

# LANGUAGE PROCESSING FOR SPOKEN DIALOGUE SYSTEMS: IS SHALLOW PARSING ENOUGH?

*Ian Lewin<sup>†</sup> Ralph Becket<sup>†</sup> Johan Boye<sup>‡</sup> David Carter<sup>†</sup> Manny Rayner<sup>†</sup> and Mats Wirén<sup>‡</sup>*

<sup>†</sup> SRI International,  
23 Millers Yard,  
Cambridge CB2 1RQ  
United Kingdom

<sup>‡</sup> Telia Research AB,  
Vitsandsgatan 9,  
S-12386 FARSTA,  
Sweden

## ABSTRACT

With maturing speech technology, spoken dialogue systems are increasingly moving from research prototypes to fielded systems. The fielded systems however generally employ much simpler linguistic and dialogue processing strategies than the research prototypes. We describe an implemented spoken-language dialogue system for a travel planning domain which supports a mixed initiative dialogue strategy. The system accesses a commercially available travel information web-server. The system architecture combines both shallow and deep linguistic processors, partly so that a robust if shallow analysis is always available to the dialogue manager, and partly so that we can begin to examine where significant gains can be made by employing more advanced linguistic processing. We present the results of a preliminary investigation using data from a Wizard of Oz experiment. The results lend limited support to our original hypothesis that deep linguistic processing will prove useful at points where the user takes the initiative in driving the dialogue forward.

## 1. INTRODUCTION

With maturing speech technology, spoken dialogue systems are increasingly moving from research prototypes to fielded systems. The fielded systems however generally employ much simpler linguistic and dialogue processing strategies than the research prototypes (for a range of example systems, see, amongst others, [2], [1], [10], [11] and [3]). For example, in the fielded systems, domain-specific keyword/phrase spotting and slot-filling techniques are preferred for utterance interpretation. At the dialogue level, these systems tend to keep the dialogue initiative to themselves by treating the user simply as an answer-supplier. Particular systems may also implement particular instances of more

sophisticated processing. However, the simple methods do dovetail simply because the more expectations that a system can impose on a dialogue, then the more those expectations can be used to aid interpretation of user utterances. Currently, there is little work which attempts to examine at what points deep linguistic processing might prove significantly useful in the sorts of spoken language dialogue system that are currently being fielded.

SRI International and Telia Research AB are developing a Swedish language spoken dialogue system for accessing a web-based travel database. The system is being built by adaptation of existing general-purpose speech recognition and language understanding components including the Nuance toolkit ([13]) and the Core Language Engine (CLE) with a domain independent Swedish grammar ([4]). The Swedish version of the CLE was originally built with a machine translation application in mind ([14]). The system also includes a dialogue manager whose role is to progress the dialogue as a whole, deciding on the best interpretation of user utterances and deciding what it should do and say next. We have also added a parallel, faster but very simple linguistic processing path. This ensures the existence of a fallback “robust” analysis. It also provides our dialogue manager with interesting strategic choices concerning the two input paths. Finally, it enables us to begin evaluating wherein lie the advantages in deep linguistic processing and when shallow analysis may be reliably used. We do not here consider system development issues. For example, one of the objectives in our current project was to examine how easily our Swedish grammar, designed to be domain independent, could be adapted to the new application. We present the results of some preliminary investigations into the relative contributions of shallow and deep analysis in our travel scenario. The results lend limited support to our original hypothesis that

deep linguistic processing will prove useful at points where the user takes the initiative in driving the dialogue forward.

## 2. SYSTEM OVERVIEW

The application domain for our system is booking a business trip within Sweden, on the basis of information about travel mode (train or plane), destinations, times and fares. The travel information itself was based on the Travellink<sup>TM</sup> system, which can be accessed at <http://www.travellink.se>.<sup>1</sup> A Wizard of Oz experiment has provided a corpus of 131 dialogues from 47 subjects (31 male and 16 female). The Wizard's conversational style was purposely chosen so as to permit mixed-initiative user strategies. Analysis of the data showed that it displayed significant variation. For example, with respect to verboseness, there is a range of behaviour stretching from consistent use of short, telegraphic-style utterances to very long, disfluent utterances. Furthermore, there are both inactive users who refrain completely from taking the initiative (in effect leaving it open to the system to cross-examine them) and active users who quickly take the initiative by means of counter-questions, keeping it more or less throughout the dialogue. There is also a range of users whose behaviours fall between these extremes. One of our immediate conclusions was that if mixed-initiative dialogues were supported, then a large proportion of the people interacting with the system would make use of this capability.

