grounding functions can be either manually defined (e.g., [54, 31]) or automatically learned (e.g., [68, 32]). In our current work, we use a set of manually defined grounding functions similar to those used in [31].

Based on our definition of the compatibility function of a matching, the optimal matching between G and G$'$ is the one with the highest compatibility score:

$$\Theta^* = \arg\max_{\Theta} f\left(\Theta\right) \tag{3.10}$$

which gives us the most feasible result of grounding the entities in the discourse graph to the objects in the vision graph.

Given our formulation of referential grounding as a graph matching problem, the next question is how to find the optimal matching between two graphs. Unfortunately, such a problem belongs to the class of *NP-complete* [78]. In practice, techniques such as $A^*$ search are commonly used to improve the efficiency (e.g. in [80, 86]). But the memory requirement can still be considerably large if the heuristic does not provide a close estimate of the future matching cost [78]. As one practical solution, we use a state-space search method [87] and apply a simple beam search algorithm [88, 89] to retain the computation tractability.

In this state-space search algorithm, a state $s$ in the search space represents the recon-

**Algorithm 1** State-space Search Algorithm for Graph Matching

1: **INPUT:**
- Two graphs G and G$'$

2: **OUTPUT:**
- A sub-optimal matching $\Theta$ between G and G$'$

3: **METHOD:**

(1) $root \leftarrow \{\}$,
$level_0 \leftarrow \{root\}$,
$M \leftarrow$ number of nodes in G,
$N \leftarrow$ number of nodes in G$'$
$\delta \leftarrow \max(M, N)$

(2) **for** $m \leftarrow 1$ to $M$ **do**

(3) **for** every state $s$ in $level_{m-1}$ **do**

(4) **for** every unmatched node index $n$ in G$'$ **do**

(5) $s_{new} \leftarrow$ add $\theta_m = x'_n$ to $s$

(6) add $s_{new}$ to $level_m$

(7) **end for**

(8) **end for**

(9) sort all the states in $level_m$ by their compatibility scores

(10) keep only the first $m\delta^2$ states

(11) **end for**

(12) **return** the first state in $level_M$

struction of a subgraph from G and a subgraph from G$'$, as well as the matching between the two subgraphs. A state $s$ can be expanded to a successive state by adding one more matched pair of nodes, namely letting $\theta_m = x'_n$ where $m$ is the index of an unprocessed node in G and $n$ is the index of an unmatched node in G$'$. A state-space search tree is constructed from the root $\Theta = \varnothing$ in a breadth-first search manner. At each level of the search tree,

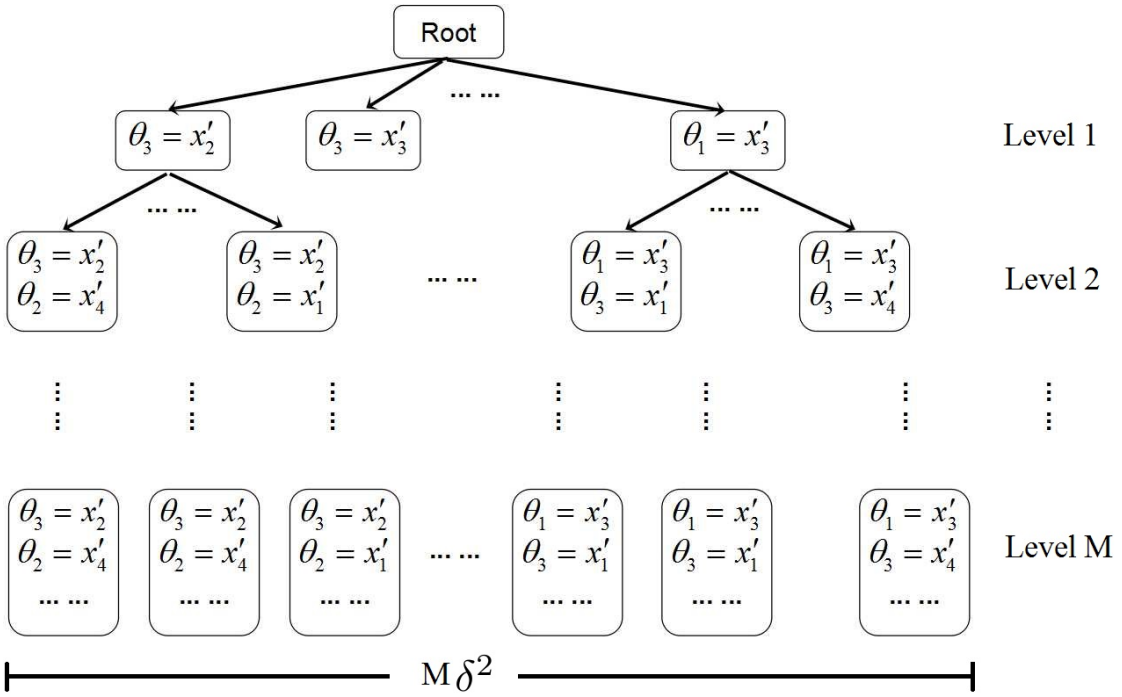



Figure 3.4 An example of the state-space search tree.

all the states are ranked based on their compatibility scores and those who fall out of a predetermined beam width are pruned. Following the assumption in [82], we set the beam size as $d\delta^2$, where $d$ is the depth of the current level of the search tree and $\delta = max\,(M, N)$ (i.e., the size of the bigger graph). Figure 3.4 illustrates the procedure of constructing the state-space search tree.

### 3.2.3 Modeling Collaborative Referring

As we have already discussed in Chapter 1, one of our goals is to explicitly incorporate collaborative referring into the graph-matching algorithm for referential grounding. As the conversation unfolds, our approach intends to incrementally build a discourse graph by keeping track of the contributions (i.e., presentation and acceptance) from both the human and the agent. This discourse graph is then matched against the perceived environment (i.e., the

vision graph) in order to resolve referring expressions from the human. Here we focus on a particularly prevalent type of collaborative referring pattern observed from our collected data[4], i.e., what we call the **agent-present-human-accept** collaboration pattern.

More specifically, our data showed that when mediating their joint perceptual basis, the director often took into consideration what the matcher saw and used that to gradually lead to the intended referents. This is demonstrated in the following example[5], where the director accepted (Turn 3) the matcher's presentation (Turn 2) through a *relevant next turn*:

(Turn 1) *D*: There is a kiwi fruit.

(Turn 2) *M*: I don't see any kiwi fruit. I see an apple.

(Turn 3) *D*: Do you see a mug to the right of that apple?

(Turn 4) *M*: Yes.

(Turn 5) *D*: OK, then the kiwi fruit is to the left of that apple.

We use this example to show how the agent-present-human-accept pattern can be incorporated to potentially improve reference resolution. Figure 3.5 gives a graphical illustration on this idea. In this example, the human and the agent (i.e., the robot) face a shared physical environment. The robot perceives the environment through computer vision algorithms and generates a graph representation (i.e., a *vision graph*), which captures the perceived objects and their spatial relations[6]. As shown in Figure 3.5(a), the kiwi is represented as an unknown object in the vision graph due to insufficient object recognition. Besides the vision graph, the robot also maintains a *discourse graph* that captures the linguistic discourse between the human and the robot.

[4]See Section 1.2 for the details of our data collection experiment.

[5]This is a clean-up version of the original example to demonstrate the key ideas.

[6]The spatial relations between objects are represented as their relative coordinates in the vision graph.

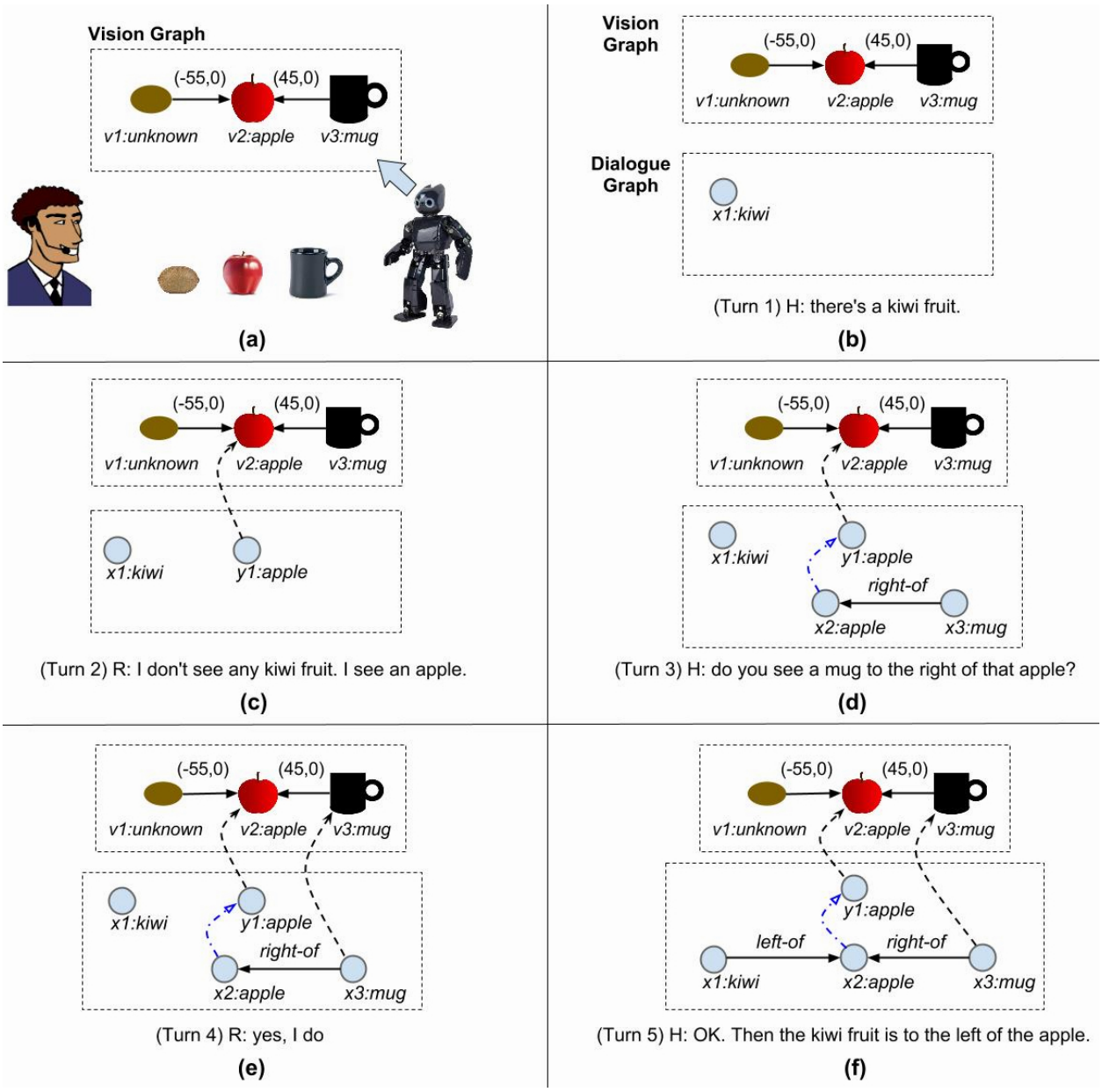



Figure 3.5 An example of incorporating collaborative efforts in an unfolding dialogue into graph representations.

At Turn 1 in Figure 3.5(b), the human says "there is a kiwi fruit". Upon receiving this utterance, through semantic processing, a node representing "a kiwi" is generated (i.e., $x_1$). The discourse graph at this point only contains this single node. Identifying the referent of the expression "a kiwi fruit" is essentially a process that matches the discourse graph to

the vision graph. Because the vision graph does not have a node representing a kiwi object, no high confidence match is returned at this point. Therefore, the robot responds with a rejection as in Turn 2 (Figure 3.5(c)) "I don't see any kiwi fruit" [7]. In addition, the robot takes an extra effort to proactively describe what is being confidently perceived (i.e., "I see an apple"). Now an additional node $y_1$ is added to the discourse graph to represent the term "an apple" [8]. Note that when the robot generates the term "an apple", it knows precisely which object in the vision graph this term refers to. Therefore, as shown in Figure 3.5(c), $y_1$ is mapped to $v_2$ in the vision graph.

In Turn 3 (Figure 3.5(d)), through semantic processing on the human's utterance "a mug to the right of that apple", two new nodes ($x_2$ and $x_3$) and their relation ($RightOf$) are added to the discourse graph. In addition, since "that apple"(i.e., $x_2$) corefers with "an apple" (i.e., $y_1$) presented by the robot in the previous turn, a coreference link is created from $x_2$ to $y_1$. Importantly, in this turn human displays his acceptance of the robot's previous presentation ("an apple") by coreferring to it and building further reference based on it. This is exactly the *agent-present-human-accept* strategy described earlier. Since $y_1$ maps to object $v_2$ and $x_2$ now links to $y_1$, it becomes equivalent to consider $x_2$ also maps to $v_2$. We name a node such as $x_2$ a **grounded node**, since from the robot's point of view this node has been "grounded" to a perceived object (i.e., a vision graph node) via the agent-present-human-accept pattern.

At this point, the robot matches the updated discourse graph with the vision graph again and can successfully match $x_3$ to $v_3$. Note that, the matching occurs here is considered *constrained graph-matching* in the sense that some nodes in the discourse graph (i.e., $x_2$) are

[7] Note that, since in this chapter we are using a dataset of human-human (i.e., the director and the matcher) dialogues, decisions from the matcher are assumed known. We leave the agent's decision making (i.e., response generation) into the future work.

[8] We use $x_i$ to denote nodes that represent expressions from the human's utterances and $y_i$ to represent nodes from the robot's utterances.

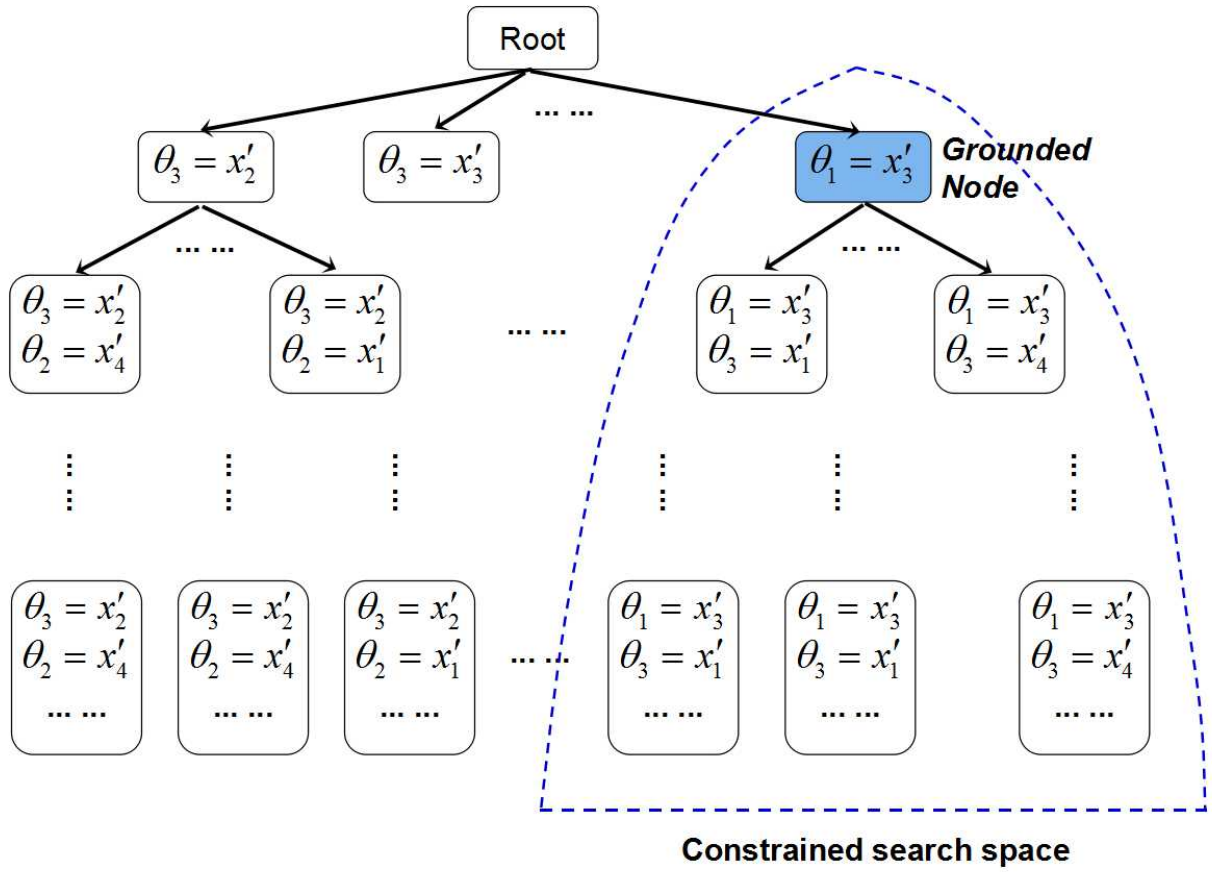



Figure 3.6 An illustration of the constrained state-space search through utilizing the *grounded nodes*.

already grounded, and the only node needs to be matched against the vision graph is $x_3$. The constrained matching utilizes additional constraints from the collaboration patterns in a dialogue and thus can improve both the efficiency and accuracy of the matching algorithm.

Based on such matching result, the robot responds with a confirmation as in Turn 4 Figure 3.5(e)). The human further elaborates in Turn 5 "the kiwi is to the left of the apple". Again semantic processing and linguistic coreference resolution will allow the robot to update the discourse graph as shown in Figure 3.5(f). Given this discourse graph, based on the context of the larger discourse graph and through constrained matching, it will be possible to match $x_1$ to $v_1$ although the object class of $v_1$ is unknown.

