Spoken Dialogue Interpretation with the DOP Model

Rens Bod
Department of Computational Linguistics
University of Amsterdam
Spuistraat 134, 1012 VB Amsterdam
rens.bod@let.uva.nl

Abstract

We show how the DOP model can be used for fast and robust processing of spoken input in a practical spoken dialogue system called OVIS. OVIS, Openbaar Vervoer Informatie Systeem ("Public Transport Information System"), is a Dutch spoken language information system which operates over ordinary telephone lines. The prototype system is the immediate goal of the NWO[1] Priority Programme "Language and Speech Technology". In this paper, we extend the original DOP model to context-sensitive interpretation of spoken input. The system we describe uses the OVIS corpus (10,000 trees enriched with compositional semantics) to compute from an input word-graph the best utterance together with its meaning. Dialogue context is taken into account by dividing up the OVIS corpus into context-dependent subcorpora. Each system question triggers a subcorpus by which the user answer is analyzed and interpreted. Our experiments indicate that the context-sensitive DOP model obtains better accuracy than the original model, allowing for fast and robust processing of spoken input.

1. Introduction

The Data-Oriented Parsing (DOP) model (cf. Bod 1992, 1995; Bod & Kaplan 1998; Scha 1992; Sima'an 1995, 1997; Rajman 1995) is a probabilistic parsing model which does not single out a narrowly predefined set of structures as the statistically significant ones. It accomplishes this by maintaining a large corpus of analyses of previously occurring utterances. New utterances are analyzed by combining subtrees from the corpus. The occurrence-frequencies of the subtrees are used to estimate the most probable analysis of an utterance.

To date, DOP has mainly been applied to corpora of trees labeled with syntactic annotations. Let us illustrate this with a very simple example. Suppose that a corpus consists of only two trees:

(1)
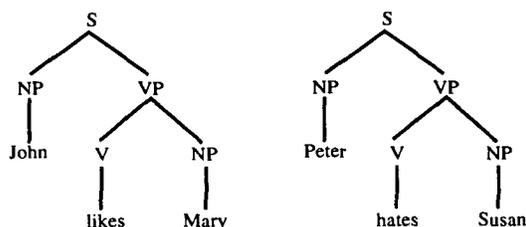
To combine subtrees, a node-substitution operation indicated as ∘ is used. Node-substitution identifies the leftmost nonterminal frontier node of one tree with the root node of a second tree (i.e., the second tree is *substituted* on the leftmost nonterminal frontier node of the first tree). A new input sentence such as *Mary likes Susan* can thus be parsed by combining subtrees from this corpus, as in (2):

(2)
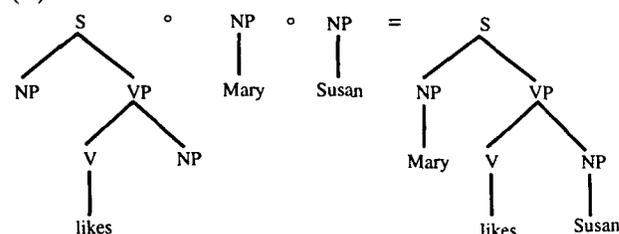
Other derivations may yield the same parse tree; for instance:

(3)
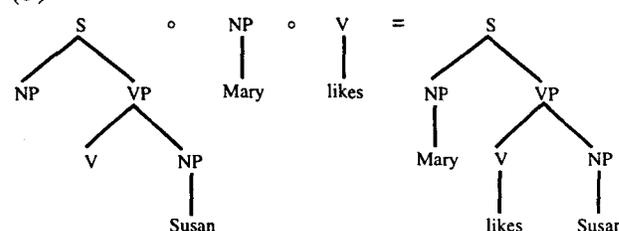
DOP computes the probability of substituting a subtree t on a specific node as the probability of selecting t among all subtrees in the corpus that could be substituted on that node. This probability is equal to the number of occurrences of t, divided by the total number of occurrences of subtrees t' with the same root label as t. Let $rl(t)$ return the root label of t then: $P(t) = \#(t) / \sum_{t': rl(t')=rl(t)} \#(t')$. The probability of a derivation is computed by the product of the probabilities of the subtrees is consists of. The probability of a parse tree is computed by the sum of the probabilities of all derivations that produce that parse tree.

