An Iterative Data Collection Approach for Multimodal Dialogue Systems

Stefan Rapp[*]**, Michael Strube**[†]

[*] Sony Intern. (Europe) GmbH, Advanced Technology Center Stuttgart
Heinrich-Hertz-Strasse 1, 70327 Stuttgart, Germany
rapp@sony.de

[†] European Media Laboratory GmbH, Villa Bosch
Schloß-Wolfsbrunnenweg 33, 69118 Heidelberg, Germany
Michael.Strube@eml.villa-bosch.de

Abstract

This paper deals with the way in which data for multimodal dialogue systems are collected. We argue that for multimodal data, an iterative data collection strategy should be followed. Instead of a single major data collection effort using a "Wizard of OZ" (WOZ) or "prompting" experimental setup, several smaller data collections should accompany the system development. We also describe the "script" experimental setup we developed. It is in between the WOZ and prompting setup, and can be used as a cost effective design for the first data collection within the iterative data collection strategy.

1. Introduction

The creation of multimodal corpora raises problems which go well beyond problems raised by the creation of spoken language corpora. The collection of data for multimodal dialogue systems is more expensive than the collection of spoken language resources (SLR), because of the more complex technical setup, the increased amount of data to be collected, and the know-how which is required from multiple fields. In addition, as the construction of multimodal user interfaces is not yet a mature field, changes to the system are likely to occur even during a project. Therefore cost-effective approaches for the collection of data and ones which foster best possible use of the data are to be preferred. This is even more important since the ability to re-use the annotated data is questionable. This is not only due to differences in the language and the domain as it is for SLR but also to differences in the modalities used. For example, user interactions on a touchscreen with a finger or with a pen can differ substantially, or, as another example, 2D gesture tracking data can be useless for 3D tracking just like mono recordings are for microphone array research. The choice of covered modalities and used analysis technology add to the variability that has to match, so it is even more difficult to re-use the data gained in one project in a second project.

Usually data collections using *Wizard-of-Oz* (WOZ) or prompting experiments are performed only once before the development of a dialogue system. We argue towards designing corpus collection strategies according to software engineering practice and introduce the *iterative* corpus collection approach. We claim it to be more useful for the system at different development stages than singleton experiments using *Wizard-of-Oz* (WOZ) or *prompting* experiments. We contrast those *Collect-Once-and-Develop* (COD) data collection approaches with the iterative data collection approach. Also, the iterative data collection approach and its first step, the *script* experiment, are considered less expensive than the COD approach with a WOZ experiment.

2. The Iterative Approach

Usually, the data collection is designed during very early stages of the development of multimodal dialog systems. This is followed by an extended data collection period. However, changes in the setup of the system are not reflected in the design of the data collection, thus questioning the usefullness of the collected data. The SmartKom data collection (Türk, 2001), a large effort in collecting data for multimodal dialog systems (Wahlster et al., 2001), is such a *Collect-Once-and-Develop* (COD) approach.

Basically, two kinds of experiments for acquiring data for the development of a multimodal dialog system are used in the COD approach:

- *Wizard-of-Oz (WOZ)* experiments and
- prompts which are common for acquiring spoken language data.

In *Wizard-of-Oz* experiments, the subject interacts with a so-called *wizard*. The wizard is a human operator simulating the system behavior. Ideally, the user is not aware that a human generates the system output. WOZ experiments are distinguished by the possibility to acquire quite natural data. The subject is not only free to choose the modality, the wording and so on, but, e.g., also the plan for solving a task. However, WOZ experiments are also the most expensive ones in the preparation and the realization phase. E.g., the *wizard* needs a detailed scheme (e.g. a flow chart) for generating the proper answers. If the wizard does not stick to the scheme, the data may be inappropriate for the development of the system. However, the most important drawback of WOZ experiments is the possibility that the collected data does not correspond to the capabilities of the system. This danger arises in particular if the data is collected (only) in the beginning of a project. An example is the known phenomenon that the *wizard* has perfect speech recognition capabilities whereas the system even in the final stages of development does not have these capabilities.

