

CLASSiC

D2.4: Two trained semantic decoders for the Appointment Scheduling task

James Henderson, François Mairesse, Lonneke van der Plas, Paola Merlo

Distribution: Public

CLASSiC

Computational Learning in Adaptive Systems for Spoken Conversation
216594 Deliverable 2.4

Jan 2011



Project funded by the European Community
under the Seventh Framework Programme for
Research and Technological Development



CogSys
Cognitive Systems



The deliverable identification sheet is to be found on the reverse of this page.

Project ref. no.	216594
Project acronym	CLASSiC
Project full title	Computational Learning in Adaptive Systems for Spoken Conversation
Instrument	STREP
Thematic Priority	Cognitive Systems, Interaction, and Robotics
Start date / duration	01 March 2008 / 36 Months

Security	Public
Contractual date of delivery	M33 = Nov 2011
Actual date of delivery	Jan 2011
Deliverable number	2.4
Deliverable title	D2.4: Two trained semantic decoders for the Appointment Scheduling task
Type	Prototype
Status & version	Final 1.0
Number of pages	6 (excluding front matter)
Contributing WP	2
WP/Task responsible	UNIGE
Other contributors	
Author(s)	James Henderson, François Mairesse, Lonneke van der Plas, Paola Merlo
EC Project Officer	Philippe Gelin
Keywords	Semantic Decoder

The partners in CLASSiC are:

Heriot-Watt University	HWU
University of Cambridge	UCAM
University of Geneva	GENE
Ecole Supérieure d'Electricité	SUPELEC
France Telecom/ Orange Labs	FT
University of Edinburgh HCRC	EDIN

For copies of reports, updates on project activities and other CLASSiC-related information, contact:

The CLASSiC Project Co-ordinator:
Dr. Oliver Lemon
School of Mathematical and Computer Sciences (MACS)
Heriot-Watt University
Edinburgh
EH14 4AS
United Kingdom
O.Lemon@hw.ac.uk
Phone +44 (131) 451 3782 - Fax +44 (0)131 451 3327

Copies of reports and other material can also be accessed via the project's administration homepage,
<http://www.classic-project.org>

No part of this document may be reproduced or transmitted in any form, or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission from the copyright owner.

Contents

- Executive Summary 1
- 1 Overview 2
- 2 University of Cambridge SLU prototype 2
- 3 Data Analysis 3
- 4 Software 4
- 5 Conclusions 4

Executive summary

This document describes the Prototype deliverable 2.4, due at month 33 of the CLASSIC project. It reports the results of Task 2.4. Task 2.4 was intended to result in two prototype Spoken Language Understanding (SLU) modules trained on data for the FT Appointment Scheduling domain. This task required spoken French data in this domain, annotated according to the agreed dialogue act scheme (D6.1.3). While some data was collected, the complexity of language produced by users of the preliminary FT system was extremely limited, making it unsuitable for training SLU models that are intended to handle the full range of the target dialogue act scheme. This made training models on real user utterances an impractical alternative.

Two contingency plans were considered. The first was to use the available data, augmented with introspective data from native French speakers, to develop an SLU module by hand. This was done, using the UCAM Phoenix parser. The Phoenix parser is a toolkit for hand-coding SLU models, based on a context-free grammar written for the domain. A grammar was developed for the French Appointment Scheduling domain and the required dialogue act scheme, and used in the academic Appointment Scheduling system, System 2 (D5.4). The second contingency plan we considered involved using the grammar developed for the Phoenix parser to generate pseudo-data, and training another SLU model on this data. Analysis of this grammar indicated that it was not suitable for this purpose.

This report describes in more detail these issues and results. It focusses on a description of the prototype French Appointment Scheduling SLU module and an analysis of the available Appointment Scheduling SLU data.

1 Overview

This deliverable reports the results of Task 2.4. Task 2.4 was intended to result in two prototype Spoken Language Understanding (SLU) modules trained on data for the FT Appointment Scheduling domain. This task required spoken French data in this domain, annotated according to the agreed dialogue act scheme (as discussed in D6.1.2 and D6.1.3). While some data was collected, the complexity of language produced by users of the deployed FT systems was extremely limited, making it unsuitable for training SLU models that are intended to handle anything like the full range of the target dialogue act scheme. This made training models on real user utterances an impractical alternative.

