

Collaborative Models for Referring Expression Generation in Situated Dialogue

Rui Fang, Malcolm Doering and Joyce Y. Chai

Department of Computer Science and Engineering

Michigan State University

East Lansing, Michigan 48824

{fangrui, doeringm, jchai}@cse.msu.edu

Abstract

In situated dialogue with artificial agents (e.g., robots), although a human and an agent are co-present, the agent's representation and the human's representation of the shared environment are significantly mismatched. Because of this misalignment, our previous work has shown that when the agent applies traditional approaches to generate referring expressions for describing target objects with minimum descriptions, the intended objects often cannot be correctly identified by the human. To address this problem, motivated by collaborative behaviors in human referential communication, we have developed two collaborative models - an episodic model and an installment model - for referring expression generation. Both models, instead of generating a single referring expression to describe a target object as in the previous work, generate multiple small expressions that lead to the target object with the goal of minimizing the collaborative effort. In particular, our installment model incorporates human feedback in a reinforcement learning framework to learn the optimal generation strategies. Our empirical results have shown that the episodic model and the installment model outperform previous non-collaborative models with an absolute gain of 6% and 21% respectively.

Introduction

Referring Expression Generation (REG) has traditionally been formulated as a problem of generating a single noun phrase (possibly with multiple modifiers and prepositional phrases) that can uniquely describe a target object among multiple objects (Dale 1995; Krahmer and Deemter 2012) so that addressees (i.e., humans) can correctly identify the intended object given this expression. Although well studied, most existing REG approaches were developed and evaluated under the assumption that humans and agents have access to the same kind of domain information.

However, this assumption no longer holds in situated dialogue with robots. In situated dialogue, robots and humans have different representations of the shared environment because of their mismatched perceptual capabilities. The robot's representation of the shared environment is often incomplete and error-prone. When a shared perceptual

basis is missing, referential communication between partners becomes difficult (Clark and Brennan 1991). Specifically for the task of REG, our previous work (Fang et al. 2013) has shown that when the human and the agent have a mismatched perceptual basis traditional approaches to REG tend to break down. A competitive algorithm that achieves over 80% accuracy (in referent identification) in a traditional setting only obtains over 40% accuracy under a mismatched perceptual basis (Fang et al. 2013). This huge performance drop calls for new approaches to REG that take into account the mismatched perceptual basis in situated dialogue.

To address this issue, motivated by collaborative behaviors in human-human referential communication (Clark and Wilkes-Gibbs 1986; Clark and Brennan 1991; Clark and Bangerter 2004) and previous computational models for collaborative references (Heeman and Hirst 1995; DeVault et al. 2005), we have developed two collaborative models for REG: an episodic model and an installment model. Instead of generating a single noun phrase (i.e., an elementary referring expression) to describe a target object as in the previous work (Dale 1995; Krahmer and Deemter 2012), our models generate multiple small noun phrases that gradually lead to the target object with the goal of minimizing the collaborative effort. In particular, our installment model incorporates human feedback in a reinforcement learning framework to learn the optimal generation strategies. Our empirical results have shown that the episodic model outperforms previous non-collaborative models with an absolute gain of 6%, and our installment model, by incorporating human feedback, achieves an absolute gain of 21%.

Background and Related Work

Referring Expression Generation. Traditional approaches for REG focus on generating a single noun phrase (with a minimum description) that uniquely describes a referent object (Dale 1995; Krahmer and Deemter 2012). Many methods have been developed including the incremental algorithm (Dale 1995), the graph-based approach (Krahmer, van Erk, and Verleg 2003), and two recent approaches that can generate a distribution of referring expressions for a referent object (Mitchell, van Deemter, and Reiter 2013; FitzGerald, Artzi, and Zettlemoyer 2013), etc.