The architecture of the system is shown in Figure 1. The modules communicate asynchronously by message passing; hence, in principle all of them could run in parallel in different processes. In the current implementation, there are four processes, which handle speech recognition, speech synthesis, database access and everything else, respectively.

The speech recognizer is a Swedish-language instantiation of the Nuance toolkit (deriving from the SRI Decipher system [12]), developed by SRI International and Telia Research. It sends an N-best speech hypothesis list to the two language processors: the Core Language Engine (deep analysis) and the Robust Parser (shallow analysis), further described in Section 3. The language processors each send their analyses to the dialogue manager (DM). After each system turn, the DM updates the language processors with limited information about the state of the discourse: the most recent question, if any, posed by the system, and the types of objects that are salient at the current point in the dialogue. At the moment, the dialogue manager waits for analyses from both linguistic processors before deciding on its future course of action. In practice, this means the dialogue manager, when waiting, is waiting for results from the CLE since deep linguis-

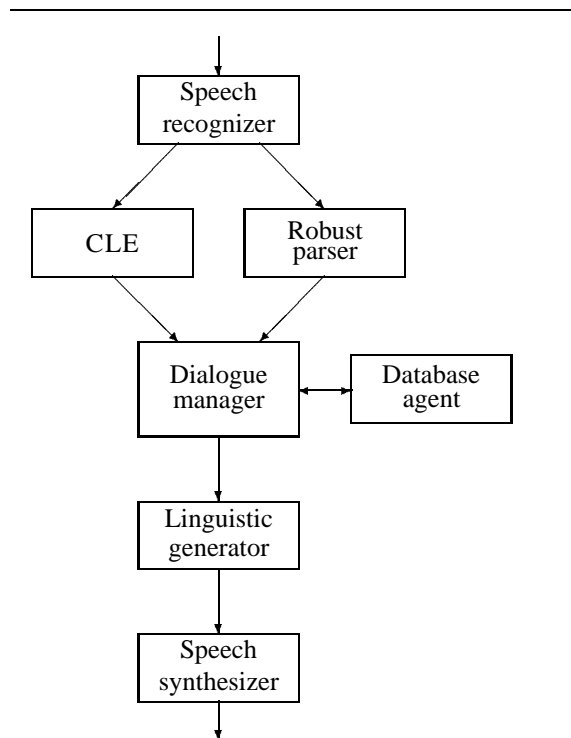


Figure 1: Architecture of the system.

tic processing takes considerably more time. In the future, we would like to tune the behaviour of the dialogue manager so that it can act quickly on just the results of shallow analysis, if those results are deemed both reliable and informative enough to merit progressing the dialogue. (This also requires those results to be suitable for updating the language processors suitably). The experiments described in Section 4 represent our initial analysis of the circumstances in which such a choice might be made.

The DM uses a two-stage heuristic selection process to advance the dialogue. First, each input analysis is categorized as a move of a certain type, and an appropriate response to that move is selected. References are resolved and contextual information is also added resulting in a further multiplication of possible moves and responses. Secondly, the relative utility of the various responses is judged, and the most productive response move is chosen. In the initial version of the system, the dialogue manager contained a strong preference to accept an analysis from the deep linguistic processor, if one existed, on the grounds that this was more likely to be reliable and that shallow analysis was primarily intended for “back-up” processing.

The generator produces the surface string representing the actual utterance, using a simple template-based approach. The surface string is then turned into speech by Telia Research's synthesizer LIPHON.

The database agent contains a web client in order to retrieve data from the Travellink database.

<sup>1</sup>We thank SMART for help in making the Travellink<sup>TM</sup> system available to us.

The system described here is fully implemented and has been permanently installed at the Telia Vision Center in Farsta/Stockholm since November 1998.

### 3. DEEP AND SHALLOW PROCESSING

The CLE is a wide-coverage state-of-the-art NL processing system with facilities for reasoning and understanding in context. The CLE incorporates a domain-independent Swedish unification grammar which maps input strings and word-lattices into Quasi-Logical Form (QLF). QLF is a function/argument representation that is intended to be useful for a wide variety of language-processing tasks, for example as a transfer level in spoken language translation ([14]). QLF representations leave some objects (e.g. pronouns, quantifier scope ambiguities) deliberately unresolved or underspecified. In the database application, the linguistically motivated QLF representations are translated into domain-dependent Flat Utterance Descriptions (Fuds), these being slot-filler expressions of the form *frame-name slot-name value* (meaning: the slot called *slot-name* is filled by *value* in the frame called *frame-name*) which are contained within wrappers giving limited expression to the following quantificational notions.

yn Are there objects with property *P*?

wh Find *X* with property *P*

wh\_agg Find the maximal/minimal *X* with property *P*

yn\_agg Does the maximal/minimal *X* with property *P* also have property *P'*?

