

# Unsupervised Learning with the DOP Model

NWO Vici-program

Extract from the original proposal

## 1. Summary

The aim of this project is to develop a general technique for the *unsupervised* learning of linguistic parsing, musical analysis and problem-solving, and to use the technique for boosting various concrete applications. To make our goal feasible, we start out with higher-level structures like *trees*. Tree-structures are compositional representations that describe how parts of a cognitive input combine into larger units and how these units combine into a whole. We will employ the *Data-Oriented Parsing* (DOP) framework, which has already been successful in integrating linguistic and musical processing. DOP analyzes new input by combining partial-analyses (e.g. subtrees) from analyses of previous input in a corpus. The most probable analysis of an input is computed from the counts of all sub-analyses in the corpus.

While DOP proposes a probabilistic memory-based view on cognition, the model does not account for the learning of *initial* analyses. We intend to show that DOP can be extended to *unsupervised* learning, *U-DOP*. The underlying idea of U-DOP is that if we do not know which tree-structures can be assigned to first input, we can just as well assume that *all* tree-structures are initially possible, and store only those subtrees that partake in analyzing new input. The main theme of this project, then, is to generalize U-DOP to other modalities and to test the model on hand-annotated corpora and concrete applications.

If unsupervised learning can compete with supervised learning, it will not only shed new light on the old controversy about the “poverty of the stimulus”, but may also overcome the time-consuming annotation of data-sets.

## 2. Description of the proposed research

### 2a. Research topic

#### Overall aim of the project

The primary goal of this project is to develop a general machine learning technique for the *unsupervised* learning of linguistic parsing, musical analysis and problem-solving. This technique will be used to boost a number of applications: speech recognition, machine translation, pitch spelling and equational reasoning. To make our goal feasible, we start out with higher-level structures such as *trees*. Tree structures are compositional representations that describe how parts of a cognitive input combine into larger units and how these units combine into a whole.

We will use the general *Data-Oriented Parsing* (DOP) framework, which has already been successful in integrating linguistic and musical processing. DOP analyzes new input by combining partial-analyses (e.g. subtrees) from analyses of previous input in a representative corpus (cf. Bod 1998, 2002a,b; Bod and Kaplan 1998; Scha 1990;

Hearne and Way 2003). The most likely analysis of an input is computed from the counts of all subtrees in the corpus. The innovative feature of DOP is that all previous subtrees - regardless of size - can play a role in the analysis of new input.

While DOP proposes a probabilistic, memory-based view on cognition, that integrates rules and exemplars, the model does not account for the learning of *initial* analyses. Recently, a first proposal to extend DOP to *unsupervised* learning of language was presented in Bod (2006a,b). The underlying idea of Unsupervised DOP (“U-DOP”) is that if we do not know which tree-structures can be assigned to first sentences, we can just as well assume that *all* tree-structures are initially possible, and store only those subtrees that partake in analyzing *new* sentences, using well-known statistical estimation methods. The main theme of this project, then, is to fully work out U-DOP for natural language, to extend it to other modalities, and to test the resulting model not only against existing hand-annotated corpora but also in the context of a number of concrete applications. If unsupervised learning can compete with supervised learning, it will not only shed new light on the old controversy of the “poverty of stimulus”, but it may also make the time-consuming annotation of data-sets superfluous, possibly resulting in improved applications.

### **Measurable key objectives**

1. To develop an extension of the all-subtrees DOP approach to unsupervised parsing of word strings (U-DOP).
2. To develop an efficient algorithm for U-DOP.
3. To apply and train U-DOP on very large data sets so as to (potentially) compete with supervised systems.
4. To use U-DOP for boosting a statistical machine translation system.
5. To use U-DOP for boosting a speech recognition system (i.e. as a structural language model for speech).
6. To extend U-DOP to music (U-DOM) and automated reasoning (U-DOR) and to test the resulting models by improving two concrete applications: pitch spelling and physics problem-solving.
7. To create a general unsupervised machine learning model for different cognitive domains.
8. To answer the research question: *Can we beat supervised parsing and learning by unsupervised parsing and learning?*

### **Scientific Background**

We will start by (briefly) explaining the *supervised* DOP approach to natural language parsing and next discuss how we intend to generalize DOP to *unsupervised* learning in different modalities. We thus take Computational Linguistics as the central, interdisciplinary field on which other fields will be built.

#### **2.1 Supervised vs Unsupervised Parsing of Language**

Natural language parsing systems seek to assign to each sentence the correct syntactic tree structure that describes which words of the sentence combine into phrases and how these phrases combine into a structure for the whole sentence. During the last few years, parsing systems have led to improvement of several natural language processing (NLP)

applications such as machine translation, database interfaces, sentence compression and speech recognition (see Lease et al. 2006 for a recent overview).

All current state-of-the-art parsing systems are *supervised*, that is, they are trained on large human-annotated data-sets (e.g. Collins 2000; Bod 2001, 2003; Charniak and Johnson 2005). Unfortunately, such hand-annotated data-sets are available for a few languages only, and their construction is extremely time-consuming often taking years of development. While *semi-supervised* methods have recently resulted in promising improvements of natural language parsing, especially self-training (cf. McClosky et al. 2006), they still need data-sets of annotated sentences to start with. A key issue in natural language processing is therefore the development of *unsupervised* methods for extracting parsing models from unlabeled raw data, of which unlimited quantities are available.

During the last few years there has been considerable progress in *unsupervised* parsing. The performance of unsupervised parsers has gone up from around 40% unlabeled f-score on the ATIS corpus (van Zaanen 2000; Clark 2001) to around 80% f-score on the Wall Street Journal (WSJ) corpus in Bod (2006b). While these scores remain behind the *labeled* f-score of around 91% of the best supervised parsers (cf. Bod 2003; Charniak and Johnson 2005; McClosky et al. 2006), it remains an open question how far we can get with a purely unsupervised approach to parsing if trained on data-sets that are several magnitudes larger than hitherto attempted. This question is not just of academic interest but is of direct relevance for improving the aforementioned NLP applications and for overcoming the laborious task of creating hand-annotated data-sets.

Yet, most if not all unsupervised parsing models limit either the lexical or the structural context that is taken into account, or both. That is, these unsupervised parsing models operate by statistically comparing *contiguous* subsequences of sentences: if substrings appear in similar lexical contexts they are likely to form a constituent of the same category (cf. van Zaanen 2000; Clark 2001; Klein and Manning 2002, 2004)<sup>1</sup>. There is an increasing insight that for building accurate unsupervised parsers it is imperative to also take into account *non-contiguous* substrings. This may be illustrated by the comparative construction "*more...than*" in the sentence *BA carried more people than cargo in 2004*. And it is well-known that there exist many other lexical dependencies which may be separated by any number of other words and which can therefore not be described by contiguous substrings. What would be needed is an "all-subtrees" approach that statistically compares all possible *subtrees* rather than all possible *substrings*.

This "all-subtrees" or "DOP" ("Data-Oriented Parsing") approach has already been highly successful in *supervised* parsing, where it resulted in very accurate parsing systems (cf. Bod 2001, 2003; Hearne and Way 2004). The key innovation of the DOP approach is to use subtrees of arbitrary size and shape as the elementary units of a probabilistic grammar (effectively resulting in a very redundant stochastic tree-substitution grammar or STSG).

Let us illustrate supervised DOP with a simple example. Suppose that we start with a very small corpus of only two sentences with their phrase-structure trees that are labeled by traditional lexical-syntactic categories, shown in figure 1. (We have left out some lexical categories to keep the example simple.)

---

<sup>1</sup> While there is also work on bootstrapping *dependency* trees (see Klein 2005 for an overview), we focus in this project on learning *constituent* trees, since virtually all state-of-the-art parsing systems used in concrete applications are based on constituent trees (including the foreseen applications in this project).

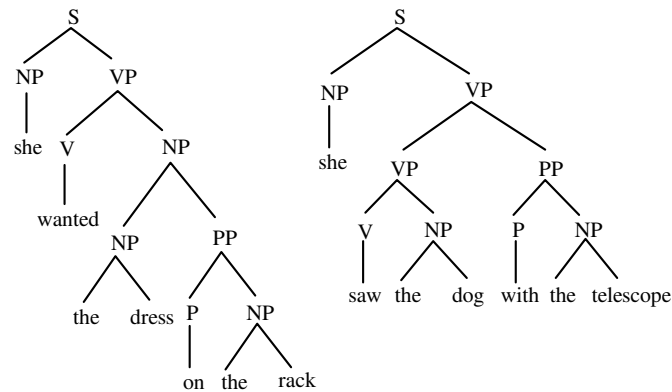


Figure 1. A small corpus of two tree structures

A new sentence, such as *She saw the dress with the telescope* may be derived by combining subtrees from the trees in the corpus by means of simple node-substitution, as shown in figure 2.