REG in Situated Dialogue. Recently, there is an increas-

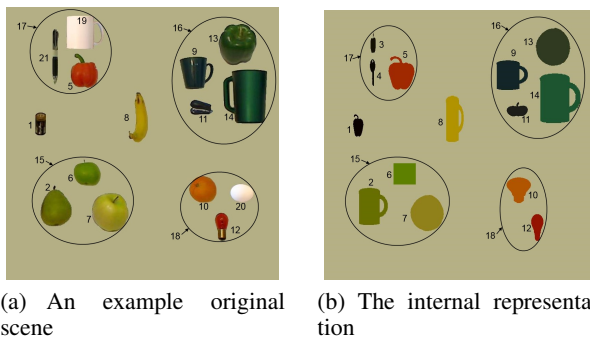


Figure 1: An original scene shown to the human and the re-rendering of the robot’s internal representation from (Fang et al. 2013)

ing interest in REG for situated dialog (Stoia et al. 2006; Kelleher and Kruijff 2006; Garoufi and Koller 2010; Striegnitz et al. 2011; Dethlefs, Cuayhuitl, and Viethen 2011). While traditional approaches work well in situated dialogue in a virtual world, they are not adequate to support situated dialogue in a physical world because of the mismatched perceptual basis between the human and the agent. To investigate the problem of REG under the mismatched perceptual basis, (Fang et al. 2013) conducted a study using artificial scenes (an example is shown in Figure 1(a)). The original scene is what a human sees, and the corresponding impoverished scene is a re-rendering of the robot’s internal representation of the same scene, created by processing the original scene with a computer vision algorithm (Zhang and Lu 2002) and a perceptual grouping algorithm (Gatt 2006). Each object/group has an ID associated with it for the identification purpose. This example demonstrates the perceptual differences between the human and the agent. For example, in the impoverished scene, some objects (e.g., object 19, 20) are missing or mis-recognized (e.g., object 2, 10), and perceptual groupings are also different (e.g., group 18 contains only 2 objects). Using these scenes, Our previous approach incorporated grouping and perception uncertainties into REG. Our experimental results have shown that, although our approach performs better than the leading approach based on regular graphs (Krahmer, van Erk, and Verleg 2003), our approach only achieved 45% accuracy (in referential identification) under the mismatched perceptual basis. Our results indicate that, if the agent applies traditional approaches to generate referring expressions, in situations where the shared perceptual basis is missing, the intended objects often cannot be correctly identified by the human. Inspired by our findings, we have developed collaborative models for REG particularly to address the mismatched perceptual basis. We use the same scenes and target objects used in (Fang et al. 2013) in our evaluation in order to have a valid comparison.

Collaborative Behaviors in Referential Communication.

Previous psycholinguistic studies have indicated that referential communication is a collaborative process (Clark and Wilkes-Gibbs 1986). To minimize the collaborative effort,

partners tend to go beyond issuing an elementary referring expression (i.e., a single noun phrase), by using other different types of expressions such as *episodic*, *installment*, *self-expansion*, etc. These collaborative behaviors from human-human referential communication have motivated previous computational models for collaborative references (Heeman and Hirst 1995; DeVault et al. 2005). Compare to these previous computational models, here we apply different approaches to collaborative models for REG with the specific goal of mediating visual perceptual differences.

More specifically, among nine different types of referring behaviors identified in (Clark and Wilkes-Gibbs 1986; Clark and Bangerter 2004), we are particularly interested in *episodic descriptions* and *installment descriptions*. Unlike a single elementary description to describe a target object, an episodic description is produced in two or more easily distinguished episodes or intonation units. Here is an example of an episodic description from (Liu et al. 2013).

A: below the orange, next to the apple, it’s the red bulb.

An installment behavior is similar to the episodic behavior in the sense that it also breaks down generation into smaller episodes. The difference is that an explicit feedback from the addressee is solicited before the speaker moves to the next episode. Here is an example of an installment from (Liu et al. 2013).

A: under the pepper we just talked about.

B: yes.

A: there is a group of three objects.

B: OK.

A: there is a yellow object on the right within the group.

The generation of episodic or installment descriptions is not to minimize the speaker’s own effort, but rather to minimize the collaborative effort so that the addressee can quickly identify the referent.

Collaborative Models for REG

Inspired by the episodic behavior and installment behavior used in human-human dialogue, we developed two collaborative computational models for REG:

- **Episodic Model** generates referring expressions (REs) in an “episodic” fashion: it generates a RE in a sequence of smaller noun phrases which lead to the target object.
- **Installment Model** generates REs in an “installment” fashion: it generates one small noun phrase, waits for partner’s response, and then generates another small noun phrase. This process iterates until the target object is reached.

