Noun Phrase Generation for Situated Dialogs

Laura Stoia and **Darla Magdalene Shockley** and **Donna K. Byron** and **Eric Fosler-Lussier**

The Ohio State University
Computer Science and Engineering
2015 Neil Ave., Columbus, Ohio 43210
`stoia|shockley|dbyron|fosler@cse.ohio-state.edu`

Abstract

We report on a study examining the generation of noun phrases within a spoken dialog agent for a navigation domain. The task is to provide real-time instructions that direct the user to move between a series of destinations within a large interior space. A subtask within sentence planning is determining what form to choose for noun phrases. This choice is driven by both the discourse history and spatial context features derived from the direction-follower's position, e.g. his view angle, distance from the target referent and the number of similar items in view. The algorithm was developed as a decision tree and its output was evaluated by a group of human judges who rated 62.6% of the expressions generated by the system to be as good as or better than the language originally produced by human dialog partners.

1 Introduction

In today's world of mobile, context-aware computing, intelligent software agents are being deployed in a wide variety of domains to aid humans in performing navigation tasks. Examples include hand-held tourist information portals (Johnston et al., 2002) campus tour guides (Yang et al., 1999; Long et al., 1996; Striegnitz et al., 2005), direction-giving avatars for visitors to a building (Cassell et al., 2002; Chou et al., 2005), in-car driving direction systems (Dale et al., 2003; Wahlster et al., 2001), and pedestrian navigation systems (Muller, 2002). These applications present an exciting and challenging new frontier for dialog agents, since attributes of the real-world setting must be combined with other contextual factors for the agent to communicate successfully.

In the current work, we focus on a scenario in which the system provides incremental directions to a mobile user who is following the instructions as they are produced. Unlike the rigid directions produced by applications like Mapquest,[1] which describes the entire route from start to finish, this task requires realtime instructions issued while monitoring the user's progress. Instructions are based on dynamic local context variables such as the visibility of and distance to reference points. In referring to items in the setting, human speakers produce a wide variety of noun phrase forms, including descriptions that are headed by a common noun and that employ a definite, indefinite, or demonstrative determiner, *one* anaphors, and pronouns such as *it*, *this* and *that*. Our goal in the current work is to model that entire space of variation, which makes the task more difficult than the noun phrase generation task defined in many previous studies that simplify the alternatives down to *description* or *pronoun*.

In order to study this process, we developed a task domain in which a human partner is directed through an interior space (a graphically-presented 3D virtual world) to perform a sequence of manipulation tasks. In the first stages of the work, we collected and annotated a corpus of human-human dialogs from this domain. Then, using this data, we trained a decision-tree classifier to utilize context variables such as distance, target object visibility, discourse history, etc., to determine lexical properties of referring expressions to be produced by the generation component of our dialog system.

2 Generation for Situated Tasks

Many previous projects, such as (Lauria et al., 2001; Moratz and Tenbrink, 2003; Skubic et al., 2002), *inter alia*, study interpretation of situated language, e.g. for giving directions to a robot. The focus of our work is rather on generating navigation instructions for a human partner to follow.

Linguistic studies have shown that speakers select noun phrase forms to refer to entities based on a variety of factors. Some of the factors are intrinsic to the object being described, while others are features of the context in which the expression is spoken. The entity's status within the discourse,

---

[1]www.mapquest.com

spatial position, and the presence of similar items from which the target referent must be distinguished, have all been found to cause changes to the lexical properties chosen for a particular referring expression (i.e. (Gundel et al., 1993; Prince, 1981; Grosz et al., 1995)). This variation is expressed in terms of the determiner chosen (e.g. *that/a*), the head noun (e.g. *that/door/one*), and the presence of additional modifiers such as prenominal adjectives or prepositional phrases.

In natural language generation, the process of generating referring expressions occurs in stages (Reiter and Dale, 1992). The process we explore in this paper is the sentence planning stage, which determines whether the context supports generating a particular referring expression as a pronoun, description, one-anaphor, etc.