Bod (1992) demonstrated that DOP can be implemented using conventional context-free parsing techniques. However, the computation of the most probable parse of a sentence is NP-hard (Sima'an 1996). The most probable parse can be estimated by iterative Monte Carlo sampling (Bod 1995), but efficient algorithms exist only for sub-optimal solutions such as the most likely derivation of a sentence (Bod 1995, Sima'an 1995) or the "labelled

recall parse" of a sentence (Goodman 1996). So far, the syntactic DOP model has been tested on the ATIS corpus and the Wall Street Journal corpus, obtaining significantly better test results than other stochastic parsers (Charniak 1996). For example, Goodman (1998) compares the results of his DOP parser to a replication of Pereira & Schabes (1992) on the same training and test data. While the Pereira & Schabes method achieves 79.2% *zero-crossing brackets* accuracy, DOP obtains 86.1% on the same data (Goodman 1998: p. 179, table 4.4). Thus the DOP method outperforms the Pereira & Schabes method with an accuracy-increase of 6.9%, or an error-reduction of 33%. Goodman also performs a statistical analysis using *t*-test, showing that the differences are statistically significant beyond the 98th percentile.

In Bod et al. (1996), it was shown how DOP can be generalized to semantic interpretation by using corpora annotated with compositional semantics. In the current paper, we extend the DOP model to spoken dialogue understanding, and we show how it can be used as an efficient and robust NLP component in a practical spoken dialogue system called OVIS. OVIS, Openbaar Vervoer Informatie Systeem ("Public Transport Information System"), is a Dutch spoken language information system which operates over ordinary telephone lines. The prototype system is the immediate goal of the NWO Priority Programme "Language and Speech Technology".

The backbone of any DOP model is an annotated language corpus. In the following section, we therefore start with a description of the corpus that was developed for the OVIS system, the "OVIS corpus". We then show how this corpus can be used by DOP to compute the most likely meaning M of a word string W: $\mathrm{argmax}_M P(M, W)$. Next we demonstrate how the dialogue context C can be integrated so as to compute $\mathrm{argmax}_M P(M, W \mid C)$. Finally, we interface DOP with speech and show how the most likely meaning M of an acoustic utterance A given dialogue context C is computed: $\mathrm{argmax}_M P(M, A \mid C)$. The last section of this paper deals with the experimental evaluation of the model.

2. The OVIS corpus: trees enriched with compositional frame semantics

The OVIS corpus currently consists of 10,000 syntactically and semantically annotated user utterances that were collected on the basis of a pilot version of the OVIS system[2]. The user utterances are answers to system questions such as *From where to where do you want to travel?*, *At what time do you want to travel from Utrecht to Leiden?*, *Could you please repeat your destination?*.

For the syntactic annotation of the OVIS user utterances, a tag set of 40 lexical/syntactic categories

---

was developed. This tag set was deliberately kept small so as to improve the robustness of the DOP parser. A correlate of this robustness is that the parser will overgenerate, but as long as the probability model can accurately select the correct utterance-analysis from all possible analyses, this overgeneration is not problematic. Robustness is further achieved by a special category, called ERROR. This category is used for stutters, false starts, and repairs. No grammar is used to determine the correct syntactic annotation; there is a small set of guidelines, that has the degree of detail necessary to avoid an "anything goes" attitude in the annotator, but leaves room for the annotator's perception of the structure of the utterance (see Bonnema et al. 1997).

The semantic annotations are based on the *update language* defined for the OVIS dialogue manager by Veldhuijzen van Zanten (1996). This language consists of a hierarchical frame structure with slots and values for the origin and destination of a train connection, for the time at which the user wants to arrive or depart, etc. The distinction between slots and values can be regarded as a special case of ground and focus distinction (Vallduvi 1990). Updates specify the ground and focus of the user utterances. For example, the utterance *Ik wil niet vandaag maar morgen naar Almere* (literally: *"I want not today but tomorrow to Almere"*) yields the following update:

(4) user.wants.(([# today];[! tomorrow]);
          destination.place.town.almere)

An important property of this update language is that it allows encoding of speech-act information (v. Noord et al. 1997). The "#" in the update means that the information between the square brackets (representing the focus of the user-utterance) must be retracted, while the "!" denotes the corrected information.

This update language is used to semantically enrich the syntactic nodes of the OVIS trees by means of the following annotation convention:

• Every meaningful *lexical* node is annotated with a slot and/or value from the update language which represents the meaning of the lexical item.