For both models the goal is to construct a sequence of objects to describe, where the target object is the final object to be described in the sequence. Both models can choose to directly describe the target object (as the traditional REG methods do) if such a choice is deemed to have the lowest overall cost. But in general these models often find an object that is “easier” to describe and then gradually lead to the target object.

For example, in Figure 1 suppose the robot wants to describe the target object 7. Note that it is hard to describe this object directly based on its features. So both models will search for a sequence of objects that lead to the target. In the episodic model, a sequence could be *object 5* \rightarrow *group 15* \rightarrow *object 7*, which could be linguistically realized as: “a red pepper, below that a group of 3 objects, and a yellow object on the right within the group”. In the installment model, once an object (e.g., object 5) is chosen, it will ask the human to provide feedback. Then based on the feedback, it will search for another object and iterate this process until the target object is reached.

Episodic Model

The episodic model is based on the Branch-and-Bound search method (Morin and Marsten 1974). It searches for the path to a target object with the overall least cost. We represent our problem space as a directed graph $G = (N, A)$, in which

$$N = \{n_i\}$$

$$A = \{a_j = \langle t_j, h_j \rangle \mid t_j, h_j \in N\}$$

N is a set of nodes, and A is a set of arcs. Each node $n \in N$ represents a perceived object or a perceptual grouping of objects by the agent, together with one of all possible concatenations of linguistic descriptors describing attributes (e.g., type, color, type with color, etc.). Each descriptor comes with a cost, which indicates how accurately the linguistic descriptor matches the underlying physical features. We use the “Uncertainty Relative Preferred” cost functions from (Fang et al. 2013)

A path in graph G is a sequence of nodes $\langle n_0, \dots, n_k \rangle$ such that $\langle n_{i-1}, n_i \rangle \in A$. The cost of a path is the sum of the cost of all the nodes and arcs on the path:

$$cost(\langle n_0, \dots, n_k \rangle) = \sum_{i=0}^k cost(n_i) + \sum_{i=1}^k cost(\langle n_{i-1}, n_i \rangle)$$

Algorithm 1 details our episodic model for generating episodic expressions. The algorithm starts with an initial state in which the path contains only the root node n_0 , the best path is none \perp , and the bound is set to infinity. The algorithm then recursively extends the path $\langle n_0 \rangle$ until the minimum cost path is found. The Boolean function $isDiscriminating()$ measures whether a path can uniquely identify object n_k in the path. The Boolean function $isGoal()$ tests whether n_k is the target node.

Installment Model

In the Installment model, we treat the collaborative referring expression generation task as a sequential decision making problem and formalize it under the reinforcement learning framework. Our goal is to learn a good strategy that can construct a series of objects in the scene which will lead to the successful identification of a target object.

State The state set is denoted as S , $s \in S$. Let O be the set of all perceived objects and perceptual groups in the scene. A state $s = \langle tar, lm, W \rangle$ of our sequential decision making problem contains the target object $tar \in O$, the landmark

```

Algorithm generateEpisodic()
   $G \leftarrow (N, A)$ 
   $bestPath \leftarrow \perp$ 
   $n_0 \leftarrow root\ node$ 
   $bound \leftarrow \infty$ 
  Return Search( $G, bestPath, bound, \langle n_0 \rangle$ )

Procedure Search( $G, bestPath, bound, \langle n_0, \dots, n_k \rangle$ )
  if  $\langle n_0, \dots, n_k \rangle$  only contains  $n_0$  then
    foreach  $\langle n_0, n_1 \rangle \in A$  do
      | Search( $G, bestPath, bound, \langle n_0, n_1 \rangle$ )
    end
  end
  if  $cost(\langle n_0, \dots, n_k \rangle) < bound$  And
   $isDiscriminating(\langle n_0, \dots, n_k \rangle)$  then
    if  $isGoal(n_k)$  then
       $bestPath \leftarrow \langle n_0, \dots, n_k \rangle$ 
       $bound \leftarrow cost(\langle n_0, \dots, n_k \rangle)$ 
    else
      foreach  $\langle n_k, n_{k+1} \rangle \in A$  do
        | Search( $G, bestPath, bound, \langle n_0, \dots, n_k, n_{k+1} \rangle$ )
      end
    end
  end
  Return  $bestPath$ 