This example demonstrates how the discourse graph can be created to incorporate the collaborative referring behaviors as the conversation unfolds and how such accumulated discourse graph can help referential resolution through constrained matching. When the "agent-present-human-accept" pattern is identified, the associated nodes (e.g., $x_2$ in the example) will be marked as *grounded nodes* and the mappings to their grounded visual entities (i.e., vision graph nodes) will be added into the discourse graph. Such information can be then incorporated into the state-space search algorithm straightforwardly: the search procedure can now start from the state that already represents the known matching of grounded nodes, instead of starting from the root. Thus it is constrained in a smaller and more promising subspace to improve both efficiency and accuracy. See Figure 3.6 for an illustration of the constrained state-space search through utilizing the grounded nodes.

## 3.3 Evaluation

### 3.3.1 Data

A total of 32 dialogues collected from our experiments (as described in Section 1.2) are used in the evaluation. For each of these dialogues, we have manually annotated (turn-by-turn) the formal semantics, discourse coreferences and grounded nodes as described earlier in this chapter. Since the focus here is on graph building and matching for referential grounding, we use these annotations to build the discourse graphs in our evaluation (Evaluation results based on completely automatic language processing will be the discussed in Chapter 4). Vision graphs are automatically generated by CV algorithms from the original images used in the experiments. In our settings, the CV algorithm's object recognition performance is rather low: only 5% of the objects in those images are correctly recognized, thus reference

resolution will need to rely on relations and collaborative strategies[9].

The 32 dialogue graphs have a total of 384 nodes[10] that are generated from directors' utterances (12 per dialogue on average), and a total of 307 nodes generated from matchers' utterances (10 per dialogue on average). Among the 307 matcher generated nodes, 187 (61%) are initially presented by the matcher and then coreferred by directors' following utterances (i.e., relevant next turns). This indicates that the agent-present-human-accept strategy is a prevalent way to collaborate in our experiment. As mentioned earlier, those director generated nodes which corefer to nodes initiated by matchers are marked as grounded nodes. In total, 187 out of the 384 director generated nodes are in fact grounded nodes.

### 3.3.2 Results

To evaluate our approach, we apply the graph-matching algorithm on each pair of discourse graph and vision graph. The matching results are compared with the annotated ground-truth to calculate the accuracy of our approach in grounding directors' referring descriptions to visual objects. For each dialogue, we have produced matching results under four different settings: with/without modeling collaborative referring (i.e., the agent-present-human-accept collaboration) and with/without using hypergraphs. When collaborative referring is modeled, the graph-matching algorithm uses the grounded nodes to constrain its search space to match the remaining ungrounded nodes[11]. When collaborative referring is not modeled, all the director generated nodes are treated as ungrounded and need to be

[9] Although CV's object recognition can be improved by employing more sophisticate algorithms and doing more trainings, it can still become unreliable given noisy or unfamiliar environment. Thus our focus is how to enable robust referential grounding when the agent's visual perception suffers such limitations

[10] As mentioned earlier in Section 3.2.1, multiple expressions that are coreferential with each other and describing the same entity are merged into a single node.

[11] Our referential grounding algorithm actually plays the role of the "matcher" in the experiment. From the matcher's/algorithm's point of view, those grounded nodes have already been grounded onto known objects via the agent-present-and-human-accept collaboration pattern

| | Object-properties only | Regular graph | Hypergraph |
|---|---|---|---|
| Not modeling collaborative referring | 31.4% | 44.1% | 47.9% |
| Modeling collaborative referring | N/A | 55.7% | 66.2% |
| Improvement | N/A | 11.6% ($p < 0.05$) | 18.3% ($p < 0.02$) |

Table 3.2 Averaged referential grounding accuracies under different settings.

grounded by the algorithm. Besides, in order to investigate the advantages of the graph-based approach, we have also applied a baseline approach which only takes object-specific properties into consideration and ignores all the spatial and discourse relations between linguistic entities [90, 91]. In this case, the discourse representations contain only "isolated nodes" but not connected graphs.

The results of these different settings (averaged accuracies on the 32 dialogues) are shown in Table 3.2. As we can see from the first row of Table 3.2, using our graph-based approach significantly improves the referential grounding accuracy by 12.7% ($p < 0.05$ based on the Wilcoxon signed-rank test [92] on the 32 dialogues). The results thus demonstrate the importance of representing and reasoning on relations between entities in referential grounding, and the graph-based approach provides an ideal solution to capture and utilize relations.

Modeling collaborative referring improves the matching accuracies for both regular graphs and hypergraphs. When regular graphs are used, it improves overall matching accuracy by 11.6% ($p = 0.05$). The improvement is even higher as 18.3% when hypergraphs are used ($p < 0.02$). The results indicate that proactively describing what the agent sees to the human to facilitate communication is an important collaborative strategy in referential grounding dialogues. Human can often ground the agent presented object via the agent-present-human-accept strategy and use the grounded object as a reference point to further describe other

| | Group 1 | Group 2 | Group 3 |
|---|---|---|---|
| Number of dialogues | 9 | 11 | 12 |
| % of grounded nodes | <30% | 30%~60% | >60% |
| Average number of object properties [a] | 20 | 21 | 12 |
| Average number of relations [b] | 11 | 13 | 8 |
| Not modeling collaborative referring | 49.7% | 49.4% | 45.3% |
| Modeling collaborative referring | 57.0% | 76.6% | 63.6% |
| Improvement | 7.3% | 27.2% | 18.3% |

[a] Specified by directors.

[b] Specified by directors. The number includes both binary and group-based relations.

Table 3.3 Matching accuracies of three groups of dialogues (all the matching results here are produced using hypergraphs).

intended object(s), and our graph-matching approach is able to capture and utilize such collaboration pattern to improve the referential grounding accuracy.

The improvement is more significant when hypergraphs are used. A potential explanation is that those group-based relations captured by hypergraphs always involve multiple (more than 2) objects (nodes). If one node in a group-based relation is grounded, all other involved nodes can have a better chance to be correctly matched. Whereas in regular graphs one grounded node can only improve the chance of one other node, since only one-to-one (binary) relations are captured by regular graphs.

To further investigate the effect of modeling collaborative referring, we divide the 32 dialogues into three groups according to how often the agent-present-human-accept collaboration pattern happens (measured by the percentage of the grounded nodes among all the director generated nodes in a dialogue). Table 3.3 shows the statistics and the matching accuracies for each of the three groups. As shown at the top part of Table 3.3, the agent-

present-human-accept pattern happened less often in the dialogues in group 1 (i.e., less than 30% of director generated nodes are grounded nodes). In the dialogues in group 2, matchers more frequently provided proactive descriptions which led to more grounded nodes. Matchers were the most proactive in the dialogues in group 3, thus this group contains the highest percentage of grounded nodes. Note that, although the dialogues in group 3 contain more proactive contributions from matchers, directors tend to specify less number of properties and relations describing intended objects (as shown in the middle part of Table 3.3).

The matching accuracies for each of the three groups are shown at the bottom part of Table 3.3. Since the agent-present-human-accept pattern appears less often in group 1, modeling collaborative referring only improves matching accuracy by 7.3%. The improvements for group 2 and group 3 are more significant compared to group 1. However, group 3's improvement is less than group 2, although the dialogues in group 3 contain more proactive contributions from matchers. This indicates that in some cases even with modeling collaborative referring, underspecified information from human speakers (directors in our case) may still be insufficient to identify the intended referents. Therefore, incorporating a broader range of dialogue strategies to elicit adequate information from humans is also important for successful human-agent communication.

## 3.4 Conclusion and Discussion

In situated dialogue, conversation partners make extra collaborative efforts to mediate a joint perceptual basis for referential grounding. It is important to model and interpret such collaborative dialogue in order to build situated conversational agents. As a first step, we have developed a graph-based representation to capture the linguistic discourse and the

visual perception. Referential grounding is then formulated as a graph-matching problem and a state-space search algorithm is applied to ground linguistic references onto perceived objects. In addition, hypergraph representations have been used to account for group-based descriptions, and one prevalent pattern of collaborative referring (i.e., *agent-present-human-accept*) has been incorporated into the search algorithm. Our empirical results have shown that, even when the perception of the environment by computer vision algorithms has a high error rate (95% of the objects are mis-recognized), our approach can still correctly ground those mis-recognized referents with 66% accuracy. As demonstrated by the results, our graph-matching approach has provided a potential solution for reliable referential grounding through modeling and utilizing spatial relations, group descriptions and collaborative referring behaviors.

The results reported in this chapter are all based on manually annotated semantics and coreference of the linguistic discourse. When the discourses are automatically processed, semantics and coreference of these discourses often will not be obtained correctly or completely as in their manual annotations. Therefore, an important next step should explore how to efficiently match hypothesized discourse graphs (from automated language processing) with vision graphs. This issue will be addressed in the later chapters.

Our current symbol grounding functions are manually defined and limited to the specific environment used in the data collection experiment. In a real world scenario, grounding a linguistic term to a visual feature will be influenced by many contextual factors such as the surrounding of the environment, the discourse history, the speaker's individual preference. Thus, it is important to explore context-based semantic grounding functions and automatic acquisition of these functions (e.g. [68, 32, 34, 35]) in future work.

The discourse graph presented in this chapter represents all the mentioned entities and

their relations that are currently available at any given dialogue status. Due to its flatten structure, however, it is difficult to model dialogue dynamics at the illocutionary level (for example, in [93, 94]). Using richer representations such as hierarchical (hyper)graphs [95, 96] will allow us to better represent and utilize the dialogue context for referential grounding and for response generation.

Nevertheless, our graph-based approach provides a well-formed basis for modeling various types of referring expressions, capturing the collaborative dynamics, and utilizing the dialogue context to constrain referential grounding. All these aspects are important for enabling collaborative dialogue agents towards situated interaction in the physical world.

# Chapter 4

# Learning to Mediate Perceptual Differences

In Chapter 3, we have introduced our graph-based approach for referential grounding, which uses ARG (Attributed Relational Graph) to capture the collaborative referring dialogue and solves referential grounding as a graph-matching problem. We have shown that the graph-based approach provides a principled way to model different types of referring expressions and incorporate the "agent-present-human-accept" collaboration pattern to improve grounding accuracy. This chapter presents our work to further extend the graph-based approach to address some other important issues in situated referential grounding:[1]

- The graph-matching we discussed so far has always been *exact-matching*, i.e., every node in the discourse graph is forced to match with a node in the vision graph. However, due to various errors that can happen in constructing both the discourse graph and the vision graph, it may not always be suitable to enforce exact matching. *Inexact-matching* criterion [78], i.e., obtaining proper matchings even if they are only partial, is more desirable to accommodate errors and allow the system to deal with problematic situations more flexibly.

[1] This chapter is based on the following publication:
C. Liu and J. Chai, "Learning to mediate perceptual differences in situated human-robot dialogue", in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 2015.

- To enable more adaptive dialogue system, we should not only focus on the interpretation of the referring expressions, but also try to acquire useful information from the interaction with human to mediate the perceptual differences. The weight-learning approach presented in this chapter demonstrates our effort towards this promising and challenging goal. Through online interaction with the human, this approach can learn a set of weights indicating how reliable/unreliable each dimension of the system's perception of the environment maps to the human's linguistic descriptors. The system can then adapt to the situation by applying the learned weights to ground follow-up dialogues and/or adjusting its language grounding models accordingly.

This chapter is organized as follows: we first discuss the motivation of our weight-learning approach to mediate the perceptual differences between agents and humans. We then present the extension to the graph-matching formulation and the optimization based weight-learning method. The empirical evaluation based on a human-robot dialogue dataset is presented next, followed by discussions on the strengths and limitations of the work. Last, we conclude the current work and discuss some future directions.

## 4.1 Motivation

As we have discussed in Chapter 3, computational approaches to referential grounding often consist of two key components [**?**, 32, 26]. The first component addresses formalisms and methods that connect linguistic terms (e.g., *red*, *left*) to the lower level numerical features (e.g., $rgb$ vectors) captured in the agent's representation of the perceived environment. Namely, this is what we called the *semantic grounding functions* in Section 3.2.2. The second component extracts all the linguistic terms from a referring expression and combines their

grounding models together to identify referents. For example, given a referring expression "*the big blue bottle to the left of the red box*", it will recognize that the intended referent has several attributes (e.g., color is blue, size is big, type is bottle) and it is to the left of another object. Then it will combine all relevant sensors' inputs and apply the corresponding semantic grounding models to identify the referent that most likely satisfies the referring expression.

Many previous works on referential grounding have focused on the first component, i.e., exploring how to learn and ground individual linguistic terms to low level physical attributes (e.g., [68, 97, 98]). Although different algorithms have been applied for the second component [99, 100], little attention has been paid to the question how to *intelligently* combine different attributes to ground references. However, this is an important question for situated dialogue since the human and the agent have mismatched representations of the shared environment. For example, the agent may not recognize any bottle, or may see something blue but not a bottle. Furthermore, the agent's perception of blue may be very different from the human's perception of blue. How should the agent utilize these different attributes? What part of its own perception should the agent trust the most when there is potential mismatch with the human's perception?

To address these questions, in this chapter we propose a computational approach that will allow the agent to learn and mediate perceptual differences during situated dialogue. The idea is that, by interacting with its human partner (e.g., through dialogue), the agent should be able to assess its perceptual differences from its human partner. In particular, the agent should be able to learn what dimension(s) of its own perception (e.g., recognition of objects or colors) are more reliable, namely more aligned with the human's perception reflected by the linguistic descriptions. To mediate the perceptual differences, the agent

should use this self-assessment to update its internal models and further improve referential grounding in follow-up communication.

Specifically, we have developed an optimization-based approach to efficiently learn a set of weights, which indicate how reliably/unreliably each dimension of the agent's perception of the environment maps to the human's linguistic descriptions. By simulating different types of mismatched perception, our empirical results have shown the weight-learning approach can successfully adjust the weights to reflect the agent's perceptual capabilities. In addition, the learned weights for specific linguistic terms (e.g., "red", "left") can further trigger automatic model updating for these words, which in turn leads to an improvement of referential grounding performance for future dialogues.

## 4.2 Weight Learning for Situated Referential Grounding

In this section, we first present a general formulation of inexact graph-matching with weighted attributes. The state-space search approach in Chapter 3 can be viewed as a special case of this general formalism. Then we propose an optimization method which utilize this formulation to learn, through the interaction with the human partner, a set of weights that indicates how reliable/unreliable each of the agent's dimensions of perception maps to the human's linguistic descriptions.

### 4.2.1 Inexact Matching with Weighted Attributes

The same as in Section 3.2.2, we first construct two ARG representations based on the language and vision processing outcomes, which are the discourse graph G and the vision graph G′.

$$\begin{aligned} &\mathrm{G} = (\mathrm{X}, \mathrm{E}) \\ &\mathrm{X} = \{x_m \mid m = 1, \ldots, M\} \\ &\mathrm{E} = \{e_i = (e_{i1}, e_{i2}) \mid i = 1, \ldots, I; e_{i1} \in \mathrm{X}, e_{i2} \in \mathrm{X}\} \end{aligned} \tag{4.1}$$

$$\begin{aligned} &\mathrm{G}' = \left(\mathrm{X}', \mathrm{E}'\right) \\ &\mathrm{X}' = \left\{x'_n \mid n = 1, \ldots, N\right\} \\ &\mathrm{E}' = \left\{e'_j = \left(e'_{j1}, e'_{j2}\right) \mid j = 1, \ldots, J; e'_{j1} \in \mathrm{X}', e'_{j2} \in \mathrm{X}'\right\} \end{aligned} \tag{4.2}$$

In an ARG, each node (e.g., $x_m$) is assigned a set of $A$ attributes $\mathrm{U}_m$ and each edge (e.g., $e_i$) is assigned a set of $B$ attributes $\mathrm{V}_i$ ($A$ and $B$ are two predetermined numbers), i.e.,

$$\begin{aligned} \mathrm{U}_m &= \{u_{ma} \mid a = 1, \ldots, A\} \\ \mathrm{V}_i &= \{v_{ib} \mid b = 1, \ldots, B\} \end{aligned} \tag{4.3}$$

Note that, we can also assign a special *UNK* value to an attribute, meaning this attribute is currently "unknown" for this node/edge. For example, when an attribute is not mentioned or the word used to describe it is "out-of-vocabulary", the value of that attribute should be assigned the value *UNK*.

As defined earlier in Section 3.2.2, a matching $\Theta$ between G and G′ is to assign each node $x_m$ in X a "label" $\theta_m$ to indicate which node in X′ that $x_m$ maps to. However, the difference here is that, we now allow $\Theta$ to be an *inexact-matching*. Namely, we first define a

set of node indices as $\breve{\mathrm{M}} = \{1, \ldots, M\}$, and two subsets of $\breve{\mathrm{M}}$ (i.e., $\breve{\mathrm{M}}_1$ and $\breve{\mathrm{M}}_2$) such that $\breve{\mathrm{M}}_1 \subseteq \breve{\mathrm{M}}$ and $\breve{\mathrm{M}}_2 = \breve{\mathrm{M}} - \breve{\mathrm{M}}_1$. And an inexact-matching matching $\Theta$ is then formally defined as:

$$\begin{aligned} \Theta &= \Theta_1 \cup \Theta_2 \\ &= \left\{\theta_{m_1} = x'_n \mid m_1 \in \breve{\mathrm{M}}_1; x'_n \in \mathrm{X}'\right\} \\ &\quad \cup \left\{\theta_{m_2} = \Lambda \mid m_2 \in \breve{\mathrm{M}}_2\right\} \end{aligned} \tag{4.4}$$

where $\Theta_1$ specifies the label/mapping of each of the nodes in X for which a proper mapping can be found in X′, and $\Theta_2$ contains the nodes in X for which **no** proper mapping can be found (i.e., $\Lambda$ denotes a special "NULL" node, thus $\theta_m = \Lambda$ means node $x_m$ is not matched with any node in X′).

