GENERALITY AND TRANSFERABILITY. TWO ISSUES IN PUTTING A DIALOGUE EVALUATION TOOL INTO PRACTICAL USE

Niels Ole Bernsen, Hans Dybkjær, Laila Dybkjær and Vytautas Zinkevicius
The Maersk Mc-Kinney Moller Institute for Production Technology
Odense University, Campusvej 55, 5230 Odense M, Denmark
emails: nob@mip.ou.dk, dybkjaer@mip.ou.dk, laila@mip.ou.dk, vytasz@ktl.mii.lt
phone: (+45) 65 57 35 44   fax: (+45) 66 15 76 97

ABSTRACT

This paper presents a first set of test results on the generality and transferability of an evaluation tool which can ensure the habitability and usability of spoken dialogues. Building on the assumption that most, if not all, dialogue design errors can be viewed as problems of non-cooperative system behaviour, the tool has two closely related aspects to its use. Firstly, it may be used for the diagnostic evaluation of spoken human-machine dialogue. Secondly, it can be used to guide early dialogue design in order to prevent dialogue design errors from occurring in the implemented system. We describe the development and in-house testing of the tool, and present results of ongoing work on testing its generality and transferability on an external corpus, i.e. an early Wizard of Oz corpus from the development of the Sundial spoken language dialogue system.

1. INTRODUCTION

Spoken language technologies are viewed as constituting one of the most important next steps towards truly natural interactive systems which are able to communicate with humans the same way that humans communicate with each other. After more than a decade of promises that versatile spoken language dialogue systems (SLDSs) using speaker-independent continuous speech recognition were just around the corner, the first such systems are now on the market. These developments highlight the needs for novel tools that can support efficient development and evaluation of SLDSs in general and their usability in particular.

It is a well-recognised fact in the field of human factors that those needs are difficult to meet. The difficulties lie not only in arriving at an initial conception of a new tool, or in tool drafting and early in-house testing. Even if these stages yield encouraging results, there is a long way to go before the tool can stand on its own and be used as an integral part of dialogue engineering best practice. Two problems stand out. First, there is the problem of *generality.* A tool which only works, or is only known to work, on a single system, in a highly restricted domain of application or in special circumstances, is of little interest to other developers. In-house testing will inevitably be done on a limited number of systems and application domains. To achieve an acceptable degree of generality, the tool must be iteratively developed and tested on systems and application domains and in circumstances that are significantly different from those available in-house. Secondly, there is the problem of *transfer.* However general the tool turns out to be eventually, it remains of little utility until other developers are able to use it with modest training and without requiring the presence or constant advice of its originators.

This paper presents test results on the generality and transfer potential of a tool which has been developed and tested on an in-house SLDSs project [2, 3]. The tool builds on the assumption that most, if not all, dialogue design errors can be viewed as problems of *non-cooperative* system behaviour. The tool has two aspects to its use. Firstly, it may be used as part of a methodology for diagnostic evaluation of spoken human-machine dialogue. Following the detection of cases of human-machine miscommunication, the tool enables a clear classification of miscommunication problems that are caused by flawed dialogue design. In addition, the tool supports the repair of those problems, preventing their occurrence in future user interactions with the system. Secondly, the tool can be used to guide early dialogue design in order to prevent dialogue design errors from occurring in the first place. In what follows, we describe the development and in-house testing of the tool (Section 2). We then present ongoing work on testing its generality (Section 3) and transferability (Section 4). Section 5 concludes the paper.

2. TOOL DEVELOPMENT

The tool was developed in the course of designing, implementing and testing the dialogue model for the Danish dialogue system. The system is a walk-up-and-use prototype SLDS for over-the-phone ticket reservation for Danish domestic flights. The system's dialogue model was developed using the Wizard of Oz (WOZ) simulation method. Based on the problems of dialogue interaction observed in the WOZ corpus, we established a set of guidelines for the design of cooperative spoken dialogue. Each observed problem was considered a case in which the system, in addressing the user, had violated a guideline of cooperative dialogue. The corpus analysis led to the identification of 14 guidelines of cooperative spoken human-machine dialogue based on analysis of 120 examples of user-system interaction problems. If those guidelines were observed in the design of the system's dialogue behaviour, we assumed, this would increase the smoothness of user-system interaction, reduce user-initiated meta-communication for clarification and repair, and improve user satisfaction with the system.

The guidelines were refined and consolidated through comparison with a well-established body of maxims of cooperative human-human dialogue which turned out to form a subset of our guidelines [2, 4]. The resulting 22 guidelines were grouped under seven different *aspects* of dialogue, such as informativeness and partner asymmetry, and split into *generic* guidelines and *specific* guidelines. A generic guideline may subsume one or more specific guidelines which specialise the generic guideline to a certain class of phenomena. Figure 1 shows shortform versions of the guidelines.