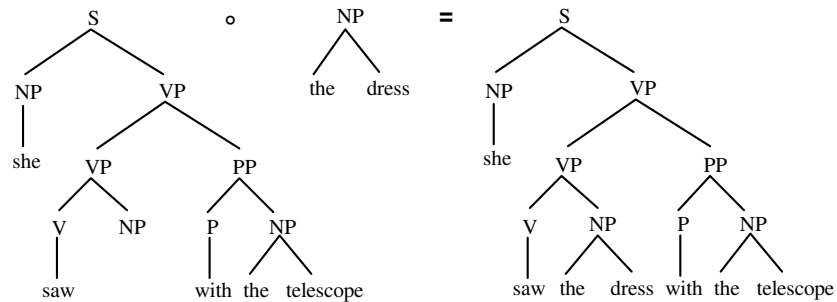


Figure 2. Analyzing a new sentence by combining subtrees from Figure 3

We can also derive an *alternative* phrase structure for this test sentence, namely by combining three (rather than two) subtrees from figure 1, as shown in figure 3.

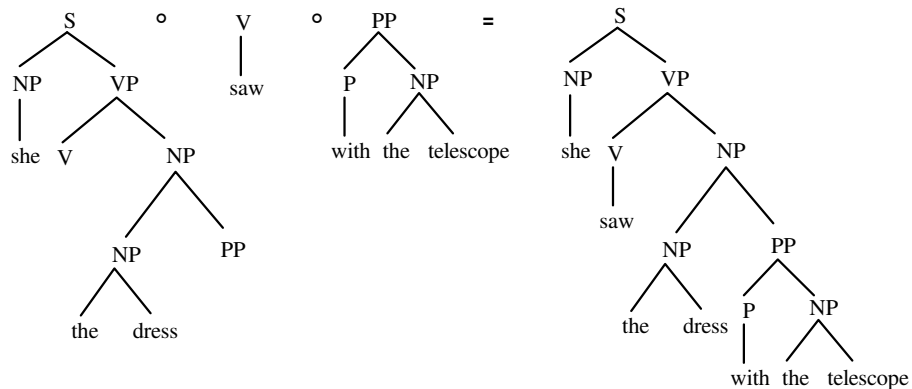


Figure 3. A different derivation for *She saw the dress with the telescope*

The sentence *She saw the dress with the telescope* is thus ambiguous in that it can be analyzed in (at least) two different ways: one analysis is analogous to the first tree figure 3 and the other analysis is analogous to the second tree. Both analyses can in principle be perceived by humans although the analysis in figure 2 is generally judged as the most plausible one. The problem of selecting the most plausible analysis of a sentence is known as the ambiguity problem, which is one of the hardest problems in NLP: Charniak (1997) estimates that an average sentence from the Wall Street Journal has over a thousand different syntactic structures.

How can we select from the possible structures of a sentence the "best" tree? DOP essentially employs a statistical methodology: the original DOP model computes the best tree from the relative frequencies of partial trees in a large corpus of previous trees. Results from psycholinguistics indeed support the idea that the frequency of occurrence of a structure is a very important factor in language comprehension (Jurafsky 2003; Gahl and Garnsey 2004). Of course, the frequency of a structure is not the only factor in syntactic disambiguation. Discourse context, semantics and recency also play an important role. It was shown elsewhere that these other factors can be integrated into DOP's probability model (see Bod 1998, 1999, 2006e).

During the last few years, many different versions of the DOP model have been developed, ranging from the use of the simple relative-frequency estimator for computing parse tree probabilities by Bod (1998) (which was shown to be an inconsistent estimator by Johnson 2002), to the use of statistically consistent estimators for DOP, either by the well-known Expectation-Maximization (EM) algorithm (Bod 2000b, 2006b) or by a new model known as DOP\* (Zollmann and Sima'an 2005). The technique has been considerably refined by Goodman (1996, 2003), who provided an efficient reduction of DOP, and by Collins and Duffy (2002), who applied kernel methods and voting to an all-subtrees model.

It has been empirically shown that the parse accuracy monotonously increases with increasing subtree size when tested on fresh data (e.g. Bod 2001; Collins and Duffy 2001; Sima'an 1999; Hearne and Way 2003). As a consequence, other parsing models have incorporated the use of arbitrarily large subtrees (e.g. Kudo et al. 2005; Henderson and Titov 2005; Moschitti et al. 2006). Several efficient algorithms have been developed that allow for fast and accurate parsing with DOP (e.g. Bod 2003; Goodman 2003; Sima'an 1999; Hearne and Way 2004).

## **2.2 Towards Unsupervised Data-Oriented Parsing**

While the supervised DOP approach has been very successful both in natural language parsing and music analysis (cf. Bod 2002a/b; van Zaanen et al. 2003; Schaefer et al. 2004), the approach is limited to domains for which human-annotated data-sets exist. As has been argued above, one of the most important challenges is to investigate whether supervised models like DOP can be generalized to unsupervised learning of structures from unlabeled data. Although we believe that semi-supervised methods, such as active learning and self-training, are also very promising, they still need annotated sentences to start with. Yet, we will investigate integrations of semi-supervised and unsupervised learning, which will be particularly important if the entirely unsupervised approach for some modality does not work well – see workplan.

In Bod (2006a), a first unsupervised version of DOP was proposed, termed *U-DOP*. Instead of using all subtrees from a set of *given* parse trees, U-DOP initially assigns all possible (binary) trees to a large data-set of initial sentences and next uses the subtrees from these trees to compute the most probable parse trees for new sentences. The underlying methodology of U-DOP is similar to (supervised) DOP: since we do not know beforehand what kind of structures are appropriate, we should not *a priori* restrict the set of possible structures, but take them all and let the statistics decide which structures (and subtrees thereof) are useful in analyzing new data. U-DOP thus allows initially for *any* partial non-contiguous string to form a syntactic group. In Bod (2006e), it was argued that (U-)DOP is congenial to usage-based and construction-grammar accounts where it is recognized that multi-word units can form exceedingly complex structures that are ubiquitous in language (cf. Fillmore et al. 1988; Croft 2003; Bybee 2006). U-DOP is also reminiscent of item-based approaches to language acquisition (cf. Tomasello 2003). However, U-DOP extends these (psycho)linguistic approaches by providing a computational model that aims to learn the syntactic structures for sentences in an automatic way and that can produce new sentences out of previous sentences.

To give an illustration of this U-DOP model, consider the following part-of-speech (p-o-s) string NNS VBD JJ NNS from the Wall Street Journal portion (WSJ) in the Penn Treebank (Marcus et al. 1993) which may correspond to the sentence *Investors suffered heavy losses* (the first implementations of U-DOP still work with p-o-s strings). U-DOP starts by assigning all possible binary trees to this string, where each root node is labeled *S* and each internal node is labeled *X*. Thus NNS VBD JJ NNS has a total of five binary trees shown in figure 4 -- where for readability we add words as well.

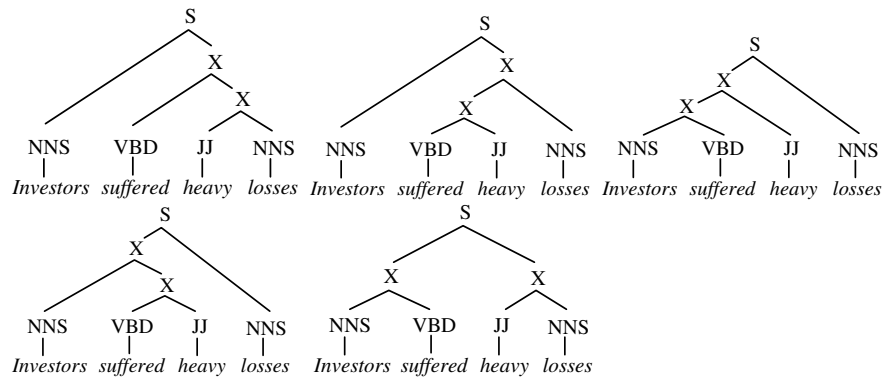


Figure 4. All binary trees for NNS VBD JJ NNS (*Investors suffered heavy losses*)

New sentences can then be parsed by combining subtrees from all possible trees for given sentences, similar to the original DOP model. We again let the frequencies decide which trees – and subtrees thereof – are most useful in analyzing fresh data. Of course, if we only had the sentence *Investors suffered heavy losses* in our corpus, there would be no difference in probability between the five parse trees in figure 4. But if we also have a different sentence where JJ NNS (*heavy losses*) appears in a different context, e.g. in *Heavy losses were reported*, its covering subtree gets a relatively higher frequency and the parse tree where *heavy losses* occurs as a constituent gets a higher total probability.