Let $\mathrm{X}_{\Theta_1}$ denote the subset of nodes that are labeled in $\Theta_1$ (i.e., $\mathrm{X}_{\Theta_1} = \left\{x_m \mid m \in \breve{\mathrm{M}}_1\right\}$, and $\mathrm{E}_{\Theta_1}$ denotes the set of edges induced by $\mathrm{X}_{\Theta_1}$ (i.e., $\mathrm{E}_{\Theta_1} = \{e_i = (e_{i1}, e_{i2}) \mid e_i \in \mathrm{E}, e_{i1} \in \mathrm{X}_{\Theta_1}, e_{i2} \in \mathrm{X}_{\Theta_1}\}$). The *compatibility function* of an inexact-matching $\Theta$ can then be defined as[2]

$$\begin{aligned} f(\Theta_1) &= \sum_{x_m \in \mathrm{X}_{\Theta_1}} g\left(x_m, x'_n\right) + \sum_{e_i \in \mathrm{E}_{\Theta_1}} h\left(e_i, e'_j\right) \\ f(\Theta_2) &= \eta_1 \left(M - \left|\mathrm{X}_{\Theta_1}\right|\right) + \eta_2 \left[M(M-1) - \left|\mathrm{E}_{\Theta_1}\right|\right] \\ &= \eta_1 \left(M - \left|\mathrm{X}_{\Theta_1}\right|\right) + \eta_2 \left[M(M-1) - \left|\mathrm{X}_{\Theta_1}\right| \left(\left|\mathrm{X}_{\Theta_1}\right| - 1\right)\right] \end{aligned} \tag{4.5}$$

where $x'_n$ and $e'_j$ are the corresponding node and edge of $x_m$ and $e_i$ according to $\Theta_1$, respectively. $\eta_1$ and $\eta_2$ are the parameters for determining whether a node should be mapped to the "NULL" node (i.e., treated as unmatched) or not.

[2]Note that, both G and G′ are now treated as *complete* graphs, in which each ordered pair of nodes are connected by an edge, thus G contains $M(M-1)$ edges. In the discourse graph G, if the relation between two nodes are not mentioned, we just set the attributes of the corresponding edge to *UNK* (i.e., "unknown").

Similar to exact-matching, the optimal matching between G and G$'$ is the one with the highest compatibility score:

$$\Theta^* = \arg\max_{\Theta} f(\Theta) \tag{4.6}$$

and the compatibility function for a matched pair of nodes and edges are based on the attributes assigned to them:

$$\begin{aligned} g(x, x') &= \frac{1}{A} \sum_{a=1}^{A} g_a(u_a, u'_a) \\ h(e, e') &= \frac{1}{B} \sum_{b=1}^{B} h_b(v_b, v'_b) \end{aligned} \tag{4.7}$$

As we have already discussed in Section 3.2.2, $g(x, x')$ and $h(e, e')$ are further decomposed to a set of stand-alone semantic grounding functions. Now we also assign each of these grounding functions a weight $\alpha_{ak}$, representing the importance or reliability of each grounding function for finding the correct matching. For example, suppose we have a set of grounding functions for colors such as "red", "green" and "yellow". If the agent's perception and model of red color is more accurate than other colors, then the grounding function of "red" should be assigned a higher weight. Also note that, when the value of $u_a$ is "unknown" (i.e., not-mentioned or out-of-vocabulary), $g_a(u_a, u'_a)$ just returns the predetermined value $\eta_1$. Thus the decomposition of $g(x, x')$ and $h(e, e')$ now become:

$$g_a(u_a, u'_a) = \begin{cases} \alpha_{a_1} g_{a_1}(u'_a) & \text{if } u_a = y_1; \\ \alpha_{a_2} g_{a_2}(u'_a) & \text{if } u_a = y_2; \\ \vdots & \vdots \\ \alpha_{a_K} g_{a_K}(u'_a) & \text{if } u_a = y_K; \\ \eta_1 & \text{else} \end{cases} \tag{4.8}$$

$$h_b\left(v_b, v_b'\right) = \begin{cases} \beta_{b_1} h_{b_1}\left(v_b'\right) & \text{if } v_b = z_1; \\ \beta_{b_2} h_{b_2}\left(v_b'\right) & \text{if } v_b = z_2; \\ \vdots & \vdots \\ \beta_{b_L} h_{b_L}\left(v_b'\right) & \text{if } v_b = z_L; \\ \eta_2 & \text{else} \end{cases} \tag{4.9}$$

Given this general form of weighted inexact-matching, we can get the special case as what we have used in Section 3.2.2 by (1) let $\eta_1 = \eta_2 = 0$, thus it becomes exact-matching; (2) let all the weights equal to 1, thus each attribute value is treated equally when the compatibility score of a matching is calculated.

We close this section by a discussion on how inexact-matching is controlled by the two parameters $\eta_1$ and $\eta_2$. Basically, inexact-matching means not every node in G has to be matched with a node in G$'$, but some nodes can be unmatched (i.e., $\Theta_2$) if no proper matchings are found for them. Suppose there is only one node in G, and it has only one attribute mentioned (e.g., color is red). For this node to be matched with a node in G$'$, there must be at least one node in G$'$ such that $\alpha_{red} g_{red}\left(u'\right) > \eta_1$, otherwise this node will be matched with $\Lambda$ (i.e., the NULL node). Suppose we further add an edge with one mentioned attribute (e.g., something is to the left of the red object) to G, then in G$'$ there must be at least one node and one relation, such that $\alpha_{red} g_{red}\left(u'\right) + \beta_{leftof} h_{leftof}\left(v'\right) > \eta_1 + \eta_2$, to find a non-NULL matching. In general, for a node $x_m$ to be matched with a non-NULL node $x_n'$, we should have $g\left(x_m, x_n'\right) + \sum_{e_i \in \mathrm{E}x_m} h\left(e_i,\, e_j'\right) > \eta_1 + \left|\mathrm{E}x_m\right| \eta_2$, where $\mathrm{E}x_m$ is the set of edges that are associated with $x_m$.

### 4.2.2 Optimization for Weight Learning

The graph-matching algorithm relies on all these attributes to find out proper matchings and filter improper ones. Importantly, because the agent can have different capacities of perceiving these different attributes and linguistic referring expressions associated with different attributes can also have different levels of ambiguities, different attributes then should not be always treated equally. Thus we develop an optimization based approach to automatically acquire a set of weights based on the matching hypotheses and the ground-truth matching.

The weights (i.e., $\alpha_a$ and $\beta_b$) are the "variables" that we aim to adjust, and our general objective is to maximize the reference grounding performance. We represent all the weights using a vector $\mathbf{w}$:

$$\begin{aligned} \mathbf{w} \quad = [ \quad & \alpha_{11}, \alpha_{12}, \ldots, \alpha_{21}, \alpha_{22}, \ldots, \alpha_{A1}, \alpha_{A2}, \ldots, \\ & \beta_{11}, \beta_{12}, \ldots, \beta_{21}, \beta_{22}, \ldots, \beta_{B1}, \beta_{B2}, \ldots, \\ & \eta_1, \eta_2 \quad ]^T \end{aligned} \tag{4.10}$$

For a given matching $\Theta$, its compatibility score $f(\Theta)$ then becomes a linear function of $\mathbf{w}$:

$$f(\Theta) = f_{\Theta}(\mathbf{w}) = \mathbf{c}_{\Theta}\mathbf{w} \tag{4.11}$$

where $\mathbf{c}_{\Theta}$ is a vector of "coefficients" that are computed from the given $\Theta$ and the grounding functions, based on our previous definitions.

Given two graphs G and G$'$, suppose $\hat{\Theta}$ is the ground-truth matching, and $\Theta_1, \Theta_2, \ldots, \Theta_D$ are the top-$D$ matching hypotheses (i.e., $\Theta_1$ is the top-1 hypothesis and so forth) generated using an initial weights vector $\mathbf{w}_0$. If $\Theta_1 \neq \hat{\Theta}$, we can try to find a new $\mathbf{w}$ that may lead to

a better matching outcome. This can be formulated into an optimization problem as:

$$\begin{aligned} &\max_{\mathbf{w},\varepsilon} \ \mathbf{c}_{\hat{\Theta}}\mathbf{w} - \gamma \sum_{d=1}^{D} \varepsilon_d \\ &\text{s. t.} \\ &\qquad \mathbf{c}_{\Theta_1}\mathbf{w} - \mathbf{c}_{\hat{\Theta}}\mathbf{w} \leq \varepsilon_1,\ \varepsilon_1 \geq 0 \\ &\qquad\qquad \vdots \\ &\qquad \mathbf{c}_{\Theta_D}\mathbf{w} - \mathbf{c}_{\hat{\Theta}}\mathbf{w} \leq \varepsilon_D,\ \varepsilon_D \geq 0 \end{aligned} \tag{4.12}$$

where $\{\varepsilon_d\}$ is a set of "slack" variables to accommodate infeasible constraints, and $\gamma$ is a penalizing factor to push the values of all $\varepsilon_d$ to be small (i.e., the closer to 0 the better).

The essence of this optimization scheme is to find a proper $\mathbf{w}$, based on which the ground-truth matching can be ranked as top as possible among the matching hypotheses generated by the state-space search algorithm. If such optimal values of $\mathbf{w}$ can be found, we expect it can facilitate the search algorithm to select out better matching hypotheses next time when a similar instance is encountered.

For example, if the referring expression is "the red apple" but in the vision graph recognition of apple is not reliable (e.g., pear is always recognized as apple), then it is likely that "red apple" will be wrongly matched with a pear. Through our learning scheme, it can be learned that the weight for "apple" need to be low and the weight for "red" need to be high for the whole expression to be correctly matched. Thus, the learning outcome (i.e., the learned weights) can tell us which attribute-values are more/less reliable for finding the correct matching results.

Actually, the optimization problem we have here is exactly a *linear programming* problem, which can be efficiently solved using algorithms such as the Interior Point methods [101].

## 4.3 Evaluation

### 4.3.1 Data

(a) Situated setup

| | |
|---|---|
| H1: | The green cup is called Bill. |
| R1: | Ok, the green cup is Bill. (point to the inferred object) |
| H2: | Do you see a car? |
| R2: | Yes, I see a car. |
| H3: | Do you see a blue can behind it? |
| R3: | I don't see a blue can. But I see a blue unknown object there. Is that the can you are talking about? (point to the inferred object) |
| H4: | Yes, it's called Mary. |
| R4: | Got it, the blue can is Mary. |

(b) Sample Dialogue

Figure 4.1 An example of situated setup and human-robot dialogue

We use the data collected from earlier experiments on human-robot dialogue to evaluate our approach. Here we first give a brief description of these experiments to help better understand the data[3]. The goal of these experiments was to investigate collaborative effort in human-robot dialogue. In these experiments, a NAO robot (with fully automated

[3]See [102] for the details of this human-robot dialogue experiment.

components for language processing, referential grounding, and dialogue management) was positioned with a human subject to play a set of naming game as shown in Figure 4.1(a). Human subjects were given some secret names for several objects in the shared environment. The task was for the human to communicate these secret names to the robot through dialogue so that the robot would know which object had what name. Because of the nature of this naming game, these experiments led to dialogues focusing on referential communication as shown in an example in Figure 4.1(b).

One controlled condition in the experiments was to simulate different perceptual capabilities of the robot, which resulted in two levels of variations:

- **High-Mismatch** simulated the situation where the human and the robot had a high mismatch in their perceptions of the shared environment. The robot's object-recognition error rate was manipulated to be very high, namely, a large portion (60% or 90%) of the objects were mis-recognized.[4]

- **Low-Mismatch** simulated the situation where the human and the robot had a low mismatch in their perceptions of the shared environment. The robot correctly recognized most of the objects, with only a small portion (10% or 30%) being mis-recognized.

Although the experiment was originally designed for a different purpose, it actually provides an ideal data set for evaluating our weight-learning approach. Since we currently do not address dialogue management, evaluating our algorithm only needs language inputs from the dialogue discourse and the corresponding visual environment. Furthermore, the systematic simulation of mismatched perceptions allows us to evaluate whether the weight-learning

[4]These mis-recognized objects were randomly selected, and their object-recognition results were randomly assigned.

| | Object-specific attribute | | | Spatial relation |
|---|---|---|---|---|
| | Type | Color | Location | |
| Number of expressions | 686 | 747 | 281 | 162 |

Table 4.1 Attributes and relations used in the referring expressions in the human-robot data

outcome is consistent with our expectation given the simulated situation. For example, we would expect the learned weight for the attribute *type* (i.e., the object-recognition result) to be smaller under the high-mismatch condition than under the low-mismatch condition. Therefore, we apply the weight-learning approach on the two subsets of data to see whether the expected weights can be learned. Before we present the results, we first briefly summarize the data.

There were a total of 147 dialogues collected from 24 human subjects. Among these dialogues, 73 were collected under the low-mismatch condition and 74 were under the high-mismatch condition. For each dialogue, the robot's perception of the environment, such as the object-recognition results, the color of each object (represented as a $rgb$ vector) and the position of each object (represented as $x$ and $y$ coordinates), were also logged. It thus provided a dataset with both discourses of human's spontaneous referring expressions and robotic perceptions of the shared environment, which is desired for the evaluation of our approach here.

Table 4.1 summarizes the most frequently used attributes/relations in human subjects' referring expressions[5]. As shown in the table, the *type* (e.g., "the bottle") and *color* (e.g., "the red object") attributes are the most commonly used to describe the referents. This conforms with psycholinguistic findings on people's preference of referring [41]. Human subjects in our experiments always intend to mention these two attributes when they refer

[5]Some other less-frequently used (appeared less than 20 times) descriptors such as *size* are excluded here.

to an object, thus the robot's ability of correctly perceiving the type and color attributes can be crucial for successful referential grounding. Furthermore, spatial location of an object (e.g., "the object in the front") is another commonly used descriptor, which can be utilized by robots equipped with advanced spatial sensors[6].

Besides these object-specific attributes, referring based on spatial relations between two objects is also observed. When one object has already been grounded, it can then serve as a reference point for referring to another object based on their relation. We mainly observed two kinds of relations, which are the projective relations (e.g., "the bottle is to the right of the box") and the relations based on object's proximity (e.g., "it is close to the box"). Although spatial relations can be very useful under certain situations, in general they are less often used than object-based attributes. In the following section, we will evaluate our weight-learning approach based on these three most frequently used object-specific attributes, i.e., *type*, *color*, and *location*.

### 4.3.2 Weight-learning Results on Object-specific Attributes

To evaluate our weight-learning method, we apply it to the dataset of 147 (i.e., low-mismatch and high-mismatch) dialogues to see what weights can be learned and how referential grounding can be potentially improved. Each dialogue in the data is represented by a discourse graph based on automatic language processing, and each discourse graph is also paired with a vision graph that represents the perceived environment in which the dialogue happened[7]. Thus a "training" instance from the data is a pair of a discourse graph and a

[6]What we call the "spatial location" here is actually one type of spatial relation within the *egocentric frame-of-reference* [59], i.e., using the viewer's own field-of-view as the reference frame. Here we treat it as an object-specific property other than a relation because it does not involve another object.

[7]The vision graphs are generated from the logged robotic vision information during the experiment.

vision graph. We then apply our weight learning method as describe in Section 4.2 to each training instance to learn a set of weights for the type, color, and location attributes. We evaluate the weight-learning at two levels, i.e., the attribute-level and the word-level (discussed later in Section 4.3.3). When learning weight at the attribute-level, we simplify the weight learning by letting $\alpha_{a_1} = \alpha_{a_2} = \cdots = \alpha_{a_K}$ and $\beta_{b_1} = \beta_{b_2} = \cdots = \beta_{b_L}$, thus we only learn one weight for each attribute. Note that, in the current evaluation, we let $\eta_1 = \eta_2 = 0$, thus only exact-matching is performed here.

Although not exactly the same, our weight learning procedure follows the online learning paradigm [103] in real interaction scenario. Given an instance (i.e., dialogue), the "learner" first generates predicted outcome (i.e., grounding results); and then it receives the true outcome for this instance[8], based on which its internal model (i.e., the weights) is updated for making better future predictions. Therefore, we conduct the learning and evaluation in a simulated online fashion to investigate how effectively and efficiently it can adapt to the situation of mismatch perceptions. In each simulation run, the weight-learner receives a sequence of 20 training dialogues that are randomly picked from the dataset. Upon receiving each training instance, it first generates a list of grounding hypotheses using the same state-space search algorithm as discussed in Section 3.2.2 based on the current weights. After the grounding hypotheses have been generated, the ground-truth is then considered given, and a new set of weights is learned using our optimization method. With the learned new weights,

[8]When learning the weights of attributes, we assume that the ground-truth mapping between the linguistic entities (i.e., discourse graph nodes) and the physical objects (i.e., vision graph nodes) is already known. Although in the original experiment, due to various kinds of errors and the time constraint, some entities could be incorrectly grounded or not grounded at all by the end of a dialogue. Here we just assume that the ground-truth mappings can always be correctly established, e.g., through endless interactions.

| | Type | Color | Location |
|---|---|---|---|
| Low-mismatch | 0.87 | 0.97 | 0.97 |
| High-mismatch | 0.45 | 0.9 | 0.71 |

Table 4.2 The final weights learned after 20 training dialogues (averaged over 100 runs of simulation)

we then updated the current weights as[9]:

$$w_t = w_{t-1} + \gamma(w_{new} - w_{t-1})$$

and move to the next instance in the training sequence. Besides, to evaluate the effect of weight-learning on referential grounding, after each training dialogue we also apply the current learned weights to ground all the other 53/54 dialogues that are not selected as training data. Such a procedure repeats itself till the entire training sequence has been gone through.