• Every meaningful *non-lexical* node is annotated with a *formula schema* which indicates how its meaning representation can be put together out of the meaning representations assigned to its daughter nodes.

In the examples below, these schemata use the variable $d1$ to indicate the meaning of the leftmost daughter constituent, $d2$ to indicate the meaning of the second daughter node constituent, etc. For instance, the full (syntactic and semantic) annotation for the above sentence *Ik wil niet vandaag maar morgen naar Almere* is given in figure (5).

Note that the top-node meaning of (5) is compositionally built up out of the meanings of its sub-constituents. Substituting the meaning representations into the corresponding variables yields the

update expression (4). The OVIS annotations are in contrast with other corpora and systems (e.g. Miller et al. 1996), in that our annotation convention exploits the Principle of Compositionality of Meaning.[3]
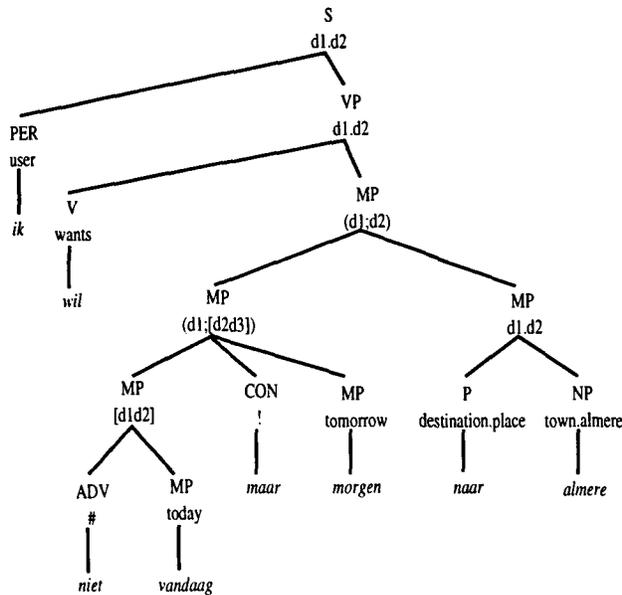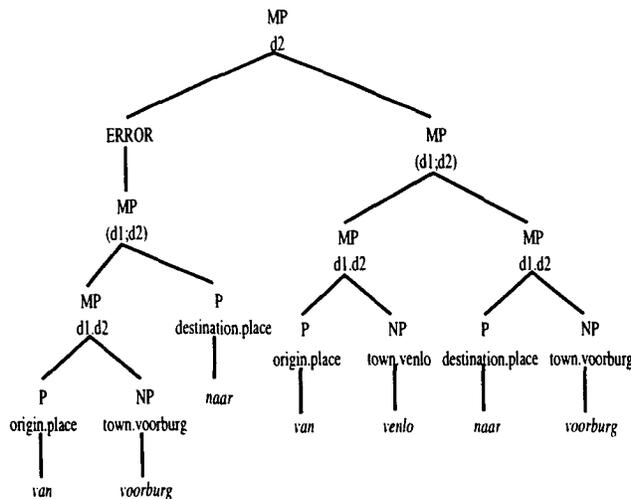
(5)

Figure (6) gives an example of the ERROR category for the annotation of the ill-formed sentence *Van Voorburg naar van Venlo naar Voorburg* ("*From Voorburg to from Venlo to Voorburg*"):

(6)

Note that the ERROR category has no semantic annotation; in the top-node semantics of *Van Voorburg*