Two contingency plans were considered. The first was to use the available data, augmented with introspective data from native French speakers, to develop an SLU module by hand. This was done, using the UCAM Phoenix parser [1]. The Phoenix parser is a toolkit for hand-coding SLU models, based on a context-free grammar written for the domain. A grammar was developed for the French Appointment Scheduling domain and the dialogue act scheme required by the dialogue manager (D5.4), and used in the academic Appointment Scheduling system (D5.4, and to be tested in D6.4).

The second contingency plan that was considered was to exploit this hand-written grammar to generate artificial data, which could then be used to train an SLU module. The hope was that the resulting statistical model would be more robust than the hand-written grammar, and could take advantage of the syntactic-semantic parser for French created in T2.3, and improved as part of T2.5. However, the Phoenix parser grammar was not written with this use in mind, in that it would vastly over-generate. Since the grammar is only designed for spotting dialogue acts, it does not enforce basic linguistic constraints like agreement and allows words to be ignored. When used as a generator, such a grammar would result in utterances that have little to do with what people might actually say, and general-purpose linguistic analysis like syntactic-semantic parsing would be ineffective. These considerations, plus the limited advantages and the substantial amount of work which such an alternative would entail, persuaded us not to pursue this contingency plan.

In the remainder of this report, we describe the prototype SLU module (also discussed in D5.4), and present an analysis of the collected Appointment Scheduling data.

2 University of Cambridge SLU prototype

The UCAM SLU module uses a handcrafted grammar for the Phoenix parser. The Phoenix parser uses a context-free grammar that specifies meaningful sub-phrases of the utterance. The non-terminals of the grammar represent various dialogue act types, item slots, and item values, as well as intermediary symbols. The parser searches the space of sub-parses over sub-phrases of the utterance, and returns the set of sub-parses resulting in the largest word coverage. By allowing sub-parses that only cover partial text fragments, the parser is more robust to noisy input utterances. The grammar is illustrated by the following grammar rules, which are used to identify ranges of days ('?' indicates optional words):

```

DAYRANGE -> FROMDAYOFWEEK TODAYOFWEEK
FROMDAYOFWEEK -> FROM DAY AT_TIME?
TODAYOFWEEK -> TO DAY AT_TIME?
FROM -> ( de ce? | du | a partir de ce? | a partir du )
TO -> ( jusque? au | jusque ce? | jusqu'a ce? )
AT_TIME -> ( a? | vers ) TIME
DAY -> ...
TIME -> ...

```

The selected set of sub-parses are then converted into a list of user dialogue acts in the appropriate format, which forms the semantic interpretation output by the Phoenix parser module. Some non-terminals in the grammar's parses map directly to components of the dialogue act scheme, as for example the `FROMDAYOFWEEK` and `TODAYOFWEEK` non-terminals above, which refer to the slots `fromday` and `today` in the dialogue act scheme.

Each input ASR hypothesis is decoded individually, computing a single semantic interpretation for each one. Identical semantic hypotheses are then merged by summing the inference evidence confidence scores of the corresponding ASR hypotheses. These inference evidence scores are interpreted as log probabilities of their hypotheses. Lets consider the following input ASR N-best list and their inference evidence scores:

```

<asrhyp score=-0.163676023> jeudi quinze heures      </asrhyp>
<asrhyp score=-2.312538147> jeudi quinze          </asrhyp>
<asrhyp score=-3.650744438> jeudi quinze heure     </asrhyp>
<asrhyp score=-5.011305809> jeudi quinze juin     </asrhyp>
<asrhyp score=-5.252517223> jeudi quinze heures     </asrhyp>
<asrhyp score=-5.345973969> jeudi quinze a         </asrhyp>
<asrhyp score=-6.132476330> jeudi jeudi quinze heures </asrhyp>
<asrhyp score=-6.539419174> le quinze heures       </asrhyp>
<asrhyp score=-6.769999504> jeudi et quinze heures  </asrhyp>
<asrhyp score=-7.219739437> jeudi en quinze heures  </asrhyp>

```

Each hypothesis in the list is decoded, and the results are merged, to produce the following distribution over semantic hypotheses. This distribution captures the uncertainty due to speech recognition errors, as well as the lack of contextual information available to the SLU module:

```

0.50: inform(dayofweek=Thursday,time="15")
0.30: inform(dayofweek=Thursday,dayofmonth="15")
0.10: inform(dayofweek=Thursday,dayofmonth="15",month=June)
0.09: inform(time="15")
0.01: inform(dayofweek=Thursday,totime="15")

```

This prototype SLU model forms part of the academic proof of concept system, System 2, described in D5.4.