We started with uniform weights (i.e., all being 1), and repeated the weight learning process throughout the sequence of the selected 20 training dialogues. Table 4.2 summarizes the final learned weights on the low-mismatch and high-mismatch data after going through the 20 training dialogues[10]. As we can see in Table 4.2, the most significant change from the low-mismatch condition to the high-mismatch condition is the drop of the learned weight for the type attribute (0.87 vs. 0.45). This is consistent with the situation (i.e., low-mismatch vs. high-mismatch) from which the data was collected. To further demonstrate the weight-learning efficiency, we plot the updated weight of the type attribute after each training dialogue, as shown in Figure 4.2(a). It shows that when the robot's object-recognition is

[9] $\gamma$ is a step-size parameter set to be 0.5.

[10] We have run the simulation 100 times, and the weights shown in Table 4.2 are the average over the 100 runs.

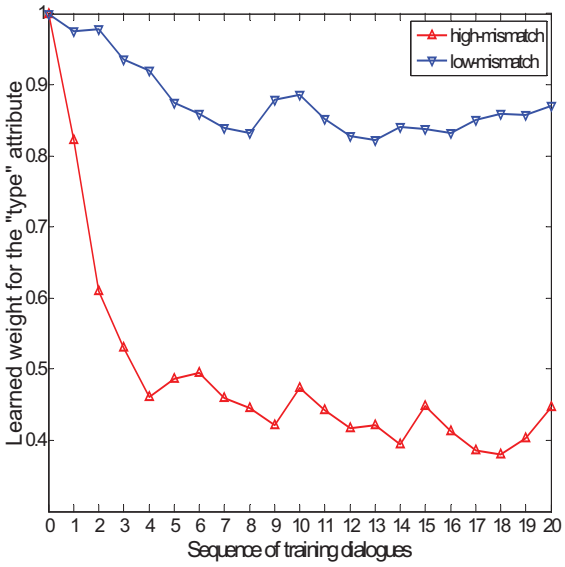
significantly mismatched with the human's perception (i.e, the high-mismatch condition), the weight for the type attribute quickly descends in the first 5 training dialogues, and after that it starts to become stable and gradually converges to the final value. The results thus demonstrate that our weight-learning approach can efficiently learn informative weights to indicate the unreliable attributes of the robot's perception.

Besides the type attribute, the learned weights of the other attributes also indicate how reliable they are for referential grounding[11]. The color attribute appears to be a reliable information source here, i.e., the color perception and grounding models are compatible with human descriptions. The spatial location attribute is less reliable compared to the color attribute, although the robot's perception of spatial information ought to be accurate. This is possibly due to the vagueness of spatial expressions themselves, since a spatial expression such as "object in the front" can often result in several objects that all conform with the description and thus difficult to resolve based on the spatial information alone.

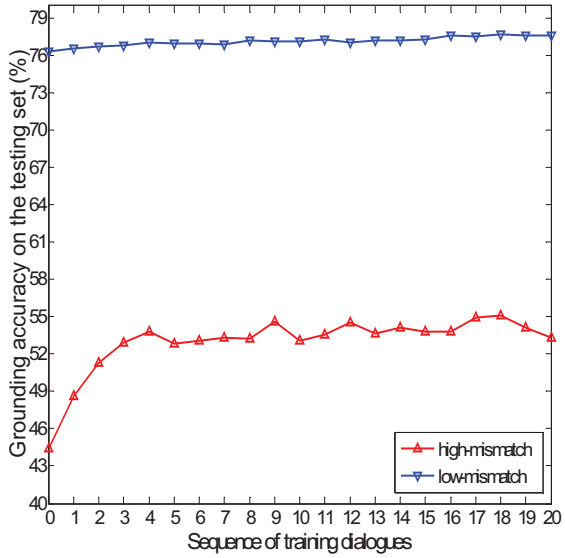
These learned weights not only indicate the robot's perceptual capabilities, but can also improve the referential grounding accuracy when applied to subsequent dialogues. To demonstrate this, we use all the remaining dialogues (i.e., those not selected as training dialogues) as the testing set[12]. After each training dialogue, we applied the current learned weights to generate referential grounding results on all the testing dialogues. The results (averaged referential grounding accuracies on the testing dialogues) are shown in Figure 4.2(b). Under the low-mismatch situation, applying the learned weights does not significantly change the grounding accuracy. This is because the learned weights are close to the initial value

[11]Note that, the weights learned from the high-mismatch data can be more informative, because our weight-learning method only updates the weights when referential grounding results are not aligned with the ground-truth, which is more often the case under the high-mismatch situation.

[12]There are 53 and 54 testing dialogues for the low-mismatch and high-mismatch conditions, respectively.




(a) Learned weights for the "type" attribute




(b) Referential grounding accuracies on the testing set by applying the learned weights

Figure 4.2 weight-learning and referential grounding results after each training dialogue (averaged over 100 runs)

(i.e., 1.0) as all the attributes were reasonably reliable. Under the high-mismatch situation, using the learned weights can improve grounding accuracy by 9.4% (from 44.4% to 53.8%) within the first 5 training dialogues. After that the grounding accuracy stays stable since

the learned weights also become stable as shown in Figure 4.2(a). Therefore the robot can reduce referential grounding errors by assigning lower weights to those unreliable attributes, once they have been learned by the weight-learning approach.

### 4.3.3 Weight-learning Results at Word Level

Besides learning weights at the attribute level as we just discussed, it will be even more useful if the weights can be learned at a lower level, i.e., to learn a weight for each of the words that are used to describe an attribute. The learned weight then indicates the reliability of the robot's perception and/or grounding model on that specific word. For example, if the robot can learn that its perception of "red" is unreliable, it can then adjust the grounding model for this specific word accordingly. This would be more useful than only knowing the overall reliability of the color attribute.

To enable learning weights at the "word level", instead of assigning only one weight for an attribute (i.e., all the words that describe one attribute always share the same weight), we need to assign each word's grounding function a unique weight (as described in Section 4.2). The same weight learning approach can then be applied to learn how well the robot's perception is aligned with the human's description for each specific word. To evaluate word level weight-learning, we again use systematic simulation of perceptual errors which allow us to easily assess whether expected weights can be learned given the simulated situation. Specifically, we use the low-mismatch data[13] and modify the robot's perception to simulate some common errors that can happen in a real situation. The modifications we make are:

- For each object's perceived color (i.e., an $rgb$ vector), we increase the intensity of the

[13]Since the low-mismatch data contain few original errors, it would be easier to see the effect of simulated errors here.

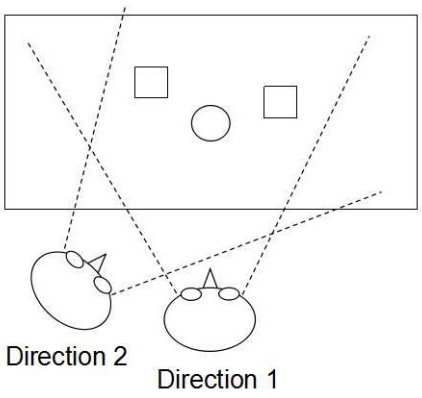



Figure 4.3 An illustration of the different viewing directions. Direction 1 is the robot's original viewing direction in the experiments, which is the same as the human subject's. Direction 2 is the misaligned viewing direction simulated by decreasing the $x$ coordinates from direction 1. Because of the narrow viewing range of the robot's camera, all the objects become closer to the left edge of its field-of-view (i.e., their $x$ coordinates decrease) under direction 2.

| Color | "orange" | "pink" | "green" | "blue" |
|---|---|---|---|---|
| Without simulated errors | 0.92 | 0.95 | 0.9 | 0.95 |
| With simulated errors | 0.47 | 0.59 | 0.88 | 0.97 |
| Location | "left" | "right" | "front" | "back" |
| Without simulated errors | 0.95 | 0.81 | 0.97 | 0.95 |
| With simulated errors | 0.19 | 0.73 | 0.77 | 0.73 |

Table 4.3 Final learned weights for some common words of the two attributes after 20 training dialogues (averaged over 100 runs of simulation)

$r$ and $g$ channels (by 100)[14] and decrease the intensity of the $b$ channel (by 100). This is to simulate color-sensing error, e.g., due to environmental lighting noise or deficient color sensor.

- For each object's perceived position (i.e., $x$ and $y$ coordinates), we decrease the $x$ coordinate (by 300 pixels)[15]. This is to simulate a different viewing direction that may affect the robot's interpretation of spatial language such as "*on the left*", as illustrated in Figure 4.3.

[14]The range of the intensity of each channel is from 0 to 255.

[15]The size of the image produced by the robot's camera is $1280 \times 960$ pixels.

With these simulated perceptual errors, we then use the same online learning scenario to evaluate the effectiveness and efficiency of weight-learning at the word level. Namely, we randomly select 20 dialogues as the training sequence and go through these dialogues one-by-one to learn and update the weights. Table 4.3 summarizes the final learned weights of some common words of the type and location attributes after 20 randomly selected training dialogues. In the table we also show the learned weights from the situation that no errors were simulated, so that the weights learned with simulated errors can be compared. As we can see from Table 4.3, there are clear correspondences between the learned weights and the simulated errors in the robot's perception:

- For the color attribute, the learned weights indicate that the robot's grounding of "orange" and "pink" is affected by the simulated error of the inflated $r$ and $g$ channel intensities and the deflated $b$ channel intensity.

- For the location attribute, the very low weight learned for "left" indicates the robot's problematic interpretation of this concept, which corresponds to the simulated error of shifting all the perceived objects to the left side.

These correspondences between the learned weights and the underlying perceptual errors again demonstrate that our weight-learning approach is capable of learning informative weights, which indicate how reliably the robot's perception maps onto the human's linguistic descriptions. For the efficiency of weight-learning at word level, it is also the same as the previous attribute-level weight learning. Figure 4.4 shows the plots of updated weights for the words "orange", "pink", and "left", after each training dialogue during the online weight-learning process. It took only 5 training dialogues for the weight of "orange" to land on its final value. The weight of "pink" and "left" took some more training dialogues to

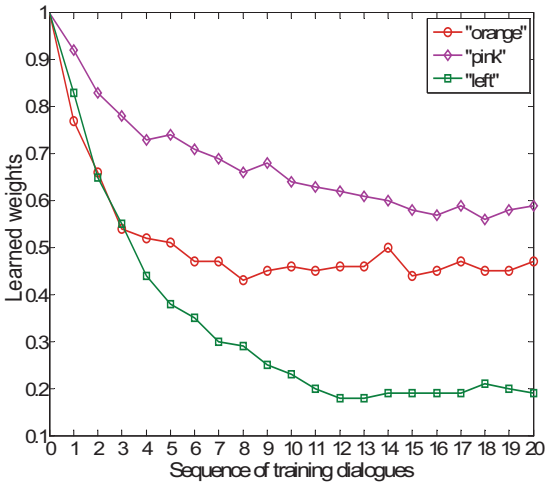


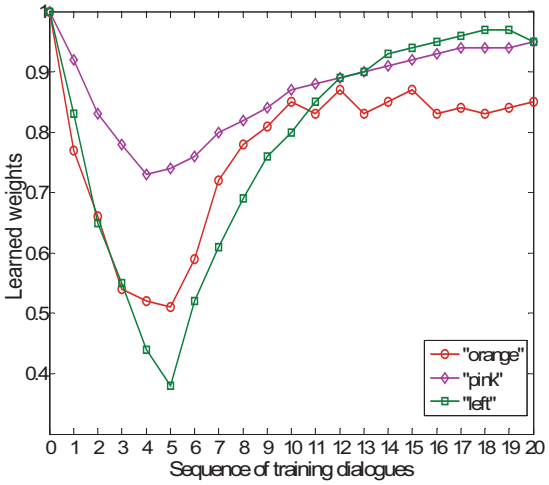

Figure 4.4 Word level weight-learning evaluation results after each training dialogue (averaged over 100 runs of simulation)

converge to the final values because they did not appear as often as the word "orange" in the data.

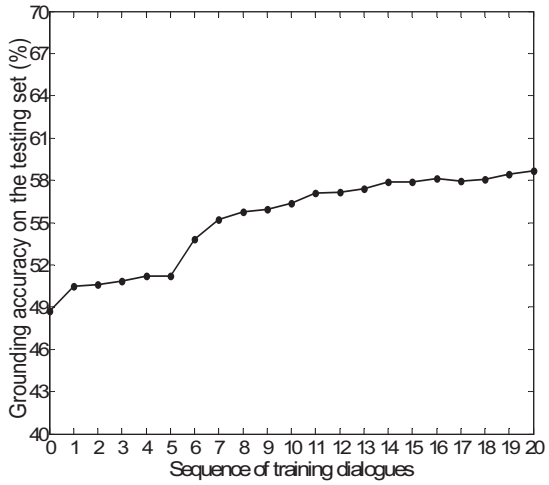
In a realistic setting, a more dynamic interaction and learning process would be expected. For example, the robot could adjust and improve its perceptual capabilities dynamically, based on the interaction with the human. Thus we further simulate one such dynamic process to see how our weight-learning responded to it. We still use the same data (i.e., low-mismatch data with simulated perceptual errors) and the online weight-learning process, but add a model-updating process after the first 5 training dialogues. This is to simulate the scenario that the robot automatically start to adjust its language grounding models for the unreliable words with low learned weights.

Initially, the grounding models for color and location terms were all defined as Gaussian distributions over the corresponding visual features [68]. To update the grounding models for the two color words (i.e., "orange" and "pink"), we follow the online word model acquisition approach as described in [104], which essentially keeps updating the mean of the Gaussian

distribution by averaging out the old mean and the new observed values. In addition to updating model parameters, the underlying models can be adjusted as well. For instance, for the word "left", the robot can switch from the Gaussian model to the exponential decay function model as in [53].




(a) Learned weights with model updating




(b) Referential grounding accuracies with model updating

Figure 4.5 Word level weight-learning results with model updating (averaged over 100 runs of simulation)

Figure 4.5(a) shows the plots of learned weights for these three words with the model

updating process. After adaptation, the weights for "orange", "pink", and "left" all become high values, indicating their grounding models became more consistent with the human's descriptions. We also evaluate referential grounding accuracy on the testing dialogues as we did earlier, but using both the updated models and weights after each training dialogue. Figure 4.5(b) further shows that the model updating process is able to improve the performance of the language grounding models which were previously affected by the simulated perceptual errors. Referential grounding accuracy is improve by 10% (from 48.7% to 58.7%) with the updated models and weights, compared to the initial state of using the original models and uniform weights. These results again demonstrate that our weight-learning approach can efficiently update the weights which consistently reflect the underlying changes of the robot's perceptual and language grounding capabilities.

As we can also see in the simulated model updating process, the robot's perceptual errors can be accommodated by adjusting language grounding models or switching to a more robust model. For color, because of the simulated perceptual errors (i.e., inflated $r$ and $g$ channel intensity and deflated $b$ channel intensity), the original grounding models became inaccurate and misleading. For example, a blue object was likely to be perceived as pink, thus it would lead to incorrect matching result. Such errors can be accommodated by re-calibrating the parameters of the color grounding models, so that the models are more aligned with the current environmental or robot's sensory condition, as demonstrated by our simulated model updating process (also see [104]). Our weight-learning approach can provide useful guidance to the model adjusting effort by indicating the models of which words are unreliable and thus need to adjusted.

For the grounding model of spatial descriptors like "left", we have two models (i.e., a Gaussian model and a exponential decay function model) and we switch from one to the

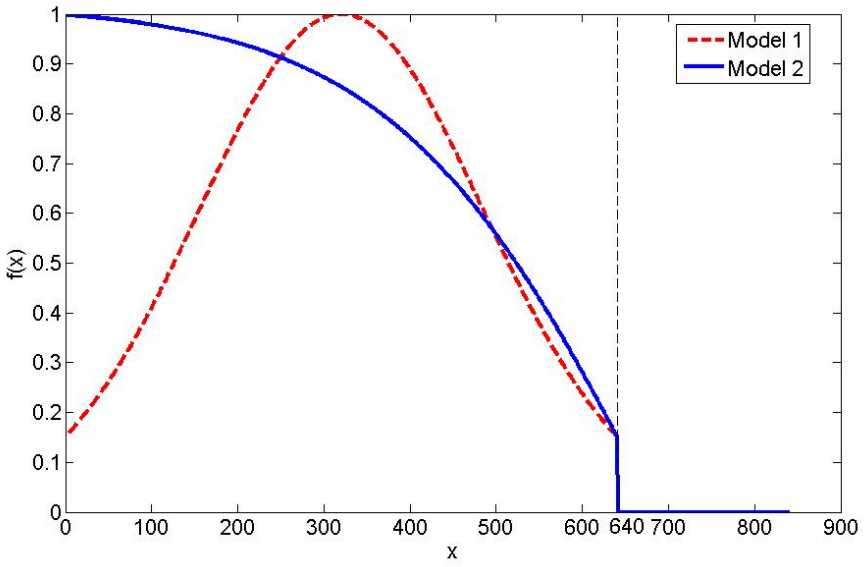



Figure 4.6 Two grounding models for "left". I.e., a grounding model here is a scoring function w.r.t. the $x$-coordinate of an object's center of mass. It measures how well an object's location conforms the descriptor "left". Model 1 is the Gaussian model; Model 2 is the exponential decay function model.

other in the model updating scenario. As indicated by the weight-learning outcome, the exponential model is more reliable than the Gaussian model. Figure 4.6 shows the two models for "left" (Model 1 is the Gaussian model and Model 2 is the exponential model. Both are defined as functions w.r.t. the $x$-coordinate of the center of mass of an object). The Gaussian model assigns higher scores to the central area of the left region in the image, whereas the exponential model assigns higher scores to the area closer to the left edge.

The exponential model is actually more robust against the "misaligned view" error (e.g., all the objects are shifted towards the left side as in our simulation). For example, assume that the original coordinates for object $o_1$ is $x_1 = 350$ and for object $o_2$ is $x_2 = 700$, and the coordinates changed to $x_1' = 50$ and $x_2' = 400$ due to the perceptual error as in our simulation. Although both models now incorrectly assign non-zero scores to $o_2$ of being "on the left", model 2 still assigns a higher score to $o_1$ and thus prefers $o_1$ as a better match than

$o_2$. As demonstrated by the case of switching spatial models, when the robot has different models to choose and needs to determine which one is better under the current situation, weight-learning again provides important guidance.

