The SAMMIE Corpus of Multimodal Dialogues with an MP3 Player

Ivana Kruijff-Korbayová*, Tilman Becker†, Nate Blaylock*, Ciprian Gerstenberger*, Michael Kaißer†, Peter Poller†, Verena Rieser*, Jan Schehl†

*Saarland University, Saarbrücken, Germany
{korbay, blaylock, gerstenb, vrieser} @coli.uni-sb.de
†DFKI, Saarbrücken, Germany
{becker, poller, schehl}@dfki.de

Abstract

We describe a corpus of multimodal dialogues with an MP3 player collected in Wizard-of-Oz experiments and annotated with a rich feature set at several layers. We are using the Nite XML Toolkit (NXT) (Carletta et al., 2003) to represent and further process the data. We designed an NXT data model, converted experiment log file data and manual transcriptions into NXT, and are building tools for additional annotation using NXT libraries. The annotated corpus will be used to (i) investigate various aspects of multimodal presentation and interaction strategies both within and across annotation layers; (ii) design an initial policy for reinforcement learning of multimodal clarification requests.

1. Introduction

In the TALK project[1] we are developing a multimodal dialogue system for an MP3 application for in-car and in-home use. The system functionalities include playback control, manipulation of playlists, and searching a large MP3 database. The system should exhibit natural, flexible interaction and collaborative behavior. In order to achieve this, it needs to provide advanced adaptive multimodal output.

To determine the interaction strategies and range of linguistic behavior naturally occurring in this scenario, we conducted two series of Wizard-of-Oz experiments: SAMMIE-1 involved only spoken interaction, SAMMIE-2 was multimodal, with speech and screen input and output.[2] Our goal was not only to collect data on user interactions with such a system, but also to observe what interaction strategies humans naturally use and how efficient they are. The experiment setup we developed for this purpose is described in (Kruijff-Korbayová et al., 2005).

In order to investigate the presentation and interaction strategies systematically, we have been annotating the corpus on several layers, representing linguistic, multimodal and context information. The annotated corpus will be used (i) to investigate various aspects of multimodal presentation and interaction strategies both within and across the annotation layers; (ii) to design an initial policy for reinforcement learning of multimodal clarifications. We use the Nite XML Toolkit (NXT) (Carletta et al., 2003) to represent and browse the data and to develop annotation tools.

Below we first briefly recapitulate our experiment goals (Section 2.), setup (Section 3.) and the collected data (Section 4.); we then describe our annotation methods and tools (Section 5.) and the annotation layers and features (Section 6.).

[1]TALK (Talk and Look: Tools for Ambient Linguistic Knowledge; http://www.talk-project.org)

[2]SAMMIE stands for Saarbrücken Multimodal MP3 Player Interaction Experiment.

2. Experiment Goals

We have so far conducted two series of Wizard-of-Oz experiments. The speech-only SAMMIE-1 experiment was essentially a pilot study aimed to get an idea of the range of linguistic and dialogue phenomena in this domain of application. We used our experience to design the more complex setup for the multimodal SAMMIE-2 experiment, which was geared towards our research questions. We briefly summarize these below.

Multimodal Presentation Strategies The main aim of the SAMMIE-2 experiment was to identify strategies for the screen output, and for the multimodal output presentation. In particular, we want to learn when and what content is presented (i) verbally, (ii) graphically or (iii) by some combination of both modes. We expect that when both modalities are used, they do not convey the same content or use the same level of granularity. These are important questions for multimodal fission and for turn planning in each modality. We also plan to investigate how the presentation strategies influence the responses of the user, in particular w.r.t. what further criteria the user specifies, and how she conveys them.