Fuds may also contain *constraints* and *references*. Constraints express numerical relations obtaining between slot-fillers and other values. References associate filler values with the linguistic information which encoded them.

For instance, the utterance “I want to arrive in Stockholm before 6 pm” is interpreted as “Find flights arriving Stockholm before 6 pm”, and is represented by the following FUD:<sup>2</sup>

```
wh(X, [ slot(trip, trip_id, X),
        slot(trip, trip_mode, plane),
        slot(trip, to_city, stockholm)
        slot(trip, arr_time, T)
        exec(before(T, 1800))])
```

The utterance “When does that flight leave?” is represented by:

```
wh(Y [ slot(trip, trip_mode, plane),
        slot(trip, trip_id, X),
        slot(trip, dep_time, T),
        ref(X, det(def, sing))])
```

<sup>2</sup>The notation used here is simplified; for example, in our implementation each filler value is typed.

where the *ref* expression represents the referential expression (“that”) in the utterance, and signals to the dialogue manager that a reference resolution has to be made.

The robust parser uses a keyword/phrase spotter to generate a list of items including filled-slots and utterance type hypotheses (e.g. *wh*-question, *yn*-question) which are then also converted into well-formed FUDs. The parser knows only the last system utterance made in order to help guide it in filling the right slot given a matching keyphrase.

### 4. EXPERIMENTAL RESULTS

How properly to evaluate dialogue systems and dialogue managers as components within them is a notoriously difficult topic but has received increasing attention in the computational linguistics community ([16, 6, 9, 8, 17, 7]).

Generally, two sorts of assessment method are distinguished. The assessment of inputs and outputs without any attempt to “look inside” and judge the system’s internal representations is called Black Box assessment. Possible metrics for this sort of analysis include, amongst others, task-completion rate, task success rate, average length of a dialogue (in terms of utterances, turns or time) and user satisfaction. The evaluation of individual components in a complete system is called Glass Box assessment. For example, word-error rates may be used to evaluate speech recognition sub-systems. In the most sophisticated recent development, the “Paradise” framework ([17]), multivariate linear regression is used to estimate a quantitative function describing how the value of one variable (user satisfaction, usually measured through users filling in a questionnaire) can be predicted from other variables such as task completion rates, word-error rates, elapsed time and so forth. In this way, one hopes to be able assess the relative contribution of different components; for example, whether word-error rates are more significant to performance than task success rates.

In this section, we report some preliminary results aimed at evaluating the relative effectiveness of our two different linguistic processing paths and the particular circumstances in which one or the other appears to have an advantage. Our evaluations are restricted in that empirical data is currently restricted to the results of the original Wizard of Oz experiment. However, the results are useful both in guiding us in system development and in guiding us in formulating better the questions to ask when the final system is evaluated in live conditions.

For our experiments, we extracted a subset of 43 dialogues (one third of the total) from our original Wizard of Oz corpus and used them for testing two instantiations of our architecture. The first instantiation contained both deep and shallow processing paths. The second contained only the shallow processing path. In

each case, tests were carried out in text mode on the transcriptions of user utterances. In the live system, speech input is used of course and the deep linguistic processor is able to select, on linguistic grounds, an n-best hypothesis that the recognizer did not itself prefer most ([15]). The shallow processor simply works on the top hypothesis from the recognizer. Advantages that accrue from this facility of the deep processor were not part of the current test.

The dialogues were minimally edited to remove those parts of the Wizard of Oz scenario which have not been implemented. In practice this only required removing the tails of dialogues covering such matters as hotel and taxi bookings and one sub-dialogue type which did not affect the flow of the dialogue. (The Wizard generally chose to ask whether flights or trains were required before database lookup; but this sub-dialogue was not included in the implemented system. We hope that future Wizard of Oz experiments will promote further user initiative taking through the omission of this sub-dialogue.) In cases where the Wizard himself requested a repetition (this was the only correction strategy employed by the wizard) only the subsequent reformulation that the Wizard accepted was used. Although the dialogues were edited by removal of some utterances, no utterance was edited through removal of words. Four dialogues were completely removed because the Wizard himself had failed to follow the original script sufficiently accurately and the structures of the resulting dialogues were too dissimilar from that of the implemented system to permit their use as a test.

