Natural Language Engineering **5** (4): 309–323. Printed in the United Kingdom

© 1999 Cambridge University Press

Context-sensitive spoken dialogue processing with the DOP model

RENS BOD

Institute for Logic, Language and Computation, University of Amsterdam, Spuistraat 134, NL-1012 VB Amsterdam, The Netherlands
and
School of Computer Studies, University of Leeds, Leeds LS2 9JT, UK
e-mail: rens@scs.leeds.ac.uk

(*Received September 1998; revised November 1999*)

Abstract

We show how the DOP model can be used for fast and robust context-sensitive 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 Priority Programme 'Language and Speech Technology'. In this paper, we extend the original Data-Oriented Parsing (DOP) model to context-sensitive interpretation of spoken input. The system we describe uses the OVIS corpus (which consists of 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 analysed 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.

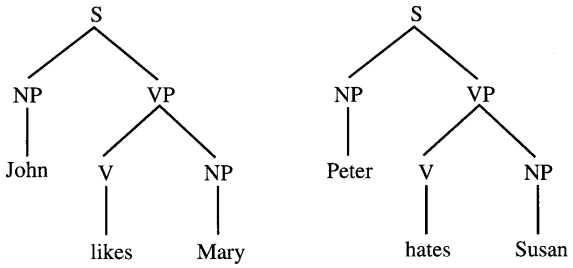
1 Introduction

The Data-Oriented Parsing (DOP) model is a corpus-based parsing model which uses subtrees from parse trees in a corpus to analyse new sentences. The occurrence-frequencies of the subtrees are used to estimate the most probable analysis of a sentence (cf. Bod 1992, 1993, 1995, 1998; Bod and Kaplan 1998; Bonnema, Bod and Scha 1997; Chappelier and Rajman 1998; Charniak 1996; Cormons 1999; Goodman 1996, 1998; Kaplan 1996; Rajman 1995; Scha 1990, 1992; Scholtes 1992; Sekine and Grishman 1995; Sima'an 1995, 1997, 1999; Tugwell 1995 and Way 1999).

To-date, DOP has mainly been applied to corpora of trees whose labels consist of primitive symbols (but see Bod and Kaplan (1998), Cormons (1999) and Way (1999) for more sophisticated DOP models). Let us illustrate the original DOP model, called DOP1 (Bod 1992, 1993), with a very simple example. Assume a corpus consisting of only two trees:

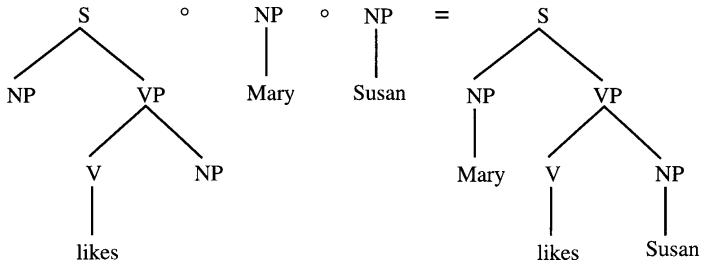
https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

(1)



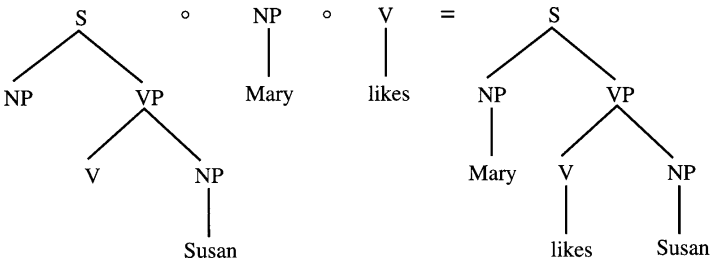
New sentences may be derived by combining subtrees from this corpus, by means of a node-substitution operation indicated as ∘. Node-substitution identifies the leftmost nonterminal frontier node of one tree with the root node of a second tree (i.e. the second tree is *substituted* on the leftmost nonterminal frontier node of the first tree). Under the convention that the node-substitution operation is left-associative, a new sentence such as *Mary likes Susan* can be derived 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 a subtree t as the probability of selecting t among all corpus-subtrees that can be substituted on the same node as t. This probability is equal to the number of occurrences of $t, |t|$, divided by the total number of occurrences of all subtrees t' with the same root label as t. Let $r(t)$ return the root label of t. Then we may write:

$$P(t) = \frac{|t|}{\sum_{t':r(t')=r(t)} |t'|}.$$

Since each node substitution is independent of previous substitutions, the probability of a derivation $D = t_1 \circ \cdots \circ t_n$ is computed by the product of the probabilities of its subtrees t_i:

$$P(t_1 \circ \cdots \circ t_n) = \prod_i P(t_i).$$

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

The probability of a parse tree is the probability that it is generated by any of its derivations. The probability of a parse tree T is thus computed as the sum of the probabilities of its distinct derivations D:

$$P(T) = \sum_{D \text{ derives } T} P(D).$$

Usually, several different parse trees can be derived for a single sentence, and in that case their probabilities provide a preference ordering.

Bod (1992) demonstrated that DOP can be implemented using conventional context-free parsing techniques by converting subtrees into rewrite rules. However, the computation of the most probable parse of a sentence is NP-hard (Sima'an 1996). The parse probabilities can be estimated by Monte Carlo techniques (Bod 1995, 1998), but efficient algorithms are only known for the most probable derivation of a sentence (the 'Viterbi parse' – see Bod (1995), Sima'an (1995)) or for the 'maximum constituents parse' of a sentence (Goodman 1996, 1998; Bod 1996).

In van den Berg, Bod and Scha (1994) and Bod, Bonnema and Scha (1996), DOP was generalized to semantic interpretation by using corpora annotated with compositional semantics. In the current paper, we extend the DOP model to context-sensitive interpretation in spoken dialogue understanding, and 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' (see Boves, Landsbergen, Scha and van Noord 1996).

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 probable meaning M of a word string W: $\text{argmax}_M \text{P}(M, W)$. Next we demonstrate how the dialogue context C can be integrated so as to compute $\text{argmax}_M \text{P}(M, W|C)$. Finally, we interface DOP with speech and show how the most probable meaning M of an acoustic utterance A given dialogue context C is computed: $\text{argmax}_M \text{P}(M, A|C)$. The last section of this paper deals with the experimental evaluation of the model.

2 The OVIS corpus: trees enriched with compositional 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.[1] 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?"

[1] The pilot version is based on a German system developed by Philips Dialogue Systems in Aachen (Aust, Oerder, Seide and Steinbiss 1995), adapted to Dutch.

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

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 overgenerates, 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 predefined grammar is used to determine the correct syntactic annotation; there is a small set of annotation guidelines, that has the degree of detail necessary to avoid an 'anything goes' attitude from 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). Each update consists of an *information state* or *form*. This form is a hierarchical structure with slots and values for the origin and destination of a train connection, for the time at which the user wants to leave or arrive, etc. Slots represent information that is already in the information form, whereas values represent information that is related to the change of information. The distinction between slots and values can be regarded as a special case of the ground and focus distinction (Vallduvi 1990; van Noord, Bouma, Koeling and Nederhof 1999), where ground relates to the given information (slot) and focus to the new information (value). For example, if the system asks *Van waar naar waar wilt u reizen?* ('*From where to where do you want to travel?*'), and the user answers with *Ik wil van Utrecht naar Leiden* (literally: '*I want from Utrecht to Leiden*'), then the update in (4) is yielded:

(4)

```
user.wants.(origin.place.town.utrecht; destination.place.town.leiden)
```

In this update, `utrecht` and `leiden` are the values of the slots `origin.place.town` and `destination.place.town`, respectively. The update language of Veldhuijzen van Zanten (1996) also allows for (a limited) encoding of speech-act information, such as denials and corrections. For instance, the user utterance *Ik wil niet vandaag maar morgen naar Almere* (literally: '*I want not today but tomorrow to Almere*') yields the following update:

(5)

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

The '#' in (5) means that the information between the square brackets is denied, while the '!' denotes the corrected information.