---

naar van Venlo naar Voorburg, the meaning of the false start *Van Voorburg naar* is thus absent:
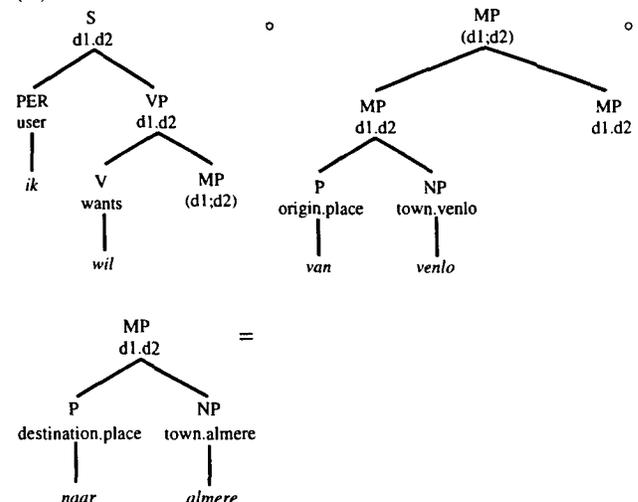
(7) (origin.place.town.venlo ;
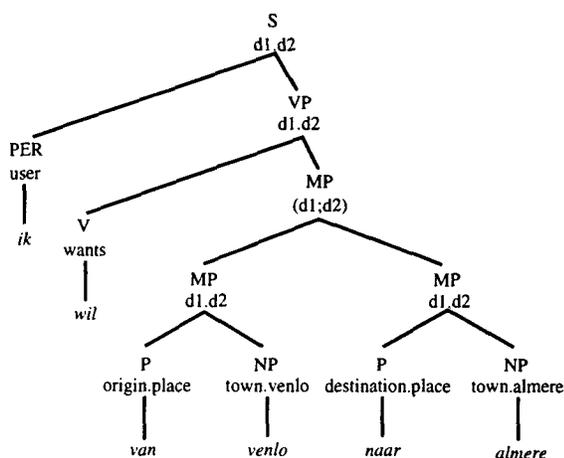   destination.place.town.voorburg)

The manual annotation of 10,000 OVIS utterances may seem a laborious and error-prone process. In order to expedite this task, a flexible and powerful annotation workbench (SEMTAGS) was developed by Bonnema (1996). SEMTAGS is a graphical interface, written in C using the XVIEW toolkit. It offers all functionality needed for examining, evaluating, and editing syntactic and semantic analyses. SEMTAGS is mainly used for correcting the output of the DOP parser. After the first 100 OVIS utterances were annotated and checked by hand, the parser used the subtrees of these annotations to produce analyses for the next 100 OVIS utterances. These new analyses were checked and corrected by the annotator using SEMTAGS, and were added to the total set of annotations. This new set of 200 analyses was then used by the DOP parser to predict the analyses for a next subset of OVIS utterances. In this incremental, bootstrapping way, 10,000 OVIS utterances were annotated in approximately 600 hours (supervision included). For further information on OVIS and how to obtain the corpus, see http://earth.let.uva.nl/~rens.

3. Using the OVIS corpus for data-oriented semantic analysis

An important advantage of a corpus annotated according to the Principle of Compositionality of Meaning is that the subtrees can directly be used by DOP for computing syntactic/semantic representations for new utterances. The only difference is that we now have composite labels which do not only contain syntactic but also semantic information. By way of illustration, we show how a representation for the input utterance *Ik wil van Venlo naar Almere* ("*I want from Venlo to Almere*") can be constructed out of subtrees from the trees in figures (5) and (6):

(8)

```
                    S
                  d1,d2
                      \
                       VP
                      d1,d2
   PER                    \
  user                     MP
    |                    (d1;d2)
   ik      V          /          \
        wants       MP            MP
          |        d1,d2         d1,d2
         wil      /    \        /    \
                 P      NP     P      NP
          origin.place town.venlo destination.place town.almere
                 |      |      |      |
                van    venlo  naar  almere
```

which yields the following top-node update semantics:

(9) user.wants.
        (origin.place.town.venlo;
        destination.place.town.almere)

The probability calculations for the semantic DOP model are similar to the original DOP model. That is, the probability of a subtree t is equal to the number of occurrences of t in the corpus divided by the number of occurrences of all subtrees t' that can be substituted on the same node as t. The probability of a derivation $D = t_1 \circ ... \circ t_n$ is the product of the probabilities of its subtrees t_i. The probability of a parse tree T is the sum of the probabilities of all derivations D that produce T. And the probability of a meaning M and a word string W is the sum of the probabilities of all parse trees T of W whose top-node meaning is logically equivalent to M (see Bod et al. 1996).

As with the most probable parse, the most probable meaning M of a word string W cannot be computed in deterministic polynomial time. Although the most probable meaning can be estimated by iterative Monte Carlo sampling (see Bod 1995), the computation of a sufficiently large number of random derivations is currently not efficient enough for a practical application. To date, only the *most likely derivation* can be computed in near to real-time (by a best-first Viterbi optimization algorithm). We therefore assume that most of the probability mass for each top-node meaning is focussed on a single derivation. Under this assumption, the most likely meaning of a string is the top-node meaning generated by the most likely derivation of that string (see also section 5).