There has been extensive research in both automatic route description and on general noun phrase (NP) generation, but few projects consider extra-linguistic information as part of the context that influences dialog behavior. (Poesio et al., 1999) applies statistical techniques for the problem of NP generation. However, even though the corpus used in that study contains descriptions of museum items visually accessible to the user, the features used in generation were mostly linguistic, and included little information about the visual or spatial properties of the referent. Another related study in statistical NP generation (Cheng et al., 2001) focuses on choosing the modifiers to be included. Again, no features derived from the situated world were used in that study. (Maass et al., 1995) use features from the world, including objects' color, height, width, and visibility, as well as the user's direction of travel and distance from objects, for generating instructions in a situated task. However, their focus is on selecting landmarks and descriptions under time pressure, rather than selecting the linguistic form to be produced.

3 Data Collection

Our task setup is designed to elicit natural, spontaneous situated language examples from human partners. The experimental platform employs a virtual-reality (VR) world in which one partner, the direction-follower (DF), moves about to perform a series of tasks, such as pushing buttons to re-arrange objects in the room, finding and picking up treasures, etc. The simulated world was presented from first-person perspective on a desk-top computer monitor. The DF had no knowledge of the world map or tasks.

DG:	you can currently see **three buttons**... there's actually **a fourth button that's kind of hidden**
DF:	yeah
DG:	by **this cabinet on the right**
DF:	I know, yeah
DG:	ok, um, so what you wanna do is you want to go in and you're gonna press **one of the buttons that's on the right hand wall**, so you wanna go all the way straight into the room and then face the wall
DF:	mhm
DG:	there with **the two buttons**
DF:	yep
DG:	um and you wanna push **the one that's on the left**

Figure 1: Sample dialog fragment and accompanying video frame

His partner, the direction-giver (DG), had a paper 2D map of the world and a list of tasks to complete. As they performed the task, the DG had instant feedback about the DF's location in the VR world, via mirroring of the DF's computer screen on the DG's computer monitor. The partners communicated through headset microphones. Our paid participants were self-identified native speakers of North American English. Figure 1 shows an example view of the world and the accompanying dialog fragment.

The video output of DF's computer was captured to a camera, along with the audio from both microphones. A logfile created by the VR software recorded the DF's coordinates, gaze angle, and the position of objects in the world 10 times per second. These data sources were synchronized using calibration markers. A technical report is available that describes the recording equipment and software used (Byron, 2005).

3.1 Corpus and Annotation Scheme

Using the above-described setup, we created a corpus consisting of 15 dialogs containing a total of 221 minutes of speech. It was transcribed and word-aligned using Praat [2] and SONIC.[3] The dialogs were further annotated using the Anvil software (Kipp, 2004) to identify a set of target referring expressions in the corpus. Because we are in-

---

[2] http://www.praat.org
[3] http://cslr.colorado.edu/beginweb/speech_recognition/sonic.html/

terested in the spatial properties of the referents of these target referring expressions, the items of interest in this experiment were restricted to objects with a defined spatial position.

Each object in the virtual world was assigned a symbolic id, and the id of each target referring expression was added to the annotation. Referring expressions with plural referents were marked as **Set**, and were labeled with a list of the members in the set. Expressions were also annotated as either *vague* when the referent was not clear at the time of utterance or *abandoned* in case the utterance was cut short. Items that did not contain a surface realization of the head of the NP (e.g., *on the left*), were marked with the tag *empty*.

The corpus contains 1736 target expressions, of which 221 were **Vague**, 45 were **Empty**, and 228 were **Set**s. The remaining 1242 form the set of test items in the experiment described below. **Vague** items were excluded since we do not wish for the algorithm we develop to reproduce this behavior. **Set** items were excluded in order to avoid the more complex calculation of spatial properties associated with plural entities.

The data used in the experiments is a consensus version on which both annotators, two of the authors, agreed on the set of target expressions and their properties. Due to the constraints introduced by the task, referent annotation achieved almost perfect agreement. The data used in this study is only the DG's language.