This update language is used to enrich the syntactic nodes of the OVIS trees with semantic annotations, which is accomplished in a *compositional* way by means of the following convention:

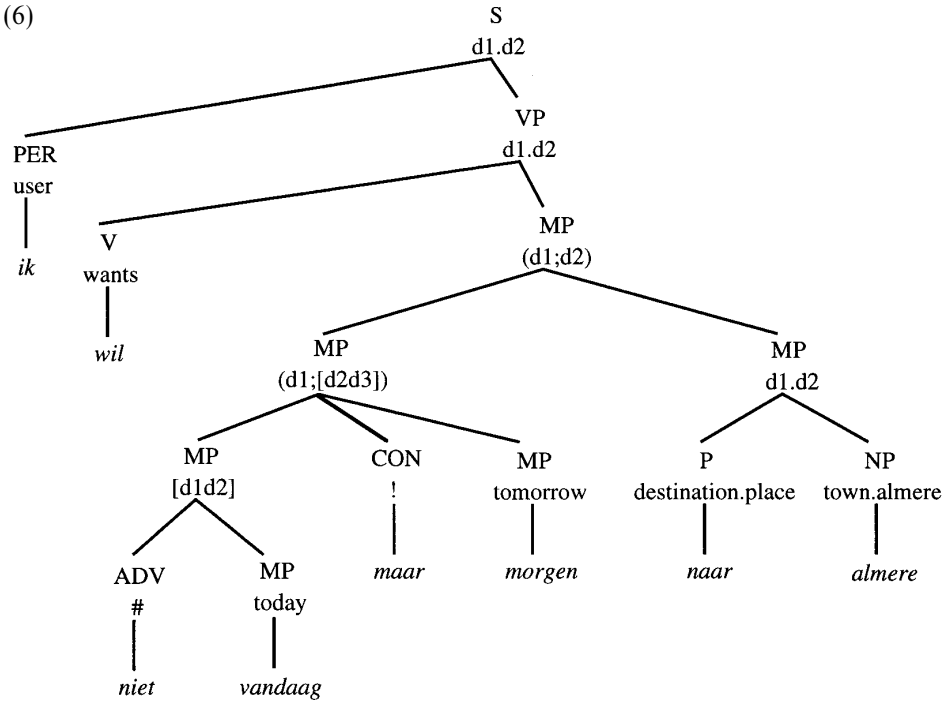
- Every meaningful *lexical* node is annotated with an *update expression* which represents the meaning of the lexical item.
- Every meaningful *non-lexical* node is annotated with a *formula schema* which

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

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 (6):

(6)



The meaning of *Ik wil niet vandaag maar morgen naar Almere* is thus compositionally built up out of the meanings of its sub-constituents. Substituting the meaning representations into the corresponding variables yields the update semantics of the top-node: `user.wants.(([# today]; [! tomorrow]); destination.place.town.almere)`.[2]

Note that our annotation convention is based on the notion of *surface compositionality*: it assumes that the meaning representation of a surface-constituent *can* in fact always be composed out of the meaning representations of its sub-constituents. This assumption is not unproblematic. To maintain it in the face of phenomena such as non-standard quantifier scope or discontinuous constituents creates complications in the syntactic or semantic analyses assigned to certain sentences and their constituents. It is unlikely, therefore, that our annotation convention can be viewed

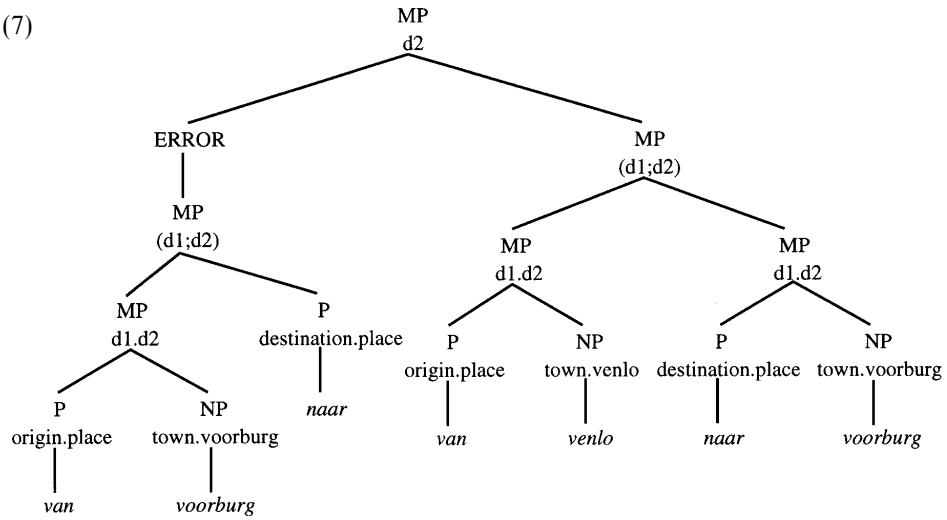
[2] The syntactic annotation of (6) may seem relatively arbitrary, as the annotator could as well have created a tree with the two '*niet vandaag maar morgen*' and '*naar almere*' MPs being direct sisters of the '*wil*' V. However, in Bonnema (1996) and Bod et al. (1996), it is shown that the DOP model is largely insensitive to inconsistencies in the syntactic annotations, as long as the semantic annotations yield the correct top-node semantics.