GG No.	SG No.	Generic or Specific Guideline
Dialogue Aspect: Informativeness		
GG1		*Say enough.

	SG1	State user commitments explicitly.
	SG2	Provide immediate feedback.
GG2		*Don’t say too much.
Dialogue Aspect: Truth and evidence		
GG3		*Don’t lie.
GG4		*Check what you will say.
Dialogue Aspect: Relevance		
GG5		*Be relevant.
Dialogue Aspect: Manner		
GG6		*Avoid obscurity.
GG7		*Avoid ambiguity.
	SG3	Ensure uniformity.
GG8		*Be brief.
GG9		*Be orderly.
Dialogue Aspect: Partner asymmetry		
GG10		Highlight asymmetries.
	SG4	State your capabilities.
	SG5	State how to interact.
Dialogue Aspect: Background knowledge		
GG11		Be aware of users’ background knowledge.
	SG6	Be aware of user inferences.
	SG7	Adapt to novices and experts.
GG12		Be aware of user expectations.
	SG8	Cover the domain.
Dialogue Aspect: Repair and clarification		
GG13		Enable meta-communication.
	SG9	Enable system repair.
	SG10	Enable inconsistency clarification.
	SG11	Enable ambiguity clarification.

Figure 1. Guidelines for cooperative system dialogue. GG means generic guideline. SG means specific guideline. The guidelines are expressed in shortform. Fullform expressions are found in [2, 3]. The generic guidelines are at the same level of generality as are the Gricean maxims (marked with an *). Each specific guideline is subsumed by a generic guideline.

The consolidated guidelines were then tested as a tool for the diagnostic evaluation of a corpus of 57 dialogues collected during a scenario-based, controlled user test of the implemented Danish dialogue system. The availability of the user-scenarios meant that problems of dialogue interaction could be objectively detected through comparison between the contents of expected and actual user-system exchanges. Each detected problem was (a) characterised with respect to its *symptom*, (b) a *diagnosis* was made, sometimes through inspection of the log of system module communication, and (c) one or several *cures* were proposed. The ‘cure’ part of diagnostic analysis suggests ways of repairing system dialogue behaviour. The diagnostic analysis may demonstrate that new guidelines of cooperative dialogue design must be added to the existing body of guidelines. We found that nearly all dialogue design errors in the user test could be classified as violations of our guidelines. Two *specific* guidelines on meta-communication, SG10 and SG11, had to be added, however. This was no surprise as meta-communication had not been simulated and therefore was mostly absent in the WOZ corpus.

3. GENERALISING THE TOOL

To test and increase the generality of the tool, we have applied it as a dialogue design guide to part of a corpus from the Sundial project [5]. The corpus comprises close to 100 early WOZ dialogues in which subjects seek time and route information on British Airways flights and sometimes on other flights as well. The corpus was produced by 10 subjects who each performed 9 to 10 dialogues based on scenarios selected from a set of 24 scenarios.

For the generality test of the tool, we selected 48 dialogues such that each subject is represented with an approximately equal number of dialogues and each scenario is used in two dialogues. Three dialogues were used for training. The remaining 45 dialogues were independently annotated and analysed by two experts in using the tool (A1 and A2) and one novice (A3). Each system utterance was analysed in isolation as well as in its dialogue context to identify violations of the guidelines. Using the Text Encoding Initiative standard (TEI), we have changed the existing markup of utterances to make each utterance unique across the entire corpus. In addition, to each utterance which reflects one or more dialogue design problems we have added markup indicating and explaining the violated guideline(s) (cf. Figure 2).

Ideally, the test will increase the generality that can be claimed for the tool in four ways: (1) the *system dialogue* is different from that of the Danish dialogue system (mixed initiative vs. system directed); (2) the *task type* is different (information vs. reservation); (3) the tool is being used as an early *dialogue design guide* rather than for diagnostic evaluation*;* and (4) *circumstances* are different because we do not have the scenarios used in Sundial. If the tool works well under circumstances (4), we shall know how to use it for the analysis of corpora produced in, e.g., field tests with implemented systems in which scenarios are entirely absent.

```
<u id="U2:7-1”> #hh yeah uhm a friend if mine is arriving er
                from caracas this morning uhm on flight two
                five eight #hh I need to know is there any de-
                lay on (th-) ehm on the time of arrival please
                (3.6)
<u id="S2:7-2”> please wait (10.6)
                flight be ay two five eight from caracas ar-
                rives at london heathrow terminal four at
                thirteen thirty (1.4)
<violation ref=“S2:7-2” principle=“SG2”> No feedback on
                arrival day.
<violation ref=“S2:7-2” principle=“GG7”> Scheduled versus
                actual arrival time not distinguished.
```

Figure 2. Markup of part of a dialogue from the Sundial corpus. The excerpt contains a user question and the system’s answer to that question. The system’s answer violates two guidelines, SG2 and GG7, as indicated in the markup.

