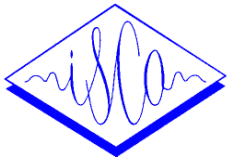

6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

A FRAMEWORK FOR EVALUATING CONTEXTUAL UNDERSTANDING*

H. Bonneau-Maynard, L. Devillers

LIMSI-CNRS, BP 133
91403 Orsay cedex, FRANCE
{hbm, devil}@limsi.fr

ABSTRACT

In this paper we propose and describe a framework for evaluating and diagnosing the understanding component of a spoken dialog system which can be applied to both literal and contextual understanding. We have observed in a previous experiment that contextual understanding performance is strongly correlated to the user satisfaction. The framework uses a glassbox approach to diagnose the interpretation modules. Results are given on a 1681 literal understanding test set and 100 contextual understanding test set.

1. INTRODUCTION

Many projects were or are currently addressing system dialog evaluation, such as the French ARC B2 of AUF, the DISC-1 and DISC-2 European projects, the American project Communicator [10] of ARPA and on a more general level EAGLES [5] and ELSE projects. We are participating in French ARC B2 of AUF [8] whose aim is to evaluate French language spoken dialogue systems on a common task of touristic information.

In a recent experiment [3] we have shown that similar spoken dialog features appear to be important features for user satisfaction across different tasks and user groups. The two tasks were PARIS-SITI [4] (Tourist information interactive kiosk), which was developed in the context of ARC B2 and ARISE [2] (train timetable telephone information server). Using PARADISE paradigm [9], we have observed that automatic speech recognition performance and contextual understanding performance explain around 40% of the variance in user satisfaction. We observed that contextual understanding performance is a more reliable predictor of user satisfaction than literal understanding.

In this paper, we describe a framework for evaluating and diagnosing the understanding component of a spoken dialog system which can be applied to both literal and contextual understanding.

This framework includes: a literal understanding test set, a contextual test set, and a scoring tool. This framework enables assessment of dialog situation and linguistics difficulties observed during a dialog such as anaphora, ellipsis, or negation. Both test sets are issued from around 200 real dialogs recorded in previous experiments [3, 4]. This framework can be also used to estimate the relative impact of recognition errors, by substituting the exact transcription with the recognized sentence. The proposed framework is used in the context of the PARIS-SITI data retrieval dialog system. PARIS-SITI allows users to obtain information such as prices, payment procedures, opening hours, address, trip, descriptions and services offered, for a variety of objects (hotels, restaurants, cinemas, department stores, museums and monuments) in Paris. Therefore, examples and results are extracted from this task.

The stucture of the paper is as follows. In section 2, we discuss about the contextual understanding evaluation. Section 3 describes the proposed framework and the scoring tool. Preliminary results are given in section 4. Discussion is then developped in section 5.

2. CONTEXTUAL UNDERSTANDING EVALUATION

The development of a methodology for performance evaluation is of major importance to the field of spoken language dialog systems. It is essential to be able to compare multiple systems or different versions of the same system. The state of the art in evaluating spoken dialog system is to evaluate the system and components in terms of a set of objective and subjective metrics. A spoken dialog system is generally composed of different modules: speech recognition, natural language understanding, generation, speech synthesis, data access and dialogue management [6]. The subjective metrics consists in judging some property of the dialog system (rarely the components) by reference to user opinion. The objective metrics are global measures such as elapsed time for task completion and success task and also direct measures on the performance of a component such as speech recognition rate and understanding rate.

We observed in previous experiments that contextual understanding performance (result of the understanding module taking into account the dialog history) is a more reliable predictor of user satisfaction than literal understanding (result of the understanding module on the exact transcription of the user utterance without considering dialog context). Thus, we are convinced that a semantic error diagnosis must be done for both literal and contextual understanding test.

Contextual understanding module evaluation assesses the actual performance of the semantic analyzer of a spoken language system. While it is possible to objectively measure recognition performance, evaluation of the under-

* This work was partially financed by the ARC B2 AUF.

standing module is not straightforward.

Two typical types of understanding system evaluation are used: glass box and black box evaluation.

In a black box methodology, only input and output are available to the evaluator. The performance is coming from the comparison between expected and actual output. So, the evaluation can be estimated as correct even though the internal interpretation is wrong. Futhermore, it does not allow precise diagnossis of the errors.

