ISSUES IN THE DEVELOPMENT OF A STOCHASTIC
SPEECH UNDERSTANDING SYSTEM

F. Lefèvre and H. Bonneau-Maynard

Spoken Language Processing Group
LIMSI-CNRS
91403 Orsay cedex, FRANCE

ABSTRACT
In the development of a speech understanding system, the recourse
to stochastic techniques can greatly reduce the need for human
expertise. A known disadvantage is that stochastic models re-
quire large annotated training corpora in order to reliably esti-
mate model parameters. Manual semantic annotation of such cor-
pora is tedious, expensive, and subject to inconsistencies. In or-
der to decrease the development cost, this work investigates the
performance of stochastic understanding models with two param-
eters: the use of automatically segmented data and the use of auto-
matically learned lexical normalisation rules.

1. INTRODUCTION
In stochastic understanding [1, 2, 3], models are automatically
trained on a large amount of data annotated with concepts (semi-
tic units). Despite the advantages offered by this approach over
rule-based approaches, for instance less extensive human expert-
tise is required, the emergence of stochastic understanding is hin-
dered by implementation difficulties. For this reason our work is
an attempt to define a precise and complete protocol for a low cost
development and evaluation of a stochastic understanding com-
ponent. In a previous study [4], we have shown that comparable
performance can be obtained from rule-based and stochastic ap-
proaches. In this paper, we are focusing on the development of
training methods to limit the need for manual intervention on both
the segmentation of the training corpus and the set up of the lexical
normalisation rules.

The overall objective of this study is to draw up task-independent
guidelines allowing a low-cost development of a stochastic under-
standing module to be used in any spoken dialog system for in-
formation retrieval. The experiments have been carried out on the
ARISE task. The LIMSI ARISE system [5] allows users to ob-
tain travel information from the French national railway’s static
timetables by telephone. The system also provides information
about services offered on the trains, reductions, fares, and fare-
related restrictions. ARISE is a good representative of a spoken
information retrieval task: it deals with a consequent number of
topics and due to the mixed-initiative strategy the user’s query are
uttered in a spontaneous way (including interruptions, disfluencies
and syntactically incorrect structures).

The paper is organized as follows. The next section gives an
overview of best-practice guidelines in the development of a task-
dependent semantic representation. In Section 3 the principles of
the stochastic understanding approach are presented. Then, after

the corpus description in Section 4, the experiments are reported
in Section 5.

2. SETTING-UP A TASK-DEPENDENT
SEMANTIC REPRESENTATION

The development of an understanding module requires first the de-
finite meaning of a semantic representation for the application domain.
The feasibility of the annotation process depends greatly on the
semantic representation. The semantic representation of an utter-
ance consists in a list of triplets of the form [mode, concept,
normalized value], where mode can be positive or negative. The
chosen concept/value frame representation (CVR) lies on a con-
cept dictionary which contains the list of concepts and for each
concept the list of its admissible values. The values are either
numeric units, proper names or semantic classes merging lexical
units which are synonyms for the task. A modal information (af-
firmative or negative) is assigned to each concept/value pair. An
example of a CVR is given in the last row of Table 1. This exam-
ple illustrates the use of the negative mode: due to the word pas
(not), the word Croisic is represented in the CVR with the place
concept assigned with a negative mode [-/place Croisic]. The
order of the concept/value pairs in the semantic representation fol-

ows their order in the user’s utterance.

The concept dictionary is task-dependent. When dealing with
information retrieval applications, defining the concepts that are
meaningful for the task and the appropriate values can be done in
a rather straightforward way. Most of the concepts are derived
directly from the information stored in the database, and most of
those required by the dialog manager are domain-independent. Ba-
sically, the concepts can be divided into four broad classes [6]:

- **Database concepts** (the most frequent) correspond to the
  attributes of the database tables, commonly referred to as the
  *constraints* (e.g. the concept place whose values are
  the city names stored in the database). In relation with
  the database concepts, other concepts are added which are
  mainly used to generalize the attribute values (e.g. range
  with values morning, evening, night... or date-relative
  with values today, yesterday...). The determination of these
  latter concepts can only be based on human expertise.