Many researchers take into account the recognition capabilities of the system, e.g. by artificially imposing recognition errors on the wizard. But still, for many applications it can not be foreseen how good the recognition rate will be for the developed system, so the suspected error rate at the beginning of a project can be drastically wrong and skew the collected data.

In prompting experiments the user gets a description of what to convey to the system and how to do that, e.g., which modality to choose, which words to use and so on. On the one hand, this kind of experiment is very useful for speech and gesture recognition experiments where the phonological or the spatial realization are studied, i.e., prompting experiments are ideal for the recognition components of a system. On the other hand, data collected in prompting experiments cannot be used for the semantic and pragmatic components of such a system, because the semantics and pragmatics of the user's utterances are fixed beforehand. The user does not have any choice with respect to these levels.

It is common practice (and we believe this to be inherited from the speech recognition community), to design a data collection in the beginning of a development effort, collect data and use this data in the design and development of the dialogue system. We want to call this approach the *Collect-Once-and-Develop* (COD) approach.

Next, we want to illustrate the COD approach with a qualitative figure (Figure 1), and, after introducing our iterative approach, we contrast the COD approach with a qualitative figure depicting our proposal. The bars in the figures represent versions of systems that occur during the process of system development. The figures are strongly oversimplified in many respects, first of all, they assume functionality is measurable as a single scalar. Next we differentiate only three classes: *S* standing for system, *F* standing for fake, and *I* standing for instruction and imagination.

Figure 1 depicts the COD approach utilizing a Wizard of Oz experiment. In the figure, we see two systems: a simulated system at the beginning of system development and the implemented system at the end. In the first system, the *I* contains e.g. the task description (in our scenario, this could be: choose a movie from tonight's TV program) and implicit restrictions (in our scenario, e.g. only a certain list of movies could be displayed for selection; the perceived (or imaginated) functionality could be any list of movies on any day), the *F* contains the actions and decisions taken by the wizards, but also for example the prepared screens that the wizards can play back etc. Some of the functionalities that the wizards can trigger are done especially for the WOZ experiment (those would be covered by the *F*-block), while other functionality is as it is used in the final system, e.g. the wizard could react on what an already existing speech recognizer has understood, or make use of other existing components like a rendering engine; this is covered by the S block. We want to note that in this informal notation, WOZ could be characterized as a system with some *F* functionality, where a human operator delivers all or some percentage of the *F* functionality. In the implemented system on the right, the developers have removed all *F* functionality, there is much more *S* functionality now, however,

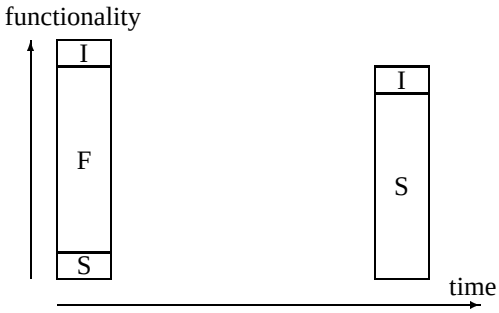

Figure 1: WOZ approach

due to some reasons, the functionality is somewhat behind the functionality of the wizard-operated system.

We contrast the COD approach with an iterative approach of collecting data. Speaking in software engineering terms, we follow the spiral model for data collection: instead of developing a full featured WOZ design at an early stage of the project, followed by one data collection phase for the implementation of the system, we want to have several data collection campaigns, where we gradually make use of more and more functionality of the system rather than to deploy wizard functionality to allow for more natural interaction than before. At the start of a project, the system usually does not have any functionality so that user interactions cannot be captured (semi-) automatically. However, this is not true if the development of the system is more advanced. In this case user interactions can be captured more or less automatically, thus reducing the cost of the annotation. If we do not want to put too much effort in the expensive wizard functionality, we have to be more restrictive in the first iterations, that is, several properties of the data like dialogue structure and task planning are fixed in the first experiments. In the following iterations the system will take over more and more functionality, e.g., it will be possible to handle dialogue structure by the system.