The MACDOW paradigm [7] for the natural language systems evaluation which has been used to evaluate the ARPA (Advanced Research Projects Agency) ATIS (Air Travel Information Services) application is a black box methodology. The evaluation consists in comparing the response of the system to a pair of minimal and maximal reference responses. The ATIS evaluation has allowed the comparison of results in the natural language processing community.

The DCR paradigm for the natural language systems evaluation proposed by [1] is also a sort of black box test that allows contextual understanding evaluation. A DCR test consists of three items: the declaration sentence D, the control sentence C (a reformulation of D) and a hand-labeled boolean reference value R. The DCR evaluation is limited to a yes or no answer whitout any error diagnosis. The choice of the tests set is then preponderant and difficult: every test must be dedicated to the assessment of a unique linguistic phenomenon and the test set needs to be as large as possible to cover all the linguistic aspects with positive and negative references.

A glass box methodology is a test in which the internal system representation can be inspected. This approach allows a real diagnosis of the capabilities and limits of the modules tested which is very important during the development phase of a specific system. Nowadays, evaluating the internal semantic representation is becoming more and more important. By example, Walker & all [10] propose to integrate in the Darpa-Communicator evaluation project concept accuracy metrics. For a comparison between different systems, the definition of a common semantic representation (the reference) is necessary, along with a tool for converting the internal representation of each system to the reference.

Therefore we propose a framework for glass box diagnosis of not only literal but also contextual understanding. Each literal understanding evaluation unit contains both an exact and an ASR transcription of a user utterance along with its semantic representation. Each contextual unit is composed of an observed dialog history context representation, the user utterance and the resulting frame.

3. PROPOSED FRAMEWORK

In this section we first describe the semantic representation format and then the framework itself which includes a literal understanding test set, a contextual test set, and a scoring tool.

3.1. Semantic Representation (AVR)

The difficulty of choosing a semantic representation lies in the finding a complete and simple representation of a user utterance meaning in a unified format. This problem is the same as finding an equivalence relationship in the user utterance meanings.

The choice of the semantic representation is the first step in the framework conception and the faisability of the evaluation process depends greatly on this representation.

We have then chosen a frame attribute/value representation (**AVR**) [5]. An example is given in Table 1. The values are either numeric units, proper names, or semantic classes that group together lexical units which are synonyms for the task. The order of the attribute/value pairs in the semantic representation corresponds directly to their order in the utterance. A modal information (positive (+) and negative (-)) is assigned to each attribute/value pair. The set of attributes may be divided into different classes:

- As far as the task corresponds to data retrieval from a database, the attributes of the representation mostly corresponds to the attributes of the tables in the database (we call them **database attributes**). (Example attribute `category` for `hotel` objects, with a number value).
- Each database attribute in the AVR is associated with a **modifier attribute** which modifies the database attribute meaning. (Example attribute `category-modifier`, with possible values `more-than`, `less-than`, `minimum`, `maximum`, `around`).
- The **discursive attributes** correspond to man-machine dialogue utterances (Example: attribute `dialogue-command` with a value list containing `annulation`, `correction`, `error specification`).
- The focused topic of the utterance is represented by the special AVR attribute **argument**. It's value is the name of the attribute focused in the sentence. The `argument` attribute is only used when no restriction is put on the focused attribute in the utterance. Example the sentence *What are the fares ?* results in `argument: fare`.
- Task ambiguities may be preserved if a value can't be desambiguated without dialog context. Example: if there is no special marker in the user utterance, `Champs-Élysée` may be interpreted as a metro station or as a street name.

This representation obviously depends on the task domain. At LIMSI we have already used a similar semantic representation for the ARISE task that is a train timetable information task. Common attributes have been re-used (hours or fares constraints for example), for PARIS-SITI and the structure of the representation for the ARISE task was well

User Query	*bonjour je veux trouver un hôtel à côté de la des Galeries-Lafayette s'il vous plaît* hello, I'm looking for an hotel near from uh from Galeries-Lafayette please
AVR	+/dialogue: introduction +/argument: hotel. +/relative-modifier: near. +/relative-name: galeries-lafayette. +/dialogue: politeness.

Table 1: Example of a LU unit.

User Query	*combien d'étoiles ont ces hôtels* which category are those hotels
AVR	+/argument: category. +/same: hotel.

Table 2: Example of a LU unit with anaphoric reference.

suited to the PARIS-SITI task. The representation must be as simple as possible so as to be corrected by a non-expert in dialog systems.