4. Extending DOP to dialogue context: context-dependent subcorpora

We now extend the semantic DOP model to compute the most likely meaning of a sentence given the previous dialogue. In general, the probability of a top-node meaning M and a particular word string W_i given a dialogue-context $C_i = W_{i-1}, W_{i-2} ... W_1$ is given by $P(M, W_i \mid W_{i-1}, W_{i-2} ... W_1)$.

Since the OVIS user utterances are typically answers to previous system questions, we assume that the meaning of a word string W_i does not depend on the full dialogue context but only on the previous (system) question W_{i-1}. Under this assumption,

$$P(M, W_i \mid C_i) = P(M, W_i \mid W_{i-1})$$

For DOP, this formula means that the update semantics of a user utterance W_i is computed on the basis of the subcorpus which contains all OVIS utterances (with their annotations) that are answers to the system question W_{i-1}. This gives rise to the following interesting model for dialogue processing: each system question triggers a context-dependent domain (a subcorpus) by which the user answer is analyzed and interpreted. Since the number of different system questions is a small closed set (see Veldhuijzen van Zanten 1996), we can create off-line for each subcorpus the corresponding DOP parser.

In OVIS, the following context-dependent subcorpora can be distinguished:

(1) place subcorpus: utterances following questions like *From where to where do you want to travel? What is your destination?*, etc.

(2) date subcorpus: utterances following questions like *When do you want to travel?, When do you want to leave from X?, When do you want to arrive in Y?*, etc.

(3) time subcorpus: utterances following questions like *At what time do you want to travel? At what time do you want to leave from X?, At what time do you want to arrive in Y?*, etc.

(4) yes/no subcorpus: utterances following y/n-questions like *Did you say that ... ? Thus you want to arrive at ...?*

Note that a subcorpus can contain utterances whose topic goes beyond the previous system question. For example, if the system asks *From where to where do you want to travel?*, and the user answers with: *From Amsterdam to Groningen tomorrow morning*, then the date-expression *tomorrow morning* ends up in the place-subcorpus.

It is interesting to note that this context-sensitive DOP model can easily be generalized to domain-dependent interpretation: a corpus is clustered into subcorpora, where each subcorpus corresponds to a topic-dependent domain. A new utterance is interpreted by the domain in which it gets highest probability. Since small subcorpora tend to assign higher probabilities to utterances than large subcorpora (because relative frequencies of subtrees in small corpora tend to be higher), it follows that a language user strives for the smallest, most specific domain in which the perceived utterance can be analyzed, thus establishing a most specific common ground.

141

5. Interfacing DOP with speech

So far, we have dealt with the estimation of the probability $P(M, W \mid C)$ of a meaning M and a word string W given a dialogue context C. However, in *spoken* dialogue processing, the word string W is not given. The input for DOP in the OVIS system are *word-graphs* produced by the speech recognizer (these word-graphs are generated by our project partners from the University of Nijmegen).

A word-graph is a compact representation for all sequences of words that the speech recognizer hypothesizes for an acoustic utterance A (see e.g. figure 10). The nodes of the graph represent points in time, and a transition between two nodes i and j, represents a word w that may have been uttered between the corresponding points in time. For convenience we refer to transitions in the word-graph using the notation $<i, j, w>$. The word-graphs are optimized to eliminate *epsilon* transitions. Such transitions represent periods of time when the speech recognizer hypothesizes that no words are uttered. Each transition is associated with an acoustic score. This is the negative logarithm (of base 10) of the acoustic probability $P(a \mid w)$ for a hypothesized word w normalized by the length of w. Reconverting these acoustic scores into their corresponding probabilities, the acoustic probability $P(A \mid W)$ for a hypothesized word string W can be computed by the product of the probabilities associated to each transition in the corresponding word-graph path. Figure (10) shows an example of a simplified word-graph for the uttered sentence *Ik wil graag vanmorgen naar Leiden* ("I'd like to go this morning to Leiden"):

(10)
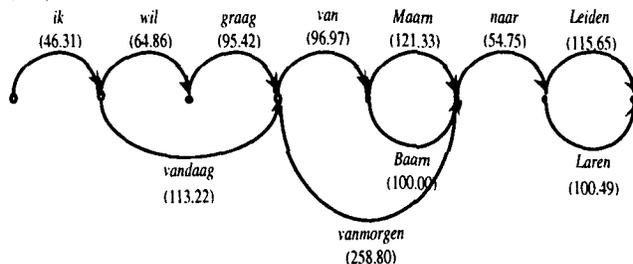
The probabilistic interface between DOP and speech word-graphs thus consists of the interface between the DOP probabilities $P(M, W \mid C)$ and the word-graph probabilities $P(A \mid W)$ so as to compute the probability $P(M, A \mid C)$ and $\text{argmax}_M P(M, A \mid C)$. We start by rewriting $P(M, A \mid C)$ as:

$$P(M, A \mid C) = \Sigma_W P(M, W, A \mid C)$$
$$= \Sigma_W P(M, W \mid C) \cdot P(A \mid M, W, C)$$