Multimodal Clarification Strategies The SAMMIE-2 experiment should also serve to identify potential strategies for multi-modal clarification behavior and to investigate individual strategy performance. The wizards' behavior will give us an initial model how to react when faced with several sources of interpretation uncertainty. In particular we are interested in what medium the wizard chooses for the clarification request, what kind of grounding level she addresses, and what "severity" she indicates. [3] In order to invoke clarification behavior we introduced uncertainties on several levels, for example, multiple matches in the database, lexical ambiguities (e.g., titles that can be interpreted denoting a song or an album), and errors on

[3]Severity describes the number of hypotheses indicated by the wizard: having no interpretation, an uncertain interpretation, or several ambiguous interpretations.

the acoustic level. To simulate non-understanding on the acoustic level we corrupted some of the user utterances by randomly deleting parts of them (Kruijff-Korbayová et al., 2005). The data gathered in the SAMMIE-2 setup is used to "bootstrap" a reinforcement learning-based clarification strategy (Rieser et al., 2005).

3. Experiment Setup

In both SAMMIE-1 and 2 the subjects performed several tasks as users of an MP3 player application simulated by a wizard. The tasks involved exploring the contents of a database of information (but not actual music) of more than 150,000 music albums (almost 1 million songs), to which only the wizard had access. [4]

In SAMMIE-1, 24 subjects participated each in one session with one of two wizards. They worked on eight tasks, for maximally 30 minutes in total. Tasks were of three types: (1) finding a specified title; (2) selecting a title satisfying certain constraints; (3) building a playlist satisfying certain constraints.

In SAMMIE-2, 42 subjects participated each in one session with one of six wizards. They worked on two times two tasks[5] for maximally twice 15 minutes. Tasks were of two types: (1) searching for a title either in the database or in an existing playlist; (2) building a playlist satisfying a number of constraints.

Both users and wizards could speak freely. The interactions were in German (although most of the titles and artist names in the database are English). In SAMMIE-2, the wizards could use speech or display only or combine speech and display, and the users could speak and/or make selections on the screen. We implemented modules to automatically calculate screen output options the wizard could select from to present search results, e.g., various versions of lists and tables (Kruijff-Korbayová et al., 2005).

In SAMMIE-1 the users and the wizards could hear each other directly, and there were no disruptions to the speech signal. In SAMMIE-2, we used a more complex setup with no direct spoken contact, in order to reproduce more realistic conditions resembling interaction with a dialogue system. The wizard's utterances were immediately transcribed and presented to the user via a speech synthesizer. The user's utterances were also transcribed and the wizard was only presented the transcript. As described in (Kruijff-Korbayová et al., 2005) we sometimes corrupted the transcript in a controlled way by replacing parts of the transcribed utterances by dots, in order to simulate understanding problems at the acoustic level.

We implemented our experimental system on the basis of the Open Agent Architecture (OAA) (Martin et al., 1999), a framework for integrating a community of software agents in a distributed environment. We made use of the OAA monitor agent to trace all communication events within the system for logging purposes.

[4]The information was extracted from the FreeDB database, freely available at `http://www.freedb.org`.

[5]For the second two tasks there was a primary task using a *Lane Change* driving simulator (Mattes, 2003).

4. Collected Data

For both SAMMIE-1 and 2 the data for each session consists of a video and audio recording and a user questionnaire; for SAMMIE-2 there also is a log file for each session[6] which consists of OAA messages in chronological order, each marked by a timestamp. The messages contain various information obtained during the experiment, e.g., the transcriptions of the spoken utterances, the wizard's database query and the number of results, the screen option chosen by the wizard, the selections made by the user in the graphical output, the wizard's online classification of clarification requests, user satisfaction and their perceived task completion, etc. The SAMMIE-1 corpus contains 24 sessions with approximately 2600 wizard and subject turns in total; the transcripts amount to approximately 248 KB plain text. The SAMMIE-2 corpus contains 21 sessions with 1700 turns; the transcripts amount to approximately 164 KB plain text. The data has been transcribed and is being annotated at multiple levels as described below.