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

as completely general. However, it turned out to be powerful enough for annotating the sentences from the OVIS corpus. Moreover, to the best of our knowledge, the OVIS is the first corpus annotated according to the *Principle of Compositionality of Meaning*.

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*') is given in (7).

(7)



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:

(8) `(origin.place.town.venlo; destination.place.town.voorburg)`

The manual annotation of 10,000 OVIS utterances may seem a laborious and error-prone process. 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).

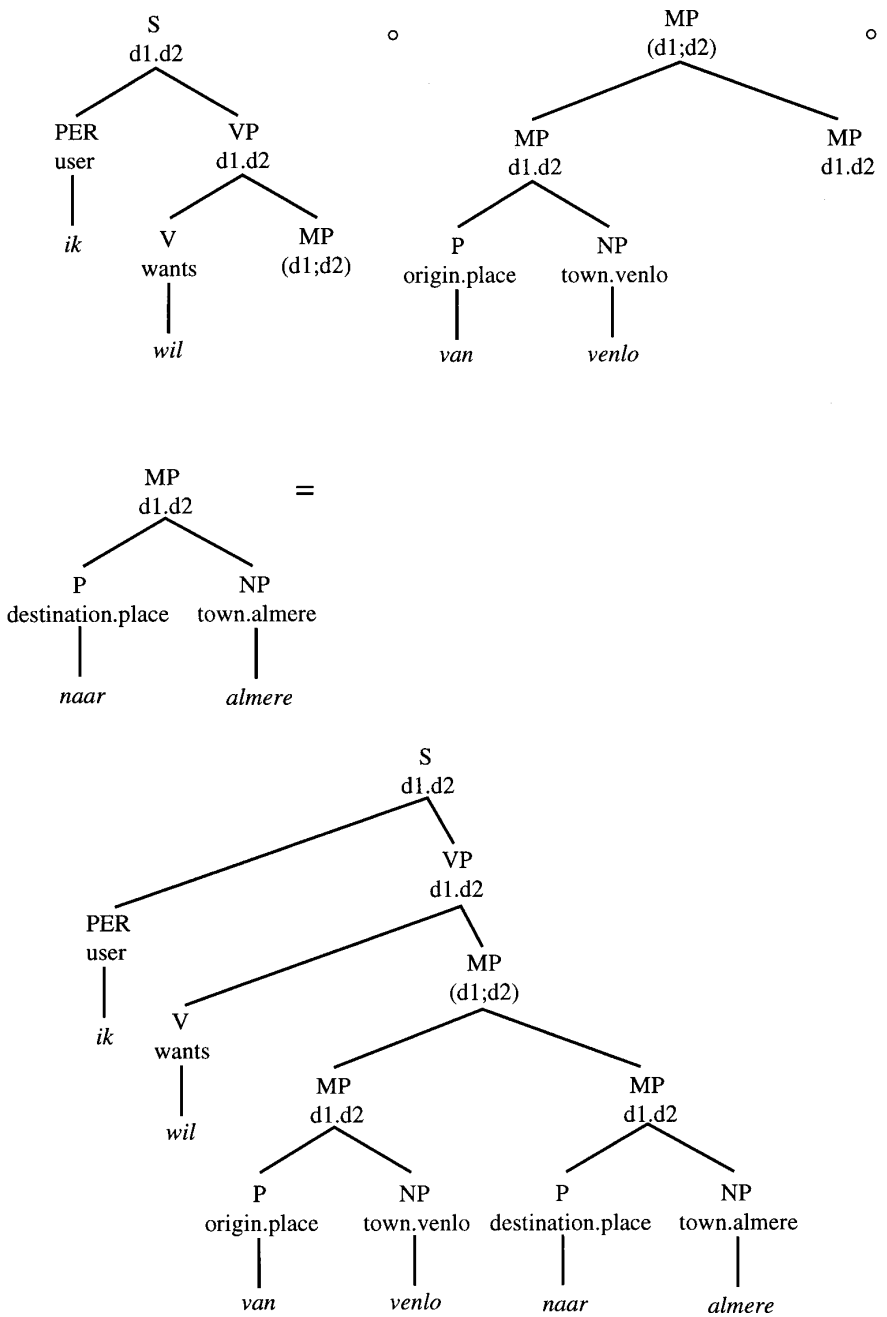
3 Using the OVIS corpus for data-oriented semantic analysis