The probability $P(M, W \mid C)$ is computed by the dialogue-sensitive DOP model as explained in the previous section. To estimate the probability $P(A \mid M, W, C)$ on the basis of the information available in the word-graphs, we must make the following independence assumption: the acoustic utterance A depends only on the word string W, and

not on its context C and meaning M (cf. Bod & Scha 1994). Under this assumption:

$$P(M, A \mid C) = \Sigma_W P(M, W \mid C) \cdot P(A \mid W)$$

To make fast computation feasible, we furthermore assume that most of the probability mass for each meaning and acoustic utterance is focused on a single word string W (this will allow for efficient Viterbi best first search):

$$P(M, A \mid C) = P(M, W \mid C) \cdot P(A \mid W)$$

Thus, the probability of a meaning M for an acoustic utterance A given a context C is computed by the product of the DOP probability $P(M, W \mid C)$ and the word-graph probability $P(A \mid W)$.

As to the parsing of word-graphs, it is well-known that parsing algorithms for word strings can easily be generalized to word-graphs (e.g. van Noord 1995). For word strings, the initialization of the chart usually consists of entering each word w_i into chart entry $<i, i+1>$. For word-graphs, a transition $<i, j, w>$ corresponds to a word w between positions i and j where j is not necessarily equal to $i+1$ as is the case for word strings (see figure 10). It is thus easy to see that for word-graphs the initialization of the chart consists of entering each word w from transition $<i, j, w>$ into chart entry $<i, j>$. Next, parsing proceeds with the subtrees that are triggered by the dialogue context C (provided that all subtrees are converted into equivalent rewrite rules -- see Bod 1992, Sima'an 1995). The most likely derivation is computed by a bottom-up best-first CKY parser adapted to DOP (Sima'an 1995, 1997). This parser has a time complexity which is cubic in the number of word-graph nodes and linear in the grammar size. The top-node meaning of the tree resulting from the most likely derivation is taken as the best meaning M for an utterance A given context C.

6. Evaluation

In our experimental evaluation of DOP we were interested in the following questions:

(1) Is DOP fast enough for practical spoken dialogue understanding?
(2) Can we constrain the OVIS subtrees without loosing accuracy?
(3) What is the impact of dialogue context on the accuracy?

For all experiments, we used a random split of the 10,000 OVIS trees into a 90% training set and a 10% test set. The training set was divided up into the four subcorpora described in section 4, which served to create the corresponding DOP parsers. The 1000 word-graphs for the test set utterances were used as input. For each word-graph, the previous system question was known to determine the particular DOP parser, while the user utterances were kept apart. As to the complexity of the word-graphs: the average number of transitions per word is 4.2, and the average number of

142

words per word-graph path is 4.6. All experiments were run on an SGI *Indigo* with a MIPS R10000 processor and 640 Mbyte of core memory.

To establish the semantic accuracy of the system, the best meanings produced by the DOP parser were compared with the meanings in the test set. Besides an exact match metric, we also used a more fine-grained evaluation for the semantic accuracy. Following the proposals in Boros et al. (1996) and van Noord et al. (1997), we translated each *update* meaning into a set of *semantic units*, where a unit is triple <CommunicativeFunction, Slot, Value>. For instance, the next example

```
user.wants.travel.destination.
                ([# place.town.almere];
                 [! place.town.alkmaar])
```
translates as:

```
<denial, destination_town, almere>
<correction, destination_town, alkmaar>
```

Both the updates in the OVIS test set and the updates produced by the DOP parser were translated into semantic units of the form given above. The semantic accuracy was then evaluated in three different ways: (1) *match*, the percentage of updates which were exactly correct (i.e. which exactly matched the updates in the test set); (2) *precision*, the number of correct semantic units divided by the number of semantic units which were produced; (3) *recall*, the number of correct semantic units divided by the number of semantic units in the test set.

