The Effect of Sensor Errors in Situated Human-Computer Dialogue

Niels Schuette
Dublin Institute of Technology
niels.schutte@student.dit.ie

John Kelleher
Dublin Institute of Technology
john.d.kelleher@dit.ie

Brian Mac Namee
Dublin Institute of Technology
brian.macnamee@dit.ie

Abstract

Errors in perception are a problem for computer systems that use sensors to perceive the environment. If a computer system is engaged in dialogue with a human user, these problems in perception lead to problems in the dialogue. We present two experiments, one in which participants interact through dialogue with a robot with perfect perception to fulfil a simple task, and a second one in which the robot is affected by sensor errors and compare the resulting dialogues to determine whether the sensor problems have an impact on dialogue success.

1 Introduction

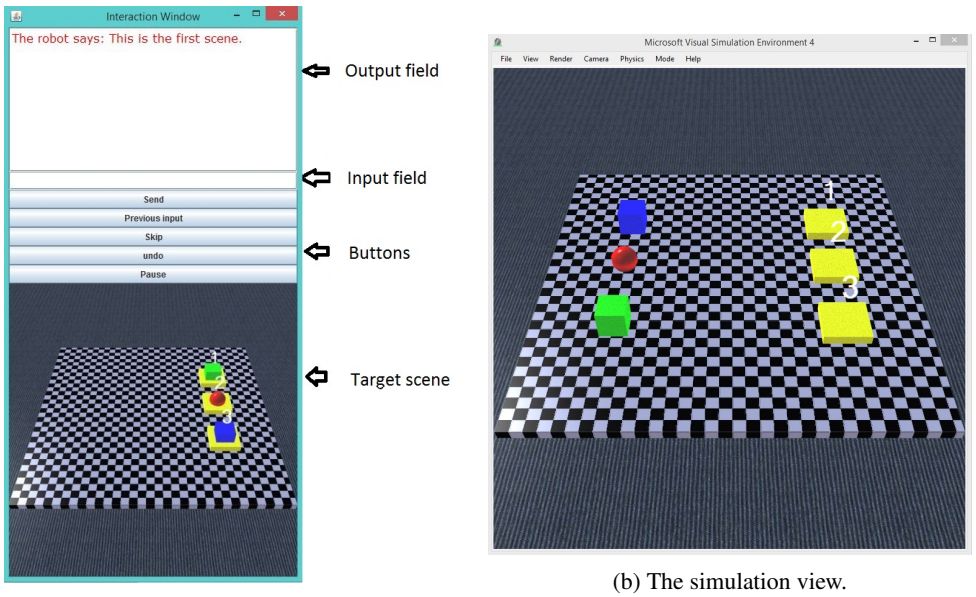
Computer systems that can engage in natural language dialogue with human users are known as **dialogue systems**. A special class of dialogue systems are **situated dialogue systems**, which are dialogue systems that operate in a spatial context. Situated dialogue systems are an active research topic (e.g. (Kelleher, 2006)). Recently opportunities for more practical applications of situated dialogue systems have arisen due to advances in the robustness of speech recognition and the increasing proliferation of mobile computer systems such as mobile phones or augmented reality glasses.

When a dialogue system operates in a situated context, it needs the ability to perceive the environment. Perception, such as computer vision, always has the potential of producing errors, such as failing to notice an object or misrecognizing an object. We are interested in the effect of perception-based errors on human-computer dialogue. If the human user and the system have shared view, false perception by the system will lead to a divergence between the user's understanding of the environment and the system's understanding. Such misunderstandings are frequent in human-human dialogue and human speakers use different strategies to establish a shared understanding or common ground (Clark and Schaefer, 1989). We investigated this problem in an earlier work based on a corpus of human dialogue (Schuette et al., 2012) and are currently moving toward the same problem in human-computer dialogue.

The problem of misunderstandings in human-computer dialogue has previously mostly been addressed under the aspect of problems arising from problems in speech recognition or language understanding (e.g. (Aberdeen and Ferro, 2003; Shin et al., 2002; López-Cózar et al., 2010)). The problem of producing referring expressions when it is not certain that the other participant shares the same perception and understanding of the scene has been addressed by (Horacek, 2005). More recently (Liu et al., 2012) performed a similar experiment in the context of human-human interaction. Their work was chiefly concerned with the generation of referring expressions.