4 Algorithm Development

Our ultimate goal is to provide input to a surface realization component for NP generation, given the ID of a target referent and a vector of context features. It is desirable for these context features to be automatically derived, to limit the reliance on human annotation, so we restricted out study to features that either were derived automatically, or required minimal human annotation.

One impact of this decision is that even though the linguistic literature predicts that syntactic features such as grammatical role are important in selecting NP forms, these features were difficult to obtain. Our corpus contains spontaneous spoken discourse, which has no sentence boundaries and relaxed structural constraints. Thus, automatic parsing was problematic. With improved parsing techniques, we may include syntactic information in the decision process for NP generation in future, but this was not included in the current study.

Following (Poesio et al., 1999), we consider the

```
det     a, the, that, none
head    it, that, one, noun, none
mod     +, -
```
The possible values of each NP frame slot

$$\begin{bmatrix} \texttt{det:} & \texttt{none} \\ \texttt{head:} & \texttt{it} \\ \texttt{mod:} & - \end{bmatrix} \qquad \begin{bmatrix} \texttt{det:} & \texttt{that} \\ \texttt{head:} & \texttt{noun} \\ \texttt{mod:} & + \end{bmatrix}$$

it   that button on the right

NP frames for *it* and *that button on the right*

Figure 2: NP frame slot values and examples

information conveyed by an NP to be divided into four slots which must be filled to be able to generate the NP form: a determiner/quantifier, a pre or post-modifier and a head noun slot. There were very few examples of premodifiers in the corpus, so we collapsed the modifier feature. Therefore, the output from our algorithm is an NP frame specifying values for the three slots for each target expression. Figure 2 shows the possible values in each slot and example slot values for two NPs. The number of occurrences in the entire corpus for the NP frame slot values are shown in Table 2.[4]

In the experimental VR world developed for this study, all items from the same category were designed to look identical. This was intended to encourage the subjects to use referring expressions that rely on spatial attributes or deictic reference such as *that one*. The spatial properties of target referents and distractors are used as inputs to the content planning algorithm. Their values in this study were calculated automatically based on geometric properties of the virtual world.

To form the training dataset, we processed each target expression with a syntactic chunker.[5] The partial parse it produced was further processed with a regular-expression matcher to isolate the values corresponding to the three slots. Parser errors caused some low-count NP frame values, so we retained only items that occurred at least 10 times in the entire corpus. Any parser errors that remained in the data were not hand corrected, in order to minimize human intervention.

4.1 Context Features

Given the restrictions that we impose over what is accessible to the learning algorithm, we developed a set of features for each referring expression that characterize both the referent and the context in which the expression was spoken. The context

---

[4]The two possible tags for **Mod** occurred in almost equal proportion (49%/51%)

[5]http://www.ltg.ed.ac.uk/software/chunk/index.html

		Dialog history features	
1.	Count and chainCount	the mention counts for the referent over the dialog and inside a reference chain[a]	
2.	DeltaTime and DeltaTimeChain	the time elapsed since it was last mentioned in the dialog overall or in a chain	
3.	PrevSpeaker	the previous speaker that mentioned the ID (either DG or DF)	
4.	$\text{Mod}_{i-1}, \text{Det}_{i-1}, \text{Head}_{i-1}$	the values of the slots of the NP frame of the prior mention of the same referent	
5.	$\text{Mod}_{i-2}, \text{Det}_{i-2}, \text{Head}_{i-2}$	the previous-1 values of the slots	
6.	WordDistance and chainWordDistance	the number of words spoken by both speakers since the last mention of the ID overall or in the chain	
7.	Type_{i-1}	indicates if the previous mention was in a Set, was Vague, or was a test item	
		Spatial/Visual features[b]	
8.	Distance	the distance between the referent and the DF's VR coordinates	
9.	Angle	the angle between the center of the DF's view angle and the center of the referent	
10.	Visible	a boolean value which indicates if the object is visible	
		Relation to other objects in the world	
11.	Visible Distractors	the number of other objects besides the target referent in the field of view	
12.	SameCatVisDistractors	the number of visible distractors of the same type as the referent	
		Object category and its information status	
13.	Cat	the semantic category of the referent: door/cabinet/button	
14.	First Locate	indicates if this is the first expression that allowed the DF to identify the object in the world. The point where joint spatial reference is accomplished.	