As to question (1), we already suspect that it is not efficient to use *all* OVIS subtrees. We therefore performed experiments with versions of DOP where the subtree collection is restricted to subtrees with a certain maximum depth. The following table shows for four different maximum depths (where the maximum number of frontier words is limited to 3), the number of subtree types in the training set, the semantic accuracy in terms of match, precision and recall (as percentages), and the average CPU time per word-graph in seconds.

subtree-depth	#subtrees	semantic accuracy			CPU time
		match	precision	recall	
1	3191	76.2	79.4	82.1	0.21
2	10545	78.5	83.0	84.3	0.86
3	32140	79.8	84.7	86.2	2.76
4	64486	80.6	85.8	86.9	6.03

Table 1: Experimental results on OVIS word-graphs

The experiments show that at subtree-depth 4 the highest accuracy is achieved, but that only for subtree-depths 1 and 2 are the processing times fast enough for practical applications. Thus there is a

trade-off between efficiency and accuracy: the efficiency deteriorates if the accuracy improves. We believe that a match of 78.5% and a corresponding precision and recall of resp. 83.0% and 84.3% (for the fast processing times at depth 2) is promising enough for further research. Moreover, by testing DOP directly on the word strings (without the word-graphs), a match of 97.8% was achieved. This shows that linguistic ambiguities do not play a significant role in this domain. The actual problem are the ambiguities in the word-graphs (i.e. the multiple paths).

Secondly, we are concerned with the question as to whether we can impose constraints on the subtrees other than their depth, in such a way that the accuracy does not deteriorate and perhaps even improves. To answer this question, we kept the maximal subtree-depth constant at 3, and employed the following constraints:

- *Eliminating once-occurring subtrees*: this led to a considerable decrease for all metrics; e.g. *match* decreased from 79.8% to 75.5%.

- *Restricting subtree lexicalization*: restricting the maximum number of words in the subtree frontiers to resp. 3, 2 and 1, showed a consistent decrease in semantic accuracy similar to the restriction of the subtree depth in table 1. The *match* dropped from 79.8% to 76.9% if each subtree was lexicalized with only one word.

- *Eliminating subtrees with only non-head words*: this led also to a decrease in accuracy; the most stringent metric decreased from 79.8% to 77.1%. Evidently, there can be important relations in OVIS that involve non-head words.

Finally, we are interested in the impact of dialogue context on semantic accuracy. To test this, we neglected the previous system questions and created one DOP parser for the whole training set. The semantic accuracy metric *match* dropped from 79.8% to 77.4% (for depth 3). Moreover, the CPU time per sentence deteriorated by a factor of 4 (which is mainly due to the fact that larger training sets yield slower DOP parsers).

The following result nicely illustrates how the dialogue context can contribute to better predictions for the correct meaning of an utterance. In parsing the word-graph corresponding to the acoustic utterance *Donderdag acht februari* ("*Thursday eight February*"), the DOP model without dialogue context assigned highest probability to a derivation yielding the word string *Dordrecht acht februari* and its meaning. The uttered word *Donderdag* was thus interpreted as the town *Dordrecht* which was indeed among the other hypothesized words in the word-graph. If the DOP model took into account the dialogue context, the previous system question *When do you want to leave?* was known and thus triggered the subtrees from the date-subcorpus only, which now correctly assigned the

143

highest probability to *Donderdag acht februari* and its meaning, rather than to *Dordrecht acht februari*.

7. Conclusions

We showed how the DOP model can be used for efficient and robust processing of spoken input in the OVIS spoken dialogue system. The system we described uses syntactically and semantically analyzed subtrees from the OVIS corpus to compute from an input word-graph the best utterance together with its meaning. We showed how dialogue context is integrated by dividing up the OVIS corpus into context-dependent subcorpora. Each system question triggers a subcorpus by which the user utterance is analyzed and interpreted.

Efficiency was achieved by computing the most probable derivation rather than the most probable parse, and by restricting the depth and lexicalization of the OVIS subtrees. Robustness was achieved by the shallow syntactic/semantic annotations, including the use of the productive ERROR label for repairs and false starts. The experimental evaluation showed that DOP's blending of lexical relations with syntactic-semantic structure yields promising results. The experiments also indicated that elimination of subtrees diminishes the semantic accuracy, even when intuitively unimportant subtrees with only non-head words are discarded. Neglecting dialogue context also diminished the accuracy.