We report on a work in progress in which we investigate the effect of sensor problems on human-computer dialogue using a dialogue system for a simulated robot. We describe two experiments we performed so far. Both experiments are based on a shared experimental platform. In the first experiment participants interact with a simulated robot using a text based dialogue interface to complete a series of tasks. In the second experiment the participants again interact with the robot, except this time errors are introduced into the robots perception. The goal of the second experiment is to investigate what effect

Proceedings of the 25th International Conference on Computational Linguistics, pages 1–8,
Dublin, Ireland, August 23-29 2014.



(a) The interaction window.

(b) The simulation view.

Figure 1: The user interface.

the presence of sensor errors has on the dialogue and the task performance and compare it to the results from the first experiment. It should be emphasized that the goal of the experiments is not to evaluate the performance of the dialogue system, but to investigate the effect of perception errors on the dialogues.

2 Experiment Methodology

The experiments were performed using an experiment system that was developed for this experiment. It consists of a simulated world and a dialogue system. The world contains a number of objects such as boxes and balls. These object can be manipulated by an abstract simulated robot arm. The dialogue system is a frame based dialogue system that uses the Stanford Parser (Klein and Manning, 2003) for parsing. The simulation environment was implement using Microsoft Robotics Studio. The system is capable of understanding and performing a range of simple to complicated spatial action instructions such as “Put the ball behind the red box” or “Pick up the red ball between the green box and the yellow box”.

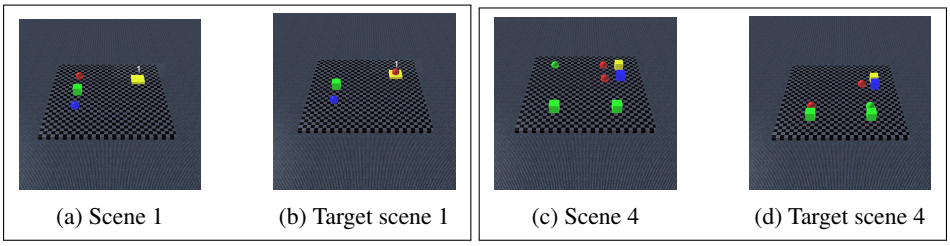
The participants interact with the system through the user interface shown in Figure 1. It consists of two elements. The **simulation window** shows a rendering of the simulation world that is updated in real time. The **interaction window** provides access to a text based chat interface that the participants use to interact with the simulated robot. When the participant sends a request to the system, the system analyses the input and attempts to perform it in the simulation world. If it can not perform the request, it replies through the user interface and explains its problem.

The robot’s perception is provided by a simulated vision system. In general its perception is correct, but sensor errors can be introduced. For example, it can be specified that the robot perceives entire objects or some of their properties incorrectly.

Each run of the experiment consisted of a sequence of test scenes. Each scene consisted of a **start scene** and a **target scene**. The start scene determined how the objects in the simulation world were arranged at the beginning of the test scene. The target scene was presented to the participants as an image in the interaction window. The participants’ task was to interact with the robot to recreate the target scene in the simulation world.

After a participant had successfully recreated the target scene, the system automatically advanced to the next scene. The participants were also offered the option to abandon a scene and go on to the next one if they thought they would not be able to complete the current scene.

All utterances by the participant and the system are transcribed and annotated with their semantic



(a) Scene 1 (b) Target scene 1 (c) Scene 4 (d) Target scene 4

Figure 2: Two scenes from Experiment 1 and their target scenes.

interpretation. The system also logs metrics that are used in the evaluation of dialogue systems to describe the cost of a dialogue, such as the task completion rate, the number of utterances, the completion time and the number of errors (Walker et al., 1997).

In the following we describe two experiments we performed with this setup so far. In the first experiment participants completed a series of tasks. In the second experiment, participants also completed a series of tasks. In this iteration however, errors were introduced into the system's perception.

3 Experiment 1

The first experiment uses the basic version of the experiment system. The purpose of the experiment was to establish how difficult the basic experiment task would be and to create a set of performance measurements that could be used to compare this version of the system to later ones.

3.1 Instructions

The participants were provided with an instruction manual that described the experiment, introduced the user interface and provided example interactions. Participants were encouraged to abandon a scene if they felt that they would not be able to complete it. After reading the instructions, the participants were shown a video recording of some example interactions with the system. This was done to prime the participants towards using language and concepts that were covered by the system. No time limit was set for experiment.