Table 1: The Context Features Used by the Algorithm

[a]mention counts are not considered over vague or ambiguous tags, or over sets.

[b]note that an **Angle** value smaller than 50^0 ensures the object is **visible**

Det			**Head**		
Value	Count	Percent	Value	Count	Percent
the	364	39%	noun	558	60%
that/this	264	29%	one	166	18%
none	253	27%	it	116	13%
a	46	5%	that	57	6%
			none	30	3%

Table 2: Distribution of **Det** and **Head** values in the corpus

v = Visible area(100°)

α = Angle to target

d = distance to target

In this scene:

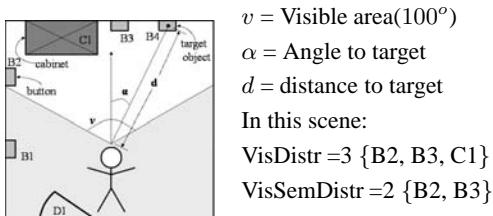VisDistr = 3 {B2, B3, C1}

VisSemDistr = 2 {B2, B3}

Figure 3: An example configuration with spatial context features. The target object is B4.

features are not only linguistic but also derived from the extralinguistic situation, including spatial relations between the referent and the DF's position and orientation at that instant. The context feature for each target expression includes these automatically-calculated attributes as well as features from the annotation described above. Table 1 describes the full set of context features, and Figure 3 shows a schematic of the spatial context features.

The mention history of any target referent is important for determining the form to use in a subsequent referring expression. Ideally, the discourse history feature should indicate whether a referent has already been discussed, and the distance between a new mention and its antecedent. But determining the discourse status of items in this world was complicated by two factors. All objects in the world of the same semantic category had identical visual features, and the VR world in which the task is conducted is a maze, which required the subjects to perform tasks, move to a different portion of the maze, and possibly return to a previously visited room. Due to the visual and spatial confusion possible in this setting, there is no guarantee that our subjects could accurately calculate whether they were discussing the same object they had encountered before, or remember whether that object had been discussed. While the subjects were focused on a task in a particular room, however, it is reasonable to expect that they could remember which items had been discussed. Therefore, the discourse histories of target objects were calculated using a re-initialization process. Each time the subjects left a VR room to pursue a different task, if more than $25s$ elapsed before the next mention of objects in that room, those subsequent expressions were considered to be in new coreference chains. This time constant was established by examining pronominal referring expressions in the training dialogs.

These features were used as input to develop a classifier to determine NP frames for unseen target referents in context. We chose decision trees due to their ease of interpretation, but we plan to test other machine learning techniques in the future. 5 dialogs were held out as unseen data and the remaining 10 were used to train and adjust the parameters of the decision process. The first procedure was to test whether the three slot values are interdependent. In contrast to previous work,

which focused on predicting the values for one of the slots at a time, we hold that due to their interdependence, these decisions should not be made separately. For example, a noun form that has the pronoun *it* as the head will never have a modifier or a determiner. If the three slots are independent, training three separate classifiers and then combining their decisions will yield better results. On the other hand, if they are dependent, better results will be obtained through training a single classifier on the combined label. Unfortunately, combining the labels is problematic due to data sparsity. To test these dependencies, we trained several decision trees, varying the independence assumptions:
Independent - a decision tree was trained for each slot and their outputs combined at the end.
Joint - a decision tree was trained for the combined label for all three slots

Conditional - three decision trees were trained in sequence, each having access to the output of the previous tree. For example, **Mod-Det-Head** means that first the **Mod** tree was trained, then a tree to classify **Det**, using the output from **Mod**, and finally a tree for **Head** , using both the **Det** and **Mod** values.