While we can efficiently represent the set of all binary trees of a string by means of a *chart*, we need to unpack the chart if we want to extract subtrees from this set of

binary trees. And since the total number of binary trees for the WSJ10 part (i.e. all WSJ sentences up to 10 words) is already 12 million, it is doubtful that we can apply the unrestricted U-DOP model to the WSJ in general. The U-DOP model in Bod (2006a) therefore randomly samples a large subset from the total number of parse trees from the chart, and next converts the subtrees from these parse trees into a compact reduction, as proposed by Goodman (2003). One of the innovations in this project will be to investigate reductions that operate directly on the parse *forest* rather than on a random subtree set (see below).

In Bod (2006b), U-DOP was extended to include EM estimation of the subtree parameters (Dempster et al. 1977), using cross-validation and taking relative frequencies as initial parameters. The resulting model obtained state-of-the-art performance on inducing tree structures for three domains: English, German and Chinese (Mandarin). It was shown that U-DOP outperformed a *supervised* parsing model on the WSJ40, i.e. a so-called binarized treebank PCFG.

Despite the success of this U-DOP model, it should be kept in mind that the model is still provisional and limited: the domains that were tested on are small compared to the test sets used by supervised parsers (and the algorithm needs to be extended with a PCFG reduction that operates directly on parse forests). Furthermore, U-DOP does not operate with word strings, it neither induces syntactic categories or verb-argument structures. Also, the evaluation against hand-annotated data is not optimal: the annotations in e.g. the Penn Treebank are very flat and unreasonably punish unsupervised parsers that usually learn more internal structure for e.g. NPs.

In order to potentially compete with supervised parsing method, we therefore propose in this project to extend U-DOP (1) to bootstrapping structures for *word* strings for large data-sets such as the so-called NANC data-set (available at the Linguistic Data Consortium, LDC) and the Europarl corpus (Koehn 2005), (2) to develop efficient algorithms for U-DOP (by creating PCFG reductions of parse forests rather than of STSGs – see workplan), (3) to induce syntactic categories and verb-argument structures, and (4) to evaluate U-DOP not only on hand-annotated data (which unreasonably favors supervised parsers), but to also assess the model in concrete task-based applications, such as machine translation and speech recognition.

We will therefore design U-DOP's test domains in such a way that the results can be directly used in concrete applications. For example, we will induce tree structures for the various languages from the multilingual Europarl corpus (Koehn 2005) which contains large extracts of the proceedings of the European Parliament in 11 European languages: Romanic (French, Italian, Spanish, Portuguese), Germanic (English, Dutch, German, Danish, Swedish), Greek and Finnish. The 'correctness' of U-DOP's induced trees can then be tested in a concrete MT system against the trees proposed by supervised parsers (see workplan), as well as in the context of non-tree-based MT systems (Och 2003). The workplan also gives details about extending U-DOP towards language modeling for speech recognition and integrating the model with semi-supervised learning (this is important as it is by no means certain that unsupervised parsing alone will actually be able to compete with supervised parsing).

## 2.3 Extending Unsupervised DOP to Other Modalities: Towards Integrating Cognition

A major working hypothesis of this project is that U-DOP can be generalized to unsupervised learning in other modalities, such as melodic structures for musical pieces (Lerdahl & Jackendoff 1983) and deductive-nomological structures for physics problems (see VanLehn 1988; Bod 2005a). The extension to other modalities is important not only for our goal to develop a general machine learning algorithm for different cognitive domains, but also to overcome the time-consuming hand-annotation of training data.

### 2.3.1 DOP and Music

It is rather straightforward to apply the DOP approach to melodic analysis. As in natural language, a listener segments a sequence of notes into groups or phrases that form a nested grouping structure for the whole piece which can again be represented by a tree (Longuet-Higgins 1976; Lerdahl & Jackendoff 1983). For example, according to Lerdahl & Jackendoff (1983: 37) a listener hears the following grouping structure for the first few bars of melody in Mozart's G Minor Symphony, K. 550.

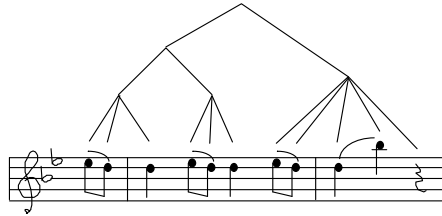


Figure 5. Grouping structure for first few bars of Mozart's G Minor symphony

Melodic trees are usually unlabeled: while in language there are syntactic constraints on how words can be combined into larger constituents, in music there are no such restrictions: in principle any note may be combined with any other note, depending on the musical style and period. This makes the problem of ambiguity in music even harder than in language (see Longuet-Higgins and Lee 1987).

For instance, the first few bars of Mozart's G Minor Symphony could also be assigned the following, alternative grouping structure (among other possible structures):

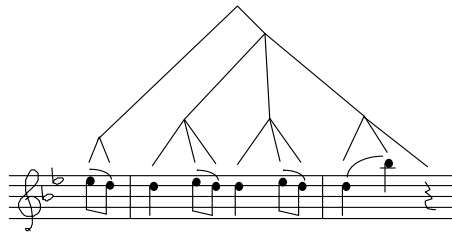


Figure 6. Alternative grouping structure for Mozart's opening theme

While this alternative structure is possible in that it *can* be perceived, it does not correspond to the structure that is actually perceived by a human listener. There is thus an important question how to select the perceived tree structure from the total set of possible tree structures of a musical input. Many systems attempt to disambiguate melodic structure in an entirely rule-based way. For example, Lerdahl and Jackendoff (1983) and Temperley (2001) use preference rules that describe Gestalt-perceptions of the kind



identified by Wertheimer (1923). However, similar to multi-word units in language, there are extremely many *multi-note units* and other idiomatic groups in music, which can only be covered by a model that also takes into account musical memory and previous musical experience (see Schaefer et al. 2004; Saffran et al. 2000; Juhász, 2004). Several approaches to melodic analysis are therefore probabilistic and exemplar-based and are trained on corpora such as the Essen Folksong Collection or EFC (Schaffrath 1995) which currently contains melodic trees for over 20,000 folksongs. The pitches in the folksongs are represented by integers added with information about duration and chromatic alteration. By automatically assigning three basic constituent labels to the nodes of the monophonic Essen folksongs (i.e.  $S$  for the whole song,  $N$  for each note and  $P$  for each intermediate phrase), we can develop supervised DOP models for music, as has been done in e.g. Bod (2002a,b) and van Zaanen et al. (2003).

However, these DOP models are (again) limited to domains for which there are actually melodically annotated corpora. There is thus an important question as to whether we can automatically bootstrap structure for music. In this project we propose that an all-subtrees approach like U-DOP will also be beneficial for bootstrapping melodic trees in music.

We propose to extend U-DOP to induce melodic structures for the 20,000+ EFC and to compare them with the human-annotated melodic structures. Since most folksongs are relatively short (smaller than 60 notes), we believe that U-DOP can in fact be applied to the EFC. We will initially assign all possible binary trees to half of the EFC and use subtrees thereof to predict the most probable parse trees for the other half, and vice versa. We can then compare the results against the human annotations in the EFC and against supervised melodic parsers (e.g. Temperley 2001) as well as unsupervised melodic parsers (e.g. Ferrand et al. 2003) (see workplan). Moreover, we will test U-DOP's annotations also on other, larger data-sets (e.g. Meredith 2006) and in two musical applications: pitch spelling and modulation prediction. We will furthermore investigate how far the U-DOP approach to music stretches by trying to induce harmonic structure for classical musical pieces. And we will explore whether a notion of “simplest structure”, which has been successfully formalized in NLP (Bod 2002b), can be used in music processing under different interpretations such as the most compact representation (Honingh and Bod 2005; Honingh 2006a).