Next, with Figure 2, we discuss an example of our iterative data collection approach. As this approach tries to minimize extra effort (*F* functionality), it starts off with more *I* (instruction and imagination) functionality and with less total functionality. There can still be some limited *F* functionality in the beginning, for example we generated HTML pages that mimicked GUI output of our hypothetical system. Characteristic of the iterative approach is that it uses several iterations, starts with reduced total functionality and avoids *F*-functionality. We think a strength of the iterative approach is that more data is collected with at least part of the final system, hopefully it is more realistic then as it takes into account also weaknesses of the system. However, there is of course also the danger that subjects adapt too much to the restrictions (e.g., users do not use modalities which are not yet covered completely or users restrict themselves in their vocabulary because they guess that simple commands are understood better than complicated sentences). So, it is necessary to control for that phenomenon. One interesting paradigm that could be helpful here is the hybrid Wizard of Oz paradigm (Cheyer et al., 1998), where an experienced user (knowing the system limitations) translates the inter-

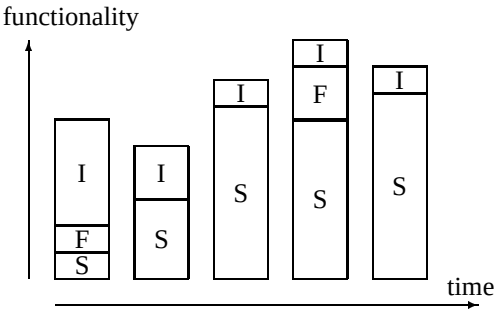

Figure 2: Iterative approach

actions of a naive user. Of course this paradigm can only be applied at an intermediate or late stage of the development. The necessity to start with a hybrid Wizard of Oz paradigm can be deduced from the observation that subjects behavior is significantly different in the beginning and towards the end of a session.

In the next section we have a closer look at the *script* experiment, a setup we developed for the first collection campaign within the iterative approach where no prototype system is available.

3. The Script Experiment

We developed the 'script' experiment as a cost-effective means for bootstrapping the system in the first iteration step of our iterative paradigm. We wanted to have a setup which is not as expensive as WOZ experiments but still provides natural data of human computer interaction. The 'script' experiment combines features of WOZ and prompting experiments. The subject gets a script with task descriptions, where each task corresponds to one dialog step. These descriptions can be prompted to the subject as for SLR. However, the descriptions neither contain the modality to be used nor the words to be said, so that interactions between spoken language and gestures can be studied. In contrast to the simple prompting setup, it is also possible to acquire a rather natural vocabulary.

It is worthwhile to note that the three kinds of experiments can be characterized by the granularity of instruction. An overview over the restrictions in the three experimental setups is given in Table 1 (not every possible level is shown, in naming we also stuck to the terminology used in the language and speech community). There are restrictions for all three kinds - even in WOZ experiments, the application area is fixed, and subjects receive a concrete task description. In prompting experiments only the phonological realization is variable while all higher levels of processing are fixed. The granularity of instruction for the script experiment is in between WOZ and prompting – phonological realization, lexical and modality choice are variable, so that it is possible to use these data for developing the system. Of the three, only WOZ experiments are suitable for research on dialogue structure and task planning. As a consequence, if using the script experiment as the initial data collection in the iterative data collection approach, data for research on dialog structure must be collected within a later campaign.

	Prompts	Scripts	WOZ
phonological realization	+	+	+
lexical choice	–	+	+
modality choice	–	+	+
dialogue structure	–	–	+
task planning	–	–	+
task/domain	–	–	–

Table 1: Granularity of Restrictions

Figure 3: Setting in the Recording Studio

4. A case study – iterative data collection for EMBASSI

To illustrate the iterative approach, we report on our experience with the collection of a corpus of multimodal human machine interaction. The investigated domain is the living room scenario, one of three application areas of the EMBASSI project (http://www.embassi.de). Here, a network links different devices to an integrated system with a common multimodal user interface utilizing speech, gesture, but also a remote control. We report on the first two campaigns, following the script experiment and the prototype evaluation experiment respectively.