All possible orderings between **Mod**, **Head** and **Det** were tested. The best result obtained was from the ordering **Mod-Det-Head**, but the differences between the orderings were not significant. The 10 fold cross-validation results are shown in Table 3. There were 632 items in the data set. The Conditional trees outperformed the Independent trees by 9%, which is significant at the level of $(p < .0002)$.

As our training data suggests, we test the **Mod-Det-Head** trees against our held out data. We decided to use a leave one out method of training/testing due to the sparsity of data.

Independent	Joint	Mod-Det-Head
22.0 %	28.8 %	31.0 %

Table 3: Testing independence of the slot values

Decision tree classifiers offer the opportunity to examine the relevance of particular features in the final decision. Algorithm 1 and 2 show example trees derived for the **Mod** and **Det** features (the **Head** tree is not shown due to space limitations). We found that there are significant dependencies between the slots in the NP form. Each time one of the slots' values was available to the decision process, it was selected as most informative feature in the next tree. The spatial features were selected as informative in all the trees, most prevelantly in the

---

Algorithm 1 An example decision tree for **Mod**

if $FirstLocate = True$ **then**
  if $VisibleDistractors = 0$ **then**
    if $Distance \leq 116$ **then**
      return Mod: -
    else
      return Mod:+
  else
    if $SameCatVisDistractors = 0$ **then**
      if $VisibleDistractors \leq 2$ **then**
        if $Angle \leq 38$ **then**
          return Mod: -
        else
          return Mod: +
      else
        return Mod: +
    else
      return Mod: +
else
  if $chainWordDistance = 0$ **then**
    if $prevMention \neq Set/AllVague$ **then**
      if $firstMention = True$ **then**
        return Mod: +
      else
        if $Angle \leq 27$ **then**
          return Mod: -
        else
          return Mod: +
    else
      if $noprevMention$ **then**
        return Mod: +
      else {$prevMention = Set/AllVague$}
        return Mod: -
  else
    return Mod: -

Algorithm 2 An example decision tree for **Det**

if $Mod : -$ **then**
  if $FirstLocate = True$ **then**
    return Det:that
  else
    if $prevMention \neq Set/AllVague$ **then**
      if $notVisible$ **then**
        if $Cat = Button/Cabinet$ **then**
          return Det:none
        else {$Cat = Door$}
          return Det:that
      else {$isVisible$}
        if $Head_{i-1} = it$ **then**
          return Det:none
        else if $Head_{i-1} = noun$ **then**
          if $DeltaTime \leq 6.3$ **then**
            if $Cat = Button/Cabinet$ **then**
              return Det:none
            else {$Cat = Door$}
              return Det:that
          else
            return Det:the
        else if $Head_{i-1} = one/none/low$ **then**
          return Det:that
        else {$Head_{i-1} = that$}
          return Det:none
    else if $noprevMention$ **then**
      return Det:that
    else {$prevMention = Set/AllVague$}
      return Det:none
else {$target$ has $modifier$}
  return Det:the

decision tree for **Mod**, suggesting that the decision of including extra information is driven largely by the spatial configuration. The information status features and discourse history, such as **First Locate**, **Type**, and attributes of the prior mention, were selected as good predictors for the **Det** slot.

5 Evaluation

We report several methods of evaluating the NP frames produced using the process given by the decision trees. First, we report the results of a strict evaluation in which the system's output must exactly match expressions produced by the human subjects. We also compare this result with a hand-crafted Centering-style generation algorithm. Requiring the algorithm to exactly match human performance is an overly-strict criterion, since in many contexts several possible referring expression forms could be equally felicitous in a given context, so we also conducted a human judgment study. The 5 test dialogs contain 295 target expressions.

5.1 Exact Match Evaluation

The output of the decision tree classifier was compared to the expressions observed in the test dialog. Table 4 reports the results of this evaluation. The accuracy obtained was 31.2%. The most frequent tag gives a 20.0% baseline performance using this strict match criterion.