As future research, we want to investigate further optimization techniques for DOP, including finite-state approximations. We want to enrich the OVIS utterances with discourse annotations, such as co-reference links, in order to cope with anaphora resolution. We will also extend the annotations with feature structures and/or functional structures associated with the surface structures so as to deal with more complex linguistic phenomena (see Bod & Kaplan 1998).

Acknowledgments

References

H. Aust, M. Oerder, F. Seide and V. Steinbiss. 1995. "The Philips automatic train timetable information system", *Speech Communication*, 17, pp 249-262.

M. van den Berg, R. Bod and R. Scha, 1994. "A Corpus-Based Approach to Semantic Interpretation", *Proceedings Ninth Amsterdam Colloquium*, Amsterdam, The Netherlands.

R. Bod, 1992. "A Computational Model of Language Performance: Data Oriented Parsing", *Proceedings COLING-92*, Nantes, France.

R. Bod, 1995. *Enriching Linguistics with Statistics: Performance Models of Natural Language*, ILLC Dissertation Series 1995-14, University of Amsterdam.

R. Bod and R. Scha, 1994. "Prediction and Disambiguation by means of Data-Oriented Parsing", *Proceedings Twente Workshop on Language Technology (TWLT8)*, Twente, The Netherlands.

R. Bod, R. Bonnema and R. Scha, 1996. "A Data-Oriented Approach to Semantic Interpretation", *Proceedings Workshop on Corpus-Oriented Semantic Analysis*, ECAI-96, Budapest, Hungary.

R. Bod and R. Kaplan, 1998. "A Probabilistic Corpus-Driven Model for Lexical-Functional Analysis", this proceedings.

R. Bonnema, 1996. *Data-Oriented Semantics*, Master's Thesis, Department of Computational Linguistics, University of Amsterdam, The Netherlands.

R. Bonnema, R. Bod and R. Scha, 1997. "A DOP Model for Semantic Interpretation", *Proceedings ACL/EACL-97*, Madrid, Spain.

M. Boros et al. 1996. "Towards understanding spontaneous speech: word accuracy vs. concept accuracy." *Proceedings ICSLP'96*, Philadelphia (PA).

E. Charniak, 1996. "Tree-bank Grammars", *Proceedings AAAI-96*, Menlo Park (Ca).

J. Goodman, 1996. "Efficient Algorithms for Parsing the DOP Model", *Proceedings Empirical Methods in Natural Language Processing*, Philadelphia (PA).

J. Goodman, 1998. *Parsing Inside-Out*, Ph.D. thesis, Harvard University, Massachusetts.

S. Miller et al. 1996. "A fully statistical approach to natural language interfaces", *Proceedings ACL'96*, Santa Cruz (Ca.).

G. van Noord, 1995. "The intersection of finite state automata and definite clause grammars", *Proceedings ACL'95*, Boston, Massachusetts.

G. van Noord, G. Bouma, R. Koeling and M. Nederhof, 1997. *Robust Grammatical Analysis for Spoken Dialogue Systems*, unpublished manuscript.

F. Pereira and Y. Schabes, 1992. "Inside-Outside Reestimation from Partially Bracketed Corpora", *Proceedings ACL'92*, Newark, Delaware.

M. Rajman 1995. "Approche Probabiliste de l'Analyse Syntaxique", *Traitement Automatique des Langues*, 36(1-2).

R. Scha 1992. "Virtuele Grammatica's en Creatieve Algoritmen", *Gramma/TTT* 1(1).

K. Sima'an, 1995. "An optimized algorithm for Data Oriented Parsing", In: R. Mitkov and N. Nicolov (eds.), *Recent Advances in Natural Language Processing 1995*, volume 136 of *Current Issues in Linguistic Theory*. John Benjamins, Amsterdam.

K. Sima'an, 1996. "Computational Complexity of Probabilistic Disambiguation by means of Tree Grammars", *Proceedings COLING-96*, Copenhagen, Denmark.

K. Sima'an, 1997. "Explanation-Based Learning of Data-Oriented Parsing", in T. Ellison (ed.) *CoNLL97: Computational Natural Language Learning*, ACL'97, Madrid, Spain.

E. Vallduvi, 1990. *The Informational Component*. Ph.D. thesis, University of Pennsylvania, PA.

G. Veldhuijzen van Zanten, 1996. *Semantics of update expressions*. Technical Report 24. NWO Priority Programme Language and Speech Technology, The Hague.