```

Algorithm 1: Episodic Model

$lm \in O$, which is confirmed by user (e.g., when the user accepts the description of the landmark object), and a set of objects W in the scene which contains the objects that have not been used but can potentially be used as landmark objects: $W = \{\langle w_1, RE_1 \rangle, \dots, \langle w_n, RE_n \rangle\}$, where $w_i \in O$, RE_i is the set of generation strategies (to be explained in the **Action** section) for w_i . In the initial state lm is none, and W contains all objects in O and their generation strategies. When a user accepts the description (a generation strategy $re_i \subset RE_i$) of an object or a group, it becomes the landmark object lm . When a user rejects the description of an object or a group, the description re_i is removed from RE_i in W . In the terminal state, the target object is reached and becomes the current landmark object $tar = lm$.

Action An action in our problem basically describes an object in relation to a landmark object. The action set is denoted as A , $a \in A$. $a = \langle o, re, sp \rangle$ is composed of an object to be described $o \in O$, its generation strategy $re \subset RE$, and a spatial relation $sp \in SP$ to the landmark object lm . Currently, RE represents the strategies that can be used to describe an object. The space of RE consists of all possible concatenations of the following dimensions:

- desType: describes the type of the object.
- desColor: describes the color of the object.
- desSize: describes the size such as “big” or “small”.
- desSpaLoc: describes the spatial location with respect to the whole scene, e.g., “on the left”, “on the right”, etc.
- desNumGroup: describes the number of objects within a group.

SP represents the strategies that can be used to describe a spatial relation between two objects, between two perceptual groups, between one object and a perceptual group or between one object and the group it belongs to. The space of SP can be one of the following dimensions:

- **desBetween**: describes the spatial relation between two objects/groups, e.g., “below”, “above”, “to the left of”, and “to the right of”.
- **desInGroup**: describes the spatial relation between one object and the group it belongs to, e.g., “on the right within”, “on the left within”, “on the bottom within”, “on the top within”.

For example, an action sequence could be $\langle \text{object 5, [desColor, desType], [none]} \rangle \rightarrow \langle \text{group 15, [desNumGroup], [desBetween]} \rangle \rightarrow \langle \text{object 7, [desColor], [desInGroup]} \rangle$. These actions capture the decisions on how to generate an expression to describe an intended object. Given an action, the surface form of an expression can be generated through some templates.

Transition $T(s, a)$ is the transition function. It takes an action $a = \langle o, re, sp \rangle$ in the current state $s = \langle tar, lm, W \rangle$ and gives the next state $s' = \langle tar, lm', W' \rangle$. Note that the action does not change the target tar . Rather the landmark object and W' are affected. Given the expression generated by a particular action, the human’s response will lead to the update of the landmark object lm' , and the remaining objects that can be used as landmark objects in the future W' .

Reward The reward function $r(s, a) : S \times A \rightarrow \mathbb{R}$ specifies a numeric reward that an agent receives for taking action a in state s . It is defined as follows:

$$r = \begin{cases} 100 & \text{If the terminal state is reached and the target object is correctly identified.} \\ 10 & \text{If the terminal state is not reached and the current object is correctly identified.} \\ -1 & \text{Otherwise.} \end{cases}$$

Policy A policy $\pi : S \rightarrow A$ is a mapping from states to actions. We can determine the expected return of a policy by estimating its action-value function. The action-value function $Q^\pi(s, a)$ of a policy π is the expected return for starting in state s , taking action a , and then following π . The optimal action-value function that yields the best expected return for each action a taken in state s is defined as

$$Q^*(s, a) = \max_{\pi} Q^\pi(s, a)$$

The optimal policy π^* is therefore the policy that chooses the action a that maximizes $Q(s, a)$ for all $s \in S$,

$$\pi^*(s) = \arg \max_a Q(s, a)$$

Basic reinforcement learning approaches use a table to represent the action-values. However, as the number of states in an environment increases, it becomes infeasible for an agent to visit all possible states enough times to find the optimal actions for those states. Furthermore, it becomes important to be able to generalize the learning experiences in a particular state to the other states in the environment. One common way to handle the large state space and generalization issue is through function approximation (Sutton and Barto 1998).