Exact match results

Predicted	All three features	**det**	**mod**	**head**
Correct	31%	48%	72%	56%

Exact match: head feature per value

Predicted	noun	it	none	one	that
Correct	65%	64%	0%	30%	38%

Exact match: det feature per value

Predicted	a	none	that	the
Correct	0%	49%	36%	66%

Table 4: Classifier results using Exact-match criterion

5.2 Comparison to Centering

For purposes of comparing the performance of our generation algorithm to existing work on generation of NPs, we performed a manual evaluation of the centering-style generation algorithm described in (Kibble and Power, 2000) against our dialog corpus. Algorithms developed according to the centering framework use discourse coherence to make decisions about pronominalization (Grosz et al., 1995), where coherence is measured in terms of topical continuity from one sentence to the next. Centering designates the *backward-looking center (Cb)* as the item in the current sentence that was most topical in the previous sentence. Therefore, to perform a centering-style evaluation, the dialogs must be broken into sentence-like units, and a ranking procedure must be devised for the items mentioned in each unit.

The current evaluation corpus, being a spoken dialog, has not been parsed to automatically determine the syntactic or dependency structure, but rather was manually segmented into utterance units, where each unit contained a main predicate and its satellites. The items mentioned in each unit were ranked according to thematic roles, using the ranking {AGENT > PATIENT > COMP > ADJUNCT}, and excluding references to the speakers themselves, which often appear in AGENT position (Byron and Stent, 1998). The Cb in each unit, if there is one, is the highest-ranked item from the prior unit's list that is repeated in the current unit's list. Following a procedure similar to that reported by Kibble and Power, our decision procedure recommends pronominalizing an item if it is the Cb of its unit and if it is in Subject position, otherwise a description is generated. Based on this rule, all items that are being mentioned for the first time in the discourse are predicted to require a description.

Although most prior studies take the recommendation to pronominalize to mean that a personal pronoun (e.g. *it*) should be generated, due to the demonstrative nature of our domain, the decision to produce a pronoun can result in either a demonstrative or a personal pronoun. Therefore, we considered the algorithm's output to match human production when the target expression in the human corpus was either a personal or demonstrative pronoun, and the algorithm generated either category of pronoun. Table 5 shows the comparison of our system's output and the output from the centering algorithm on anaphoric mentions. The 5 dialogs used for testing in this study contained 145 such items. Both algorithms obtained a similar accuracy (64.8% our system vs. 64.1% centering) and over-generated pronoouns. Although our algorithm does not outperform centering, it assumes less structural analysis of the input text.

5.3 Human Judgment Evaluation

Evaluating generation studies by calculating their similarity to human spontaneous speech may not be the ideal performance metric, since several different realizations may be equally felicitous in a

	Pron	Desc	Total
Human Production	28	117	145
Predicted by Our Algorithm	55	90	
Predicted by Centering	64	81	

Table 5: Comparison to Coherence-based Generation
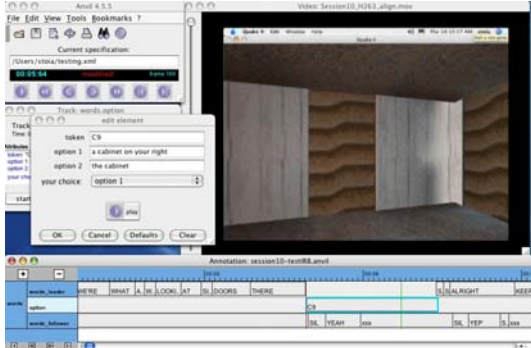
Figure 4: The Anvil software tool used for judging

All Items	
System compared to Human	Trials: 577
equal	45.9%
system preferred	16.6%
(system equal or preferred to human)	**(62.6%)**
human preferred	37.4%
System compared to Random	Trials: 289
equal	24.2%
system preferred	53.3%
(system equal or preferred to random)	**(77.5%)**
random preferred	22.5%
Random compared to Human	Trials: 292
equal	23.3%
random preferred	13.0%
(random equal or preferred to human)	**(36.3%)**
human preferred	65.7%
Items with two judges & judges agreed	
System against Human	Trials: 197
equal	37.3%
system preferred	19.8%
(system equal or preferred to human)	**(57.1%)**
human preferred	36.6%

Table 6: Results of Human Judging

given context. Therefore, we also performed a human judgement evaluation. In this evaluation, judges compared the NPs generated by our algorithm to the NPs produced by human subjects, and to NPs with randomly generated feature assignments. Judges viewed test NPs in the context of the original test corpus.