5. Annotation Methods and Tools

Since we are interested in investigating various aspects of the multimodal presentation and interaction strategies, including aspects of contextually adaptive linguistic and graphical realization, we are annotating a rich set of features at multiple layers. Each layer is annotated independently, but subsequent investigations involve exploration and automatic processing of the integrated data across layers. Among the existing toolkits that support multi-layer annotation, it was decided to use the Nite XML Toolkit (NXT) (Carletta et al., 2003)[7] in the TALK project.

We created our NXT-based corpus in several steps: **(1)** The speech data was manually transcribed using the Transcriber tool.[8] **(2)** We automatically extracted features at various annotation layers by parsing the OAA messages in the log files. **(3)** We automatically converted the transcriptions and the information from the log files into our NXT-based data representation format; features annotated in the transcriptions and features automatically extracted from the log files were assigned to elements at the appropriate layers of representation during this step.

For the annotation of additional features we use a mixture of manual and (semi-)automatic annotation techniques.

Manual annotation: We use tools specifically designed to support the particular annotation tasks. We briefly describe them below.

As already mentioned, we used Transcriber for the manual transcriptions. We also performed certain relatively simple annotations directly on the transcriptions and coded them in-line. This includes the identification of (i) self-speech; (ii) utterances that convey the results of database queries; (iii) expressions referring to domain objects (e.g., songs, artists and albums) and (iv) their phonetic transcription.

For several other manual annotation tasks we have been building specialized tools based on the NXT library of rou-

[6]Due to data loss caused by a technical failure, complete data (video, audio and log files) only exists for 21 of the 42 sessions.