We define a mapping ϕ that assigns a feature vector to each state-action pair. Then, the action value $Q(s, a)$ of a state-action pair (s, a) is obtained by linearly combining the components of $\phi(s, a)$ with the weights θ :

$$Q(s, a) = \theta^T \phi(s, a)$$

#	Feature Value Description	Learned Weights
1	normalized sum of vision confidence of all descriptors in re	0.92
2	is spatial location descriptor in re ?	0.54
3	visual confidence of spatial relation descriptor	0.52
4	vision confidence of type descriptor	0.51
5	vision confidence of spatial location descriptor	0.48
6	is type descriptor in re ?	0.21
7	number of descriptors in re	0.19
8	vision confidence of size descriptor	0.13
9	is size descriptor in re ?	0.13
10	is color descriptor in re ?	0.10
11	vision confidence of color descriptor	0.09
12	can re together with sp to lm uniquely identify an object?	0.88
13	is there a sp between o and tar ?	0.51
14	number of spatial links from lm	0.23
15	number of spatial links to o (in degree)	0.01
16	number of spatial links from o (out degree)	0.01
17	is o and lm in the same group?	1
18	is o a group?	0.97
19	is lm is a group and o in lm ?	0.96
20	is o a group and tar in o ?	0.90
21	is o and tar in the same group?	0.51
22	is o a group and lm in o ?	0.11

Table 1: Features used in the installment model. We use the following abbreviations, o : an object to describe, re : generation strategy for an object, lm : landmark object, tar : the target object, sp : spatial relation

Here we represent the Q -function as a linear combination of the features, and learn the weights which most closely approximate the true expected reward.

Features We use features from both the state and the action to model the Q -function as summarized in Table 1. Features 1-11 are inspired by (Fang et al. 2013) which demonstrate that encoding visual confidence of a symbolic descriptor into the cost function improves performance. Feature 12 models the discriminating power of a referring expression, which is the key requirement for traditional REG approaches. Features 13-16 measure the spatial relations between objects. Features 17-22 are inspired by previous work indicating that group descriptions are important to REG (Funakoshi et al. 2004; Fang et al. 2013).

Learning To learn these weights θ we follow the method in (Vogel and Jurafsky 2010), which uses SARSA (Sutton and Barto 1998) with linear function approximation. The learning model is shown in Algorithm 2. We use $\Pr(a_0|s_0; \theta) = \frac{\exp(\theta^T \phi(s_0, a_0))}{\sum_{a'} \exp(\theta^T \phi(s_0, a'))}$ to choose the best action based on the current estimation of θ , with ϵ -greedy ($\epsilon = 0.2$) for the exploration (meaning 20% of the time, we randomly choose an action). The learning rate α_t is set to $\frac{30}{30+t}$ and we stop training when the magnitude of updates $\|\theta_{t+1} - \theta_t\|$ is smaller than 0.0001.

Input : Object identification task set I
 Feature function ϕ
 Transition function T
 Reward function $r(s, a)$
 Learning rate α_t

Initialize θ to small random values

```

while  $\theta$  not converge do
  foreach task  $i \in I$  do
    Initialize  $s$  to initial state
    Select  $a \sim \Pr(a|s; \theta)$  using  $\epsilon$  greedy
    while  $s$  not terminal do
       $s' = T(s, a)$ 
      Select  $a' \sim \Pr(a'|s'; \theta)$  using  $\epsilon$  greedy
       $\Delta \leftarrow r(s, a) + \theta^T \phi(s', a') - \theta^T \phi(s, a)$ 
       $\theta \leftarrow \theta + \alpha_t \phi(s, a) \Delta$ 
    end
  end
end

```

Output: Estimate of feature weights θ

Algorithm 2: Installment Model

Empirical Evaluations

Experimental Setup

To evaluate the performance of both the episodic model and the installment model for REG, we conducted an empirical study using crowd-sourcing from the Amazon Mechanical Turk. Through a web interface we displayed an original scene and a description generated to refer to an object in the scene. We asked each turker to choose the object he/she believed was referred to by the description. They can also choose if none of the objects or multiple objects were considered to be referred to. Similar to (Fang et al. 2013), each description received three votes regarding its referent from the crowd¹. The referent with a majority voting was taken as the identified referent and was used to calculate the performance metric: the accuracy of referent identification (i.e., the percentage of generated referring expressions where the target object is correctly identified). If all three votes were different, then the referent was evaluated as not correctly identified for that expression.