3 Data Analysis

Two corpora of data were provided by FT, one collected with System 3 and one collected with System 3.5, which was an extension of System 3. This data became available soon after the writing of D6.1.3.

The System 3 data included 12,214 transcribed user utterances, hand annotated with dialogue acts according to the agreed dialogue act scheme (D6.1.3). Of these 12,214 utterances, 11,380 were assigned dialogue

acts. This is a good amount of data, but the diversity and complexity of this data was extremely limited. 10,865 of these utterances were simply annotated with *affirm()* (e.g. “oui”) or *negate()* (e.g. “non”), leaving 515 more complex utterances. 392 utterances are individual inform acts, mostly consisting of simple noun phrases that specify days or times. Only 35 utterances (0.3%) contain predicative verbs (i.e. not “être”, “avoir”, or “faire”). Almost all of these are requests for something to be repeated. So analysing these utterances with a semantic role labeller would provide almost no useful information. Of the total set of 18 dialogue act types, only 8 occur in the corpus.

System 3.5 was designed with slightly more open prompts, in the hope that users would produce more varied utterances. An initial dataset of 170 user utterances was collected, SLU outputs logged, and transcribed, but not hand annotated. The result was a little more varied, with 147 utterances tagged other than *affirm()* or *negate()*. But these are almost all dates or times, with only 3 of these utterances (2%) including predicative verbs. Again, this collection setup did not produce sufficiently varied utterances to be used for training a general-purpose Appointment Scheduling SLU module.

We believe that this lack of diversity in the collected data was something we could not anticipate during the planning of the project. Because these systems were publicly deployed, their design was highly constrained by departments of FT that are not involved in the CLASSiC project. This was the main reason the systems were not designed to illicit more complex user utterances, where the system would be more likely to have difficulty. Also, the original grant proposal included the more complex Self Help domain instead of the Appointment Scheduling domain, which was changed due to the same constraints.

4 Software

The UCAM French Appointment Scheduling SLU prototype forms part of the academic proof of concept system, System 2, described in D5.4. It can be run as part of prototype D5.4.

The software for this UCAM SLU prototype is in the *SemIO* directory. The code for the Phoenix decoder is in the directory *SemIO/Phoenix*. The mapping from the output of the decoder to the dialogue act is done in *SemIO/SemIO/SemIPhoenix.cpp*.

The *resources/PhxGrammar/AS_French* directory includes files that specify the Phoenix grammar for the French Appointment Scheduling domain. The file *AS_French.gra* is the grammar. The syntax of the grammar is quite simple: ‘[]’ specified a global node (e.g. any phrase can refer to it), ‘*’ precedes optional words, and words in caps are only matched with local patterns defined within a given node. The mapping from parses in this grammar to dialogue acts is specified in the file *frames*, which list the slots that gets labelled in the output dialogue acts.

5 Conclusions

This task has achieved its main objective, which was to provide an SLU module for use in the academic proof of concept system for the French Appointment Scheduling domain, System2. A prototype SLU module was completed on schedule and integrated in System 2. Unfortunately this prototype did not fulfil the other objectives of Task 2.4, in that it was not trained on data, and there was not a version that exploited syntactic-semantic parsing. This change was necessitated by the limited quantity and lack of diversity in the available data for the domain, which we did not envisage during the planning of this project. An alternative contingency plan based on generating pseudo-data was considered, but it was determined that

only the contingency plan based on a hand-coded grammar for the Phoenix parser was a viable option. To date, no publications have resulted from this task.

In future work, we plan to analyse the user utterances currently being collected as part of the evaluations of the Appointment Scheduling systems (D6.4). If they are appropriate, we will be able to use them in training and/or evaluating SLU systems, including the UCAM Phoenix parser discussed above.

Bibliography

- [1] Wayne H. Ward. The Phoenix system: Understanding spontaneous speech. In *Proceedings of ICASSP*, 1991.