### 2.3.2 DOP and Problem-Solving/Reasoning

What counts for linguistic and melodic analysis also counts for problem solving and reasoning: given a problem or theorem, there can be (extremely) many different possible solutions or derivations. As a case study, we will concentrate on derivations for physics problems as developed in Bod (2005a,b, 2006c) and Lacerda (2006), where it was shown that subtrees from derivation trees of *any* size can be important in predicting the correct derivation tree for new problems. As an example consider the following solution for the problem of deriving the Earth's mass from the Earth-Moon system as given in a classical physics textbook (Alonso and Finn 1996: 247):

Suppose that a satellite of mass  $m$  describes, with a period  $P$ , a circular orbit of radius  $r$  around a planet of mass  $M$ . The force of attraction between the planet and the satellite is  $F = GMm/r^2$ . This force must be equal to  $m$  times the centripetal acceleration  $v^2/r = 4\pi^2r/P^2$  of the satellite. Thus,

$$4\pi^2 mr/P^2 = GMm/r^2$$

Canceling the common factor  $m$  and solving for  $M$  gives

$$M = 4\pi^2 r^3/GP^2.$$

This rather textual derivation can be formalized by means of a proof tree or derivation tree in figure 7 (which is often represented in an upside-down fashion compared to linguistic or musical trees – see Baader and Nipkow 1998 or Russell and Norvig 2002).

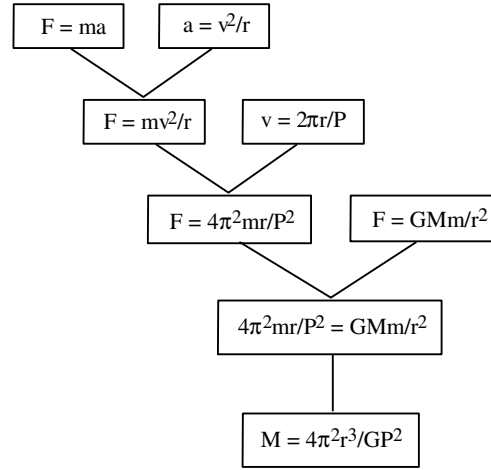


Figure 7. Derivation tree for the Earth's mass

Derivation trees form the standard representations in models like Explanation-Based Learning (Mitchell et al. 1986), Inductive Logic Programming (Cussens 2001), Stochastic Logic Programs (Muggleton 1996), Statistical Relational Learning (Mooney and Zelle 1994) and Case-Based Reasoning (Carbonell 1993), and are also used in models of human problem-solving (cf. Makachev et al. 2004). Moreover, a derivation tree corresponds to the notion of a *deductive-nomological* (D-N) explanation in philosophy of science (Hempel and Oppenheim 1948) which was improved by Kitcher (1989) in his “unificationist account” of explanation.

In Bod (2005a, 2006c) it was shown how derivation trees for new problems or theorems can be constructed out of derivation trees of previous problems by means of a new DOP model which is extended with a mathematical component that can solve an equation for a certain variable (such as *Mathematica* or *TKSolver*). Moreover, it was shown in Bod (2005a, 2006c) that subtrees of arbitrary size are important for predicting the derivation tree assigned by humans (i.e. by 19 third-year physics students).

Thus while (scientific) problem-solving is not entirely equivalent to linguistic or musical parsing (the substitution operation “o” has a somewhat different meaning in that it substitutes a formula in a certain variable rather than in a whole node – see Bod 2005a for the definition), it is fascinating that we can use the same underlying formalism, i.e. an implementation of a stochastic tree-substitution grammar (STSG), for these three modalities (see Bod 2006c,d). This raises the question as to whether we can also create an *unsupervised* learning model based on U-DOP for problem-solving.

Previous approaches to statistical learning of reasoning and problem-solving were limited in that they are based on stochastic enrichments of context-free grammars (CFGs)

or definite-clause grammars (DCUs). These approaches, known for instance as Stochastic Logic Programs or Statistical Relational Learning (De Raedt and Kersting 2004; Muggleton 1996; Cussens 2001) cannot cover all possible dependencies in a proof tree. In Bod (2006c) it was shown that, as with language and music, there may be arbitrarily distant dependencies in problem-solving, both structurally and sequentially. The examples discussed in Bod (2006c) came from the field of fluid mechanics. But distant dependencies already occur in derivations for much simpler systems, such as Galileo's pendulum (between leaf nodes and formulas later in the derivation tree). Entire subtrees must be preserved, otherwise they cannot be reapplied for solving new problems (since one loses the particular dependency). It is well-known that students of physics typically have to go through various example-derivations ("exemplars") before they can successfully solve new problems by themselves, usually by modeling the new problem on similar, previously solved problems (see Kuhn 1970; Giere 1999; Bod 2006c).

In order to take into account partial derivations of any size, it seems likely that we need an STSG for modeling the unsupervised learning of problem-solving rather than stochastic extensions of CFGs or DCUs. We will extend the EM training techniques proposed in e.g. De Raedt and Kersting (2004) to STSGs and next test whether U-DOP's all-subtrees approach can also be used for derivational reasoning. We will initially use the data-sets developed in Bod (2005a) for our evaluation. These data-sets were extracted from the widely used textbook of Alonso and Finn (1996) and have also been used by Lacerda (2006). Note that we will not try to induce the laws from classical physics, but the reasoning and problem-solving process with such laws (see workplan for more details). As before, we will assign all possible, mathematically consistent trees to a set of given theorems and next select those trees – and subtrees thereof – that could actually be used to solve new problems.

Note that (U-)DOP is congenial to Explanation-Based Learning or EBL (Mitchell et al. 1986) which also allows for reusing large derivational chunks. But whereas EBL needs at least some full-fledged explanations to begin with, we aim at bootstrapping these first explanations (i.e. derivation trees) as well -- see workplan.

## **Originality**

It has become clear that (semi-)supervised approaches to language, music and other domains are reaching a limit: they are dependent on *hand*-annotated data which are available for few languages and few domains only. Yet, annotated data do improve NLP applications, as shown by the overview paper by Lease et al. (2006), as well as musical applications (Honingh 2006b). A key issue is therefore the development of unsupervised learning models that can bootstrap annotations from unlabeled raw data. Until quite recently all unsupervised models limited either the sequential or the structural context that is taken into account, or both (e.g. van Zaanen 2000; Clark 2001; Klein and Manning 2002, 2004; Klein 2005). Moreover, these models were tested on short input and small data-sets only. The current project is timely and novel as it aims to develop efficient algorithms for one of the richest unsupervised learning technique possible, which uses *all* subtrees from *all* possible trees, thus limiting neither the sequential nor the structural context. We believe that our project is especially timely since first experiments with U-DOP have already obtained excellent results on benchmarks. But to make the unsupervised all-subtrees approach widely useful for NLP and other modalities, we must

develop efficient algorithms for U-DOP and show that they can be employed in concrete applications in language (machine translation and speech recognition), music (pitch spelling, modulation prediction) and problem-solving (solving physics problems).

### **Significance**

If successful, this research may show that insights and techniques from computational linguistics, in particular statistical NLP and DOP, are much more widely applicable than previously thought: not only can they be generalized to unsupervised learning of syntactic structures for speech and translation but also to music and reasoning, resulting in a general unsupervised machine learning model for different cognitive modalities. Our results will also be important to the Machine Learning community, especially if we can show that unsupervised learning competes with supervised methods. Our results will for instance be relevant to Memory-Based Learning (Daelemans and van den Bosch 2005).

Furthermore, our results may make the time-consuming annotation of data-sets, and perhaps even the laborious development of grammars, superfluous. It may very well turn out that massively unsupervised learning from large amounts of data provides the next major step in improving concrete applications in language, music and problem-solving. Since most models have reached an asymptote in these fields, new directions should be tried out.

Our research may shed new light on the problem known as the “poverty of the stimulus”. There is still a fierce debate whether a universal, innate grammar is needed to learn a language or whether all structure can be learned from data alone. U-DOP may be seen as a concrete implementation of constructionist and usage-based approaches to language that seem to favor the view that all structure can be statistically inferred from data. In order to show that U-DOP can actually learn full-fledged language production and comprehension, we need to extend the experiments to the very large (task-based) domains proposed in this project.