To re-create the context in which the original expression was produced, the video, audio, and dialog transcript were played for the judges using the Anvil annotation tool (Kipp, 2004). The judges could play or pause the video as they wished. Using the word-alignments established during the data annotation phase, the audio of the test NPs was replaced by silence, and the words were removed in the transcript shown in the timeline viewer. For each test item, the judges were presented with a selection box showing two possible referring expressions that they were asked to compare using a qualitative ranking (option 1 is better, option 2 is better, or they are equal), given a particular target ID and the context. Figure 4 shows a screen-shot of the judges' annotation tool. The judges did not know the source of the expressions they evaluated (system, human production, or random). The 10 judges were volunteers from the university community who were self-identified native speakers of English. They were not compensated for their time.

The decision tree selected NP-frame slot values which were converted into realized NPs. The **Det** and **Head** choices were directly translated into surface forms (for **Head=noun** we chose a consistent common noun for each semantic class: *but-*

ton, *door* or *cabinet*. If the system's selection of **Mod** feature matched the value from the corpus, we used the expression produced by the original speaker. If the original expression did not include a modifier, but the system selected **Mod:+**, we lexicalized this feature to a simple but correct spatial description like *on the right*, *on the left* or *in front*.

Table 6 shows the results of human judging. The system's output was either equal or preferred to the original spontaneous language in 62.6% of cases where these two choices were compared directly. Interestingly, the randomly-generated choice was preferred over the original spontaneous language in 13.0% of trials, and was preferred over the system's output in 22.5% of trials. Aggregating over all judges, the system's performance was judged to be much better than random, but not as good as the original human language.

Trials were balanced among judges so that each target item was seen by four judges: with two comparing the system's response to the original human language, one comparing the system to random, and one comparing the human to random. There were 282 trials for which 2 judges saw the identical pair of choices. Out of these, the two judges' responses agreed in 197 cases, producing an inter-annotator reliability (kappa score) of 0.51, with raw agreement of 69% and expected agreement of 37%. Although this is a relatively low kappa value, we believe that the aggregate judgments of all of the judges over all of the test items are still informative, since the scores of items for which we have two judgements follow a very sim-

ilar pattern to the overal distribution of responses. The low inter-annotator agreement may be due to the substitutability of the expressions.

6 Conclusions and Future Work

In this paper we describe a generation study for situated dialog and a novel evaluation setup of the system's output. The algorithm decides upon the determiner, head and modifier values to be produced in a noun phrase describing an object in a particular moment in the dialog. The decision is influenced by dialog history, spatial and visual relations and information status of the ID to be described. Our algorithm achieved 31.2% exact match with human language, but human evaluators judged the output as good as or better than the original human language 62.6% of the time.

For our future work, we intend to develop the generation module of a dialog system that performs the direction giver's role. We plan to incorporate the results of this study in an extension of (Reiter and Dale, 1992) algorithm that would take into account other types of properties of the objects like visual salience, temporal attributes (for example time elapsed between mentions), if it participated in an action (like the case of a door opening, or a button being pushed) or its importance to the overall task completion.

Acknowledgments

References

D. Byron and A. Stent. 1998. A preliminary model of centering in dialog. In *Proceedings of ACL '98*, pp. 1475–1477.

D. Byron. 2005. The OSU Quake 2004 corpus of two-party situated problem-solving dialogs. Technical Report OSU-CISRC-805-TR57, The Ohio State University Computer Science and Engineering Department, September.