Applying the tool to the Sundial corpus led to the identification of a large number of dialogue design problems all of which could be classified as violations of existing guidelines. Thus, the different system dialogue (1) and the different task type (2) compared to the Danish dialogue system did not reveal a need for additional guidelines.

Using the tool as an early dialogue design guide (3) is not significantly different from using it for diagnostic evaluation as was done in the Danish dialogue project. The main difference is that early WOZ dialogues appear to produce more, and often more complex, violations. In the 45 trial dialogues, the two experts found and agreed on 354 violations in the system’s utterances (cf. Figure 3). Many of these violations were complex in the sense that one system utterance violates several different guidelines. Typically, it appears, the expert analyser discovers the general problem raised by the utterance but only classifies one or two of the violations it produces. Two experts may thus find three or four violations arising from the same general problem. For instance, the unsatisfactory nature of the Sundial system's opening statement gave rise to 4 guideline violations. In the user test corpus from the Danish dialogue system (Section 2), experts A1 and A2 found and agreed on 117 violations in 57 dialogues, and at most two different guidelines were found violated in the same utterance. Thus it seems likely that complex violations occur less frequently in corpora from later systems development phases. This hypothesis will be tested on the Philips field trial corpus [1] in which we expect to find a further decrease in number and complexity of violations compared to the Sundial WOZ corpus and the Danish user test corpus.

The important generalisation (4) poses a particular problem. When, as in controlled user testing, the scenarios used are available, it is relatively straightforward to detect the dialogue design errors that are present in the transcribed corpus using objective methods. When, as in many realistic cases in which the tool might be used, no scenarios are available, the problem arises of whether the corpus analysers are actually able to detect *the same* problems in a dialogue prior to classification. In the Sundial case, objectivity of detection was tested by investigating if the two experts actually did detect the same problems.

Objectivity of detection was tested as follows. The two experts independently analysed 30 dialogues. Each detected violation was then discussed in detail and a typology of violations established. This, highly task dependent, typology provides an overview of the different ways in which each individual guideline was violated in the corpus. The typology is useful for revising the dialogue model. The number of *individual* violations may support estimates of system performance and acceptability but is of little importance otherwise, as many violations are identical. In a corpus containing as many guideline violations as the Sundial corpus, it will be very time consuming if not practically impossible to find all the individual violations. It is also unnecessary, because what is needed for repairing the dialogue design are the types of guideline violations that occur. As shown in Figure 4, many individual violations were found by both experts (identities) but even more were found by either A1 or A2 (complementarity). However, all agreed violations could be classified under 24 different types. Of these, 15 were found by both experts whereas 9 types were found by either A1 or A2. Upon closer analysis, the cases belonging to 6 of the 9 complementary types turned out to be part of complex violations which had been discovered by both experts. The remaining 3 types only covered 1 case each.

Guideline	**No. of agreed violations (in 30/15 dialogues)**	**No. of types**
GG1	13/9	6/5
SG1	Not relevant in information systems	
SG2	16/10	3/3
GG2	3/0	3/0
GG3	15/13	1/3
GG4	1/0	1/0
GG5	13/3	6/2
GG6	3/3	2/3
GG7	17/9	6/4
SG3	69/45	1/1
GG8	The system is successful in this respect	
GG9	The system is successful in this respect	
GG10	Massively violated in SG4 and SG5	
SG4	39/22	1/1
SG5	30/15	1/1
GG11	The “system” understands	
SG6	too well	
SG7	for these to be violated	
GG12	Violated in SG8	
SG8	9/4	1/1
GG13	Violated in SG10 and SG11	
SG9	The system has human capabilities of understanding	
SG10	0/2	0/2
SG11	0/1	0/1

Figure 3. Cases and types of dialogue design errors found in 30 + 15 Sundial dialogues analysed and sorted by guideline violated. Note that Figure 3 does not include the cases and types that were either undecidable, disagreed, or rejected (see text and Figure 4). Figure 3 does include, on the other hand, cases and types that were classified in different ways (under different guidelines) by A1 and A2.

Having discussed and classified 30 dialogues, the experts analysed another 15 dialogues from the Sundial corpus using the corpus dependent typology established during the analysis of the first 30 dialogues. This facilitated dialogue annotation which could be reduced to references to a growing table of types. Each detected violation was checked against the table. If a corresponding type was found, the violation was categorised as a case of this type, otherwise a new type was introduced. As shown in Figure 4, many more identical cases were found by the two experts in the last 15 dialogues. This is probably a result of their having discussed the findings in the first 30 dialogues. Slightly more type identities were found but also slightly more type complementarities.