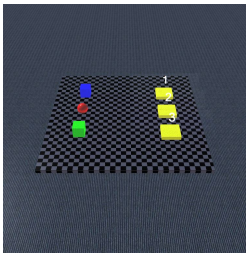
3.2 Test Scenes

The set of test scenes contained 10 scenes in total. Figure 2 shows some of the start scenes together with their respective target scenes. Scene 1 (Figure 2a) is an example of a simple scene. Scene 4 (Figure 2c) is an example of a more complex scene.

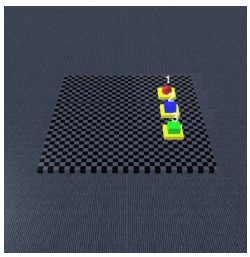
The scenes were presented in fixed order. The two initial scenes contained simple tasks. Their main purpose is to allow the participants to gain practical experience with interacting with the system before approaching the actual test scenes. The remaining scenes were designed to elicit specific **referring expressions**. To transform a scene into its target scene, the participants had to move a number objects from their original location to their respective target location as specified in the target scene. To get the robot to move a target to a location, the participants had to specify which target the robot should move (e.g. "Take the red ball"), and specify where to move it (e.g. "Put it behind the green box on the left"). The complexity of the this task depends on the objects contained in the scene and their placement in relation to each other. We were particularly interested in getting the participants to use specific objects as **landmarks** in their referring expressions, and designed the scenes in such a way that participants were influenced towards specific expressions. This was done with the motive of using landmark objects as targets for perception errors in the second experiment. For each scene a set of target conditions was specified that determined when a scene was complete.

3.3 Participants

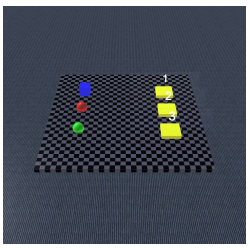
In total 11 participants participated in the experiment. Most of them were native English speakers or non-native speakers who had been speaking English for a number of years. Two of the participants were



(a) The start scene.



(b) The target scene.



(c) The start scene as perceived by the robot.

Figure 3: One of the scenes from Experiment 2.

female, the rest were male. The participants were between 20 and 50 years of age. All were college sciences graduates who worked with computers on a daily basis.

3.4 Results

In total 11 participants completed the experiments. This resulted in a total of 110 interactions, two of which had to be discarded due to recording problems. A summary of the recorded metrics for this experiment is given in Table 1. It shows for each scene:

- How many instructions the participants used on average to complete it.
- How long the participants needed to complete each scene on average.
- How many of the instructions the participants produced contained a reference that was either ambiguous (it could not be resolved to a unique referent) or unresolved (no referent that matched the referring expression was found).
- The final column show how often each scene was abandoned.

For the current investigation the last two columns are of primary interest. Participants had been instructed to abandon a scene if they thought that they would not be able to complete it. The fact that this only occurred three times in 108 interactions indicates that the task was not very difficult and that the dialogue system's performance was adequate for the task. The percentage of unresolved references in the second to last column is also interesting because it indicates how often participants made references that the system was not able to resolve. Since there were no errors introduced at this stage, the figures can be seen as a baseline for the system's ability to understand referring expressions.

4 Experiment 2

The main purpose of the second experiment was to investigate how the introduction of sensor errors would influence the interactions and the outcome.

4.1 Instructions

The participants were provided with an extended version of the instruction manual as well as the introduction video from the first experiment. The manual was identical to the manual from Experiment 1 except for a small section that was added to explain that errors could occur in some of the scenes. The participants were encouraged to either try to work around the errors or to abandon the scene if the thought they would not be able to finish it. Again, no time limit was set.

4.2 Test Scenes

The set of test scenes was based on the set of test scenes for Experiment 1, except that this time sensor errors were introduced. We investigated three possible error conditions. In the **missing object** condition, the perception system did not register an object at all. In the **colour misclassification**, the system did

Scene name	Average number of actions per scene	Average time per scene	Percentage of ambiguous or unresolved references	Number of times abandoned
Scene 1	2.9	00:00:56	0	0
Scene 2	2.3	00:00:54	0	0
Scene 3	8.7	00:01:45	2.1	0
Scene 4	5.9	00:01:52	10.8	0
Scene 5	2	00:00:28	0	0
Scene 6	5.2	00:01:23	5.2	1 ($\approx 9\%$)
Scene 7	2.6	00:00:40	0	0
Scene 8	5.8	00:01:06	3.1	1 ($\approx 9\%$)
Scene 9	5.3	00:01:12	8.4	0
Scene 10	6.8	00:01:30	6.7	1 ($\approx 9\%$)
Average	5.1	00:01:14	6	0.3

Table 1: Summary of the cost metrics for Phase 1. Few scenes were abandoned. The percentage of unresolved references forms a baseline for the resolution performance of the system.