[7]`http://www.ltg.ed.ac.uk/NITE/`

[8]`http://trans.sourceforge.net/`

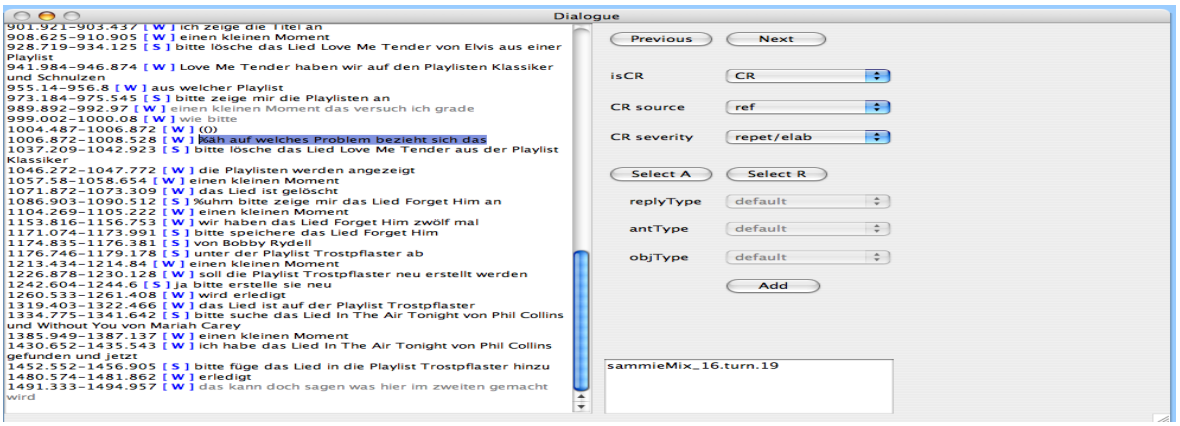

Figure 1: NXT-based tool for annotating CRs

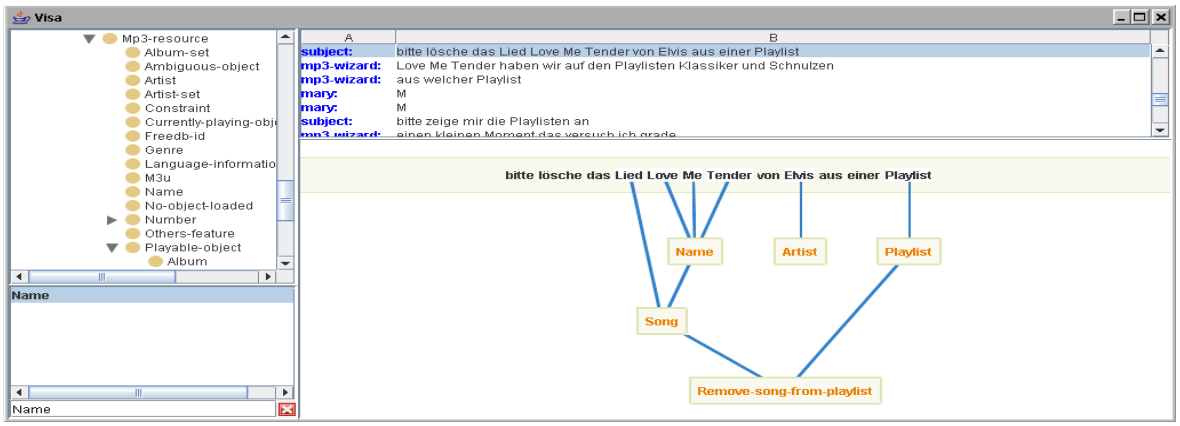

Figure 2: NXT-based tool VISA for annotating propositional content

tines for building displays and interfaces based on Java Swing (Carletta et al., 2003). Although NXT comes with a number of example applications, these are tightly coupled with the architecture of the corpora they were built for. We therefore developed core NXT-based tool libraries for our own corpus; using these libraries, we implement specialized tools for different annotation tasks (the annotation of clarification requests, syntactic-clause units and their features, dialogue acts, task segmentation and completion, referring expressions and the relations between them). To facilitate tool development, NXT provides GUI elements linked directly to corpora elements and support for handling complex multi-layer corpora. This proved very helpful.

For illustration, Figure 5. shows a screenshot of our clarification request (CR) annotation tool. It allows one to select an utterance in the left-hand side of the display by clicking on it, and then choose the attribute values from the pop-down lists on the right-hand side; one can also create and annotate relations between elements by clicking on “Create A” (to create a CR antecedent) and “Create R” (to create a CR reply).

For annotating propositional content we are going to use VISA, an NXT-based annotation tool developed at DFKI within the AMI project[9]. It loads an OWL-based ontology and corpus data from the word- and utterance-layer in order to annotate propositional content by assigning given ontological concepts to words, word-groups or other concepts. Figure 5. shows a screenshot of a preliminary version of the VISA tool.

Automatic annotation using indexing: NXT also provides a facility for automatic annotation based on NiteQL query matches (Carletta et al., 2003). Some of our features, e.g., the dialogue history ones, can be easily derived via queries.

6. Annotation Layers and Features

Our corpus consists of the following layers. Two base layers: words and graphical output events; both are time-aligned. On top of these, structural layers correspond to one session per subject, divided into task sections, which consist of turns, and these consist of individual utterances, containing words. Graphical output events will be linked to turns at a featural layer.

Further structural layers are defined for syntactic clauses and clause-like units, domain objects and discourse entities (units are expressions consisting of words), and for CRs and dialogue acts (units are utterances). We keep independent layers of annotation separate, even when they could in

[9] AMI (Augmented Multi-party Interaction; http://www.amiproject.org)