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

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

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 (6) and (7):

(9)



https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

which yields the following top-node update semantics:

(10)
```
user.wants.
  (origin.place.town.venlo;
   destination.place.town.almere)
```

The probability calculations for this semantic DOP model are similar to the original DOP model, i.e. 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 $D = t_1 \circ \cdots \circ t_n$ 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. Finally, 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 of a word string cannot be computed in deterministic polynomial time. Although the most probable meaning can be estimated by Monte Carlo techniques (see Bod 1995, 1998), the computation of a sufficiently large number of random derivations is currently not efficient enough for a practical application. To-date, only the *most probable derivation* can be computed in near-to-real-time (by a best-first Viterbi optimization algorithm – see Sima'an 1995). We therefore assume that most of the probability mass for each top-node meaning is focused on a single derivation. Under this assumption, the most probable meaning of a string is the top-node meaning generated by the most probable 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 $C_i = W_{i-1}, W_{i-2} \ldots W_1$ is given by $\mathrm{P}(M, W_i | W_{i-1}, W_{i-2} \ldots W_1)$. 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 upon the full dialogue context, but only on the previous (system) question W_{i-1}. Under this assumption,

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

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 analysed 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:

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

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 in general: 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 an utterance tends to be interpreted by the smallest, most specific domain in which it can still be analysed.

5 Interfacing DOP with speech

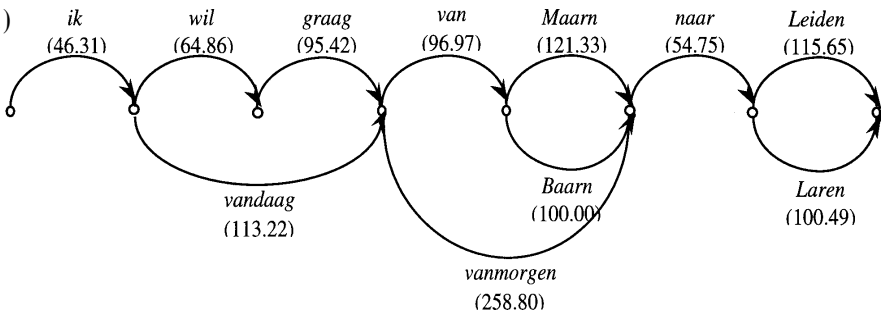
So far, we have dealt with the estimation of the probability $\mathrm{P}(M, W|C)$ of a meaning M and a word string W given a dialogue context C. However, in *spoken* dialogue processing, the word string W is not given. The input for DOP in the OVIS system are *word-graphs* produced by the speech recognizer (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 (e.g. see (11)). 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 $\langle i, j, w \rangle$. 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 $P(a|w)$ for a hypothesized word w normalized by the length of w. Converting these acoustic scores into their corresponding probabilities, the acoustic probability $P(A|W)$ 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. An example of a simplified word-graph for the uttered sentence *Ik wil graag vanmorgen naar Leiden*

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

('*I'd like to go this morning to Leiden*') is shown in (11):

(11)



The probabilistic interface between DOP and speech word-graphs thus consists of the interface between the DOP probabilities $\mathrm{P}(M, W|C)$ and the word-graph probabilities $\mathrm{P}(A|W)$ so as to compute the probability $\mathrm{P}(M, A|C)$ and $\mathrm{argmax}_M \mathrm{P}(M, A|C)$.

We start by rewriting $\mathrm{P}(M, A|C)$ as:

$$\begin{aligned}\mathrm{P}(M, A|C) &= \sum_W \mathrm{P}(M, W, A|C) \\ &= \sum_W \mathrm{P}(M, W|C) \cdot \mathrm{P}(A|M, W, C).\end{aligned}$$

The probability $\mathrm{P}(M, W|C)$ is computed by the dialogue-sensitive DOP model, as explained in the previous section. To estimate the probability $\mathrm{P}(A|M, W, C)$ on the basis of the information available in the word-graphs, we must make the following independence assumption: the acoustic utterance A depends only upon the word string W, and not on its context C and meaning M (cf. Bod and Scha 1994). Under this assumption:

$$\mathrm{P}(M, A|C) = \sum_W \mathrm{P}(M, W|C) \cdot \mathrm{P}(A|W).$$

To make fast computation feasible, we further 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):

$$\mathrm{P}(M, A|C) = \mathrm{P}(M, W|C) \cdot \mathrm{P}(A|W).$$

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 $\mathrm{P}(M, W|C)$ and the word-graph probability $\mathrm{P}(A|W)$.