Scene name	Average number of actions per scene	Average time per scene	Percentage of ambiguous or unresolved references	Number of times abandoned
Scene 1	2.29	00:00:59	2.6	0 (0%)
Scene 2	3.29	00:00:56	3.6	2($\approx 11.8\%$)
Scene 3	9.12	00:02:13	9.7	3 ($\approx 17.6\%$)
Scene 4	9.88	00:01:58	10.1	5 ($\approx 29.4\%$)
Scene 5	10.35	00:01:46	9.7	2 ($\approx 11.8\%$)
Scene 6	12.82	00:02:43	7.3	9 ($\approx 52.9\%$)
Scene 7	4.82	00:01:08	14.6	2 ($\approx 11.8\%$)
Scene 8	3.35	00:00:47	8.8	1 ($\approx 5.9\%$)
Scene 9	9.88	00:01:34	9.5	4 ($\approx 23.5\%$)
Scene 10	9.59	00:01:47	9.8	5 ($\approx 29.4\%$)
Scene 11	10.82	00:02:08	5.4	3 ($\approx 17.6\%$)
Scene 12	7	00:01:21	8.4	1 ($\approx 5.9\%$)
Scene 13	6.65	00:01:29	8	2 ($\approx 11.8\%$)
Scene 14	11.7	00:03:10	8.5	17 (100%)
Scene 15	5.18	00:01:02	15.9	1 ($\approx 5.9\%$)
Scene 16	4.88	00:01:04	14.5	1 ($\approx 5.9\%$)
Scene 17	6.82	00:01:01	1.7	0 (0%)
Scene 18	8.65	00:02:00	6.8	1 ($\approx 5.9\%$)
Scene 19	9.4	00:01:45	7.8	0 (0%)
Scene 20	6	00:01:17	6.9	0 (0%)
Average	7.6	00:01:36	8.5	2.95
Average (scenes w/o errors)	6.1	00:01:20	4.9	0.5
Average (scenes w/ errors)	8.3	00:01:44	10	4

Table 2: Summary of the cost metrics for Phase 2. Scenes that contained no errors are highlighted in green. Compared to Table 1, scenes that contained errors were more often abandoned, and resolution problems were more frequent.

perceive the affected object but determined its colour incorrectly. A green ball for example, might be mistaken for a red ball. In the **type misclassification** condition, the system also perceives the object, but determines the object's type incorrectly, for example, a green ball might be mistaken for a green box. We restricted the errors so that at most one object was affected per scene. This was done to create scenes that contained errors, but would still be solvable in most cases without major communication breakdowns. The impact a sensor error has on the interaction greatly depends on which object it affects, the context the object appears in, and the role the object plays in the task. For example, if an object is affected that does not need to be moved and that is unlikely to be mentioned as a landmark, it is likely that the error will not be noticed by the participant, and have no influence on the dialogue at all. On the other hand, if an error affects an object that absolutely needs to be moved in order to complete the task in such a way that it becomes impossible to interact with the object (e.g. because the robot does not see the object at all), it becomes effectively impossible to complete the task. In less severe cases, errors may introduce problems that can be solved. For example, if the first attempt at a reference fails because a landmark is not available to the system, the participant may reformulate the expression with a different landmark. This highlights the fact that sensor errors can have different effects depending on the circumstances.

We therefore decided to design each scene and the errors for the second phase manually in order to make sure that examples for as many problem combinations as possible were presented to the participants. We based the design of the scenes on our experiences from Experiment 1. We selected suitable scenes and introduced errors such that the preferred expressions used in Experiment 1 would be affected. Each new scene created this way together with the original scene formed a **corresponding scene pairs**. Members of a pair can be compared against each other to assess the impact of errors in Experiment 2. The final set of scenes contained 14 scenes with sensor errors. We added four more scenes without errors to the test set. Their purpose was to complement the data from the first experiment, and to check if the presence of errors in other scenes would influence the behaviour of the participants in non-error scenes. We also added the two introductory scenes from the first experiment. They were always presented as the first scenes. The remaining scenes were presented in randomized order to prevent learning effects. Therefore each participant was presented with a set of 20 scenes. In total there were 22 corresponding scene pairs.