3.2. Literal understanding test set

The literal understanding (LU) test set is composed of 1681 utterances extracted from 200 dialogues recorded by 22 subjects during a preceeding experiment [4]. Semantic labeling of reference dialogues might be a very costly procedure, so a semi-automatic procedure was used to prepare the reference labeled corpora in three steps. First the exact transcription of the LU test set is given to the PARIS-SITI understanding component which carries out a case-frame analyzis to produce a semantic frame representation. Second this representation is converted by a translation tool into the AVR format, and finally the AVR representation is hand-corrected. Examples of LU units are given in Table 1, 2 and 3. The total number of AVR attributes of the literal test set is 3991, so the mean number of AVR attibutes per sentence is 2.4, with a 7.7 mean number of words per utterance. In order to observe the ability of the systems to deal with recognition errors each LU unit also contains the ASR transcription of the user utterance which occured during the dialogue from which the user utterance was taken.

3.3. Contextual understanding test set

We have observed in a previous experiment that user satisfaction is strongly correlated to contextual understanding performances and that literal understanding performance is

User Query	*je ne cherche pas un hôtel de luxe mais un hôtel pas cher* I am not looking for a luxurious hotel but for a cheap one
AVR	-/argument: hotel -/description: luxurious +/fare-modifier: cheap

Table 3: Example of a LU unit with negative information.

Context paraphrase	*je voudrais un hôtel 4 étoiles dans le neuvième* **I would like a 4 category hotel in the ninth**

	`+/argument: hotel`
	`+/district: 9`
	`+/category: 4`

User Query	*la même catégorie dans un autre arrondissement* **the same categorie in another district**
	`+/other: district` `+/same: category`
AVR	**+/argument: hotel** **+/argument: district** **+/category: 4**

Table 4: Example of a contextual understanding unit composed of a Context paraphrase, a User query and the resulting AVR. AVR of Context paraphase and User Query are also given in `typewriting mode`. Ellipsis (*"in the ninth"*) and anaphora (*"same category"*, *"another district"*) may be observed.

in some cases a less reliable predictor [3]. Actually, contextual understanding (CU) evaluation is more informative than literal evaluation: it informs about the capability of the system to take into account the dialog history context in order to properly interpret the user query.

Therefore one important aspect of the proposed evaluation framework is to include CU evaluation. Obviously, contextual understanding is hard to perform because it is dependent on the dialog strategy of the system, nevertheless we propose a simple methodology which is able to evaluate local contextual interpretation problems.

The dialog contexts have been extracted from actual dialogs (the same that were used for literal test set) in three steps. First, the internal semantic frames representing the dialog context are automatically extracted from the log files of the recording sessions. Secondly, the dialog history semantic frames are converted into the AVR format and then hand-corrected. Then, the last step consists in writing a sentence (the **Context paraphrase**) which results into the same AVR representation as the dialog context. An example of a contextual unit is given in Table 4. So, each contextual unit is composed of the context paraphrase, the user utterance and the resulting AVR. For this experiment, we have chosen a 100 CU test set focused on the main linguistic difficulties such as anaphoric references, negation and ellipsis.

3.4. Scoring Tool

A dedicated scoring tool was developed to allow AVR comparison. This program is able to perform a comparison between a reference corpus and a tested corpus. Both corpora must have the form of a list of fixed lengh records. In our experiment we used 3 fields records: the first field consists in the mode annotation (-/+), the second field consists in

the attribute name and the third field is the attribue value. The alignment consists in applying a set of predefined operators assigned with a cost value. The alignment process looks for operators list to be applied to the test frame to obtain the reference frame that minimizes the final cost value. One advantage of the scoring tool is that it is very easy to define new operators. It is then possible to use the classical operators from speech evaluation (including DELetion, INSertion, and SUBStitution), but also to define especially adapted operators in order to distinguish between different types of errors.