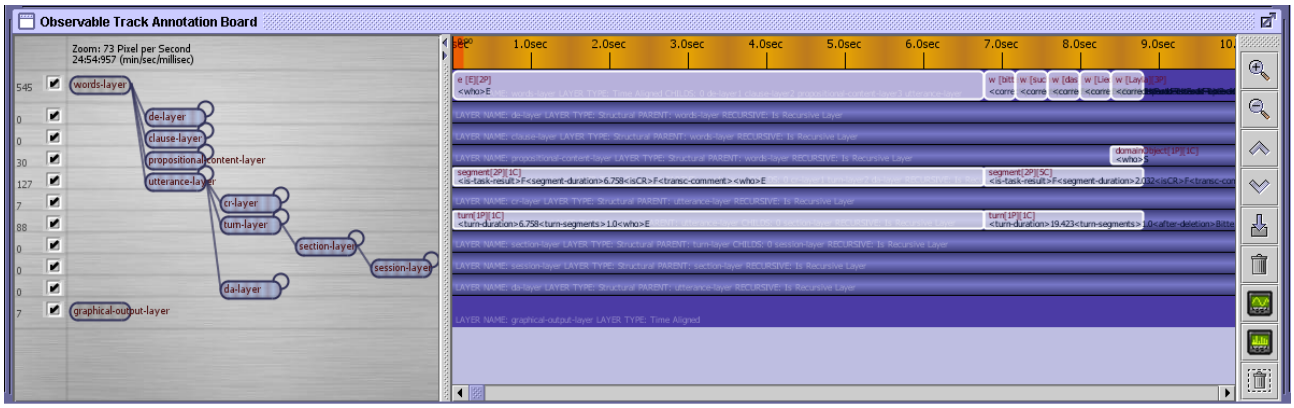

Figure 3: SAMMIE-2 corpus displayed in Amigram

principle be merged into a single hierarchy.

NXT uses a stand-off XML data representation format that consists of several XML files that point to each other. The NXT data model is a multi-rooted tree with arbitrary graph structure. Each node has one set of children, and can have multiple parents. Figure 3 shows a screenshot made with Amigram (Lauer et al., 2005), a generic tool for browsing and searching NXT data. On the left-hand side one can see the dependencies between the layers. The elements at the respective layers are displayed on the right-hand side.

We describe the features we have been annotating at the various layers in more detail below. Detailed annotation guidelines can be found in (Blaylock et al., 2006).

Word-layer features Words and other types of sounds are time-stamped; the annotation includes features for self-speech, pronunciation, word form, lemma and part-of-speech and deletion status (to reflect whether a user's word was sent to the wizard). All features, except lemma and part-of-speech, were automatically extracted from the transcriptions and converted into NXT-based data.

Graphical output features Graphical output events are time-stamped, and their features encode the type and amount of information displayed, the display option selected by the wizard, and the user's choices. In addition the graphical output layer holds also information of the database search, if any, that preceded the a shown graphical output, namely the query constraints and the number of matches. All information was automatically extracted from the log files and converted into NXT-based data.

Utterance features Annotation at the utterance level includes duration, error rates due to word deletion and the information whether the utterance presents database search results. All features were automatically extracted from the transcriptions and converted into NXT-based data.

Syntactic features of clauses We are manually annotating clauses and clause-like units within utterances with various features describing the syntactic structure, e.g., mood, polarity, diathesis, complexity and taxis, the presence of marked syntactic constructions such as ellipsis, fronting, extraposition, cleft, etc.

The main purpose of the features at this level is to get a detailed description of the syntactic constructions used in the SAMMIE corpora, including grammatical functions and some semantic features. This description will allow us to systematically investigate the use of a range of constructions in different contexts, and use the results to motivate and/or refine sentence planning decisions (cf. also (Poesio, 2004), (Poesio, 2000) and (Jordan and Walker, 2005)).

For illustration, consider the range of alternative realizations in (1a–1e).[10] The features `type`, `ellipsis` and `diathesis` characterize the global syntactic pattern, according to traditional grammar. For instance, the unit in (1a) is **imperative**, **nonElliptical** and **active**, (1b) is **declarative**, **nonElliptical** and **active**, and (1c) is **declarative**, **nonElliptical** and **passive**. The feature `verbed` reflects the existence of a verbal part in the unit: **true** in (1d) and **false** in (1e).