### **2b. Approach**

As explained above, we propose to use an all-subtrees methodology to unsupervised learning. We assign all trees to initial data and let the statistics decide which (sub)trees are most useful in analyzing fresh data. This methodology has already been successful in *supervised* analysis of language, music and problem-solving, and the time has come to explore its usefulness to *unsupervised* learning. First experiments with this U-DOP model show promising results and warrant further research (Bod 2006a,b).

We will start out by using the relative frequencies of subtrees which will next be iteratively reestimated by an instantiation of the expectation-maximization (EM) algorithm until the cross-entropy becomes negligible (see Bod 2006b). Cross-validation will be used to avoid overlearning. The main reason to use an all-subtrees approach is its high context-sensitivity: it has been shown at various places that *larger subtrees lead to better results* (Bod 2001; Collins and Duffy 2001; Hearne and Way 2003). To create efficient algorithms for U-DOP we will investigate PCFG-reductions that operate directly on the parse forest of utterances rather than on the resulting DOP grammar that can be extracted from the parse forest (see workplan for details).

We will evaluate U-DOP's induced trees both on hand-annotated data, as in Klein and Manning (2002, 2004) and Bod (2006a,b), and as a component in concrete applications, including machine translation, speech recognition, musical pitch spelling

and physics problem solving. For the evaluation on hand-annotated data we will have to binarize all parse trees otherwise the precision and recall measures are (almost) meaningless (see Klein 2005). Given the problems with comparing unsupervised to supervised models on *pre*-annotated data, we will also compare the two paradigms in concrete wide-coverage applications. That is, we will isolate the contribution of the various methods in improving the accuracy of these applications, by measuring the difference in accuracy of such applications under different parsing methods. We believe that such a task-based evaluation is important since it does not assess the internal annotated tree structures but the accuracy of the output only (e.g. the transcribed acoustic utterance or the translated sentence or the spelled pitches). To make our system comparison meaningful for the scientific community at large, we will use the standard metrics (i.e. Word Error Rate or WER, Perplexity, BLEU score, pitch spelling score).

## **2c. Innovation of the approach**

The main innovation of the approach is the *all-subtrees methodology* for *unsupervised* learning, applied to *different modalities*. Despite the widely recognized advantages of an all-subtrees approach (not only within the DOP framework but also in tree-kernels and other approaches – see e.g. Kudo et al. 2005; Collins and Duffy 2002; Henderson et al. 2005; Moschitti et al. 2006), such an approach has to the best of our knowledge never been generalized to wide-coverage unsupervised learning.

Another important innovation of our approach is the use of concrete applications for comparing unsupervised systems against supervised ones, so as to abstract from the theory-dependent tree annotations. Our approach of testing across several modalities is also novel.

## **2d. Plan of work and description of the subprojects**

*NB: This workplan has undergone various changes, and should be taken only as indicative.*

There are three general subprojects in this proposal: unsupervised NLP (three-four researchers), unsupervised music processing (one researcher) and unsupervised problem solving (one researcher). As already mentioned above, most of the research effort will go into the first subproject. Each subproject is divided into workpackages (WPs). Sufficient time will be reserved at the end of each WP to write and submit papers for conferences and journals.

### **Subproject 1. Unsupervised Natural Language Processing**

- 1.1 Postdoc: Unsupervised Data-Oriented Parsing (**U-DOP**)
- 1.2 PhD student: Unsupervised Data-Oriented Translation (**U-DOT**)
- 1.3. PhD student: Unsupervised Structural Language Models for Speech (**U-DOS**)

### **Subproject 2. Unsupervised Music Processing**

Postdoc: Integrating U-DOP with Music Analysis (**U-DOM**)

### **Subproject 3. Unsupervised Reasoning and Problem-Solving**

Postdoc/PhD student: Integrating U-DOP with Reasoning (**U-DOR**)

**Overall Supervision:** *Is there one unsupervised learning algorithm for different modalities? Can Unsupervised Learning outperform Supervised Learning in Concrete Applications?*

## **Subproject 1. Unsupervised Natural Language Processing**

### **Sub-subproject 1.1: Postdoc: Unsupervised Data-Oriented Parsing (U-DOP)**

#### **WP1.1.1: Literature review and familiarization with in-house U-DOP system**

#### **WP1.1.2: Develop efficient algorithms for U-DOP**

The current U-DOP system is based on a rather simple algorithm: it first assigns all binary trees to each sentence (efficiently stored in a chart), and next creates a PCFG reduction of a (large) random *subset* from these binary trees (which is trained by EM in Bod 2006b). It would be advantageous for various reasons (accuracy, efficiency, reproducibility) to create a PCFG reduction *directly* for the *entire* parse forest of all binary trees. This is the topic of the current workpackage. Instead of creating a PCFG reduction for a *list* of trees, we wish to develop a PCFG reduction for a *hypergraph* of trees in the vein of Klein and Manning (2001) and Huang and Chiang (2005).

This workpackage is one of the most significant parts of the research project. We believe that chances of success are high, since efficient algorithms for decoding parse forests have in part already been developed and published at recent conferences (Huang and Chiang 2005; May and Knight 2006). In their NAACL paper, May and Knight (2006) even perform a set of first experiments of their technique with a subset of DOP's subtrees. The extension of their technique to U-DOP seems to be rather straightforward, given that the only difference is U-DOP's use of a full chart containing all binary trees.

#### **WP1.1.3: Implement efficient algorithms for U-DOP**

#### **WP1.1.4: Evaluate U-DOP on unrestricted word strings from WSJ and NANC**

Both the old U-DOP algorithms (Bod 2006a,b) and new ones developed in WP1.1.2 will be tested on word strings from the WSJ and NANC corpora, first by extending U-DOP with the unsupervised part-of-speech tagger developed by Clark (2000) and Schuetze (1995), that induce lexical categories by distribution clustering, and second by also abstracting from any lexical category assignment (though U-DOP might induce them later in WP1.1.5). (At this stage the researchers in the other subprojects may directly use the U-DOP algorithms and software.)

#### **WP1.1.5: Bootstrap syntactic categories**

U-DOP will be extended to bootstrapping syntactic categories by randomly assigning categories  $X_1$ ,  $X_2$ ,  $X_3$ ... to all possible trees which are next trained by Expectation-Maximization (as in Bod 2006b) on an additional part of the respective corpora. We already performed preliminary experiments with this approach using two categories only, which led to a clear distinction between NPs and VPs. Category induction is important as it has been shown at various places that trees with categories outperform trees without categories for many applications.

#### **WP1.1.6: Bootstrap verb-argument structures with U-DOP**

There is an interesting research question whether we can apply the same strategy for learning predicate-argument structures. In this WP we investigate the success of assigning randomly all possible verb-argument structures to U-DOP's induced trees (and their syntactic categories) and to next test which of these assignments emerge as the best ones by maximizing the likelihood on a held-out data set. There has been some previous successful probabilistic learning of verb-argument structures (Gildea 2002).

#### **WP1.1.7: Extend U-DOP with active learning and try to integrate U-DOP and DOP by self-training**

We will investigate extensions of U-DOP/DOP towards active learning, sample selection and co-training (Steedman et al. 2003), the latter on the multilingual Europarl corpus. We will develop entropy measures to predict novel sentences that are likely to contain new constructions with respect to the corpus. Active learning usually leads to a more efficient way of handling very large data sets. We will also try out to use the reranking method proposed in Bod (2003) in combination with self-training, i.e. where a parser is trained on the trees provided by that same parser (cf. McClosky et al. 2006). This workpackage will be important as a backoff if it turns out that previous WPs do not lead to substantial accuracy improvement.

#### **WP1.1.8: Collaborate with PhD students to use full-fledged U-DOP for machine translation, speech recognition and music processing (U-DOT, U-DOS and U-DOM)**

### **Sub-subproject 1.2: PhD student: Unsupervised Data-Oriented Translation (U-DOT)**

The Data-Oriented Translation (DOT) model, which uses DOP for statistical machine translation, was first developed by Poutsma (2000) and subsequently extended by Hearne and Way (2003, 2006). DOT uses DOP's all-subtrees idea to exploit bilingually aligned treebanks for machine translation. It has led to the best published results on Xerox PARC's HomeCentre corpus (Hearne and Way 2006). However, the HomeCentre corpus is extremely small compared to the Europarl corpus currently used by most MT systems (see Koehn 2005). One goal of this sub-subproject is to exploit the trees that will be automatically learned by U-DOP for the Europarl languages. This will be especially important for languages for which currently no accurate parsers are available.