To test the systems, the user-utterances from the dialogues were fed into the two versions of the system and the points at which divergences arose from the original scripts were noted. At these points, it was determined which, if any, of the two systems had taken the better path. The most salient reason for the better path was noted.

In most cases (approximately 66% of the time), the two systems took identical paths. In these cases, the robust parser and the deep linguistic processor extracted similar information from the input material. The paths were not necessarily *good* paths. In some cases, the dialogue manager simply made the same decision about what to do what next given a certain input, but this decision did not match the Wizard's decision and hence a divergence from the original script arose.

In those cases where there was a difference, the shallow processor actually led to the better path more often than the deep linguistic processor did (25% and 9% respectively). The overwhelming reason for this success was the success of shallow processing in extracting information from particularly long or multiple utterances. Since the shallow processor is only looking for very local pieces of information and ignoring any global structure, it is not affected by the length of the input string. In contrast, the deep linguistic pro-

cessor does attempt to find a structure incorporating the whole input string. If it cannot (perhaps the input *is* ungrammatical or perhaps the grammar is inadequate), then some strategy on fragmentary parsing is required. The policy at the time of the test was simply selecting the longest parsable substring. However, this policy leads to parsable information in other parts of the string simply being overlooked.

There was no clear pattern to the cases where the deep linguistic processor led to the better path. However, this is not least because it is more difficult in principle to analyse a failure by a shallow analyser. The problem is delimiting those cases that a shallow analyser *ought* to be able to handle from those that it is unreasonable to expect it to handle. Is the shallow analyser merely lacking a rule that would generally be successful or is there actually no such rule available? Answering these questions is a significant topic in its own right. For example, in one of our dialogues, the parameters (destination, origin, date, time) of the journey are established and the system announces that there is a suitable flight together with its departure and arrival times. The user then asks whether there is an earlier *train*. The shallow analyser failed to understand this utterance but it is far from clear whether this is a failure in principle of shallow analysis. On the one hand, there may be a possible rule that will successfully cover this and similar cases without interfering too much in other examples. Alternatively, it may be that, even though there is a possible rule, it is not one it is reasonable to expect to discover in advance. The very rarity of the example may precisely illustrate the value of a compositional grammar based approach.

Our original hypothesis had been that shallow analysis would in fact be more suitable for earlier, system directed portions of the dialogue than latter parts of the dialogue which involved more negotiation. This expectation was borne out to some extent in that, in those cases where the deep processor took the better path it was mostly at the negotiative stage. (Interestingly, the other cases occurred in answers to the opening question "How can I help you", which are also conversationally freer points in the dialogue). However, it is not clear how much this is due to increased user initiative and an increase in linguistic sophistication. For instance, after the system has suggested a particular journey, then the user is handed the initiative in progressing the dialogue. He may accept the suggestion or start a negotiation. However, negotiative moves, at least in the scenario under examination, appear to be themselves reasonably predictable. Nearly all are requests for either earlier or later flights and the simple occurrence of particular words ("senare" *later* and "tidigare" *earlier*) proved excellent predictors of these moves. Similarly, some "advanced linguistic features", e.g. discourse reference, also generally arise only at the negotiative stage ("Hmm när går flyget?" "*Hmmm when does that flight leave?*"). But

these cases too may often be predictable on shallow grounds given the current corpus. We will shortly be carrying out a further iteration of data collection based on use of the implemented system rather than the Wizard of Oz experiment, which may have circumscribed the sorts of user follow-up queries too tightly.

Our hypothesis that shallow analysis would prove useful as a “back-up” mode for use when deeper analysis failed has also not been strongly supported. In fact, we had simply underestimated the degree to which deep linguistic processing would itself produce only partial results which might require careful selection. We are currently putting considerable effort into developing fragmentary analysis selection in the deep linguistic processor so that decisions can be made statistically from the results of supervised training over already parsed corpora. The technique is integrated into our general tool for customizing the disambiguation component of a language processor [5].

## 5. CONCLUSIONS

The architecture of a spoken dialogue system combining both shallow and deep linguistic processors has been described. The architecture is designed to enable us to have a fallback strategy in case deep linguistic processing fails (or, possibly, takes too long) and also to begin evaluating at which points sophisticated analysis does significantly improve such a dialogue system. We have presented the results of a preliminary investigation using data from a Wizard of Oz experiment. The results lend limited support to our original hypothesis that deep linguistic processing will prove useful at a particular point in our scenario where the dialogue changes from a situation where the system primarily holds the initiative to one where the user takes it.