Next, the features `vfContent`, `hasSplitNP`, `hasExtraP` and `hasScrambledMF` represent information about marked syntactic constructions in terms of the Topologial Field Model for Germanic languages (cf. (Höhle, 1983)), respectively: which kind of element is fronted (e.g., a **prepositional phrase** in (1b) vs. the **subject** in (1c); whether a unit exhibits split NPs as in (1b) and (2a); whether a unit exhibits extraposition, e.g., the PP in (2a); and finally, whether a unit exhibits scrambling in the Mittelfeld (middle field) as in (3).

(1) a. Subj: suche Titel von Madonna
Subj: search titles by Madonna
Subj: Search titles by Madonna!

b. MP3: von Madonna habe ich 711 Titel gefunden
MP3: by Madonna have I 711 titles found
MP3: I've found 711 songs by Madonna.

c. MP3: 711 Titel von Madonna wurden gefunden
MP3: 711 titles by Madonna were found
MP3: 711 songs by Madonna have been found.

d. MP3: 711 Titel gefunden
MP3: 711 titles found
MP3: 711 songs found.

e. MP3: 711 Titel von Madonna
MP3: 711 titles by Madonna
MP3: 711 songs by Madonna found.

[10] All examples correspond to utterances found in the corpus. In some cases we changed the lexical content for ease of comparison.

(2) a. Subj: ich möchte eine Playlist erstellen mit drei Liedern
Subj: I like a playlist build with three songs
Subj: I'd like to build a playlist with three songs!

b. MP3: wie soll denn die Playlist heissen
MP3: how should then the playlist be called
MP3: How to name the playlist?

(3) ich möchte aus diesen drei Liedern eine Playliste erstellen
I wish out-of these three songs a playlist make
I'd like to make a list with these three songs.

This level of annotation is currently in progress.

Domain objects and discourse entities In order to investigate systematic reference phenomena in our domain, so that we can incorporate the findings in the natural language generation module of the SAMMIE system, and thus improve the quality of the system output, we annotate expressions that introduce discourse entities. For this purpose, we annotate various properties of referring expressions and coreference/bridging links between them.
Each discourse entity is annotated with the following features: `deType` (e.g., **song**, **artist**, **album**); `typeMention` (**true** in *das Lied Yesterday*, **false** in *Yesterday*); `properNameMention` (**true** in *das Lied Yesterday*, **false** in *ein Lied*); `npForm` (e.g., **defNP** in *das Lied Yesterday*, **indefNP** in *ein Lied*); `gFunction` (e.g., **directObject** for *eine Playlist* in (2a), **subject** for *die Playlist* in (2b)); and `informationStatus` (e.g., **new** for *eine Playlist* in (2a) or **old** for *die Playlist* in (2b)).
This level of annotation is currently in progress.

Propositional content We plan to annotate propositional content of utterances by assigning concept instances to expressions using the OWL-based ontology of our multimodal dialogue system for in-car application. As noted above, we will use the VISA annotation tool designed specifically for this kind of annotation (see Figure 5.).

Dialogue acts For the annotation of dialogue acts we will use a taxonomy inspired by existing schemes, mainly (Traum and Hinkelman, 1992), DAMSL (Core and Allen, 1997), DIT (Bunt, 2005), and DATE (Walker and Passoneau, 2001). We will distinguish between *speech acts* (corresponding to DAMSL forward-looking functions) and *grounding acts* (corresponding closely to DAMSL backward-looking functions) The third dimension, reflecting *task-specific actions* (as in DATE and DIT) corresponds to and is annotated at the propositional content level. Within each dimension we use a hierarchical organization to allow for introducing finer distinctions if needed in the future. Annotation at this level is in preparation stage.