4. PRELIMINARY RESULTS

Results are given in Table 5. The understanding accuracies are computed as the ratio between the sum of the number of deleted, inserted and substituted attributes, and the total number of AVR attributes in the test set. The literal understanding accuracy on exact transcription is 93.5%. The framework allows to analyze results on specific aspects (for example, the identification accuracy of the user query's topic is 94.3% and 95.7% for the `modifers` attributes). Due to a 26.5% ASR error rate, the LU accuracy goes down to 72% after ASR transcription. The contextual understanding accuracy on the 100 test units is 86.8%. The anaphoric references are well solved, with 84.4% accuracy on the 50 concerned units. The anaphoric referenced object is generally well identified (the errors are often due to other incorrect constraint management). Same remarks are available for the ellipsis resolution (accuracy 85.3%). The lowest score is encountered in case of constraints annulation (79.0%). This case corresponds to some major difficulties. For example when the user asks *"and the others on Champs-Élysées avenue"* in the paraphrased context *"I would like to eat seafood"*.

5. DISCUSSION

We have proposed and described a framework for both literal and contextual understanding evaluation. Based on an AVR semantic representation of the task domain, this framework allows an automatic evaluation of the understanding component. As far as the scoring tool points out the errors, this framework is well suited to analyze the understanding modules performances. This framework allows to easily diagnose the capabilities of local contextual interpretation in case of anaphoric, negative phenomenons and when adding and relaxing constraints. The "flat" structure of the AVR may have some limitations in case of long-time dialog dependencies because it does not memorize all the steps of the dialog. For example, if the speaker says first "*I would like a 2 stars hotel*", then "*no I prefer 3 stars*" and finally says "*give me again my first choice*", the CU unit can't take into account this succession of queries. However, this kind of interaction is rarely observed in the dialogue corpora: the user usually repeats the constraint value ("*give me again a 2 stars hotel*").

	#units	#attributes	accuracy
LU exact transc.	1681	3991	93.5%
LU ASR transc.	1681	3991	72.0%
Topic ident.	680	833	94.3%
Modifiers ident.	323	445	95.7%
CU exact transc.	100	430	86.8%
Anaphoric resolution	50	245	84.4%
Ellipsis resolution	25	106	85.3%
Constraint annulation	11	38	79.0%

Table 5: Literal Understanding attribute accuracy on both exact and ASR transcription, and Contextual Understanding attribute accuracy. Second column indicates the number of units included in the test set (i.e. # of user utterances), third culumn gives the total number of attributes in the correct AVR test sets. Details are given for Topic (`argument`) and `modifiers` attributes identification for LU on exact transcription, and for anaphoric reference and ellipsis resolution, and constraint annulation for CU.

Even if of course we believe that global and users evaluation are necessary for dialog systems comparison [3, 9, 10], we are convinced that such a framework can be an interesting complementary tool to better analyze the differences between the capabilities of the compared systems. We are at the moment working on this evaluation framework in the context of ARC B2 of AUF to compare different understanding approches.

6. ACKNOWLEDGMENTS

Thanks are due to S. Rosset and A. Fotopoulos for their work on the semantic representation and to Y. Kercadio for the development of the scoring tool.

7. REFERENCES

1. J.Y. Antoine & all, "Obtaining predictive results with an objective evaluation of spoken dialogue systems: experiments with the DCR assessment paradigm", LREC Athens, May 2000.
2. S. Bennacef, L. Devillers, S. Rosset, L. Lamel, "Dialog in the RAILTEL telephone-based system", ICSLP-96 and ISSD-96, October 1996.
3. H. Bonneau-Maynard, L. Devillers, S. Rosset, "Predictive performance of dialog systems", LREC 2000, Athenes, May 2000.
4. L. Devillers, H. Bonneau-Maynard "Evaluation of dialog strategies for a tourist information retrieval system", ICSLP 98, Sydney, October 1998.
5. King M. & all, "Evaluation of Natural Language Processing Systems - EAGLES Final Report, EAG-WEG-PR.2, ISBN-87-90708-00-8, October 1996, pp 1-271.
6. L. Lamel, P. Paroubek, W. Minker, "DISC, SLDS Best Practice", Natural Language Engineering, Special Issue on Best Practice in Spoken Language Dialogue Systems Engineering, in print.
7. MADCOW, "Multi-Site Data Collection for A Spoken Language Corpus", Proc. DARPA Speech and Natural Language Workshop, February 1992, pp. 7-14.
8. J. Mariani "The Aupelf-Uref Evaluation-Based Language Engineering Action and Related Projects", LREC, May 1998.
9. M. Walker & all, "Paradise: a general framework for evaluating spoken dialog agents", ACL/EACL 1997.
10. M. Walker, L. Hirchman, J. Aberdeen "Evaluation for Darpa Communicator spoken dialogue systems", LREC 2000, Athens, May 2000.