One limitation of the current weight-learning approach is it only learns about false-positive errors, but not false-negative errors. For instance, if the robot's object-recognition constantly mis-recognize bottles as "boxes", a low weight can be learned for the word "box" but not "bottle". Namely, when the grounding model of "box" often produced false-positive errors, the approach learned a low weight to indicate the problem. But it can not respond to the false-negative errors (i.e., failed to recognize a bottle). Similarly, low weights were learned for "pink" and "orange" but not for "blue" when blue color was likely to be mis-recognized as pink or orange, and a low weight was learned for "left" but not for "right" when objects on the right were likely to be mis-detected as on the left. How to develop a systematic approach that can learn from both false-positive and false-negative errors is an interesting topic for future work.

## 4.4 Conclusion and Discussion

Towards enabling robust and adaptive human-robot dialogue, we have developed an optimization based weight-learning method that can mediate the perceptual differences between a robot and its human partner for referential grounding. Since audio perception (i.e., speech recognition) related issues have been addressed before (e.g., [21]), we focus on the robot's visual perception of the physical environment here.

As demonstrated by our empirical evaluations, our weight-adjusting mechanism is capable of learning informative weight values that reflect the alignment or misalignment between

the robotic visual perception and the human's description of the shared physical environment. Our method can efficiently and reliably adapt to new situations through just a small number of interactions, as soon as the change occurs. The learned weights can be applied to referential grounding and/or word grounding model learning algorithms to improve the referential grounding performance. They can also be utilized by referring expression generation algorithms (e.g., [105, 106]) to facilitate referential communication between robots and humans.

While many previous works mainly focused on learning semantic grounding models. The work presented in this chapter addresses language grounding from a different angle. The focus here is on assessing and adapting existing semantic grounding models for a given situation. The reliabilities of the semantic grounding models can be affected by many situational factors, such as noisy environment, faulty sensors, and human speakers individual differences. Thus, even previously well-performed models can become unreliable in a different environment. We address this challenge by proposing a weight-learning mechanism that quickly learns a set of weights to indicate the reliabilities of the semantic grounding models under the current situation.

For example, suppose a color-blind person speaks to the robot and swaps red and green. In this case, our approach can quickly learn new weights for color words, indicating the inconsistency between the current color models and this specific speaker. It will allow the robot to discount the color terms (since they are not reliable for this user) and rely on other more reliably terms (e.g., spatial terms) to ground linguistic expressions to the environment. When the system turns to a normal person again, it can also quickly update the weights back to normal values, even if the weights have been offset by the color-blind person. Our approach can learn and remember specific weights for each specific person/situation in an

online manner through a couple of training dialogues as demonstrated by the evaluation results.

There are several interesting future directions that can be further explored. First, our current evaluation is based on several simplifications, including the simulation of perceptual errors and the strong assumption that the correct grounding information can be provided to the robot through dialogue with a human. Using simulated visual perceptual errors enables us to do controlled evaluations (e.g., manipulating color and spatial perception in a systematic way) without concerning about the effects of many nuisance factors that may occur in real-time physical setting. It allows us to focus on evaluating whether the algorithm performs the way expected (e.g., when the color perception changes whether the algorithm correctly responds to such change). Now we have evaluated the algorithm using the simplified setting, the next step is to integrate this algorithm into our dialogue system, and evaluate our algorithm with actual perceptual errors from real environments (e.g., varying lighting conditions, camera angles, etc.). Besides, the assumption that the ground truth is provided at the end of dialogue is certainly a strong assumption, which points to the importance of engaging humans in providing feedback to the robot during dialogue. This again calls for sophisticated and integrated models to support collaborative referential communication between humans and agents.

Second, we have only evaluated exact graph-matching results so far. To enable more robust situated referential grounding, learning useful values of parameter $\eta_1$ and $\eta_2$ for controlling inexact-matching should be further explored. The role of $\eta_1$ and $\eta_2$ in the matching process is like two "threshold" values that will determine whether a node can be found a proper matching or not. Inexact-matching can signal the problematic situation (by assigning a node the "NULL" label) in which a discourse graph node's matching compatibility is

too low with any node in the vision graph. Such situation can happen due to errors from both language and vision processing. For example, one situation is there can be a "missing object" which the agent cannot see at all due to CV segmentation error. In such case, only inexact-matching can produce a proper matching, i.e., to match human's description on a missing object to NULL. In general, allowing NULL-matching with inexact-matching makes referential grounding more flexible and thus provides the dialogue manager better choices to deal with problematic situations.

Last but not least, it is important to explore a broader range of information and knowledge that can be dynamically acquired to further mediate perceptual differences between the human and the agent. For example, it can be very helpful to learn context-dependent models for language interpretation and grounding, or to acquire knowledge about out-of-vocabulary words. Furthermore, it is also important to investigate how to utilize different kinds of machine learning mechanisms (e.g., online learning, reinforcement learning, or unsupervised learning) to better serve the purposes of supporting *situated interaction* in the open world and *long-time relationship* between agents and their human users.

# Chapter 5

# A Probabilistic Labeling Approach

The graph-matching approach for referential grounding we have discussed so far is based on the state-space search algorithm to obtain proper grounding results. Although it has made meaningful progress in addressing collaborative referential grounding under mismatched perceptions as demonstrated in Chapter 3 and 4, the state-space search based approach has two major limitations. First, it is neither flexible to obtain multiple grounding hypotheses, nor flexible to incorporate different hypotheses incrementally for follow-up grounding. Second, the search algorithm tends to have a high time complexity for optimal solutions. Thus, the state-space search based approach is not ideal for collaborative and incremental dialogue systems that interact with human users in real time.

To address these limitations, this chapter presents a new approach to referential grounding based on probabilistic labeling.[1] This approach aims to integrate different types of evidence from the collaborative referential communication discourse into a unified probabilistic scheme. It is formulated under the Bayesian reasoning framework to easily support incorporation and generation of multiple grounding hypotheses for follow-up processes. Our empirical results have shown that the probabilistic labeling approach significantly outperforms the state-space search approach in both grounding accuracy and efficiency.

In this chapter, we first discuss some motivations for developing a probabilistic approach

[1] This chapter is based on the following publication:
C. Liu, L. She, R. Fang, and Y. J. Chai, "Probabilistic labeling for efficient referential grounding based on collaborative discourse," in *Proceedings of the 52nd ACL conference*, pp. 13–18, 2014.

to situated referential grounding. We then give a detailed account on the key limitation of the state-space search based approach. The technical details of the probabilistic labeling based approach is presented next, followed by the evaluation of this new approach. Lastly, we conclude this section with some discussions on future directions.

## 5.1 Motivation

As we have already discussed in Chapter 1, one key challenge for referential communication in situated settings is the mismatched capabilities between the human and the agent to perceive and reason about the shared environment. Because of the mismatched capabilities, the human's and the agent's knowledge and representations of the shared world can be significantly different. Object-specific properties such as object-class, color and shape now become unreliable and insufficient for communicating the intended objects. Referential communication thus becomes more difficult and extra efforts have to be made to mediate between the mismatched perceptions [107].

To overcome this challenging problem, collaborating through extensive dialogue between the human and the agent becomes an important strategy [17, 108, 26, 27]. As demonstrated by our experiments using simulated mismatched perceptions in Chapter 1, there are rich dynamics that can be observed from human partners' referential grounding dialogues. Such dialogue always unfolds as a well-formed structure of *presentation* and *acceptance* phases contributed by both partners.

For example, speakers often refer to the intended object(s) through multiple "episodes" of actions, such as an initial *installment* followed by further *refashioning* (e.g., *expansion*, *replacement* or *repair*) based on the hearer's immediate feedback. The hearer, on the other

hand, can also refashion what the speaker just said in a *relevant next turn.*[2] It is through utilizing collaborative dialogues, human partners can often succeed in referential communication even under mismatched perceptions. Therefore, it is important that a situated dialogue system can also capture and utilize the dialogue dynamics and engage in the dialogue collaboratively.

Besides, to build robust situated dialogue systems, a computational framework should be able to handle uncertainties and errors that can arise from every step of the its decision making process. For instance, these uncertainties and errors can come from:

- Speech and language processing, such as acoustic speech recognition, parsing, and coreference resolution.
- Computer vision, such as segmentation, object recognition, and feature extraction.
- Vagueness/flexibility of human language, for example, how to map language to numeric metrics.
- Individual differences among human users, for example, differences in personal experiences, differences in language using habits.
- Situational/environmental changes, that can often affect language and perception from time to time.

Therefore, a computational approach to situated referential grounding needs to fuse the uncertainties from different sources to generate and evaluate the most likely grounding hypotheses. It is desirable that the approach is based on a probabilistic framework, under which the information from different sources can be incorporated and reasoned in a unified

[2]See the discussions in Section 1.2.2 and Section 1.2.3 on these collaborative referring patterns.

and principled manner. As the information from different sources is often interrelated (e.g., the dialogue content is always related to the surrounding environment in situated dialogue), incorporating the information from one source into the processing of another information source can potentially reduce the uncertainties and errors in the latter (e.g., vision processing results can help with language processing, and vice versa). A unified scheme will make it much easier to share information between different components and integrate them in a principled way to make better decisions.

## 5.2 Limitation of the State-Space Search Based Approach

In Chapter 3, we introduced the ARG (Attributed Relational Graph) based model and the state-space search algorithm for situated referential grounding. It showed some encouraging results as a first step towards modeling and interpreting episodic and collaborative dialogues of situated referential communication. However, there are two major limitations:

(1) The state-space search based graph-matching algorithm can only produce "hard" and "one-to-one" matching results (see the later discussion in this section for more details). We believe such an algorithm is not adequate to produce meaningful multiple grounding hypotheses, which is the key for generating collaborative responses. In this chapter, we present a *probabilistic relaxation labeling* algorithm [109], which permits "soft" and "many-to-one" graph matching outcome. It significantly outperforms the state-space search algorithm in generating multiple matching hypotheses.

(2) The evaluation in Chapter 3 was not based on automatic dialogue processing and the

graph representations were built based on manually annotated information. This makes the overall referential grounding problem much easier and unrealistic, since the uncertainties arise from automatic dialogue processing is often one key challenge that a real system has to deal with. In this chapter, we present the work of building an end-to-end system using all automatic components. We especially tackle the issue of handling the uncertainties of automatic dialogue processing through the probabilistic approach.

Here we give a detailed account on the key limitation of the state-space search algorithm to referential grounding, i.e., it only supports "hard" (or binary) matching between two graphs. More specifically, if we organize the matching $\Theta$ that we have defined in Section 3.2.2 in the form of a matching matrix as:

$$\Theta = \begin{array}{c|cccc} & x'_1 & x'_2 & \cdots & x'_N \\ x_1 & p_{11} & p_{12} & \cdots & p_{1N} \\ x_2 & p_{21} & p_{22} & \cdots & p_{2N} \\ \vdots & \vdots & \vdots & & \vdots \\ x_M & p_{M1} & p_{M2} & \cdots & p_{MN} \end{array}$$

Namely, $\Theta$ now becomes a $M \times N$ matrix and each element $p_{mn}$ in this matrix can be viewed as $P\left(\theta_m = x'_n\right)$, i.e., the probability of matching node $x_m$ with node $x'_n$. The state-space search algorithm only supports binary matching because it produces matching matrices in which each $p_{mn}$ can only be either 1 or 0, as illustrated in the following example:

$$\Theta = \begin{array}{c|cccc} & x'_1 & x'_2 & \cdots & x'_N \\ \hline x_1 & 1 & 0 & \cdots & 0 \\ x_2 & 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \vdots & & \vdots \\ x_M & 0 & 1 & \cdots & 0 \end{array}$$

Thus, for the state-space search based matching, only binary decisions can be made on whether a node $x_m$ is matched with node $x'_n$ or not. Such a binary matching nature makes it difficult to incorporate multiple hypotheses from other components and to generate multiple matching hypotheses.

For instance, machine-learning based coreference resolution often produces multiple hypotheses that are ranked by probabilities. For the state-space search based graph-matching, the only way to handle multiple coreference hypotheses is to keep different "versions" of the discourse graph, each of which represented one coreference hypothesis (e.g., node $x_4$ should be merged with $x_3$ or with $x_1$ needs to be represented by two different versions of a discourse graph). Furthermore, state-space search is also inadequate for generating multiple matching hypotheses for each individual node. For example, if there are $M$ discourse entities, state-space search will need to generate at least $3^M$ states to produce 3 different grounding hypotheses for each entity. However, this is not computationally feasible.

To address these key limitations, we have developed a probabilistic approach that handles the uncertainties of automatic dialogue processing and produces "soft" grounding results. Specifically, we extend the ARG model to capture not only the discourse of referring expressions, but also the dialogue dynamics in situated referential communication. To cope with the uncertainties that arise from mismatched perceptions and other sources (e.g., language

and dialogue processing), we employ a probabilistic graph-matching algorithm to combine different sources of information and produce multiple reference grounding hypotheses.

The empirical evaluation results demonstrate that the probabilistic graph-matching based approach is capable to incorporate dialogue processing uncertainties and generate multiple grounding hypotheses, whereas the state-space search based approach fails to do so. It outperforms the state-space search algorithm in both grounding accuracy and efficiency. Thus, our probabilistic graph-matching based approach provides a better framework for interpreting the unfolding dialogue in real time and generating collaborative agent's responses in situated referential communication.

## 5.3 Probabilistic Labeling for Referential Grounding

In this section, we first describe the automatic language processing components we use to build an end-to-end referential grounding system. Then the procedure of the probabilistic labeling algorithm and how the outcomes of different components are integrated under its unified scheme are presented in detail.

### 5.3.1 Automatic Language Processing

One necessary step in building the discourse graph is the semantic parsing of human utterances. As shown in the examples in Section 1.2.2, these utterances are often informal, fragmented, and contain various types of disfluencies. Therefore we apply a partial parsing approach based on *Combinatory Categorical Grammar* (CCG) [110]. We have defined a set of basic CCG lexicon rules, which covers key expressions in our domain, such as describing object properties, spatial relations, and so forth. Given a human utterance, the CCG

parser [111] searches for the longest word sequences (i.e., chunks) covered by the basic CCG grammar iteratively until the end of the utterance. A semantic interpretation (represented in logic forms) is then generated for each chunk.

For example, given the utterance *"ok then to the right of the green object I have got a giant coffee cup"*, the parser can extract all key chunks. It generates the semantic representation $Rightof(x, y) \wedge Color(y, Green)$ for the first chunk *"to the right of the green object"*, and the semantic representation $Size(z, Giant) \wedge Isa(z, Cup)$ for the second chunk "*a giant coffee cup*". The semantic representations of these chunks will be further combined to form the final representation as following:

$$[x, y], [Size(x, Giant) \wedge Isa(x, Cup) \wedge$$
$$Color(y, Green) \wedge RightOf(x, y)]$$

This final semantic representation is in the form of the *Discourse Representation Structure* (DRS) [112], which contains a list of discourse entities introduced by the utterance, and a list of first-order-logic predicates specifying the properties and relations between these entities. Then for each discourse entity, a node is added to the discourse graph. Unary predicates become the attributes of corresponding nodes, and binary predicates become the attributes of corresponding edges in the graph. Group-based descriptions are analyzed in the same way and incorporated into the graphs as hyperedges (see Section 3.2.1 for details).

To assess the performance of automatic semantic parsing on our data, we compared the parsing outcome with manually annotated semantics of all the utterances. Our partial parser can achieve 66.95% recall and 74.4% precision in extracting the correct first-order-logic predicates from the utterances in our data.

### 5.3.2 Tracking Dialogue Dynamics via Coreference

As we have already discussed in Section 1.2.2, situated referential communication is often a highly incremental and collaborative process between two dialogue participants. Because of the incremental and collaborative nature of these dialogues, keeping track of the dialogue dynamics can be important for correctly grounding all the interrelated referring expressions in the dialogue.

Based on the observations from our data, we simplify tracking the dynamics of situated referential communication as a *coreference resolution* problem. We now use the same examples in in Section 1.2.2 to illustrate the idea (For the convenience of reading, here we show the two examples from Section 1.2.2 again):

Example 1:

$D$: what I am seeing on my screen are three apples (1)

$D$: and there is an apple that is directly below, slightly to the right of the battery (2)

$M$: ok (3)

$D$: and then there is an apple to the right of that (4)

$D$: and there is an apple below that last apple (5)

$M$: ok (6)

$D$: so the apple directly below the battery is called Alexis (7)

$M$: ok, this is Alexis (8)

$D$: and then to the right of Alexis is an apple (9)

$D$: and then below that apple is (10)

$D$: I am sorry, actually that is a red pear (11)

$D$: but it looks like an apple (12)

. . . . . .

Example 2:

$D$: there is basically a cluster of four objects in the upper left, do you see that (1)

$M$: yes (2)

$D$: ok, so the one in the corner is a blue cup (3)

$M$: I see there is a square, but fine, it is blue (4)

$D$: alright, I will just go with that, so and then right under that is a yellow pepper (5)

$M$: ok, I see apple but orangish yellow (6)

$D$: ok, so that yellow pepper is named Brittany (7)

$M$: uh, the bottom left of those four? Because I do see a yellow pepper in the upper right (8)

$D$: the upper right of the four of them? (9)

$M$: yes (10)

$D$: ok, so that is basically the one to the right of the blue cup (11)

$M$: yeah (12)

$D$: that is actually an apple (13)

$D$: that is a green apple and it is named Ashley (14)

. . . . . .

In Example 1, for instance, utterance (4) expands utterance (2) by adding new information about a spatial relation ("right of") between two referents. Such a relation can be very informative for identifying the referred objects, but first we need to link this new piece of information to what we already have. Here the pronoun "that" in utterance (4) serves as the key: since we can infer this "that" corefers to the phrase "an apple" in utterance (2), then we can update the snapshot of the current dialogue to be something like "*apple*_1 is

below and to the right of *battery_1*, *apple_2* is to the right of *apple_1*". With such updated information, we can further search through the environment to find objects that conform to the described situation.