4.1. First Data Collection Campaign

In the first campaign we followed the cost effective script experiment. Subjects were prompted for dialogue steps using written instructions displayed on the TV screen. During the subject's interaction, the TV showed either a still picture indicating normal TV operation, or a static on-screen GUI, both generated by a PC showing prepared HTML pages.

The subjects were recorded in a sound treated room at the Sony Advanced Technology Center Stuttgart. We tried to achieve a more realistic living room atmosphere by putting armchairs, table, floor lamp, sideboard, a TV set, a set-top-box and two VCRs into the room (Figure 3). User utterances were recorded by several microphones, a camera was registering the face and lips and two cameras were recording pointing gestures. In addition, the subject

had a standard infrared remote control at their disposal. Seven channel audio was routed through a digital mixer and recorded both on ADAT tapes and on hard disk. The face image was recorded on DV tape and on the hard disk of another PC. The data is useful for lip reading as well as face and gaze tracking experiments. Two additional PCs were used to record the uncompressed two video streams showing the pointing gestures on hard disk. Everything was controlled by two operators using an extended version of Sony's speech recording system that triggered the prompting PC, the recording of the video sequences and the log of the remote control.

This architecture was choosen because it allows a distributed preparation of the recording software. A simple protocol was jointly defined and small trigger sender and receiver programs were distributed to develop and check functionality independently. By this, we could build on the experience of selecting appropriate recording hardware, interface it to a computer and access it from software to write data to files.

In consequence, the data collection campaign could be organized in less than 2 months. The recording studio was occupied for about three weeks. During the two weeks of intensive recordings, we collected multimodal interactions of 41 age group balanced subjects (19 m, 22 f). Each subject attended the studio for about one hour. They acted on one to five scripts each or 172 scripts in total.

Although we limited the recordings of the two cameras for the gesture to only those prompts where a gesture was likely to occur, we had to handle more than 153 GB that were written (in chunks of about 15 GB) to DDS-4 tape over night between two recording days. If the data collection would have just considered mono speech recordings it would have easily fit into a half gigabyte.

4.2. Second data collection campaign

For the second data collection campaign, we exhibited an early prototype to the subjects. It could treat a small but core part of the system (browsing an electronic program guide (EPG) and selecting a program for recording), either by GUI, speech or a multimodal combination of both. Audio was captured as for the first campaign. As the people interested in the vision aspects had enough data from the first campaign, we could avoid handling huge data masses and were able just to record the overall scene on DV tape and MPEG-4 for annotation purposes. For annotation of the system's activities we developed an XML-based centralized logging facility.

The campaign was combined with an evaluation of the influence of different output strategies (Krämer and Nitschke, 2002), so people had to fill in questionnaires as well during their 1.5 hour stay. Different subject groups had to interact with three versions of the system, each having specific output capabilities, i.e. GUI only, GUI and synthesis, GUI and synthesis and animated face. The were asked to solve 3 task: record a specific programme, browse and select any interesting programme for recording, and again record a specific programme. This time, we recorded 65 age and gender balanced subjects. As we recorded more subjects this time, we scheduled the recordings over three weeks. An experimenter introduced the subjects to the task, and two people controlled the technical equipment.

4.3. Data annotation

The annotation of both collection efforts was done as follows: In parallel to subsequent recordings or after the collection was finished, the orthographic transliteration of speech was done manually. Next, an automatic segmentation into word, syllable and phoneme files was performed using the system reported in (Rapp, 1995). Schmid's part-of-speech tagger (Schmid, 1995) was applied to yield morpho-syntactic information.

All of these automatically derived annotations were then converted into XML. In doing so, related data from different levels (like phonemes, words and turns) was distributed on different files, with the relations between elements being represented by means of *standoff annotation* (Thompson and McKelvie, 1997) (a notion introduced by (Ide and Priest-Dorman, 1996) as *remote markup*). In this technique, embedding of one element within another is expressed through a special attribute of the latter, which has as its value the *ID* of the embedded element and which is interpreted as a pointer to this element. Apart from the practical advantage of directly supporting the collaboration of diverse sites on the same data (as outlined above), keeping apart data from different levels of description is preferable for methodological reasons as well. For display and annotation, the original data is reconstructed from the distributed files. For this purpose, a tool (Müller and Strube, 2001) was developed which directly supports standoff annotation creation and resolution. With this tool, our multimodal corpus was manually annotated for coreference.