- **Modifier concepts** are associated to the database concepts.
  Their values are used by the dialog manager to modify the
  database concept value interpretation (e.g. the concept
  modifier-time with possible values before, after, mini-
The aim of stochastic understanding is to find the sequence of concepts which maximizes the a posteriori probability, rewritten according to the Bayes formula:

\[
\hat{C} = \text{arg max}_C \text{Pr}(C|W) = \text{arg max}_C \text{Pr}(W|C) \text{Pr}(C)
\]

The term \(\text{Pr}(W|C)\) is estimated by means of \(n\)-gram probabilities of words given the concept associated to word \(i\):

\[
\text{Pr}(W|C) \approx \prod_{i=1}^{N} \text{Pr}(w_i|w_{i-1}, \ldots, w_{i-n}, c_i)
\]

and \(\text{Pr}(C)\) is estimated in terms of \(m\)-gram probabilities of concepts:

\[
\text{Pr}(C) \approx \prod_{i=1}^{N} \text{Pr}(c_i|c_{i-1}, \ldots, c_{i-m})
\]

Based on this formulation, two approaches are considered depending on the orders of the models used to produce the estimates of \(\text{Pr}(W|C)\) and \(\text{Pr}(C)\). Our baseline understanding model is limited to \(n = 0\), giving word unigrams conditioned on a concept \(\text{Pr}(w_i|c_i)\), and \(m = 1\), giving concept bigrams \(\text{Pr}(c_i|c_{i-1})\). A second model configuration with \(n = 1\) giving word bigrams conditioned on a concept \(\text{Pr}(w_i|w_{i-1}, c_i)\) is also investigated.

A set of lexical classes, derived from the database attribute values, can be used in the \(n\)-gram estimation. In the case of the ARISE task, a class \text{station} can be used, but also more general classes like days, months, numbers, etc. Only the class \text{station} is used in the system as preliminary experiments have shown that it is the only class to have a real impact on the performance.

A block diagram of the stochastic speech understanding procedure is shown in Figure 1. An example of the representation used at different steps in the decoding process is given in Table 1. The speech recognizer transforms the acoustic signal into the most probable sequence of words (second row in Table 1). No prior transduction, such as lexical parsing, is performed. The conceptual decoding is then carried out to decode the sentence into a sequence of concepts (third row in Table 1). Then the concept value normalization module is used to transduce each word sequence assigned to a given concept into its normalized form, according to the CVR concept value list given in the concept dictionary and a set of transformation rules. In the example, the normalization module modifies the sequence \text{dans la matinée} (\text{in the morning}) assigned to the concept \text{range-dep} to the normalized form \text{matin} (fourth row in Table 1). The resulting CVR proposed by the whole understanding process for the example is given in the last row of Table 1.

### 3. STOCHASTIC UNDERSTANDING

The aim of stochastic understanding is to find the sequence of concepts \(C = c_1, c_2, \ldots, c_N\) that will represent the meaning of the sentence, assuming that there is a sequential correspondence between the concept and word sequences [1].

Given \(W = w_1, w_2, \ldots, w_N\) the sequence of words in the sentence, the understanding process consists of finding the sequence of concepts which maximizes the a posteriori probability, rewritten according to the Bayes formula:

\[
\hat{C} = \text{arg max}_C \text{Pr}(C|W) = \text{arg max}_C \text{Pr}(W|C) \text{Pr}(C)
\]

### 4. CORPUS DESCRIPTION

The training set used in our experiments contains 14,582 sentences. These utterances have been extracted from the LIMSI ARISE corpus, which has over 10k dialogs of users interacting with the system. This corpus has been semi-manually annotated in terms of concepts. There is a total of 53 concepts for the ARISE domain.
### Table 2. Corpus description: number of utterances, words and CVR concepts in the training, development, and test sets. Word error rate of the recognized utterances is given for test sets.