Example 2 demonstrates another interesting collaborative pattern. In utterance (4), the matcher proactively describes what he perceives (i.e. "a blue square"). Then in utterance (5) the director confirmatively accepts this presentation and further expands it by introducing a new object that is "under" the matcher's squarish object. An utterance like (5) is called a *relevant next turn* [16], which is often an effective way of moving the dialogue forward. From the matcher's point of view, utterance (5) also can be easier to ground since it is built on what has already been jointly accepted. He now only needs to look at the area below his blue square to find the next referent (i.e. "a yellow pepper").[3] Notice that, this again relies on correctly linking the second "that" in utterance (5) to "a blue square" in utterance (4).

As we can see from these two examples, tracking the dialogue dynamics for the referential grounding purpose boils down to the *coreference resolution* problem (e.g., [113]). Namely, we want to infer whether a referring expression introduces a new referent (i.e., an object that is not mentioned before) or it (co)refers to the same referent as a previous referring expression does. With such inferences, we can link the information across multiple referring expressions or overwrite old information with new information, and then search for proper referents based on all the accumulated information.

We use the similar machine learning based approach for coreference resolution[4] (e.g., as in [114, 115]). Formally, let $x_i$ be a discourse entity extracted from the current referring

[3]Recall that, in Section 3.2.3, we have already discussed this prevalent collaboration pattern and given it a name as "agent-present-human-accept".

[4]To serve our referential grounding purpose, we specifically focus on coreference resolution on mentions of the physical objects in the shared environment, but not mentions of people, event, and so on.

expression, and $x_j$ is a discourse entity from a previous expression. We then perform coreference resolution as binary classification on the entity pair $(x_i, x_j)$, i.e., we train a maximum entropy classifier[5] [116] to predict whether $x_i$ and $x_j$ should both refer to the same object (i.e. *positive*) or each of them should refer to a different object (i.e. *negative*). The features we use for the classification include the distance between $x_i$ and $x_j$, the determiners associated with them, the associated pronouns, the extracted object-specific properties, the syntactic roles, who is the speaker, etc.

Note that, although our coreference tracking task is similar to the pairwise classification step in regular coreference resolution (e.g., as in [117]; [118]; [119]), it does have some unique characteristics:

- Since its purpose is to link between separated pieces of information (e.g., due to the dialogue dynamics, fragmented utterances, or partial parsing results), coreference tracking directly deals with the extracted discourse entities but not the linguistic mentions as regular coreference resolution does. For example, our parser can extract a discourse entity as $[x, red(x)]$ from a one-word fragment "*red*", which even does not count as a mention. Furthermore, it only focuses on expressions referring to physical objects in the shared environment, but not people, event, etc.

- The situated context (i.e., the shared environment) plays an important role. Recall the earlier Example 1, the director was trying to communicate the identities of three objects (two apples and a pear). However the phrase "an apple" appeared five times plus another "a red pear". This is contradictory to non-situated discourse, where after an entity has already been evoked the definite determiner "the" should be used

[5] `http://nlp.stanford.edu/downloads/classifier.shtml`

in the following referring expressions. Thus coreference tracking based on linguistic information alone may be difficult and the situational context needs to be taken into consideration. Actually, solving coreference resolution and referential grounding as a joint problem in situated dialogue itself is an interesting future work to pursue.

- Since our dialogue graph use edges to represent only binary relations between discourse entities, entity-pairwise coreference tracking is sufficient and it is not necessary for us to look at coreference chains.

To incorporate the coreference tracking results into the dialogue graph, we add a special type of "coreference edges" into the graph. Such an edge (i.e., denoted as $\overline{x_i x_j}$) exists between each pair of nodes in the graph. It is also assigned an attribute to encode the coreference tracking result (i.e., positive or negative, as well as the classification probability) on this pair of discourse entities.[6]

### 5.3.3 Iterative Probabilistic Labeling Algorithm

The probabilistic relaxation labeling algorithm [109] is formulated in the Bayesian framework for contextual label assignment. It provides a unified probabilistic evidence-combining scheme to integrate unary attributes, binary relations and prior knowledge for updating the labeling probabilities (i.e. $P\left(\theta_m = x'_n\right)$). In essence, the probabilistic relaxation labeling algorithm is based on several conditional independence assumptions, such as the matching of a node $x_m$ is only dependent on itself and its neighbors but not all other unrelated nodes.

Given the same kind of graph representations (i.e., the discourse graph and the vision graph) as described in Section 3.2.1, probabilistic relaxation labeling finds proper labelings

[6] If the distance between a pair of discourse entities is greater than a predefined window size, we just assign it to the majority class (i.e. negative) based on the class prior probability.

(i.e., matchings) in an iterative manner. It first initiates the labeling probabilities by considering only the unary attributes of each node, and then efficiently and effectively updates the labeling probability of each node based on the labeling of its neighbors and the relations with them. The detailed algorithm is as follows:

**Initialization:**

Compute the initial labeling probabilities:

$$P^{(0)}(\theta_m = x'_n) = \frac{P\left(x_m \mid \theta_m = x'_n\right) \hat{P}\left(\theta_m = x'_n\right)}{\sum\limits_{x'_{n_2} \in \mathrm{X}'} P\left(x_m \mid \theta_m = x'_{n_2}\right) \hat{P}\left(\theta_m = x'_{n_2}\right)} \tag{5.1}$$

in which $\hat{P}\left(\theta_m = x'_n\right)$ is the prior probability of labeling $x_m$ with $x'_n$. The prior probability can be used to encode any prior knowledge about possible labelings. Especially in incremental processing of the dialogue, the prior can encode previous grounding hypotheses (e.g., the agent-present-human-accept pattern as discussed in Section 3.2.3), and other information from the collaborative dialogue such as confirmation, rejection, or replacement (see the collaboration patterns summarized in Section 1.2.3).

$P\left(x_m \mid \theta_m = x'_n\right)$ is called the "compatibility coefficient" between $x_m$ and $x'_n$, which is computed based on the attributes of $x_m$ and $x'_n$:

$$P\left(x_m \mid \theta_m = x'_n\right) \approx \prod_a P\left(u_{ma} \mid \theta_m = x'_n\right) \tag{5.2}$$

and we further define

$$
\begin{aligned}
P\left(u_{ma} \mid \theta_m = x'_n\right) &= P\left(u_{ma} \mid u'_{na}\right) \\
&= \frac{p\Big(u'_{na}|u_{ma}\Big)p(u_{ma})}{\sum\limits_{u_{ma} \in L_a} p(u'_{na}|u_{ma})p(u_{ma})}
\end{aligned}
\tag{5.3}
$$

where $L_a$ is the "lexicon" for the $a$-th attribute of a discourse graph node, e.g., for the *color* attribute:

$$
L_{color} = \{red, green, blue, \ldots\}
$$

and $p\left(u'_{na} \mid u_{ma}\right)$ is what we have called the "semantic grounding function" earlier in Section 3.2.2[7], i.e., the probability of observing $u'_{na}$ given the word $u_{ma}$. It judges the compatibilities between the symbolic attribute values from the discourse graph and the numeric attribute values from the vision graph.

**Iteration:**

At each iteration (i.e., in the equation, superscript $(r)$ means at the $r$-th iteration) and for each possible labeling, compute the "support function" as:

$$
\begin{aligned}
&Q^{(r)}\left(\theta_m = x'_n\right) = \\
&\prod_{m_2 \in \breve{\mathrm{M}}_m} \sum_{x'_{n_2} \in \mathrm{X}'} P^{(r)}\left(\theta_{m_2} = x'_{n_2}\right) P\left(x_m x_{m_2} \mid \theta_m = x'_n, \theta_{m_2} = x'_{n_2}\right)
\end{aligned}
\tag{5.4}
$$

in which the set of indices $\breve{\mathrm{M}}_m$ is defined as:

$$
\breve{\mathrm{M}}_m = \{1, 2, \ldots, m-1, m+1, \ldots, M\},
$$

The support function $Q^{(r)}\left(\theta_m = x'_n\right)$ here expresses how the labeling $\theta_m = x'_n$ at the

[7] Here we use the same set of manually defined semantic grounding functions as introduced in Section 3.2.2.

$r$-th iteration is supported by the labeling of $x_m$'s neighbors, taking into consideration the binary relations that exist between $x_m$ and these neighbors (Here we denote an edge between node $x_m$ and node $x_{m_2}$ directly as $x_m x_{m_2}$).

Then update the probability of each possible labeling as:

$$P^{(r+1)}(\theta_m = x'_n) = \frac{P^{(r)}\left(\theta_m = x'_n\right) Q^{(r)}\left(\theta_m = x'_n\right)}{\sum\limits_{x'_{n_2} \in \mathrm{X}'} P^{(r)}\left(\theta_m = x'_{n_2}\right) Q^{(r)}\left(\theta_m = x'_{n_2}\right)} \tag{5.5}$$

Similar to the node compatibility coefficient, the edge compatibility coefficient between $x_m x_{m_2}$ and $x'_n x'_{n_2}$ , namely the $P\left(x_m x_{m_2} \mid \theta_m = x'_n, \theta_{m_2} = x'_{n_2}\right)$ for computing $Q^{(r)}\left(\theta_m = x'_n\right)$, is also based on the attributes of the two edges and their corresponding semantic grounding functions. For an edge that represents the spatial relation between two entities, this is similar to the state-space search approach as described in Section 3.2.2.

The discourse coreference relations, however, can be handled differently under the unified scheme of the probabilistic labeling method. For the state-space search method, two coreferential nodes need to be merged for them to be grounded jointly, but in this way a discourse graph can only represent one hypothesis of the coreference relation between two nodes. To be able to utilize multiple hypotheses from the coreference resolution component, the probabilistic labeling approach can directly incorporate them as the edge compatibility coefficients.

Recall that in Section 5.3.2 we have treated coreference resolution as a classification problem on discourse entities pairs: for each pair of discourse entities (i.e., discourse graph nodes) $\left(x_m, x_{m_2}\right)$, we trained a statistical classifier to predict whether $x_m$ and $x_{m_2}$ should both refer to the same object (i.e., *positive*) or each of them should refer to a different object

(i.e., *negative*). To incorporate the coreference results into the discourse graph, we now add a special type of "coreference edges" into the graph. Such an edge (i.e., denoted as $\overline{x_m x_{m_2}}$) exists between each pair of nodes in the graph. It is also assigned an attribute to encode the coreference tracking result (i.e., positive or negative, and the classification probability) on this pair of discourse entities.

Suppose $\overline{x_m x_{m_2}}$ is assigned a *positive* attribute value by the coreference classifier, the compatibility coefficient then can be computed as:

$$\begin{aligned}
&P\left(\overline{x_m x_{m_2}} = + \mid \theta_m = x'_n, \theta_{m_2} = x'_{n_2}\right) \\
&= \frac{P\left(\theta_m{=}x_m, \theta_{m_2}{=}x'_{n_2} | \overline{x_m x_{m_2}}{=}+\right) P\left(\overline{x_m x_{m_2}}{=}+\right)}{P\left(\theta_m{=}x_m, \theta_{m_2}{=}x'_{n_2}\right)}, \text{ where} \\
&P\left(\theta_m = x_m, \theta_{m_2} = x'_{n_2}\right) \\
&= P\left(\theta_m = x_m, \theta_{m_2} = x'_{n_2} \mid \overline{x_m x_{m_2}} = +\right) P\left(\overline{x_m x_{m_2}} = +\right) \\
&+P\left(\theta_m = x_m, \theta_{m_2} = x'_{n_2} \mid \overline{x_m x_{m_2}} = -\right) P\left(\overline{x_m x_{m_2}} = -\right)
\end{aligned} \tag{5.6}$$

in which $P\left(\overline{x_m x_{m_2}} = +\right)$ is the coreference classifier's output probability of assigning the entity pair $\left(x_m, x_{m_2}\right)$ to the positive class, and $P\left(\overline{x_m x_{m_2}} = -\right) = 1 - P\left(\overline{x_m x_{m_2}} = +\right)$ since it is a binary classification. Furthermore, we use the coreference classifier's *precision* to estimate $P\left(\theta_m = x'_n, \theta_{m_2} = x'_{n_2} \mid \overline{x_m x_{m_2}} = +\right)$ and $P\left(\theta_m = x'_n, \theta_{m_2} = x'_{n_2} \mid \overline{x_m x_{m_2}} = -\right)$. Namely, when the classifier assigns an entity pair to the positive or negative class, how likely that they are truly mapping to the same object (i.e. when $\theta_m = \theta_{m_2}$) or mapping to different objects (i.e. when $\theta_m \neq \theta_{m_2}$). The compatibility coefficient for a negative coreference edge can be computed in a similar way.

**Termination:**

Terminate the algorithm if any one of the following conditions is true:

- For each $m$, one of $P(\theta_m = x'_n)$ exceeds $1 - \epsilon_1$, where $\epsilon_1 \ll 1$;
- In the last iteration, none of $P(\theta_m = x'_n)$ changed by more than $\epsilon_2$, where $\epsilon_2 \ll 1$;
- The number of iterations has reached a specified limit.

In practice, we find that this algorithm often converges very fast ($\leq 5$ iterations).

## 5.4 Evaluation

### 5.4.1 Data

We use the same data as in Section 3.3 and compare the referential grounding performances between the probabilistic labeling approach and the state-space search approach. As described earlier in Section 3.3, the dataset we use for evaluation purposes contains the transcriptions of human-human (i.e., a director and a matcher) dialogues on a object-naming task, along with the images that were used to collect each dialogue. This time we build the discourse graph completely based on automatic language processing and coreference resolution. It thus allows us to use more data collected from the experiment, since we don't need to manually annotate the semantics and coreferences for each dialogue.

Our dataset for evaluation now has 62 dialogues, each of which contains an average of 25 valid utterances from the director. We first apply our CCG-based parser to extract semantic representations (i.e., the DRS representation as described in Section 5.3.1) from these utterances. An average of 33 discourse entities per dialogue (1.3 per utterance) are extracted through automatic parsing. Since the matchers in these dialogues actually played the role of a computer-vision based agent, we treat all their decisions as known to our system[8]

[8]We leave the agent's decision making (e.g., response generation) into our future work.

and focus on only the processing of the directors' utterances.

Based on the semantic parsing outcome, we need to further infer the coreference relations between each pair of extracted discourse entities. As discussed earlier in Section 5.3.2, such coreference relations become the key for tracking the collaborative dynamics in the dialogues of this dataset, thus building the dialogue graph representation and generating the grounding (i.e., graph-matching) results rely on coreference resolution. To infer the pairwise coreference between discourse entities, for each dialogue we create a list of entity pairs by pairing each discourse entity with each of the previous discourse entities within a predefined window. On average over 300 entity pairs are created for each dialogue. We then use each dialogue as one testing set and all others as training set (i.e., in the "leave-one-out" manner) to train and evaluate the coreference classifier.

### 5.4.2 Results

We apply both the probabilistic labeling algorithm and the state-space search algorithm to ground each of the director's discourse entities onto an object perceived from the image. The referential grounding performances of the two algorithms are compared in Table 5.1.

Note that, the results in Table 5.1 are calculated and organized based on three ideas:

(1) Besides the accuracy of the top-1 grounding hypothesis, we also measure and compare the accuracies of the top-2 and top-3 grounding hypotheses. The "accuracy" of the top-2 grounding hypotheses is measured in this way: suppose $\hat{x'}$ is the ground-truth mapping of $x_m$, and $T_2 = \{x'_1, x'_2\}$ is the top-2 grounding hypotheses generated by the labeling algorithm. The matching of $x_m$ is counted as a "hit" if $\hat{x'} \in T_2$, otherwise a "miss". And the overall accuracy is the percentage of all the hits among all the discourse entities

| | Top-1 | Top-2 | Top-3 |
|---|---|---|---|
| Random Guess[a] | 7.7% | 15.4% | 23.1% |
| S.S.S. | 19.1% | 19.7% | 21.3% |
| P.L. | 24.9% | 36.1% | 45.0% |
| Gain[b] | 5.8% ($p < 0.01$) | 16.4% ($p < 0.001$) | 23.7% ($p < 0.001$) |
| P.L. using annotated coreference | 66.4% | 74.8% | 81.9% |

[a] Each image contains an average of 13 objects.

[b] $p$-value is based on the Wilcoxon signed-rank test [92] on the 62 dialogues.

Table 5.1 Comparison of the reference grounding performances of a random guess baseline, Probabilistic Labeling (P.L.) and State-Space Search (S.S.S.), and P.L. using manually annotated coreference.

being evaluated. The accuracy of the top-3 grounding hypotheses is measured similarly.

(2) The reference grounding results are evaluated in an incremental manner to resemble the situation in a real dialogue: For each dialogue, we go through the director's utterances one-by-one from the beginning to the end. At each utterance, we first update the dialogue graph (e.g., add in new discourse entities and relations), and then (re-)evaluate the updated grounding results of all the so far encountered discourse entities[9]. The final accuracy then is the averaged accuracy over all the "evaluation points" (i.e., each utterance is a evaluation point).

As shown in Table 5.1, probabilistic labeling (P.L.) significantly outperforms state-space search (S.S.S.), especially with regard to producing meaningful multiple grounding hypotheses. The state-space search algorithm actually only results in multiple hypotheses for the

[9] Note that, when a "new" discourse entity is added into the graph, it can change the grounding results of previous discourse entities with the new information it brings in. This is why we re-evaluate the grounding results of all the encountered discourse entities each time after new information is added.

overall matching, and it fails to produce multiple hypotheses for many individual discourse entities. Multiple grounding hypotheses can be very useful to generate responses such as clarification questions or nonverbal feedback (e.g. pointing, gazing). For example, if there are two competing hypotheses, the dialogue manager can utilize them to generate a response like "I see two objects there, are you talking about this one (pointing to) or that one (pointing to the other)?". Such proactive feedback is often an effective way in referential communication, as we discussed earlier in Section 1.2.3 and 3.2.3.