Clarification requests (CRs) A clarification object is a triple of three related utterances; one utterance being the CR itself, the antecedent (what caused the CR) and the reply to that CR. For each of these three utterances we are annotating additional attributes. For the CR itself we annotate the (`source`) and degree of uncertainty (`severity`) as indicated by the speaker. Furthermore we are interested whether the wizard showed a graphic when asking a CR. We get this information from the graphical layer (`graphic`). The problem source of the clarification request describes the type of understanding problem which caused the need to clarify. Its attributes map to the level of understanding as defined by (Clark, 1996), the `channel, acoustic, reference, intention` level. The problem severity describes which type of feedback the CR-initiator requests from the other dialogue participant (`repetition, elaboration, confirmation, disambiguation`). These values also reflect how many hypotheses are available to the CR-initator.
For the antecedent we are interested in its dialogue act and the discourse entities which were referred to. Both of these attributes are available from other annotation layers.
The reply is classified according to its information gain and the complexity of the underlying language model. These features reflect that a good clarification strategy for spoken dialogue systems should elicit responses which maximize the information gain while minimizing recognition errors. We summarize those features into an attribute `replyType` with `y/n, repeat, paraphrase, add, omit, add/omit, select, change-topic` as possible values.

(4) Subj: [Bitte die Playliste anzeigen]
Subj: Please show me the playlist.
Antecedent: `speech_act=request action_type=show-playlist`

MP3: **[Welche Playliste willst du sehen?]**
MP3: Which playlist do you want to see?
CR: `source=reference, severity=elaboration, graphic=no`

Subj: [Beatles.] Subj: Beatles.
Reply: `replyType=addition`

Antecedent and reply features provide input to the user model, and CR features to the action space used for reinforcement learning (Rieser et al., 2005).
The CR annotation is done manually. We chose a cascaded annotation process (Carletta et al., 1997), which enabled us to achieve very reliable CR identification and annotation with $\kappa = 0.826$. For the CR antecedent and reply and their respective attributes we are currently evaluating reliability.

Turn features The turn level comprises several features which were automatically computed from the log files: the turn duration and the number of utterances in the turn, the text of the user's turn after potential deletion of its parts and the text of a wizard's turn as sent to the text-to-speech synthesis module. In order to use the corpus for extracting Information-State-Update (ISU)-based dialogue strategies (Traum and Larsson, 2003), we additionally annotate dialogue history features by an NXT query. Dialogue history features are time delay, dialogue duration so far, number of CRs etc. These values accumulate over time, and will be computed automatically on the basis of other features.

Task features The annotation includes a set of features for estimating user satisfaction as a reward for reinforcement learning (Rieser et al., 2005). We elicited via user questionnaire subjective task satisfaction and perception of task completion for each task, as well as a final user satisfaction, following the PARADISE framework, (Walker et al., 1997). For each dialogue we also manually annotate the objective overall and sub-task completion, whether a (sub-)task was resumed, how the task was terminated (i.e.

if terminated due to time constraints, or abandoned by the user), whether the user was operating the driving simulator, the overall task duration, etc. Annotation test runs for task features and the following session features already showed promising results.

Session features The annotation comprises subject and wizard information, user questionnaire answers, and accumulating attribute values from other layers (by NXT query).

7. Summary

We described a corpus of multimodal dialogues with an MP3 music player application, gathered through Wizard-of-Oz experiments. The corpus is represented and annotated using NXT-based tools. Our multi-layer data model relates linguistic and graphical realization to a rich set of context features and represents structural, hierarchical interactions between different annotation layers. We combined different annotation methods to construct the corpus. Many features have been automatically extracted from the transcriptions and converted into NXT-based data. Manual annotation and annotation evaluation is on-going. The corpus will be used (i) to investigate multimodal presentation and interaction strategies with respect to dialogue context and (ii) to design an initial policy for reinforcement learning of multimodal clarification strategies.

8. Acknowledgments

This work has been carried out in the TALK project, funded by the EU 6th Framework Program, project No. IST-507802. For help in carrying out the experiments and annotating the data we would like to thank D. Steffen from the CLT company and several students of the Saarland University: B. Fromkorth, M. Grác, A. Moos, and M. Wirth.