DOT is part of what has been called statistical machine translation (SMT) which has led to considerable progress in the field of MT (see the various workshops at recent ACL/EACL/NAACL/COLING conferences). The time seems to have come to try out for the first time an all-subtrees approach to *large-scale* SMT on the Europarl. Since this project uses some of the algorithms developed in the previous sub-subproject by the postdoc, we propose to start with this project one year later.

#### **WP1.2.1: Literature review and familiarization with in-house U-DOP and DOT systems**

#### **WP1.2.2: Induce tree structures for the Europarl corpus with U-DOP**

### **WP1.2.3: Develop unsupervised method for automatically aligning subsentential fragments**

The method described by Groves et al. (2004) will be used as a starting point to automatically assign alignments between subtrees for the structures induced for the Europarl corpus. Groves et al.'s method can induce alignments once the trees are available.

### **WP1.2.4: Extend U-DOP induction algorithm to integrate with DOT**

(U-)DOT uses paired (or aligned) subtrees from two parallel languages to compute the most probable translation of a target sentence given a source sentence. It will be necessary to extend the algorithm developed in WP1.1.2 to parse with paired subtrees. We will also incorporate the efficiency improvement described in (Huang and Chiang 2005).

### **WP1.2.5: Evaluate U-DOT on Europarl**

We will employ the BLEU evaluation metric to quantitatively compare the Unsupervised and Supervised DOT systems. Of course, we can only make such a comparison for the languages for which supervised parsers exist (e.g. English, German), but we will compare U-DOT against other unsupervised systems for more languages (cf. Koehn 2005), and against other syntax-based SMT systems, in particular Chiang (2005).

### **WP1.2.6: Extend U-DOT with EM training and evaluate on Europarl**

One of the recent insights of U-DOP is that by iterative reestimation based on expectation-maximization (EM) the performance is considerably improved (Bod 2006b). It is therefore important to develop and test an EM-training algorithm for subtree-pairs.

### **WP1.2.7: Extend U-DOT with bootstrapped verb-argument structures and evaluate on Europarl**

Investigate whether the bootstrapped verb-argument structures in WP1.1.6 can improve translation quality. It is well-known that verb-argument structures are important for correct translations (e.g. for distinguishing the subject and object in the translation pair *John misses Mary*  $\Leftrightarrow$  *Mary manque a John*). At this stage we will also test U-DOT using the trees predicted by the unsupervised version of *TIG-DOP* developed in sub-subproject 1.3 (see below). If successful, we believe U-DOP/U-TIG-DOP may overcome the extremely time-consuming annotation of corpora by hand.

### **WP1.2.8: Write up PhD thesis**

## **Sub-subproject 1.3: PhD student: Unsupervised Structural Language Models for Speech-Recognition (U-DOS)**

DOP has been used as a language model for speech recognition in Bod (1998, 1999) and Sima'an (1999). A language model is a crucial component of any speech recognizer and assigns scores to the candidate output strings of the speech recognizer. The idea to use tree structures in developing language models is known as "structural language models" (cf. Chelba and Jelinek 1998). It is widely felt that structural tree-based models are



among the most promising language models (see Charniak and Goodman 2002). However, while *all-subtrees* approaches have been applied to parsing, they have surprisingly never been used to boost structural language models for speech on large benchmarks such as the Switchboard speech data – to the best of our knowledge.

#### **WP1.3.1: Literature review and familiarization with in-house (U-)DOP system**

#### **WP1.3.2: Create incremental version of DOP that can compute probabilities of substrings**

To make DOP adequate for speech, it is convenient to create an incremental parser for DOP by employing the extension of DOP created by Hoogweg (2003) which uses an additional combination operation between subtrees known as “insertion” (i.e. a limited form of the adjunction operation from tree-adjoining grammar, still resulting in cubic time processing). This effectively leads to a stochastic tree-insertion grammar, known as TIG-DOP. The insertion operation is important as it is well-known that a simple TSG cannot be fully lexicalized while maintaining the same strong generative capacity.

#### **WP1.3.3: Develop efficient algorithm to compute most probable (sub)string by TIG-DOP**

While the problem of computing the most probable string is NP-hard (Sima'an 1999), there exists an efficient Viterbi  $n$ -best method to estimate the most probable string from among the  $n$  most probable derivations. Moreover, the most probable string can also be estimated by its most probable derivation for which efficient algorithms do exist.

#### **WP1.3.4: Use U-DOP for inducing trees for Switchboard sentences**

We will use the Switchboard data set so that we can compare our results with (one of) the best published results in language modeling by Roark et al. (2006). We will of course only induce trees for the training set (roughly 277.000 transcribed utterances). But we will also compare U-DOP's language model against DOP's by using the (fewer) Switchboard annotations in the Penn Treebank.

#### **WP1.3.5: Use induced trees to train TIG-DOP resulting into unsupervised language model U-DOS**

#### **WP1.3.6: Test U-DOS on Switchboard speech data (and on WSJ)**

U-DOS will be tested on the 20854 utterances of Switchboard test set also used in Roark et al. (2006). Note that both our model and that of Roark et al. are unsupervised (though our model induces trees rather than  $n$ -grams). We will therefore also compare U-DOS against Chelba and Jelinek's (1998) structured language model and against DOP.

#### **WP1.3.7: Investigate extensions of U-DOP towards using insertion: “U-TIG-DOP”**

For a flying start, this subproject directly used U-DOP for the induction of trees. In this WP we will create unsupervised generalizations of TIG-DOP rather than of DOP. Experimental investigation will be carried out on Switchboard and WSJ, but also on the Europarl together with WP 1.2.7 (see above).

### **WP1.3.8: Write up PhD thesis**

## **Subproject 2. Unsupervised Music Processing**

### **Postdoc: Integrating U-DOP with Music Analysis (U-DOM)**

#### **WP2.1: Literature review and familiarization with in-house U-DOP system**

#### **WP2.2: Use U-DOP to induce melodic structures for Essen Folksong Collection (EFC): *U-DOM***

#### **WP2.3: Evaluate *U-DOM*'s application to melodic analysis**

Evaluation will be carried out by n-fold testing/cross-validation and U-DOP will be compared against DOP's supervised parser in Bod (2002a,b), other unsupervised approaches to music (e.g. Ferrand et al. 2003), and rule-based melodic parsers such as Temperley (2001). We will try to integrate our accuracy metrics with some recent ideas about music evaluation as well (see H-J. Honing 2006).

#### **WP2.4: Integrate U-DOM with *convexity* and *compactness***

Similar to the preference for the "simplest" structure in linguistic parsing proposed in Bod (2000b), Honingh and Bod (2005) showed that there is also a preference for simplicity in music: if we represent the frequency ratios of pitches in a two dimensional space, then there is a very strong tendency that scales, chords, harmonic reductions and virtually any other musical items (from all over the world) form *convex* and *compact* structures. This is one of the most robust outcomes of A. Honingh's thesis. The preference for the most compact and convex structure can be easily combined with U-DOP and DOP: instead of computing the most probable melodic structure (of e.g. a folksong), we might compute the most probable structure from the most compact, convex structures. We will also try out a reranking method by selecting the most compact, convex structure from among the  $n$  most probable structures.

#### **WP2.5: Use the induced melodic trees for improving modulation prediction and pitch spelling**

This WP builds on Honingh (2006b), where convexity and compactness are used as systems that predict modulations in musical pieces and the spelling of pitches (the problem of assigning note names to specific pitches). Most of the errors in Honingh's application are due to insufficient knowledge of (melodic) phrase boundaries (Honingh 2006b). This WP will investigate the contribution of U-DOM (and other unsupervised melodic parsers) against supervised melodic parsers in automatic pitch spelling and modulation prediction. We will use the very large data sets in Meredith (2006) that have been used to test the various pitch spelling systems in the literature.

#### **WP2.6: Bootstrap harmonic structure (and investigate semi-supervised methods)**

Analogous to U-DOP with respect to inducing verb-argument structure, there is also a similar question as to how far U-DOM stretches. That is, can we apply the same approach

to learning harmonic structure in an unsupervised way? To the best of our knowledge, there is no previous work about bootstrapping harmony in an entirely unsupervised way, and we will therefore also try out semi-unsupervised methods (as described in WP1.1.7) if the entirely unsupervised approach does not generate success.