Figure 3 contains an example of a scene from the second experiment that contained a perception error. Figure 3a show the start scene as presented to the participant. Figure 3b shows the target scene that was presented to the participant. Figure 3c shows the start scene as it was perceived by the robot (it mistakes the green box for a ball).

Each scene was annotated with a set of target conditions and a set of sensor error specifications.

4.3 Participants

17 participants were recruited for the experiment from roughly the same demographic as the first experiment. About half of the participants had participated in the first experiment. A space of about 60 days was left between the first experiment and the second experiment to minimize any influence between the experiments.

4.4 Results

In total 17 participants completed the experiment. This results in a total of 340 interactions. Two interactions were lost, resulting in a set of 338 interactions. The results for this experiment are given in Table 2. The highlighted rows (Scene 1,2,17,18,19 and 20) refer to scenes in which no errors were introduced.

As in the first experiment, the two last columns are the most interesting ones. Overall it can be observed that more scenes were abandoned than in the first experiment. Every scene except for the ones without errors was abandoned at least once (Scene 14 was abandoned by all participants. This was expected because it was designed to be not completable due to the errors). This indicates that the task with the errors was more difficult than the one in the first experiment.

It also appears that unresolved or ambiguous references were more frequent than in the first experiment. At the bottom of the table we present overall averages for the different metrics. It appears that scenes with sensor errors generally show higher values than scenes without.

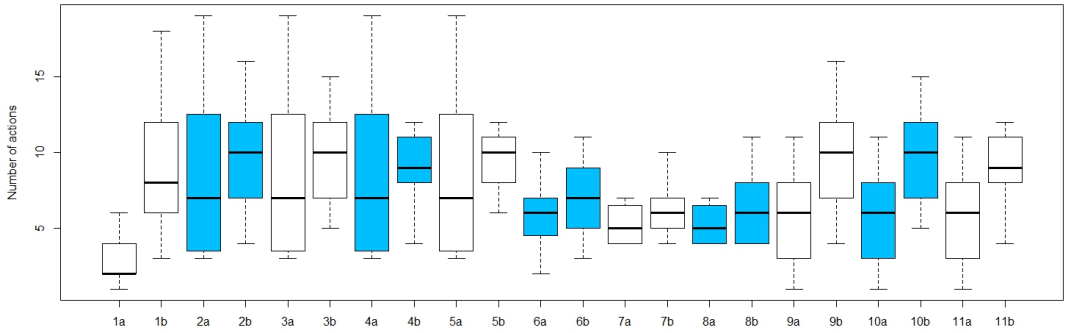


Figure 4: A boxplot comparing the number of actions between scenes from Experiment 1 and 2. Paired plots with the same colour refer to corresponding scenes (continued in Figure 5).

5 Discussion and Analysis

Overall the results indicate that the introduction of sensor errors increases the difficulty of the task. The results show that the participants had to abandon scenes with errors more often than scenes without errors. On average they used more actions to complete scenes with errors. A possible explanation can be found in the higher percentage of unresolved references. Participants attempted to refer to an object, but the system was unable to interpret it as expected due to a sensor error. This forced the participants to try a different expression to progress with the task. It should be noted that the number of unresolved and ambiguous references at the present does not account for references that were resolved to an object that was not the object intended by the speaker. We may approach this problem at a later stage.

Figure 4 and 5 visualize the distribution of the number of actions for all the corresponding scene pairs. They are numbered 1 to 22. Plots labelled with *a* correspond to scenes without errors, plots labelled with *b* to their counterparts with errors. For easier visual comprehension, we coloured pairs alternatingly in blue and white.

In general it can be observed that the median number of actions is generally higher for scenes with errors than for their non-error counterparts, and that the interquartile range also tends to be higher. The distributions appear to be fairly spread out. This suggests that there is considerable variation between participants. We performed t-tests between corresponding scenes to determine whether the differences between corresponding scenes were significant. The test shows that 12 out of 22 pairs were significantly different with a p-value below 0.05. We will investigate at a later stage how much the strength of the correspondence depends on the type of the error that was introduced.