The probabilistic labeling algorithm not only produces better grounding results, it also runs much faster (with a running-time complexity of $O\left(M^2N^2\right)$,[10] comparing to $O\left(N^4\right)$ of the state-space search algorithm[11]). Figure 5.1 shows the averaged running time of the two algorithm algorithms (the red curve is the state-space search algorithm and the blue curve is the probabilistic labeling algorithm) on a Intel Core i7 1.60GHz CPU with 16G RAM computer. As we can see, when the size of the dialogue graph becomes greater than 15, state-space search takes more than 1 minute to run, whereas probabilistic labeling only takes 1 second or less even for large graphs. The efficiency of the probabilistic labeling algorithm thus makes it more appealing for real-time interaction applications.

Although probabilistic labeling significantly outperforms the state-space search, the grounding performance is still rather poor (less than 50%) even for the top-3 hypotheses. With no surprise, the coreference resolution performance plays an important role in the final grounding performance (see the grounding performance of using manually annotated coreference in the bottom part of Table 5.1). Due to the simplicity of our current coreference classifier and the flexibility of the human-human dialogue in the data, the pairwise coreference resolution

[10] $M$ is the number of nodes in the vision graph and $N$ is the number of nodes in the dialogue graph.

[11] Beam search algorithm is applied to reduce the exponential $O\left(M^N\right)$ to $O\left(N^4\right)$.

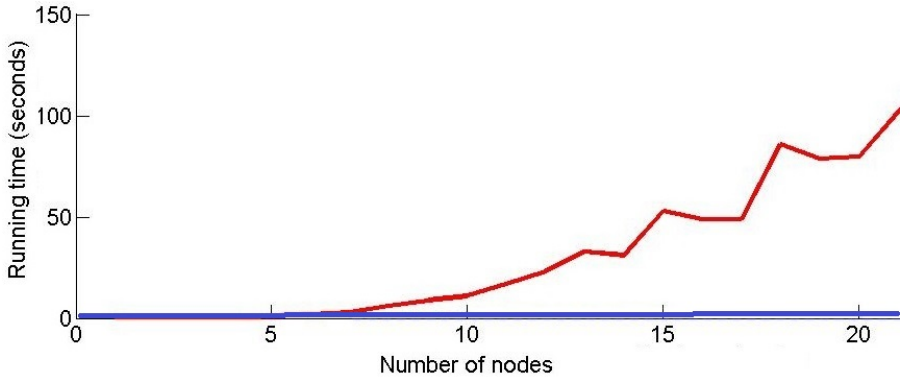



Figure 5.1 Average running time of the state-space search algorithm with respect to the number of nodes to be grounded in a dialogue graph. The red curve is the state-space search algorithm and the blue curve is the probabilistic labeling algorithm.

only achieves 0.74 in precision and 0.43 in recall. The low recall of coreference resolution makes it difficult to link interrelated referring expressions and resolve them jointly. So it is important to develop more sophisticated coreference resolution and dialogue management components to reliably track the discourse relations and other dynamics in the dialogue to facilitate referential grounding.

## 5.5 Conclusion and Discussion

In this chapter, we have presented a probabilistic labeling based approach for referential grounding in situated dialogue. This approach provides a unified scheme for incorporating different sources of information. Its probabilistic scheme allows each information source to present multiple hypotheses to better handle uncertainties. Based on the integrated information, the labeling procedure then efficiently generates probabilistic grounding hypotheses, which can serve as important guidance for the dialogue manager's decision making.

As demonstrated by the empirical evaluation results, the probabilistic labeling approach has some desirable advantages compared to the state-space search algorithm based approach:

- The probabilistic labeling approach is designed to incorporate uncertainties from different sources through its unified evidence-combing scheme. Its outputs are directly multiple-hypotheses with probabilities. For the state-space search approach, maintaining multiple-hypotheses is very cumbersome and inefficient.

- The probabilistic labeling approach outputs an individual score/probability for each possible matching of each discourse entity. Individual matching scores and multiple-hypotheses can provide important guidance for dialogue management.

- The probabilistic labeling approach is more efficient than the state-space search approach, and incremental grounding can be easily implemented. It is thus more suitable for building a system that interacts with human in real-time.

One future direction is to extend the probabilistic labeling approach to handle group-based (i.e., $n$-ary) relations, as what we have done in Section 3.2.1 using hypergraph representations. The issue with the probabilistic labeling approach is that its iterative labeling scheme only takes binary relations into consideration. How to incorporate $n$-ary relations into probabilistic labeling in a theoretically sound and computational efficiently way is an interesting research question to address. One straightforward approach to try is to decompose a $n$-ary relations into several binary relations without loss of expressive power. Such a method may work well for "inter-group" relations (such as "the apple is behind the cluster of four"), but may not for "intra-group" relations (such as "the apple is in the middle of the cluster of four").[12] Nevertheless, a general approach that can account for any kind of binary and n-ary relations is desirable.

In Chapter 4, we introduced a weight-learning method that is based the state-space

[12]See [55] for some discussion on different types of group-based relations.

search algorithm. One open question is whether we can also design a similar weight-learning mechanism for the probabilistic labeling algorithm. In general, we can always assign some weights to different attributes and attribute-values to specify how reliable/important they are for referential grounding. Just the same as the state-space search, probabilistic labeling also relies on the combination of all the attributes to find proper matchings. Thus it should benefit from meaningful weights in the same way as state-space search does. However, weight learning under the probabilistic labeling framework may not be easy to design and implement, because of the non-linear calculation and iterative procedure it involves. More complex optimization techniques (such as quadratic programming) may need to be considered. One possible simpler solution is to use the state-space search for weight-learning and apply the learned weights in probabilistic labeling for real-time grounding. If the weights learned by our weight-learning method make general senses, such a "hybrid" approach may produce better results than using either of the two approaches alone.

Finally, another important future direction is to tackle coreference resolution and referential grounding in situated dialogue as a joint problem. As we discussed earlier in Section 5.3.2, coreference is often the key to track the relations at the locutionary level (i.e., the relations among the formal semantic representations of different referring expressions) and to build a well-connected discourse graph for referential grounding. It may also serve as one important source of information to track the dynamics at the illocutionary level (i.e., identifying and tracking dialogue acts) for sophisticated dialogue management. Our empirical results have already demonstrated that, on one hand, referential grounding relies on good coreference resolution to achieve better performance, but on the other hand coreference resolution in situated dialogue is difficult and needs to incorporate the situational context and other information from the interaction. As we have mentioned earlier at the beginning of this

chapter, information from different sources is often interrelated (e.g., the dialogue content is always related to the surrounding environment in situated dialogue), incorporating one source of information into another can potentially reduce the uncertainties and errors within the latter one. Thus, based on the probabilistic and iterative scheme of probabilistic labeling, we should further investigate how to solve coreference resolution and referential grounding jointly and iteratively, and expect that the performances of both can be jointly leveraged.

# Chapter 6

# Conclusion and Future Work

In situated dialogue, one significant challenge is the agent (i.e., the dialogue system) needs to perceive and make sense of the shared environment simultaneously during conversation. The agent's representation of the world is often limited by its perceptual and reasoning capabilities. Therefore, although co-present, humans and agents do not share a joint perceptual basis, and the lack of such a joint basis will jeopardize referential communication between humans and agents. It will become more difficult for the agent to identify referents in the physical world that are referred to by the human, i.e., the problem of referential grounding. The work presented in this dissertation has focused on developing computational approaches to enable robust and adaptive referential grounding in situated dialogue.

In Chapter 3, graph-based representations are utilized to model the linguistic discourse of collaborative referring dialogue and the visual perception of the physical environment. Referential grounding is then formulated as a graph-matching problem and a state-space search algorithm is applied to ground linguistic references onto perceived objects. In addition, hypergraph representations are introduced to account for group-based descriptions, and the most prevalent pattern of collaborative communication observed from dialogue data is incorporated into the search algorithm. The empirical results show that, even when the perception of the environment by computer vision algorithms has a high error rate (95% of the objects are mis-recognized), this approach can still correctly ground those mis-recognized referents with 66% accuracy, whereas a object-properties-only baseline just obtains 31%

grounding accuracy. As demonstrated by the results, the graph-matching approach has provided a potential solution to robust referential grounding through modeling and utilizing spatial relations, group descriptions, and collaborative referring behaviors.

In Chapter 4, an optimization based approach is further developed to allow the agent to detect and adapt to the perceptual differences through interaction and learning. Through interaction with the human, the agent can learn a set of weights indicating how reliably/unreliably each dimension (object type, object color, etc.) of its perception of the environment maps to the human's linguistic descriptors. The agent can then adapt to the situation by applying the learned weights to the grounding algorithm, and adjust its word grounding models accordingly. The empirical evaluation shows this weight-learning approach can effectively adjust the weights to reflect the agent's perceptual insufficiencies. When the perceptual difference is high (i.e., the agent can only correctly recognize 10-40% of objects in the environment), applying the learned weights with updated word grounding models significantly improves referential grounding performance by an absolute gain of 10%.

In Chapter 5, a probabilistic-labeling based approach is developed to better support collaborative and incremental dialogue systems that interact with human users in real time. This approach provides a unified scheme for incorporating different sources of information. Its probabilistic scheme allows each information source to present multiple hypotheses to better handle uncertainties. Based on the integrated information, the labeling procedure then efficiently generates probabilistic grounding hypotheses, which can serve as important guidance for the dialogue manager's decision making. Evaluated on the same dataset, the probabilistic labeling approach significantly outperforms the state-space search approach in both grounding accuracy and efficiency.

As we have already discussed in Section 3.4, Section 4.4, and Section 5.5, there are a broad

range of future directions that can be further explored. One very interesting and important future work is to solve coreference resolution and referential grounding as a joint problem in situated dialogue. Coreference resolution in situated settings is to determine whether two linguistic entities refer to the same physical object or not, and referential grounding is to find out which physical object an linguistic entity refers to. It should not be difficult to see that these two problems are intertwined and need to be solved jointly in situated dialogue.

However, in our current work they are treated as two separate problems and solved individually. For coreference resolution, we used the same kind of classification method and feature set as in the previous text-based work. Unsurprisingly, the performance is rather low because the key characteristic of situated dialogue, i.e., the situational context, was not taken into consideration. And unreliable coreference results have further impacted the performance of referential grounding because the latter relies on the coreference to combine information across the collaborative discourse.

Our probabilistic labeling approach has provided a unified and iterative scheme to update beliefs on the matchings between linguistic entities and physical objects, and on the relationships between linguistic entities as well[1]. Thus, probabilistic labeling may provide a potential solution for jointly updating both the coreference resolution and referential grounding beliefs throughout its iterative evidence-combining procedure. If we can develop such a joint approach and demonstrate its effectiveness in improving both coreference resolution and referential grounding performances, it will be an interesting and meaningful achievement.

To enable situated dialogue systems in the open world, sophisticated machine-learning techniques are indispensable. Recent works on joint learning (e.g., [120, 35]) have demonstrated the nice idea of jointly acquiring/updating models of different components from the

[1]I.e., this is captured by the "supporting function" as described in Section 5.3.3.

system's direct inputs and outputs, without providing supervised data for each individual component. Such kind of learning methods is desirable for our goal, i.e., building systems that can learn and adapt from the interactions with human users and the environment.

While most of the previous work on situated referential grounding only focused on the locutionary level (i.e., interpretation of single referring expressions), processing information at the illocutionary level (i.e., collaborative discourses) should also play an important role in building interactive systems. Although we have incorporated a prevalent collaborative pattern into our graph-based approach for referential grounding, it is still an ad-hoc solution. The probabilistic labeling approach provides a unified scheme for incorporating different source of information for referential grounding, and supports agent's response generation by providing probabilistic grounding hypotheses to the dialogue manager component. But we just have focused on only the interpretation aspect.

Can we have a more general framework, under which the interpretation of referring expressions, incorporation of dialogue dynamics, generation of collaborative actions, and mediation of perceptual differences can be all handled in a unified manner? Furthermore, such a framework should be based on or integrated with advanced machine-learning techniques, so that it can continuously adapt to changing environments and situations through learning from the interaction with human users. One important step along this direction is to utilize some well-established machine-learning techniques/frameworks, such as POMDP and reinforcement learning [121, 122], and tailor them to best address our focused problem here – situated referential communication.

# REFERENCES

# REFERENCES


[1] D. Bohus and E. Horvitz, "Dialog in the open world: platform and applications," in *Proceedings of the 2009 international conference on Multimodal interfaces*, pp. 31–38, ACM, 2009.

[2] H. I. Christensen, G. M. Kruijff, and J. Wyatt, eds., *Cognitive Systems*. Springer, 2010.

[3] M. F. McTear, "Spoken dialogue technology: enabling the conversational interface," *ACM Computing Surveys*, vol. 34, no. 1, pp. 90–169, 2002.

[4] A. Raux, B. Langner, D. Bohus, A. W. Black, and M. Eskenazi, "Lets go public! taking a spoken dialog system to the real world," in *in Proc. of Interspeech 2005*, Citeseer, 2005.

[5] M. Johnston, S. Bangalore, G. Vasireddy, A. Stent, P. Ehlen, M. Walker, S. Whittaker, and P. Maloor, "MATCH: An architecture for multimodal dialogue systems," in *Proceedings of the ACL'02*, pp. 376–383, 2002.

[6] J. Chai, P. Hong, M. Zhou, and Z. Prasov, "Optimization in multimodal interpretation," in *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, p. 1, Association for Computational Linguistics, 2004.

[7] D. K. Byron, "Understanding referring expressions in situated language: some challenges for real-world agents," in *Proceedings of the First International Workshop on Language Understanding and Agents for Real World Interaction*, pp. 39–47, 2003.

[8] P. Gorniak and D. Roy, "Situated language understanding as filtering perceived affordances," *Cognitive Science*, vol. 31, no. 2, pp. 197–231, 2007.

[9] R. J. Ross, *Situated Dialogue Systems: Agency and Spatial Meaning in Task-Oriented Dialogue*. PhD thesis, University of Bremen, 2009.

[10] H. H. Clark, *Using language*. Cambridge University Press, 1996.

[11] H. H. Clark and S. E. Brennan, "Grounding in communication," *Perspectives on socially shared cognition*, vol. 13, no. 1991, pp. 127–149, 1991.

[12] H. H. Clark and D. Wilkes-Gibbs, "Referring as a collaborative process," *Cognition*, vol. 22, no. 1, pp. 1–39, 1986.

[13] M. Skubic, D. Perzanowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska, and D. Brock, "Spatial language for human-robot dialogs," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 34, no. 2, pp. 154–167, 2004.

[14] T. Tenbrink, "Identifying objects on the basis of spatial contrast: An empirical study," in *Spatial cognition iv. reasoning, action, interaction*, pp. 124–146, Springer Berlin Heidelberg, 2005.

[15] R. Moratz, "Intuitive linguistic joint object reference in human-robot interaction," in *Proceedings of the Twenty-first National Conference on Artificial Intelligence (AAAI), Boston, MA.*, AAAI, Menlo Park, CA, 2006.

[16] H. H. Clark and E. F. Schaefer, "Contributing to discourse," *Cognitive science*, vol. 13, no. 2, pp. 259–294, 1989.

[17] P. G. Edmonds, "Collaboration on reference to objects that are not mutually known," in *Proceedings of the 15th Conference on Computational Linguistics-Volume 2*, pp. 1118–1122, Association for Computational Linguistics, 1994.

[18] P. A. Heeman and G. Hirst, "Collaborating on referring expressions," *Computational Linguistics*, vol. 21, no. 3, pp. 351–382, 1995.

[19] D. DeVault and M. Stone, "Learning to interpret utterances using dialogue history," in *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 184–192, Association for Computational Linguistics, 2009.

[20] E. Levin and R. Passonneau, "A woz variant with contrastive conditions," in *Proceedings of the Interspeech Sattelite Workshop, Dialogue on Dialogues: Multidisciplinary Evaluation of Advanced Speech-based Interactive Systems*, 2006.

[21] R. J. Passonneau, S. L. Epstein, and J. B. Gordon, "Help me understand you: Addressing the speech recognition bottleneck.," in *AAAI Spring Symposium: Agents that Learn from Human Teachers*, pp. 119–126, 2009.

[22] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, no. 285-296, pp. 23–27, 1975.

[23] D. Zhang and G. Lu, "An integrated approach to shape based image retrieval," in *Proceedings of 5th Asian Conference on Computer Vision (ACCV), Melbourne, Australia*, 2002.

[24] R. Moratz and T. Tenbrink, "Spatial reference in linguistic human-robot interaction: Iterative, empirically supported development of a model of projective relations," *Spatial cognition and computation*, vol. 6, no. 1, pp. 63–107, 2006.

[25] T. Tenbrink and R. Moratz, "Group-based spatial reference in linguistic human-robot interaction," in *Proceedings of EuroCogSci*, vol. 3, pp. 325–330, 2003.

[26] C. Liu, R. Fang, and J. Chai, "Towards mediating shared perceptual basis in situated dialogue," in *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, (Seoul, South Korea), pp. 140–149, Association for Computational Linguistics, July 2012.

[27] C. Liu, R. Fang, L. She, and J. Chai, "Modeling collaborative referring for situated referential grounding," in *Proceedings of the SIGDIAL 2013 Conference*, (Metz, France), pp. 78–86, Association for Computational Linguistics, August 2013.

[28] C. Liu and J. Chai, "Learning to mediate perceptual differences in situated human-robot dialogue.," in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 2015.

[29] C. Liu, L. She, R. Fang, and Y. J. Chai, "Probabilistic labeling for efficient referential grounding based on collaborative discourse," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 13–18, Association for Computational Linguistics, 2014.