Training of the Installment Model

In order to learn the weights for features in the installment model, we first created 32 different training scenes similar to the scenes used in (Fang et al. 2013), then used the Amazon Mechanical Turk to solicit feedback from the crowd. The training was divided into sessions where each session was used to identify only one target object. More specifically, in each session, the system applied Algorithm 2 to pick an action. Then a referring expression was generated based on this action and shown to the user. The user was then asked to identify the referent based on this expression. Based on the user’s feedback, the internal state would be updated and the system would pick the next action accordingly. This process

¹To have a fair comparison, we use the same quality control of crowdsourcing as used in (Fang et al. 2013).

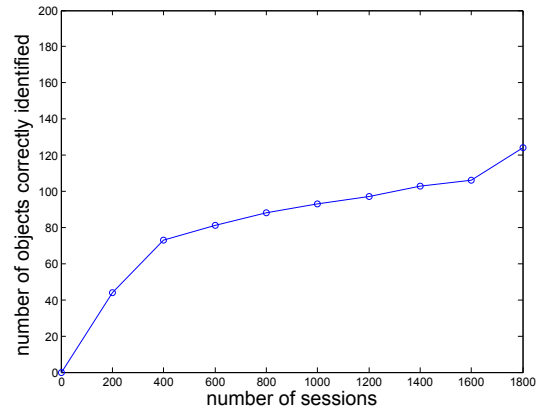


Figure 2: The number of objects correctly identified at an interval of every 200 training sessions

iterated until one of the following three conditions was met: (1) The system reaches the target object and describes it. (2) There is no more action available for the system to take. (3) The number of interactions exceeds 13, which is the average number of objects in a scene. When one training session ended, a new training session for a new object would start. We stopped training when the algorithm converged. We had a total number of 1860 training sessions and a total number of 6775 interactions with the user, which resulted in an average of 3.6 interactions per session.

Figure 2 shows the number of objects correctly identified during training at an interval of 200 sessions. From Figure 2, we observe a trend that the number of correctly identified objects gradually increases, indicating that the system is learning in the right direction toward the true underlying parameter θ . Note that we stopped the training when the algorithm converged. However, as shown in Figure 2, the performance seems to pick up during the last 200 sessions. This suggests that further training may improve the performance.

The weights learned for each feature are shown in the third column of Table 1. We can observe that group-based features (e.g., features 17 - 20) are assigned relatively higher weights. Through the interaction with the user, the system learns that a strategy of describing an object with reference to groups has a higher probability of correctly identifying the target object. This learning result is consistent with previous work showing that group descriptions are important. However, in contrast to previous work, our system can learn the importance of the group descriptions through interaction and gradually assign higher weights to them. Visual confidence (feature 1) and the discriminating power (feature 12) of a referring expression also receive relatively high weights.

Evaluation Results

In the testing phase, we applied the learned parameters to the testing scenes and evaluated how well the turkers² can

²We randomly recruit the turkers from Amazon Mechanical Turk using the same quality control criteria. The turkers are presented with the same interface as in the training phase.

	Accuracy
Non-collaborative model (Fang et al. 2013)	47.2%
The episodic model	53.6%
The installment model	68.9%

Table 2: Comparison of accuracy for referent identification based on expressions generated from three models.

identify target objects based on the expressions generated by the learned policy. We used the same 48 scenes used in (Fang et al. 2013) for evaluation. Each testing scene has 13 objects on average and there are 583 objects in total³. The turkers went through all the testing scenes and we recorded the number of responses the turkers made while identifying the target objects, as well as the total number of target objects that were correctly identified. Each expression received three votes from the crowd and the referent with the majority voting was used in calculating the accuracy.