As to the parsing of word-graphs, it is well-known that chart 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 $\langle i, i+1\rangle$. For word-graphs, a transition $\langle i, j, w\rangle$ 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 (11)). It is easy to see that, for word-graphs, the initialization of the chart consists of entering each word w from transition $\langle i, j, w\rangle$ into chart entry $\langle i, j\rangle$. Next, parsing proceeds with the subtrees that are triggered by the dialogue context C. That is, every subtree t is converted into a context-free rewrite rule: $root(t) \rightarrow yield(t)$, and every such rule is indexed to maintain the link to its original subtree

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

(see Bod 1995). The rules obtained in this way are used to construct a chart-like parse forest for the input word-graph. The most probable derivation from the chart is computed by a dynamic programming algorithm based on the well-known Viterbi algorithm for CKY (Sima'an 1995, 1997). This algorithm has a time complexity which is cubic in the number of word-graph nodes and linear in the grammar size. Sima'an's algorithm is an optimization over previous statistical disambiguation algorithms (such as Jelinek, Lafferty and Mercer 1990), in that Sima'an's algorithm has a time complexity which is linear instead of quadratic in the grammar size. Finally, the top-node meaning of the tree resulting from the most probable 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 context-sensitive spoken language understanding?
2. Can we constrain the OVIS subtrees without losing 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 words per word-graph path is 4.6, and the average number of transitions per word (i.e. the number of transitions in the word-graph divided by the actual number of words in the acoustic utterance) is 4.2. 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, Eckert, Gallwitz, Görz, Hanrieder and Niemann (1996) and van Noord et al. (1999), 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>
```

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

Table 1. *Experimental results on OVIS word-graphs*

		Semantic accuracy			
Subtree-depth	# Subtrees	Match	Precision	Recall	CPU time
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

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; and (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 suspected that it was 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. Table 1 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.

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 as 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 is caused by the ambiguities in the word-graphs (i.e. the multiple paths).

Secondly, we were 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.

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

- *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.

These results thus indicate that the accuracy decreases if we restrict the subtree set by frequency, number of lexical items, or non-head words.

Finally, we were 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 the correct prediction for the 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 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 highest probability to *Donderdag acht februari* and its meaning, rather than to *Dordrecht acht februari.*

7 Conclusions

We have shown how the DOP model can be used for efficient and robust context-sensitive interpretation of spoken input in the OVIS spoken dialogue system. The system we described uses syntactically and semantically analysed 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 analysed and interpreted.

The experimental evaluation showed that DOP's combination of lexical, syntactic, semantic and dialogue dependencies yields promising results. The experiments also indicated that any restrictions on the subtree-set diminish the accuracy, even when intuitively unimportant subtrees with only non-head words were 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 functional structures associated with the surface structures so as to deal with more complex linguistic phenomena and feature generalizations (see Bod and Kaplan 1998).

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

Acknowledgements

This research was supported by NWO, the Netherlands Organization for Scientific Research (Priority Programme Language and Speech Technology). We are grateful to Khalil Sima'an for using his optimized DOP parser, and to Remko Bonnema for using SEMTAGS and the relevant semantic interfaces. The OVIS corpus was annotated by Mike de Kreek and Sascha Schütz.

References