[30] T. Winograd, *Procedures as a representation for data in a computer program for understanding natural language*. PhD thesis, Massachusetts Institute of Technology, 1970.

[31] P. Gorniak and D. Roy, "Grounded semantic composition for visual scenes," *J. Artif. Intell. Res. (JAIR)*, vol. 21, pp. 429–470, 2004.

[32] A. Siebert and D. Schlangen, "A simple method for resolution of definite reference in a shared visual context," in *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, pp. 84–87, Association for Computational Linguistics, 2008.

[33] D. L. Chen and R. J. Mooney, "Learning to interpret natural language navigation instructions from observations," in *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, pp. 859–865, 2011.

[34] C. Matuszek, N. FitzGerald, L. Zettlemoyer, L. Bo, and D. Fox, "A joint model of language and perception for grounded attribute learning," in *Proceedings of the 29th International Conference on Machine Learning*, 2012.

[35] J. Krishnamurthy and T. Kollar, "Jointly learning to parse and perceive: Connecting natural language to the physical world," *Transactions of the Association for Computational Linguistics*, vol. 1, no. 2, pp. 193–206, 2013.

[36] S. Guadarrama, L. Riano, D. Golland, D. Gouhring, Y. Jia, D. Klein, P. Abbeel, and T. Darrell, "Grounding spatial relations for human-robot interaction," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 1640–1647, IEEE, 2013.

[37] R. M. Krauss and S. Weinheimer, "Concurrent feedback, confirmation, and the encoding of referents in verbal communication.," *Journal of Personality and Social Psychology*, vol. 4, no. 3, p. 343, 1966.

[38] R. M. Krauss and S. Weinheimer, "Effect of referent similarity and communication mode on verbal encoding," *Journal of Verbal Learning and Verbal Behavior*, vol. 6, no. 3, pp. 359–363, 1967.

[39] H. P. Grice, *Logic and conversation*. Harvard Univ., 1970.

[40] M. Mitchell, K. van Deemter, and E. Reiter, "Natural reference to objects in a visual domain," in *Proceedings of the 6th international natural language generation conference*, pp. 95–104, Association for Computational Linguistics, 2010.

[41] S. Kriz, J. G. Trafton, and J. M. McCurry, "The role of spatial information in referential communication: Speaker and addressee preferences for disambiguating objects," in *Proceedings of the 29th Annual Cognitive Science Society*, 2007.

[42] H. H. Clark, *Arenas of Language Use*. University of Chicago Press, 1992.

[43] H. H. Clark and D. Wilkes-Gibbs, "Referring as a collaborative process," in *Cognition*, no. 22, pp. 1–39, 1986.

[44] H. H. Clark and S. E. Brennan, "Grounding in communication," in *Perspectives on socially shared cognition* (L. B. Resnick, R. M. Levine, and S. D. Teasley, eds.), pp. 127–149, 1991.

[45] D. Traum, *A Computational Theory of Grounding in Natural Language Conversation*. PhD thesis, University of Rochester, 1994.

[46] J. Cassell, T. Bickmore, L. Campbell, and H. Vilhjalmsson, "Human conversation as a system framework: Designing embodied conversational agents," in *Embodied conversational agents* (J. Cassell, J. Sullivan, S. Prevost, and E. Churchill, eds.), MIT Press, 2000.

[47] T. Fong, I. R. Nourbakhsh, C. Kunz, L. Fluckiger, J. Schreiner, R. Ambrose, R. Burridge, R. Simmons, L. Hiatt, A. Schultz, J. Trafton, M. Bugajska, and J. Scholtz, "The peer-to-peer human robot interaction project," *Proceedings of AIAA Space 2005*, 2005.

[48] K. Stubbs, P. Hinds, and D. Wettergreen, "Autonomy and common ground in human-robot interaction," in *IEEE Intelligent Systems*, pp. 42–50, 2007.

[49] K. Stubbs, D. Wettergreen, and I. Nourbakhsh, "Using a robot proxy to create commmon ground in exploration tasks," in *Proceedings of the HRI08*, pp. 375–382, 2008.

[50] M. K. Brown, B. M. Buntschuh, and J. G. Wilpon, "Sam: A perceptive spoken language-understanding robot," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 22, no. 6, pp. 1390–1402, 1992.

[51] K. Nagao and J. Rekimoto, "Ubiquitous talker: Spoken language in-teraction with real world objects," in *Proceeding of the International Joint Conference on Artificial Intelligence*, 1995.

[52] A. Mojsilovic, "A computational model for color naming and describing color composition of images," *Image Processing, IEEE Transactions on*, vol. 14, no. 5, pp. 690–699, 2005.

[53] T. Regier and L. A. Carlson, "Grounding spatial language in perception: an empirical and computational investigation.," *Journal of Experimental Psychology: General*, vol. 130, no. 2, p. 273, 2001.

[54] S. S. Dhande, "A computational model to connect gestalt perception and natural language," Master's thesis, Massachusetts Institute of Technology, 2003.

[55] K. Funakoshi, S. Watanabe, T. Tokunaga, and N. Kuriyama, "Understanding referring expressions involving perceptual grouping," *IEIC Technical Report (Institute of Electronics, Information and Communication Engineers)*, vol. 105, no. 204, pp. 19–24, 2005.

[56] M. Skubic, P. Matsakis, G. Chronis, and J. Keller, "Generating multi-level linguistic spatial descriptions from range sensor readings using the histogram of forces," *Autonomous Robots*, vol. 14, pp. 51–69, January 2003.

[57] R. Moratz and T. Tenbrink, "Instruction modes for joint spatial reference between naive users and a mobile robot," in *Proc. IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, vol. 1, pp. 43–48, 2003.

[58] S. C. Levinson, *Space in language and cognition: explorations in cognitive diversity.* Cambridge University Press, Cambridge, 2003.

[59] C. Liu, J. Walker, and J. Chai, "Ambiguities in spatial language understanding in situated human robot dialogue," in *AAAI 2010 Fall Symposium on Dialogue with Robots*, (Arlington, Virginia, USA), November 2010.

[60] J. G. Trafton, N. L. Cassimatis, M. D. Bugajska, D. P. Brock, F. E. Mintz, and A. C. Schultz, "Enabling effective human-robot interaction using perspective-taking in robots," vol. 35, pp. 460–470, 2005.

[61] T. Tenbrink, V. Maiseyenka, and R. Moratz, "Spatial reference in simulated human-robot interaction involving intrinsically oriented objects," in *Proceedings of the Symposium Spatial Reasoning and Communication at AISB'07 Artificial and Ambient Intelligence*, 2007.

[62] K.-y. Hsiao, N. Mavridis, and D. Roy, "Coupling perception and simulation: Steps towards conversational robotics," in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 1, pp. 928–933, IEEE, 2003.

[63] D. Roy, K.-Y. Hsiao, and N. Mavridis, "Mental imagery for a conversational robot," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 34, no. 3, pp. 1374–1383, 2004.

[64] J. Chai, P. Hong, and M. Zhou, "A probabilistic approach to reference resolution in multimodal user interfaces," in *Proceedings of the 9th international conference on Intelligent user interfaces*, pp. 70–77, ACM, 2004.

[65] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 4, pp. 377–388, 1996.

[66] T. Regier, "The human semantic potential," 1996.

[67] D. K. Roy and A. P. Pentland, "Learning words from sights and sounds: A computational model," *Cognitive science*, vol. 26, no. 1, pp. 113–146, 2002.

[68] D. K. Roy, "Learning visually grounded words and syntax for a scene description task," *Computer Speech & Language*, vol. 16, no. 3, pp. 353–385, 2002.

[69] C. Yu and D. H. Ballard, "A multimodal learning interface for grounding spoken language in sensory perceptions," *ACM Transactions on Applied Perception (TAP)*, vol. 1, no. 1, pp. 57–80, 2004.

[70] S. Tellex, T. Kollar, S. Dickerson, M. R. Walter, A. G. Banerjee, S. J. Teller, and N. Roy, "Understanding natural language commands for robotic navigation and mobile manipulation.," in *AAAI*, 2011.

[71] C. Sutton and A. McCallum, "An introduction to conditional random fields for relational learning," *Introduction to statistical relational learning*, pp. 93–128, 2006.

[72] S. Tellex, P. Thaker, J. Joseph, and N. Roy, "Learning perceptually grounded word meanings from unaligned parallel data," *Machine Learning*, vol. 94, no. 2, pp. 151–167, 2014.

[73] R. J. Kate and R. J. Mooney, "Learning language semantics from ambiguous supervision," in *AAAI*, vol. 7, pp. 895–900, 2007.

[74] T. Kwiatkowski, L. Zettlemoyer, S. Goldwater, and M. Steedman, "Lexical generalization in ccg grammar induction for semantic parsing," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 1512–1523, Association for Computational Linguistics, 2011.

[75] D. DeVault, N. Kariaeva, A. Kothari, I. Oved, and M. Stone, "An information-state approach to collaborative reference," in *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, pp. 1–4, Association for Computational Linguistics, 2005.

[76] M. Richardson and P. Domingos, "Markov logic networks," *Machine learning*, vol. 62, no. 1-2, pp. 107–136, 2006.

[77] C. Kennington and D. Schlangen, "Markov logic networks for situated incremental natural language understanding," in *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, (Seoul, South Korea), pp. 314–323, Association for Computational Linguistics, July 2012.

[78] D. Conte, P. Foggia, C. Sansone, and M. Vento, "Thirty years of graph matching in pattern recognition," *International journal of pattern recognition and artificial intelligence*, vol. 18, no. 03, pp. 265–298, 2004.

[79] D. Conte, P. Foggia, C. Sansone, and M. Vento, "How and why pattern recognition and computer vision applications use graphs," in *Applied Graph Theory in Computer Vision and Pattern Recognition*, pp. 85–135, Springer, 2007.

[80] W. Tsai and K. Fu, "Error-correcting isomorphisms of attributed relational graphs for pattern analysis," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 9, no. 12, pp. 757–768, 1979.

[81] A. Sanfeliu and K. S. Fu, "A distance measure between attributed relational graphs for pattern recognition," *IEEE transactions on systems, man, and cybernetics*, vol. 13, no. 3, pp. 353–362, 1983.

[82] M. A. Eshera and K. S. Fu, "A graph distance measure for image analysis," *IEEE transactions on systems, man, and cybernetics*, vol. 14, no. 3, pp. 398–410, 1984.

[83] M. Eshera and K. Fu, "An image understanding system using attributed symbolic representation and inexact graph-matching," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 5, pp. 604–618, 1986.

[84] G. Gallo, G. Longo, S. Pallottino, and S. Nguyen, "Directed hypergraphs and applications," *Discrete applied mathematics*, vol. 42, no. 2, pp. 177–201, 1993.

[85] D. Roy, Y. Ghitza, J. Bartelma, and C. Kehoe, "Visual memory augmentation: Using eye gaze as an attention filter," in *Proceedings of the IEEE International Symposium on Wearable Computers*, 2004.

[86] W.-H. Tsai and K.-S. Fu, "Subgraph error-correcting isomorphisms for syntactic pattern recognition," *Systems, Man and Cybernetics, IEEE Transactions on*, no. 1, pp. 48–62, 1983.

[87] W. Zhang, *State-space search: Algorithms, complexity, extensions, and applications*. Springer-Verlag New York Inc, 1999.

[88] R. M. Cesar Jr, E. Bengoetxea, I. Bloch, and P. Larrañaga, "Inexact graph matching for model-based recognition: Evaluation and comparison of optimization algorithms," *Pattern Recognition*, vol. 38, no. 11, pp. 2099–2113, 2005.

[89] K. Sambhoos, R. Nagi, M. Sudit, and A. Stotz, "Enhancements to high level data fusion using graph matching and state space search," *Information Fusion*, vol. 11, no. 4, pp. 351–364, 2010.

[90] Z. Prasov and J. Chai, "What's in a gaze?: the role of eye-gaze in reference resolution in multimodal conversational interfaces," in *Proceedings of the 13th international conference on Intelligent user interfaces*, pp. 20–29, ACM, 2008.

[91] Z. Prasov and J. Chai, "Fusing eye gaze with speech recognition hypotheses to resolve exophoric references in situated dialogue," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pp. 471–481, Association for Computational Linguistics, 2010.

[92] F. Wilcoxon, S. Katti, and R. A. Wilcox, "Critical values and probability levels for the wilcoxon rank sum test and the wilcoxon signed rank test," *Selected tables in mathematical statistics*, vol. 1, pp. 171–259, 1970.

[93] S. Sitter and A. Stein, "Modeling information-seeking dialogues: The conversational roles (cor) model," *RIS: Review of Information Science (online journal)*, vol. 1, no. 1, pp. 165–180, 1996.

[94] H. Shi, R. J. Ross, T. Tenbrink, and J. Bateman, "Modelling illocutionary structure: combining empirical studies with formal model analysis," in *Computational Linguistics and Intelligent Text Processing*, pp. 340–353, Springer, 2010.

[95] V. N. Kasyanov and I. A. Lisitsyn, "Hierarchical graph models and visual processing," in *Proc. Int. Conf. on Software: Theory and Practice, 16th World Comput. Congr. IFIP*, pp. 119–123, 2000.

[96] W. Palacz, "Algebraic hierarchical graph transformation," *Journal of Computer and System Sciences*, vol. 68, no. 3, pp. 497–520, 2004.

[97] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. Jordan, "Matching words and pictures," *Journal of Machine Learning Research*, vol. 3, pp. 1107–1135, 2003.

[98] D. Roy, "Grounding words in perception and action: computational insights," *TRENDS in Cognitive Sciences*, vol. 9, no. 8, pp. 389–396, 2005.

[99] C. Liu, R. Fang, and J. Chai, "Towards mediating shared perceptual basis in situated dialogue," in *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, (Seoul, South Korea), pp. 140–149, Association for Computational Linguistics, July 2012.

[100] C. Liu, L. She, R. Fang, and Y. J. Chai, "Probabilistic labeling for efficient referential grounding based on collaborative discourse," Proceedings of ACL'14 (Volume 2: Short Papers), pp. 13–18, 2014.

[101] C. Roos, T. Terlaky, and J.-P. Vial, *Interior point methods for linear optimization*. Springer, 2006.

[102] J. Y. Chai, L. She, R. Fang, S. Ottarson, C. Littley, C. Liu, and K. Hanson, "Collaborative effort towards common ground in situated human-robot dialogue," in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pp. 33–40, ACM, 2014.

[103] N. Cesa-Bianchi, *Prediction, learning, and games*. Cambridge University Press, 2006.

[104] R. Fang, C. Liu, and J. Y. Chai, "Integrating word acquisition and referential grounding towards physical world interaction," in *Proceedings of ICMI '12*, pp. 109–116, 2012.

[105] R. Fang, C. Liu, L. She, and J. Y. Chai, "Towards situated dialogue: Revisiting referring expression generation.," in *Proceedings of EMNLP'13*, pp. 392–402, 2013.

[106] R. Fang, M. Doering, and J. Y. Chai, "Collaborative models for referring expression generation in situated dialogue," in *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[107] H. H. Clark, *Using language*. Cambridge, UK: Cambridge University Press, 1996.

[108] P. A. Heeman and G. Hirst, "Collaborating on referring expressions," *Computational Linguistics*, vol. 21, no. 3, pp. 351–382, 1995.

[109] W. J. Christmas, J. Kittler, and M. Petrou, "Structural matching in computer vision using probabilistic relaxation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 17, no. 8, pp. 749–764, 1995.

[110] M. Steedman and J. Baldridge, "Combinatory categorial grammar," *Non-Transformational Syntax*, pp. 181–224, 2011.

[111] C. Bozşahin, G.-J. M. Kruijff, and M. White, "Specifying grammars for openccg: A rough guide," in *Included in the OpenCCG distribution*, 2005.

[112] H. Kamp and U. Reyle, *From discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory*. No. 42, Springer, 1993.

[113] W. M. Soon, H. T. Ng, and D. C. Y. Lim, "A machine learning approach to coreference resolution of noun phrases," *Computational linguistics*, vol. 27, no. 4, pp. 521–544, 2001.

[114] M. Strube and C. Müller, "A machine learning approach to pronoun resolution in spoken dialogue," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pp. 168–175, Association for Computational Linguistics, 2003.

[115] Y. Versley, S. P. Ponzetto, M. Poesio, V. Eidelman, A. Jern, J. Smith, X. Yang, and A. Moschitti, "Bart: A modular toolkit for coreference resolution," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Demo Session*, pp. 9–12, Association for Computational Linguistics, 2008.

[116] C. Manning and D. Klein, "Optimization, maxent models, and conditional estimation without magic," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Tutorials - Volume 5*, NAACL-Tutorials '03, (Stroudsburg, PA, USA), pp. 8–8, Association for Computational Linguistics, 2003.

[117] W. M. Soon, H. T. Ng, and D. C. Y. Lim, "A machine learning approach to coreference resolution of noun phrases," *Computational linguistics*, vol. 27, no. 4, pp. 521–544, 2001.

[118] Y. Versley, S. P. Ponzetto, M. Poesio, V. Eidelman, A. Jern, J. Smith, X. Yang, and A. Moschitti, "Bart: A modular toolkit for coreference resolution," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Demo Session*, pp. 9–12, Association for Computational Linguistics, 2008.

[119] G. Durrett and D. Klein, "Easy victories and uphill battles in coreference resolution," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, (Seattle, Washington), Association for Computational Linguistics, October 2013.

[120] L. S. Zettlemoyer and M. Collins, "Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars," *CoRR*, vol. abs/1207.1420, 2012.

[121] M. Gašic, C. Breslin, M. Henderson, D. Kim, M. Szummer, B. Thomson, P. Tsiakoulis, and S. Young, "Pomdp-based dialogue manager adaptation to extended domains," in *Proceedings of SIGDIAL*, 2013.

[122] A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 1998.