Table 2 shows the results of our two collaborative models compared with the non-collaborative model (Fang et al. 2013). Table 2 shows that two collaborative models significantly outperform the non-collaborative approach with an absolute performance gain of over 6% and 21% respectively. The installment model further significantly outperforms the episodic model with an absolute performance gain of 15% (Pearson’s Chi-square test, $\chi^2 = 20.11$, $p < 0.001$ with Tukey’s Post Hoc test). The average number of turns in the installment model is 2.35 (Standard Deviation = 1.07).

Comparison of the Episodic Model with the Non-Collaborative Model. The episodic model outperforms the non-collaborative model. This is mainly due to two reasons. First, instead of directly describing the target object, the episodic model provides more options to describe the target object, and these options have less cost than the direct description of the target object. In the episodic model 72% of the target objects are not described directly (i.e., one long noun phrase to describe the target), which indicates that, in these cases describing other objects first and then gradually approaching the description of the target object has a lower overall cost. The second reason is due to the complexity of interpreting a description. The average number of descriptors in a referring expression for the non-collaborative model (Mean=4.3, SD=1.23) is larger than that in each small noun phrase in the episodic model (Mean=2.2, SD=0.78) (t-test, $p < 0.001$). In the non-collaborative model, the target object can be described in relation to other objects. Although as a whole the overall description should be able to distinguish the target from the rest of the objects, the expressions to describe each of the objects (in relation to the target) do not need to be distinguishing. In contrast, the smaller descriptions generated by the episodic model can already distinguish the intended object from the rest. We believe these behaviors contribute to less complexity in interpreting episodic expressions.

³Here we remove 38 split objects from the target object set, so the adjusted result shown in Table 2 is a little different from the reported results in (Fang et al. 2013).

Comparison of the Installment Model with the Episodic Model. There are two main reasons that the installment model outperforms the episodic model. First, the installment model incorporates many more features than the episodic model, and the weight of those features are learned through on-line interaction with users. The episodic model only relies on two features when searching for the best path to a target object: vision uncertainty and the discriminating power of a path. These two features roughly corresponds to feature 1 and 12 in Table 1. Second, in the episodic model there is no intermediate feedback from humans, whereas in the installment model the system selects the next object to describe based on the intermediate feedback from the human partner. Among all the successful sessions (where the target object is correctly identified), 18.7% of them in fact encountered some problems. (The turkers could not identify the object at the intermediate step.) However, the system was able to get around the problem by choosing to describe another object. It is the intermediate feedback that guides the installment model to generate referring expressions that lead to the correct identification of target objects.

Conclusion and Future Work

In situated dialogue, humans and robots have mismatched perceptions of the shared world. To facilitate successful referential communication between a human and a robot, the robot needs to take into account such discrepancies and generate referring expressions that can be understood by its human partner. Motivated by collaborative behaviors in human-human referential communication, we developed two collaborative models for REG. In contrast to previous non-collaborative models which tend to generate a single long description to describe a target object, our models generate multiple short expressions describing easier-to-identify landmark objects that eventually lead to the target object. The goal of these models is to minimize the collaborative effort between the human and the robot. Our empirical results have shown that the two collaborative models significantly outperform the non-collaborative model.

Although our current models, especially the installment model, have yielded encouraging results, several problems need to be addressed in the future. First, the cost function we used for each descriptor is predefined. As the environment changes, the agent may have different confidence in capturing different dimensions of perception. Thus, one future direction is to automatically learn and adjust these cost functions in new environments. Furthermore, our current evaluation is only based on the simplified scenes without addressing complexities in true human-robot dialogue. Our future work will extend the current work to real human-robot dialogue and incorporate non-verbal modalities (e.g., gaze direction from the human) as intermediate feedback for generating referring expressions.

Acknowledgement

This work was supported by N00014-11-1-0410 from the Office of Naval Research and IIS-1208390 from the National Science Foundation. We also thank anonymous re-

viewers for their valuable comments and suggestions.