## 6. REFERENCES

- [1] J. Allen, B.W. Miller, E.K. Ringger, and T. Sikorski. A robust system for natural spoken dialogue. In *Proceedings of 34th ACL Santa Cruz*, pages 62–70, 1996.
- [2] H. Aust, M. Oerder, F. Siede, and V. Steinbiss. A spoken language enquiry system for automatic train timetable information. *Philips Journal of Research*, 49(4):399–418, 1995.
- [3] P. Bohlin, J. Bos, S. Larsson, I. Lewin, C. Matheson, and Milward D. Survey of existing interactive systems. Technical Report Deliverable D1.3, Trindi: Task Oriented Instructional Dialogue (European Telematics Applications Programme project LE4-8314), 1999.
- [4] D. Carter, J. Kaja, L. Neumeyer, Rayner M., F. Weng, and Wirén M. Handling compound nouns in a swedish speech-understanding system. In *Proceedings of ICSLP-96*, 1996. see also: SRI Technical Report CRC-062 available from <http://www.cam.sri.com>.
- [5] D.M. Carter. The treebanker: a tool for supervised training of parsed corpora. In *ACL Workshop on Computational environments for Grammar Development and Linguistic Engineering (Madrid 1997)*, 1997. see also: SRI Technical Report CRC-068 available from <http://www.cam.sri.com>.
- [6] M. Danieli and E. Gerbino. Metrics for evaluating dialogue strategies in a spoken language system. In *Proceedings of the 1995 AAAI Spring Symposium on Empirical Methods in Discourse Interpretation*, pages 34–39, 1995.
- [7] W. Eckert, E. Levin, and R. Pieraccini. Automatic evaluation of spoken dialogue systems. In J. Hulstijn and A. Nijholt, editors, *TWLT 13: Formal Semantics and Pragmatics of Dialogue*, Twente Workshop on Language Technology, pages 99–110. Enschede, Universiteit Twente, Faculteit Informatica, 1998.
- [8] D. Gibbon, R. Moore, and Winski R. *Handbook of Standards and Resources for Spoken Language Systems*. Mouton de Gruyter, Berlin & New York, 1997.
- [9] L. Hirschman and Thompson H. Overview of evaluation in speech and natural language processing. In R. Cole, editor, *Survey of the State of the Art in Human Language Technology*. Cambridge University Press, 1996.
- [10] L. Lamel, S. Rosset, J.L. Gauvin, S. Bennacef, Garnier-Rizet M., and B. Prouts. The lims ARISE system. In *Proceedings of IEEE 4th Workshop on Interactive Voice Technology for Telecommunications Applications, Torino, Italy*, pages 209–214, 1998.
- [11] D.J. Litman, S. Pan, and M.A. Walker. Evaluating response strategies in a web-based spoken dialogue agent. In *Proceedings of ACL-COLING 98: 36th Annual Meeting of the Association of Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 780–787, 1998.
- [12] H. Murveit, J. Butzberger, V. Digilakis, and M. Weintraub. Large vocabulary dictation using sri’s decipher(tm) speech recognition system: Progressive search techniques. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP II*, pages 319–322, 1993.

- [13] Nuance-Communications. Nuance speech recognition system, version 6, developer's manual. Technical report, Nuance Communications, Menlo Park, California, 1997. <http://www.nuance.com>.
- [14] M. Rayner and D.M. Carter. Hybrid language processing in the spoken language translator. In *Proceedings of ICASSP-97, Munich*, 1997. see also: SRI Technical Report CRC-064 available from <http://www.cam.sri.com>.
- [15] M. Rayner, D.M. Carter, V. Digilakis, and P. Price. Combining knowledge sources to re-order n-best speech hypothesis lists. In *Proceedings of the ARPA workshop on Human Language Technology, Princeton, New Jersey*, 1994. see also: SRI Technical Report CRC-044 available from <http://www.cam.sri.com>.
- [16] A. Simpson and N. Fraser. Black box and glass box evaluation of the sundial system. In *Proceedings of the 3rd European Conference on Speech Communication and Technology*, pages 1423–1426, 1993.
- [17] M. Walker, D. Litman, C. Kamm, and A. Abella. Paradise: A general framework for evaluating spoken dialogue agents. In *Joint Proceedings of 35th ACL and 8th European ACL*, pages 271–280, 1997.