J. Cassell, T. Stocky, T. Bickmore, Y. Gao, Y. Nakano, K. Ryokai, D. Tversky, C. Vaucelle, and H. Vilhjalmsson. 2002. MACK: Media lab Autonomous Conversational Kiosk. In *Proceedings of IMAGINA'02*, Monte Carlo, January.

H. Cheng, M. Poesio, R. Henschel, and C. Mellish. 2001. Corpus-based NP modifier generation. In *NAACL '01*, pp. 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

S. Chou, W. Hsieh, F. Gandon, and N. Sadeh. 2005. Semantic web technologies for context-aware museum tour guide applications. In *Proceedings of the 2005 International Workshop on Web and Mobile Information Systems*.

R. Dale, S. Geldof, and J. Prost. 2003. CORAL: Using natural language generation for navigational assistance.

In M. Oudshoorn, editor, *Proceedings of the 26th Australasian Computer Science Conference*, Adelaide, Australia.

B. Grosz, A. Joshi, and S. Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–226.

J. Gundel, N. Hedberg, and R. Zacharski. 1993. Cognitive status and the form of referring expressions in discourse. *Language*, 69(2):274–307.

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

R. Kibble and R. Power. 2000. An integrated framework for text planning and pronominalisation. In *Proceedings of INLG'2000*, pp. 77–84.

M. Kipp. 2004. *Gesture Generation by Imitation - From Human Behavior to Computer Character Animation*. Dissertation.com.

S. Lauria, G. Bugmann, T. Kyriacou, J. Bos, and E. Klein. 2001. Training personal robots using natural langauge instruction. *IEEE Intelligent Systems*, 16(5):2–9.

S. Long, R. Kooper, G. Abowd, and C. Atkesonet. 1996. Rapid prototyping of mobile context-aware applications: The cyberguide case study. In *2nd ACM International Conference on Mobile Computing and Networking (MobiCom'96)*, November 10-12.

W. Maass, J. Baus, and J. Paul. 1995. Visual grounding of route descriptions in dynamic environments.

R. Moratz and T. Tenbrink. 2003. Instruction modes for joint spatial reference between naive users and a mobile robot. In *Proc. RISSP 2003 IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, Special Session on New Methods in Human Robot Interaction*.

C. Muller. 2002. Multimodal dialog in a pedestrian navigation system. In *Proceedings of ISCA Tutorial and Research Workshop on Multi-Modal Dialogue in Mobile Environments*.

M. Poesio, R. Henschel, J. Hitzeman, and R. Kibble. 1999. Statistical NP generation: A first report. Utrecht, August.

E. Prince. 1981. On the inferencing of indefinite *this* NPs. In Aravind K. Joshi, Bonnie Lynn Webber, and Ivan Sag, editors, *Elements of Discourse Understanding*, pp. 231–250. Cambridge University Press.

E. Reiter and R. Dale. 1992. A fast algorithm for the generations referring expressions. In *Proceedings of COLING '92*, pp. 232–238.

M. Skubic, D. Perzanowski, A. Schultz, and W. Adams. 2002. Using spatial language in a human-robot dialog. In *2002 IEEE International Conference on Robotics and Automation*, Washington, D.C.

K. Striegnitz, P. Tepper, A. Lovett, and J. Cassell. 2005. Knowledge representation for generating locating gestures in route directions. In *Proceedings of Workshop on Spatial Language and Dialogue (5th Workshop on Language and Space)*, Delmenhorst, Germany, October.

W. Wahlster, N. Reithinger, and A. Blocher. 2001. Smartkom: Towards multimodal dialogues with anthropomorphic interface agents. In *International Status Conference: Lead Projects HumanComputer -Interaction*, Saarbruecken, Germany.

J. Yang, W. Yang, M. Denecke, and A. Waibel. 1999. Smart sight: a tourist assistant system. In *Proceedings of the 3rd International Symposium on Wearable Computers*, pp. 73–78, San Francisco, California, 18-19 October.