<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Utterances</td>
<td>14582</td>
<td>496</td>
</tr>
<tr>
<td>#Words</td>
<td>72380</td>
<td>2880</td>
</tr>
<tr>
<td>#Concepts (in CVR)</td>
<td>44812</td>
<td>923</td>
</tr>
<tr>
<td>Word Error Rate</td>
<td>-</td>
<td>14.3%</td>
</tr>
</tbody>
</table>

(106 with modality), among them 39 are database or database-derived concepts. Only 9 modifier concepts are used as in practice not every concept can be modified. Along with the argument concept, a set of 4 discursive concepts are used. The average number of words per utterance is 5. The total number of concept occurrences in the training corpus is 44,812, giving an average number of 3 CVR concepts per utterance.

The evaluation is performed on a test set of 496 utterances randomly selected from the remaining portion of the ARISE corpus. An iterative approach has been used to derive the reference CVR of the test set [6]. A scoring tool is used to compare two semantic representations in terms of deletions, insertions, and substitutions. The scoring is done on the whole triplet including mode, concept name and concept value. Table 2 summarizes the characteristics of the training and test sets.

### 5. EXPERIMENTS

Three sets of experiments are reported. First an automatic segmentation procedure is used allowing the use, during the decoding process, of the information brought by the keyword context. Then the use of automatically derived normalization rules is evaluated in comparison with hand-written rules. Finally, an attempt is made to use a bigram model for the word level probability estimation.

#### Automatic vs. manual concept segmentation

Adjacent words of a concept keyword provide useful information to disambiguate between concepts. In *from Paris to Marseille*, for instance, the preposition *from* implies that *Paris* should be associated to *place-from* and *to* implies that *Marseille* should be associated to *place-to*. In previous experiments [4], concept markers were used to tag adjacent words. The marker set has been established *a priori* and was based on a human expertise. In these conditions, the manual annotation was based on a complete segmentation of the training utterances in terms of concepts and concept markers. To facilitate the annotation procedure, we propose to limit the manual segmentation to the dictionary concepts (first two columns of Figure 2). The segmentation of the adjacent words is derived automatically in a subsequent stage.

The technique used in our experiments consists in associating a set of pre- and post-markers to each concept. These concept markers are distinguished by their distance to the concept. A pre-marker (resp. post-marker) of order b (resp. a) is attributed to every word appearing b (resp. a) words before (resp. after) a concept keyword. The concept modality is propagated from the concept to its markers. An example of automatic annotations in terms of markers is given in Figure 2 for various orders.

Table 3 gives the results of the experiments performed on the automatic transcription of the test set, along with the total number of tags (concepts and markers) used by the understanding model. Without marker, the error rate is 26.9%. A 20% relative gain is obtained with a pre-marker of order 1. Increasing the pre-markers to higher orders doesn’t lead to further improvement. A gain is observed with the introduction of post-markers of order 1. The best understanding error rate is finally obtained with a combination of pre- and post-markers of order 2 and 1. These experiments show that a manual annotation limited to the basic concepts combined with an automatic segmentation in concept markers results in only 0.7% loss in performance compared with a model trained on a fully manual annotation (the best error rate is 19.8% to be compared with 19.1%). This small performance gap is maintained when only 2k training utterances are used (22.3 vs. 21.5% with the manual concept markers). At the same time, the development cost of the system is reduced by a diminution of nearly 30% of the number of tags considered during the manual annotation.

#### Automatic vs. manual lexical normalization

As shown in Figure 1, a normalization procedure is applied after the decoding step to transduce the sequence of words associated to one concept into the corresponding normalized baseform expected in the concept dictionary. In our first experiments, the normalization process was based on hand-written rules. We propose to use an automatic method to derive the normalization rules from the training corpus.

---

**Fig. 2.** Automatic concept markers. The pre-markers have an order of b words and the post-markers of a words. From left to right, the marker orders are increased (newly introduced markers in bold).