References

- Clark, H., and Bangerter, A. 2004. *Changing ideas about reference*. Experimental pragmatics. Palgrave Macmillan. 25–49.
- Clark, H., and Brennan, S. 1991. Grounding in communication. *Perspectives on socially shared cognition* 13:127–149.
- Clark, H. H., and Wilkes-Gibbs, D. 1986. Referring as a collaborative process. *Cognition* 22:1–39.
- Dale, R. 1995. Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive Science* 19:233–263.
- Dethlefs, N.; Cuayhuil, H.; and Viethen, J. 2011. Optimising natural language generation decision making for situated dialogue. In *The 12th annual SIGdial Meeting on Discourse and Dialogue*, 78–87.
- DeVault, D.; Kariaeva, N.; Kothari, A.; Oved, I.; and Stone, M. 2005. An information-state approach to collaborative reference. In *Proceedings of the ACL 2005 on Interactive Poster and Demonstration Sessions*.
- Fang, R.; Liu, C.; She, L.; and Chai, J. Y. 2013. Towards situated dialogue: Revisiting referring expression generation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 392–402. Seattle, Washington, USA: Association for Computational Linguistics.
- FitzGerald, N.; Artzi, Y.; and Zettlemoyer, L. 2013. Learning distributions over logical forms for referring expression generation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 1914–1925. Seattle, Washington, USA: Association for Computational Linguistics.
- Funakoshi, K.; Watanabe, S.; Kuriyama, N.; and Tokunaga, T. 2004. Generation of relative referring expressions based on perceptual grouping. In *20th International Conference on Computational Linguistics*.
- Garoufi, K., and Koller, A. 2010. Automated planning for situated natural language generation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 1573–1582. The Association for Computer Linguistics.
- Gatt, A. 2006. Structuring knowledge for reference generation: A clustering algorithm. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics*, 321–328.
- Heeman, P. A., and Hirst, G. 1995. Collaborating on referring expressions. *Computational Linguistics* 21:351–382.
- Kelleher, J. D., and Kruijff, G.-J. M. 2006. Incremental generation of spatial referring expressions in situated dialog. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, ACL-44*, 1041–1048. Stroudsburg, PA, USA: Association for Computational Linguistics.
- Krahmer, E., and Deemter, K. V. 2012. Computational generation of referring expressions: A survey. *computational linguistics* 38(1):173–218.
- Krahmer, E.; van Erk, S.; and Verleg, A. 2003. Graph-based generation of referring expressions. *Computational Linguistics* 29(1):53–72.
- Liu, C.; Fang, R.; She, L.; and Chai, J. 2013. Modeling collaborative referring for situated referential grounding. In *Proceedings of the SIGDIAL 2013 Conference*, 78–86. Metz, France: Association for Computational Linguistics.
- Mitchell, M.; van Deemter, K.; and Reiter, E. 2013. Generating expressions that refer to visible objects. In *Proceedings of NAACL-HLT 2013, pages 1174-1184*.
- Morin, T. L., and Marsten, R. E. 1974. *Branch-and bound strategies for dynamic programming*. MIT.
- Stoia, L.; Shockley, D. M.; Byron, D. K.; and Fosler-Lussier, E. 2006. Noun phrase generation for situated dialogs. In *Proceedings of the Fourth International Natural Language Generation Conference*, 81–88. Sydney, Australia: Association for Computational Linguistics.
- Striegnitz, K.; Denis, A.; Gargett, A.; Garoufi, K.; Koller, A.; and Theune, M. 2011. Report on the second second challenge on generating instructions in virtual environments (give-2.5). In *Proceedings of the Generation Challenges Session at the 13th European Workshop on Natural Language Generation, Nancy, France*, 270–279. Nancy, France: Association for Computational Linguistics.
- Sutton, R. S., and Barto, A. G. 1998. *Introduction to Reinforcement Learning*. Cambridge, MA, USA: MIT Press, 1st edition.
- Vogel, A., and Jurafsky, D. 2010. Learning to follow navigational directions. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL '10*, 806–814. Stroudsburg, PA, USA: Association for Computational Linguistics.
- Zhang, D., and Lu, G. 2002. An integrated approach to shape based image retrieval. In *Proc. of 5th Asian Conference on Computer Vision (ACCV)*, 652–657.