Aust, H., Oerder, M., Seide, F. and Steinbiss, V. (1995) The Philips automatic train timetable information system. *Speech Communication* **17**: 249–262.

van den Berg, M., Bod, R. and Scha, R. (1994) A corpus-based approach to semantic interpretation. *Proceedings Ninth Amsterdam Colloquium* Amsterdam, The Netherlands.

Bod, R. (1992) A computational model of language performance: Data oriented parsing. *Proceedings COLING-92* Nantes, France.

Bod, R. (1993) Using an annotated language corpus as a virtual stochastic grammar. *Proceedings AAAI'93* Morgan Kauffman, Menlo Park, CA.

Bod, R. (1995) Enriching linguistics with statistics: performance models of natural language. *ILLC Dissertation Series 1995-14*. University of Amsterdam.

Bod, R. (1996) Efficient algorithms for parsing the DOP model? A reply to Joshua Goodman. *Computation and Language Archive* cmp-lg/9605031.

Bod, R. (1998) *Beyond Grammar: An Experience-Based Theory of Language*. CSLI Publications, Cambridge University Press.

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

Bod, R., Bonnema, R. and Scha, R. (1996) A data-oriented approach to semantic interpretation. *Proceedings Workshop on Corpus-Oriented Semantic Analysis* (*ECAI-96*) Budapest, Hungary.

Bod, R. and Kaplan, R. (1998) A probabilistic corpus-driven model for lexical-functional analysis. *Proceedings COLING-ACL'98* Montreal, Canada.

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

Bonnema, R., Bod, R. and Scha, R. (1997) A DOP model for semantic interpretation. *Proceedings ACL/EACL-97* Madrid, Spain.

Boros, M., Eckert, W., Gallwitz, F., Görz, G., Hanrieder, G. and Niemann, H. (1996) Towards understanding spontaneous speech: word accuracy vs. concept accuracy. *Proceedings ICSLP'96* Philadelphia, PA.

Boves, L., Landsbergen, J., Scha, R. and van Noord, G. (1996) *Projectplan NWO Priority Programme Language and Speech Technology*. NWO, The Hague.

Chappelier, J. and Rajman, M. (1998) Extraction stochastique d'arbres d'analyse pour le modèle DOP. *Proceedings TALN 1998* Paris, France.

Charniak, E. (1996) Tree-bank Grammars. *Proceedings AAAI'96* Portland, Oregon.

Cormons, B. (1999) Analyse et désambiguisation: Une approche à base de corpus (Data-Oriented Parsing) pour les répresentations lexicales fonctionnelles. *PhD thesis*, Université de Rennes, France.

Goodman, J. (1996) Efficient algorithms for parsing the DOP model. *Proceedings Empirical Methods in Natural Language Processing* Philadelphia, PA.

Goodman, J. (1998) Parsing Inside-Out. *PhD thesis*, Harvard University, MA.

Jelinek, F., Lafferty, J. and Mercer, R. (1990) Basic Methods of Probabilistic Context Free Grammars. *Technical Report IBM RC 16374* (*#72684*), IBM, Yorktown Heights, NJ.

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press

Kaplan, R. (1996) A probabilistic approach to lexical-functional analysis. *Proceedings of the 1996 LFG Conference and Workshops*. CSLI Publications, Stanford, CA.

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

van Noord, G., Bouma, G., Koeling, R. and Nederhof, M. (1999) Robust grammatical analysis for spoken dialogue systems. *Natural Language Engineering* **5**(1): 45–93.

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

Scha, R. (1990) Language theory and language technology; Competence and performance. In: Q. A. M. de Kort and G. L. J. Leerdam (eds.), *Computertoepassingen in de Neerlandistiek*. Almere: Landelijke Vereniging van Neerlandici (LVVN-Jaarboek) (in Dutch).

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

Scholtes, J. (1992) Resolving linguistic ambiguities with a neural Data-Oriented Parsing (DOP) system. In: I. Aleksander and J. Taylor (eds.), *Artificial Neural Networks 2*, Vol. 2, Elsevier.

Sekine, S. and Grishman, R. (1995) A corpus-based probabilistic grammar with only two non-terminals. *Proceedings Fourth International Workshop on Parsing Technologies* Prague, Czech Republic.

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

Sima'an, K. (1996) Computational complexity of probabilistic disambiguation by means of tree grammars. *Proceedings COLING-96* Copenhagen, Denmark.

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

Sima'an, K. (1999) *Learning Efficient Disambiguation*. ILLC Dissertation Series 1999-02. Utrecht/Amsterdam.

Tugwell, D. (1995) A state-transition grammar for Data-Oriented Parsing. *Proceedings EACL'95* Dublin, Ireland.

Vallduvi, E. (1990) The Informational Component. *PhD thesis*, University of Pennsylvania, PA.

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

Way, A. (1999) A hybrid architecture for robust MT using LFG-DOP. *Journal for Experimental and Theoretical Artificial Intelligence* **11** (Special Issue on Memory-Based Language Processing).

https://doi.org/10.1017/S1351324900002291 Published online by Cambridge University Press