	First 30 dialogues	Last 15 dialogues
Case identities (found by both experts)	81	92
Case complementarity (found by one expert)	133	41
Alternatives (different classifications)	7	3
Undecidable	1	0
Disagreements	21	2
Rejects	18	3
Type identities	15	17
Type complementarity	9	12

Figure 4. Results from the analysis of two sets of Sundial dialogues by two experts in using the evaluation tool.

However, all cases belonging to 8 of the 12 complementary types were part of complex violations that had been discovered by both experts. The remaining 4 types only covered one case each.

Results on objective (complex) problem identification are thus encouraging. Still, improvements in objective type identification would be desirable. At least two issues will have to be addressed in order to solve this problem. The concept of (corpus dependent) “types” needs elaboration and we have to construct more thorough explanations of each guideline and it use.

4. TRANSFERABILITY OF THE TOOL

In an early test of tool transferability, we trained a visiting researcher (A3) in how to use the tool. A3’s background is in language technology (computational morphology) and he has never designed dialogue models for SLDSs. He therefore appears representative of novice dialogue designers who want to use the tool as a dialogue design guide. By way of introduction, A3 received the cooperativity guidelines (cf. Figure 1), a paper on their background and development, expanding on what was said in Section 2 above and including examples of guideline violations, and a detailed tool application walkthrough of three Sundial dialogues. The complete analysis of one of these dialogues was given to him on paper. Having independently analysed a first set of 15 dialogues, A3 asked for, and had, a joint walkthrough of one of those. A3 received no detailed written information on how to use the guidelines.

We analysed the correspondence between the findings of the two experts and those of the novice. Since the two experts had thoroughly discussed their findings after having analysed 30 of the 45 dialogues, thereby improving their performance on the last 15 dialogues, the following novice/expert comparison is based on the first 30 dialogues alone (cf. Figure 4, Column 2).

A3 found a total of 154 cases and 14 types, i.e. 80% of the average number of cases found by A1 and A2, and 72% of the average number of types found by A1 and A2. A3 found 10, or 42%, of the 24 types found by A1 and A2, and he found 4 new types. Three of these were part of complex violations that already had been observed by A1 and/or A2. The last type which covered only one case was not found by the two experts. Of the 154 cases found by A3, 26 cases were rejected, disagreed with or considered undecidable by A1 and A2. This should be compared to an average of 20 such cases found by the two experts.

Taking into account that A3 never received any formal instructions on how to use the guidelines but had to generalise from examples, his performance would seem acceptable. We now have to find out how to improve it further. The next step will be to introduce A3 to the use of corpus dependent violation typologies and then have A1, A2 and A3 analyse 10 dialogues from the remaining Sundial corpus, one per subject and each based on a different scenario. If the performance of A3 improves to the extent that transferability has been successful, we have a less abstract and more operational transfer problem in front of us. It is to formalise what A3 needed to learn, thereby defining an explicit and simple training scheme for how to become an expert in using the tool without assuming person-to-person tuition. If this problem can be solved, the tool would have taken a significant step towards transferability.

5. CONCLUSION

We find the results reported in this paper encouraging. The tool has generalised well with respect to the Sundial corpus. A high degree of objectivity has been demonstrated with respect to the identification of complex dialogue design problems. Somewhat less objectivity was found in corpus dependent type identification. As regards transferability, the results obtained seem encouraging taking into account the nature and amount of introductory material provided to A3. We clearly need a more systematic and elaborate way of demonstrating and explaining the use of each individual guideline. This is ongoing work whose results, we hope, will also help the experts improve their corpus dependent type identification.

We plan to continue the tests of generality, objectivity and transferability of our tool on a small sub-corpus of the Philips corpus which comprises 13.500 field test dialogues concerning train timetable information [1]. This will add a new dialogue type, a new task type, and the circumstances of a field trial to the generality test of the tool.

REFERENCES

[1] Aust, H., Oerder, M., Seide, F. and Steinbiss, V.: The Philips Automatic Train Timetable Information System. *Speech Communication* 17, 1995, 249-262.

[2] Bernsen, N.O., Dybkjær, H. and Dybkjær, L.: Cooperativity in human-machine and human-human spoken dialogue. *Discourse Processes*, Vol. 21, No. 2, 1996, 213-236.

[3] Bernsen, N.O., Dybkjær, H. and Dybkjær, L.: What should your speech system say to its users, and how? Guidelines for the design of spoken language dialogue systems. To appear in *IEEE Computer*, 1997.

[4] Grice, P.: Logic and conversation. In Cole, P. and Morgan, J.L., Eds. *Syntax and Semantics,* Vol. 3, *Speech Acts,* New York, Academic Press, 41-58, 1975.

[5] Peckham, J.: A new generation of spoken dialogue systems: Results and lessons from the SUNDIAL project. *Proceedings of Eurospeech ‘93,* Berlin, 1993, 33-40.