#### **WP2.7: Bootstrap metrical structure**

Analogous to WP 2.7 (cf. Desain and Honing 1995; Van Zaanen et al. 2003).

#### **WP2.8: Test the resulting, full-fledged U-DOM on the pitch spelling application**

#### **WP2.9: Try to integrate results on TIG-DOP with music**

This WP will use TIG-DOP in sub-subproject 1.3 to the various tasks. It would be fascinating to explore whether the insertion operation might also be beneficial to music.

### **Subproject 3. Unsupervised Reasoning and Problem-Solving**

#### **Postdoc/PhD student: Integrating U-DOP with Reasoning in Problem-Solving (U-DOR)**

This subproject explores stochastic reasoning models that go beyond the commonly used models of Stochastic Logic Programs, Statistical Relational Learning and Probabilistic Logic Learning which are mainly based on simple probabilistic CFGs or DCUs. (NB: since we will use an additional mathematical component and take the laws of classical physics as given, this subproject can only be called “unsupervised” if it refers to the problem-solving process of creating derivation trees. As a model for classical physics in general, it would only be *semi*-unsupervised.)

#### **WP3.1: Extend U-DOP to induce derivational trees for physics problems in Bod (2005a, 2006c)**

The DOP model for problem-solving in Bod (2006b) will be extended to unsupervised reasoning (U-DOR). We will *assign all mathematically correct trees to physics problems and use all subtrees to predict the best trees for new problems* (taken from Bod 2005a), just as with unsupervised DOP models for language and music. Note that U-DOR is extended with a mathematical component (*TKSolver*, release 5.0, also used in Bod 2005a, 2006c).

#### **WP3.2: Evaluate U-DOR on physics test corpus (Bod 2006c,d) and compare with Stochastic Logic Programming and Statistical Relational Learning**

Once the extended U-DOP model can induce new derivational structures, we can compare the resulting U-DOR model against the manually assigned derivational trees in Bod (2006d; 2005a) and test them against other stochastic reasoning approaches (Cussens 2001; De Raedt and Kersting 2004). We will investigate whether we actually need large subtrees in unsupervised reasoning or whether we can do the same job with constrained subtree-sets or by probabilistic versions of DCUs.

### **WP3.3: Investigate EM training of STSGs for U-DOR and compare against EM training for DCUs and PCFGs**

### **WP3.4: Explore U-DOR's predictive power for the problems in Alonso and Finn (1996)**

This consists of about 300 exemplary problem solutions (“exemplars”), most of which build on previous problem solutions. Although this data-set is small compared to the data-sets used in NLP and music, humans appear to use much fewer exemplars in physics reasoning than in e.g. language processing. Giere (1999) estimates that the number of exemplars that physics-experts have readily access to is in the order of a few hundreds to maximally a couple of thousands. A corpus of 300 problem solutions may thus not be far from the number of readily available exemplars of experienced students – while the number of exemplars (e.g. fixed phrases and collocations) in natural language is estimated as hundreds of thousands of items (Erman and Warren 2000) and in music cognition as several thousands (Juhasz 2004).

### **WP3.5: Investigate usefulness of integrating U-DOR with Shortest Derivation and TIG-DOP**

The insertion of subtrees may be especially helpful for describing dependencies between initial conditions and approximations occurring later in the tree whilst adding any kind of other subtree in between by the insertion operation (see section 2.3.2 above for more details).

## **2e. Literature references**

- M. Alonso and E. Finn. 1996. *Physics*, Addison-Wesley.
- F. Baader and T. Nipkow, 1998. *Term Rewriting and All That*, Cambridge University Press.
- W. Bechtel and G. Graham, 1999. *A Companion to Cognitive Science*, Blackwell.
- R. Bod, 1998. *Beyond Grammar: An Experience-Based Theory of Language*, Stanford: CSLI Publications (Lecture notes number 88), distributed by Cambridge University Press.
- R. Bod, 1999. Context-Sensitive Spoken Dialogue Processing with the DOP Model. *Natural Language Engineering* 5(4), 309-323.
- R. Bod, 2000a. Parsing with the Shortest Derivation. *Proceedings COLING-2000*, Saarbrücken, Germany.
- R. Bod, 2000b. Combining semantic and syntactic structure for language modeling. *Proceedings ICSLP 2000*.
- R. Bod, 2001. What is the minimal set of fragments that achieves maximal parse accuracy? *Proceedings ACL 2001*.
- R. Bod, 2002a. Memory-Based Models of Melodic Analysis: Challenging the Gestalt Principles. *Journal of New Music Research*, 31(1), 27-37.
- R. Bod, 2002b. A unified model of structural organization in language and music, *Journal of Artificial Intelligence Research*, 17(2002), 289-308.
- R. Bod, 2003. An efficient implementation for a new DOP model. *Proceedings EACL'03*, Budapest.
- R. Bod, 2005a. Modeling Scientific Problem-Solving by DOP. *Proceedings CogSci'05*, Stresa, Italy.
- R. Bod, 2005b. Exemplar-Based Explanation. In L. Magnani and R. Dossena (eds.), *Computing, Philosophy and Cognition*. Kluwer, 329-348.
- R. Bod, 2006a. Unsupervised Parsing with U-DOP. *Proceedings CONLL 2006*, New York.
- R. Bod, 2006b. An All-Subtrees Approach to Unsupervised Parsing. *Proceedings ACL-COLING 2006*, Sydney.
- R. Bod, 2006c. Towards a General Model of Applying Science. *International Studies in the Philosophy of Science* 20(1), 5-25. (*Symposium on Applying Science*)
- R. Bod, 2006d. The Data-Oriented Paradigm and its Application. In J. Fulcher and L. Jain (eds.), *Handbook of Computational Intelligence*, Springer Verlag. (in press)
- R. Bod, 2006e. Exemplar-Based Syntax: How to Get Productivity from Examples. *The Linguistic Review* 23(3), *Special Issue of Exemplar-Based Models in Linguistics*, 289-318.
- R. Bod and R. Kaplan, 1998. A Probabilistic Corpus-Driven Model for Lexical-Functional Analysis. *Proceedings ACL-COLING'98*, Montreal, Canada, 145-152.