A comparison of the distribution of the completion times was less conclusive. For some correspondence pairs, the median completion time is higher for error scenes, for other pairs it is lower. We conjecture that there is some sort of self-selection mechanism at work where participants who were less confident with the task in the first place task abandoned scenes earlier than confident participants when they encountered problems, but this will require further investigation.

To summarize: The presence of sensor errors appears to increase the difficulty of the task, although the effect appears to be small in some cases. This was in some way to be expected because the errors were designed to pose solvable problems and not lead to major communication breakdowns.

6 Future Work

The results from this experiment are still very fresh, and this paper represents the first step in their analysis. In the next step we are going to try to identify strategies the participants employed once they encountered an error and see how well they match up with the strategies we described for the human-human domain (Schuette et al., 2012). We are also interested in finding out how strategies evolved over the course of the experiment, and in how much variation there is between individual participants.

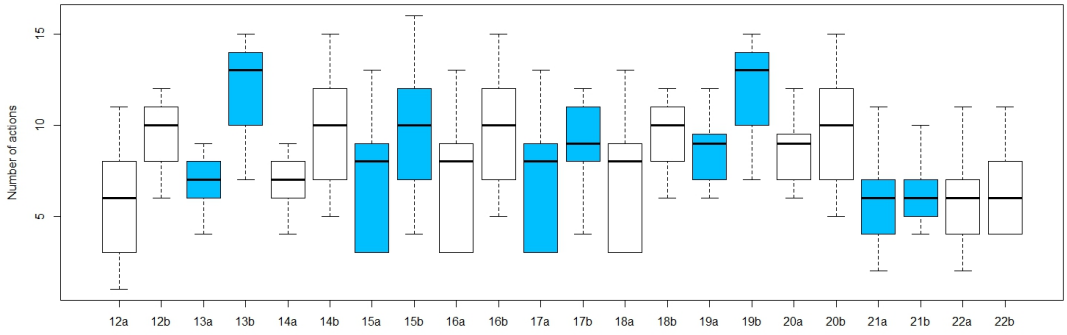


Figure 5: A boxplot comparing the number of actions between scenes from Experiment 1 and 2. Paired plots with the same colour refer to corresponding scenes (continued from Figure 5).

We are currently preparing a third experiment based on the experiment setup. In this experiment, the participants will be offered different ways of accessing the robot's understanding of what it sees to the participant. For example, in one condition, the system will be able to generate descriptions of how it perceives the scene. The results of this third experiment will be evaluated in the context of the first and second experiment.

References

John Aberdeen and Lisa Ferro. 2003. Dialogue patterns and misunderstandings. In *ISCA Tutorial and Research Workshop on Error Handling in Spoken Dialogue Systems*.

Herbert H. Clark and Edward F. Schaefer. 1989. Contributing to discourse. *Cognitive Science*, pages 259–294.

Helmut Horacek. 2005. Generating referential descriptions under conditions of uncertainty. In *Proceedings of the 10th European Workshop on Natural Language Generation (ENLG)*, pages 58–67. Citeseer.

J. D. Kelleher. 2006. Attention driven reference resolution in multimodal contexts. *Artificial Intelligence Review*, 25(1):2135.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, page 423430. Association for Computational Linguistics.

Changsong Liu, Rui Fang, and Joyce Y. Chai. 2012. Towards mediating shared perceptual basis in situated dialogue. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 140149. Association for Computational Linguistics.

Ramón López-Cózar, Zoraida Callejas, and David Griol. 2010. Using knowledge of misunderstandings to increase the robustness of spoken dialogue systems. *Knowledge-Based Systems*, 23(5):471–485, July.

Niels Schuette, John Kelleher, and Brian Mac Namee. 2012. A corpus based dialogue model for grounding in situated dialogue. In *Proceedings of the 1st Workshop on Machine Learning for Interactive Systems: Bridging the Gap Between Language, Motor Control and Vision (MLIS-2012).*, Montpellier, France, August.

Jongho Shin, Shrikanth S. Narayanan, Laurie Gerber, Abe Kazemzadeh, Dani Byrd, and others. 2002. Analysis of user behavior under error conditions in spoken dialogs. In *INTERSPEECH*.

Marilyn A. Walker, Diane J. Litman, Candace A. Kamm, and Alicia Abella. 1997. PARADISE: a framework for evaluating spoken dialogue agents. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, page 271280. Association for Computational Linguistics.