9. References

N. Blaylock, B. Fromkorth, C. Gerstenberger, I. Kruijff-Korbayová (ed.), O. Lemon, P. Manchón, A. Moos, V. Rieser, C. del Solar, and K. Weilhammer. 2006. Annotators handbook. Deliverable D6.2, TALK Project.

H. Bunt. 2005. A framework for dialogue act specification. Paper presented at the Joint ISO TC 37/SC 4/TDG 3 and ACL-SIGSEM WG Workshop on the Representation of Multimodal Semantic Information, Tilburg, January 2005.

J. Carletta, A. Isard, S. Isard, J. C. Kowtko, G. Doherty-Sneddon, and A. H. Anderson. 1997. The reliability of a dialogue structure coding scheme. *Computational Linguistic*, 1(23):13–31.

J. Carletta, S. Evert, U. Heid, J. Kilgour, J. Robertson, and H. Voormann. 2003. The NITE XML Toolkit: flexible annotation for multi-modal language data. *Behavior Research Methods, Instruments, and Computers, special issue on Measuring Behavior*. Submitted.

H. Clark. 1996. *Using Language*. Cambridge University Press.

M. Core and J. Allen. 1997. Coding dialogs with the damsl annotation scheme. In *Proceedings of the AAAI Fall Symposium on Communicative Action in Humans and Machines*, pages 28–35, Boston, MA, November.

T. Höhle. 1983. *Topologische Felder*. University of Cologne.

P. Jordan and M. Walker. 2005. Learning Content Selection Rules for Generating Object Descriptions in Dialogue. *Journal of Artificial Intelligence Research*, 24:157 – 194.

I. Kruijff-Korbayová, T. Becker, N. Blaylock, C. Gerstenberger, M. Kaißer, P. Poller, J. Schehl, and V. Rieser. 2005. An experiment setup for collecting data for adaptive output planning in a multimodal dialogue system. In *Proc. of ENLG*, pages 191–196.

C. Lauer, J. Frey, B. Lang, T. Becker, T. Kleinbauer, and J. Alexandersson. 2005. Amigram - a general-purpose tool for multimodal corpus annotation. In *Proc. of MLMI*.

D. L. Martin, A. J. Cheyer, and D. B. Moran. 1999. The open agent architecture: A framework for building distributed software systems. *Applied Artificial Intelligence: An International Journal*, 13(1–2):91–128.

S. Mattes. 2003. The lane-change-task as a tool for driver distraction evaluation. In *Proc. of IGfA*.

M. Poesio. 2000. Annotating a Corpus to Develop and Evaluate Discourse Entity Realization Algorithms: Issues and Preliminary Results. In *Proc. of LREC*, pages 211 – 218, Athens, May/June.

M. Poesio. 2004. The MATE/GNOME Scheme for Anaphoric Annotation, Revisited. In *Proc. of the SIGDIAL*, pages 154–162, Boston, April.

Verena Rieser, Ivana Kruijff-Korbayová, and Oliver Lemon. 2005. A corpus collection and annotation framework for learning multimodal clarification strategies. In *Proc. of SIGdial6-2005*, pages 97–106.

D. R. Traum and E. A. Hinkelman. 1992. Conversation acts in task-oriented spoken dialogue. *Computational Intelligence*, 8(3):575–599. Also available as University of Rochester Department of Computer Science Technical Report 425.

D. Traum and S. Larsson. 2003. The information state approach to dialogue management. In *Current and New Directions in Discourse & Dialogue*. Kluwer Academic Publishers.

M. Walker and R. Passoneau. 2001. DATE: A dialogue act tagging scheme for evaluation. In *Human Language Technology Conference*, pages 1–8.

M. Walker, D. Litman, C. Kamm, and A. Abella. 1997. Paradise: a general framework for evaluating spoken dialogue agents. In *Proc. of the 35th Annual Meeting of the Association for Computational Linguistics, ACL/EACL*, pages 271–280.