5. Conclusions and Future Work

This paper explains the design considerations for the data collection strategy for a multimodal dialogue system. The key point of the iterative approach is that we do several small campaigns rather than one single large data collection. We reported on the experience that we gained in the first iteration for which we developed the script experiment, that stands between the Wizard of Oz and the prompting experiments. We did not try to hide from the users the fact that the dialogue structure and task planning is totally fixed by the script. Yet, some subjects seem to be unaware of the fact that all system reactions were totally predetermined, as one of the subjects mentioned after the recordings. In the second collection iteration we exhibited a (functionally severely limited) prototype to subjects and could collect realistic data of the core task.

Characteristic of the iterative approach is that it starts with reduced total functionality and avoids fictive functionality. The corpus collection can be accomodated to changes during development, so the risk of producing a large quantity of (partly) unrealistic data is reduced. Another strength is that more data is collected with at least part of the final system, so that weaknesses of the system are also taken into account. However, there is also the danger that subjects adapt too much to the restrictions (e.g., users do not use modalities which are not yet covered completely).

We have decided to follow the iterative approach in the EMBASSI project, and are confident of adhering to the iterative paradigm in future projects.

6. Acknowledgments

The work presented here has been partially funded by the German Ministry of Research and Technology as part of the EMBASSI project (01 IL 904 S 8, 01 IL 904 D/2), by Sony International (Europe) GmbH and by the Klaus Tschira Foundation.

We would like to thank all the people involved in the two collection campaigns, especially Hanno Braun, Andreas Haag, Günther Schlienz, Jürgen Schimanowski, Georg Michelitsch, Thorsten Tödtmann, Christian Küblbeck, Ulrich Dieckmann, Klaus Dorfmüller, Vítor Sá, Christoph Müller, Nicole Krämer, Julia Nitschke and Christoph Meyer zu Kniendorf.

7. References

A. Cheyer, L Julia, and J.C. Martin. 1998. A unified framework for constructing multimodal experiments and applications. In *CMC'98*, pages 63–69, Tilburg, The Netherlands.

Nancy Ide and Greg Priest-Dorman. 1996. The corpus encoding standard. http://www.cs.vassar.edu/CES.

Nicole Krämer and Julia Nitschke. 2002. Ausgabemodalitäten im Vergleich: Verändern sie das Eingabeverhalten der Benutzer? In R. Marzi, editor, *Bedienen und Verstehen. 4. Berliner Werkstatt Mensch–Maschine–Systeme*, Düsseldorf. VDI–Verlag.

Christoph Müller and Michael Strube. 2001. MMAX: A tool for the annotation of multi-modal corpora. In *Proceedings of 2nd IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, Seattle, Wash., 5 August, pages 45–50.

Stefan Rapp. 1995. Automatic phonemic transcription and linguistic annotation from known text with Hidden Markov Models/An aligner for German. In *Workshop "Integration of Language and Speech in Academia and Industry"*, Moscow, November. ELSNET goes east and IMACS.
http://www.ims.uni-stuttgart.de/~rapp/aligner.ps.

Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to German. In *Proceedings of EACL SIGDAT-Workshop*, Dublin, Ireland.

Henry S. Thompson and David McKelvie. 1997. Hyperlink semantics for standoff markup of read-only documents. In *Proceedings of SGML Europe '97*, Barcelona, Spain, May.

Ulrich Türk. 2001. The technical processing in SmartKom data collection: a case study. In *Proceedings of Eurospeech 2001 Scandianavia*, Aalborg, Denmark, 3-7 September, 2001, pages 1541–1544.

Wolfgang Wahlster, Norbert Reithinger, and Anselm Blocher. 2001. SmartKom: Multimodal communication with a life-like character. In *Proceedings of Eurospeech 2001 Scandianavia*, Aalborg, Denmark, 3-7 September, 2001, pages 1547–1550.