<table>
<thead>
<tr>
<th>Manual Annotation</th>
<th>Automatic Markers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><em>je souhaiterais</em> (I would like)</td>
<td></td>
</tr>
<tr>
<td><em>aller</em> (to go)</td>
<td></td>
</tr>
<tr>
<td><em>de</em> (from)</td>
<td></td>
</tr>
<tr>
<td><em>Grenoble</em></td>
<td>+/-place_from</td>
</tr>
<tr>
<td><em>à</em> (to)</td>
<td></td>
</tr>
<tr>
<td><em>Paris</em></td>
<td>+/-place_to</td>
</tr>
<tr>
<td><em>première</em> (first)</td>
<td>+/-classe</td>
</tr>
<tr>
<td><em>classe</em> (class)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The technique used in our experiments consists in associating a set of pre- and post-markers to each concept. These concept markers are distinguished by their distance to the concept. A pre-marker (resp. post-marker) of order b (resp. a) is attributed to every word appearing b (resp. a) words before (resp. after) a concept keyword. The concept modality is propagated from the concept to its markers. An example of automatic annotations in terms of markers is given in Figure 2 for various orders.
A rewriting rule is derived from each observation of a concept in the training corpus. In the example of Figure 2, where the manual annotation of the sentence is shown in the second and third columns, the rewriting rule \((\text{first} \rightarrow 1)\) can be derived for the class concept. Concept-dependent sets of rules are then obtained. To improve generalization, automatically derived rules are shared between modalities (+/-) of the same concept and also for concepts referring to the same database attribute (e.g. for range, range-departure and range-arrival).

Comparative results between automatic and manual normalization are given in the second and third rows of Table 4. Both methods have the same level of performance, only slight differences of 0.3% and 0.6% for the manual and automatic transcriptions are observed. The same experiment using only 2k training utterances has been carried out and has led to the same conclusion.

**Bigram models**

The current manual annotation of the training corpus is keyword-based (i.e. a concept is only associated to the word which gives its value). In this condition, the probability \(\Pr(W | C)\) is approximated by an unigram. The automatic segmentation and normalization procedures introduced above allow us to investigate the use of a bigram model (i.e. word bigrams conditioned on concepts). In this case, instead of considering concept markers, the association of the concept to a keyword is propagated to its adjacent words. Then, for each concept, the automatic normalization scheme allows to derive suitable rules for the new sequences of words associated to the concept.

The performance of the bigram model are shown in the last row of Table 4. In accordance with the best results obtained with an unigram, the keyword context is limited to the 2 previous words and the word following. Despite a 11% relative improvement observed with the bigram model on the manual transcriptions of the test set, the unigram and bigram models obtain comparable performance on the automatic transcriptions.

**Table 3.** Understanding error rate (%) and total number of tags (concepts and markers) used in the models, for various pre and post marker orders \(b - a\) obtained on the automatic transcription of the test set. First column gives results without markers, last column gives the results with manual markers.

| Seg., Norm., \(\Pr(W | C)\) | Manual | Auto |
|-----------------------------|--------|------|
| Man, Man, 1g                | 9.4    | 19.1 |
| Auto, Man, 1g               | 11.2   | 19.8 |
| Auto, Auto, 1g              | 10.9   | 19.2 |
| Auto, Auto, 2g              | 9.7    | 19.1 |

**Table 4.** Understanding error rates (%) on the manual and automatic transcriptions of the test set for various configurations of the understanding model: automatic of manual training segmentation, manual or automatic normalization and \(\Pr(W | C)\) estimated by 1 or 2-grams.

In this paper, issues in the development of a low cost and efficient stochastic speech understanding system have been investigated. First a procedure allowing the automatic concept segmentation of the training utterances has been studied as an alternative to the use of concept markers during the manual annotation of the training corpora. This technique leads to a nearly 30% reduction of the number of tags considered during the manual annotation for a relative performance loss of only 3%. Afterwards a procedure deriving automatically the rewriting rules used for the lexical normalization has been introduced with no loss in performance (compared to the use of \textit{a priori} hand-written rules).

Using these two techniques, an extension of our baseline model to a second order model has been tested but didn’t succeed in improving the system performance on automatic transcriptions of the user’s utterances. A plausible reason could be a too crude concept segmentation of the training utterances due to the automatic procedure. An iterative training scheme of the models will be tested using successive refinement of the utterance segmentations, as proposed in [7].

The methods introduced in this study allow to reduce the development cost of the understanding system and as a consequence increase its flexibility and portability. The validation of this approach on new domains and languages is under progress in the context of the AMITIES project.

**6. CONCLUSIONS**

**7. REFERENCES**