- R. Bod and R. Kaplan, 2003. A DOP Model for Lexical-Functional Representations. In R. Bod, R. Scha and K. Sima'an (eds.), *Data-Oriented Parsing*, The University of Chicago Press.
- R. Bod, R. Scha and K. Sima'an (eds.) 2003. *Data-Oriented Parsing*, The University of Chicago Press.
- R. Bod and R. Scha, 2004. Data-Oriented Parsing: An Overview. In Sampson and McCarthy (eds.), 2004, *Corpus Linguistics*, 304-325.
- R. Bonnema, R. Bod and R. Scha, 1997. A DOP Model for Semantic Interpretation, *Proceedings ACL/EACL-97*, Madrid, Spain.
- J. Bybee, 2006. From usage to grammar: the mind's response to repetition. *Language* 82, to appear.
- J. Carbonell, 1993. Derivational Analogy: A Theory of Reconstructive Problem Solving and Expertise Acquisition. In Michalski et al. (eds.), *Machine Learning*, Vol. II, Morgan Kaufmann, 371-392.
- E. Charniak, 1997. Statistical Techniques for Natural Language Parsing, *AI Magazine*, Winter 1997, 32-43.
- E. Charniak and J. Goodman 2002. The State of the Art in Language Modeling. Tutorial Presented at AAAI 2002, Edmonton.
- E. Charniak and M. Johnson, 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. *Proceedings ACL 2005*.
- C. Chelba and F. Jelinek 1998. Exploiting Syntactic Structure for Language Modeling. *Proceedings ALC-COLING 1998*, Montreal.
- D. Chiang, 2000. Statistical parsing with an automatically extracted tree adjoining grammar, *Proceedings ACL'2000*, Hong Kong, China.
- D. Chiang, 2005. A hierarchical phrase-based model for statistical machine translation. *Proceedings ACL 2005*.
- N. Chomsky, 1965. *Aspects of the Theory of Syntax*, Cambridge (Mass.), The MIT Press.
- A. Clark, 2000. Inducing Syntactic Categories by Context Distribution Clustering, *Proceedings CoNLL 2000*.
- A. Clark, 2001. Unsupervised induction of stochastic context-free grammars using distributional clustering. *Proceedings CoNLL 2001*.
- M. Collins, 2000. Discriminative Reranking for Natural Language Parsing. *Proceedings ICML 2000*.
- M. Collins and N. Duffy, 2001. Convolution Kernels for Natural Language. *Proceedings NIPS*, Vancouver.
- M. Collins and N. Duffy, 2002. New Ranking Algorithms for Parsing and Tagging: Kernels over Discrete Structures, and the Voted Perceptron. *Proceedings ACL'2002*, Philadelphia, PA.
- B. Cormons, 1999. Efficient Monte Carlo Sampling for LFG-DOP. PhD thesis, University of Rennes, France.
- B. Croft. *Radical Construction Grammar*. Oxford University Press.
- J. Cussens, 2001. Parameter estimation in stochastic logic programs. *Machine Learning*, 44(3):245-271.
- W. Daelemans and A. van den Bosch, 2005. *Memory-Based Language Processing*. Cambridge Univ. Press.
- L. De Raedt and K. Kersting 2004. Probabilistic Inductive Logic Programming. *Proceedings Algorithmic Learning Theory (ALT) 2004*.
- P. Desain and H. Honing, 1995. Computational Models of Beat Induction: the Rule-Based Approach. *Artificial Intelligence and Music*, G. Widmer (ed.), Montreal, IJCAI.
- A. Dubey, 2004. *Statistical Parsing for German: Modeling syntactic properties and annotation differences*. PhD thesis, Saarland University, Germany.
- B. Erman, Britt and B. Warren, 2000. The idiom principle and the open choice principle. *Text* 20(1), 29-62.
- B. Falkenhainer, K. Forbus and D. Gentner 1989. The Structure-Mapping Engine: Algorithm and Examples. *Artificial Intelligence*, 41, 1-63.
- M. Ferrand, P. Nelson, G. Wiggins, 2003. Unsupervised Learning of Melodic Segmentation: A Memory-based Approach - *Proceedings of 5th Triennial Conference of the European Society for the Cognitive Sciences of Music (ESCOM5)*.
- S. Gahl and S. Garnsey, 2004. Knowledge of grammar, knowledge of usage: syntactic probabilities affect pronunciation variation. *Language* 80(4), 748-775.
- R. Giere, 1999. *Science without Laws*, The University of Chicago Press.
- D. Gildea, 2002. Probabilistic Models of Verb-Argument Structure. *Proceedings COLING 2002*, Taipei.
- A. Goldberg, 2006. *Constructions at Work: the nature of generalization in language*. Oxford University Press.
- J. Goodman, 1996. Efficient Algorithms for Parsing the DOP Model, *Proceedings Empirical Methods in Natural Language Processing*, Philadelphia, PA.
- J. Goodman, 2003. Efficient Algorithms for the DOP Model, In R. Bod, R. Scha and K. Sima'an (eds.), *Data-Oriented Parsing*, The University of Chicago Press.
- D. Groves, M. Hearne and A. Way, 2004. Robust sub-sentential alignment of phrase-structure trees. *Proceedings COLING 2004*, Geneva.
- M. Hearne and A. Way, 2003. Seeing the Wood for the Trees: Data-Oriented Translation. *Proceedings of MT Summit IX*, New Orleans.
- M. Hearne and A. Way, 2004. Data-Oriented Parsing and the Chinese Penn Treebank. *Proceedings of The First International Joint Conference on Natural Language Processing, March 2004, Hainan Island, China*.
- M. Hearne and A. Way, 2006. Disambiguation strategies for data-oriented translation. In *Proceedings of the 11th Conference of the European Association for Machine Translation*, Oslo.
- C. Hempel and P. Oppenheim, 1948. Studies in the Logic of Explanation, *Philosophy of Science*, 15: 135-175.

- J. Henderson and I. Titov, 2005. Data-defined kernels for parse reranking derived from probabilistic models. *Proceedings ACL 2005*, Ann Arbor.
- H. Honing 2006. The role of surprise in theory testing: Some preliminary observations. *Proceedings of the International Conference on Music Perception and Cognition* (pp. 38-42). Bologna: Italy
- A. Honingh and R. Bod 2005. Convexity and the Well-Formedness of Musical Objects. *Journal of New Music Research* 34(3), 293-303.
- A. Honingh, 2006a. Convexity and Compactness as Models for the Preferred Intonation of Chords. *Proceedings ICMPC 2006* Bologna.
- A. Honingh, 2006b. *The Origin and Well-Formedness of Tonal Pitch Structures*. PhD thesis, University of Amsterdam (to be defended on 20 October 2006).
- L. Hoogweg, 2003. Extending DOP with Insertion. In R. Bod, R. Scha and K. Sima'an (eds.). *Data-Oriented Parsing*, The University of Chicago Press.
- L. Huang and D. Chiang 2005. Better  $k$ -best parsing. *Proceedings IWPT 2005*, Vancouver.
- Z. Juhász, 2004. Segmentation of Hungarian Folk Songs Using an Entropy-Based Learning System. *Journal of New Music Research*, 33(1), 5-15.
- D. Jurafsky, 2003. Probabilistic Modeling in Psycholinguistics: Comprehension and Production. In R. Bod, J. Hay and S. Jannedy (eds.), *Probabilistic Linguistics*, The MIT Press.
- P. Kitcher, 1989. Explanatory unification and the causal structure of the world", in Kitcher, Philip, and Salmon, Wesley (eds.), *Scientific Explanation*, University of Minnesota Press, 410-505.
- D. Klein, 2005. *The Unsupervised Learning of Natural Language Structure*. PhD thesis, Stanford University.
- D. Klein and C. Manning, 2001. Parsing and Hypergraphs. *Proceedings of the 7th International Workshop on Parsing Technologies (IWPT-2001)*.
- D. Klein and C. Manning 2002. A general constituent-context model for improved grammar induction. *Proceedings ACL 2002*, Philadelphia.
- D. Klein, and C. Manning 2004. Corpus-based induction of syntactic structure: models of dependency and constituency. *Proceedings ACL 2004*, Barcelona.
- P. Koehn, 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. *Proceedings MT Summit 2005*.
- T. Kudo, J. Suzuki and H. Isozaki, 2005. Boosting-based parse reranking with subtree features. *Proceedings ACL 2005*, Ann Arbor.
- G. Lacerda, 2006. The exemplar-based approach to scientific reasoning. Manuscript under review.
- M. Lease, E. Charniak, M. Johnson, and D. McClosky, 2006. A Look at Parsing and its Applications. *Proceedings of AAAI 2006*.
- F. Lerdahl and R. Jackendoff, 1983. *A Generative Theory of Tonal Music*. Cambridge, The MIT Press.
- M. Leyton, 2001. *A Generative Theory of Shape*. Springer Verlag.
- H. Longuet-Higgins, 1976. Perception of Melodies. *Nature* 263, October 21, 646-653.
- H. Longuet-Higgins and C. Lee, 1987. The Rhythmic Interpretation of Monophonic Music. In: *Mental Processes: Studies in Cognitive Science*, Cambridge, The MIT Press.
- C. Manning and H. Schütze, 1999. *Foundations of Statistical Natural Language Processing*. Cambridge, The MIT Press.
- M. Marcus, B. Santorini and M. Marcinkiewicz, 1993. "Building a Large Annotated Corpus of English: the Penn Treebank", *Computational Linguistics* 19(2).
- J. May and K. Knight, 2006. A Better N-Best List: Practical Determinization of Weighted Finite Tree Automata, *Proceedings NAACL-HLT 2006*.
- D. McClosky, E. Charniak and M. Johnson 2006. Effective self-training for parsing. *Proceedings NAACL-HLT 2006*.
- D. Meredith, 2006. The *ps13* pitch spelling algorithm. *Journal of New Music Research*, 35(2), pp. 121-159.
- T. Mitchell, R. Keller, and S. Kedar-Cabelli, 1986. Explanation-based learning: A unifying view. *Machine Learning*, 1, 47-80.
- J. Mooney and J. Zelle. Integrating ILP and EBL. *SIGART Bulletin*, 5, 12-21.
- A. Moschitti, D. Pighin and R. Basili, 2006. Semantic role labelling via tree kernel joint inference. *Proceedings CONLL 2006*.
- S. Muggleton, 1996. Stochastic Logic Programs. *Proceedings of the 5th International Workshop on Inductive Logic Programming*.
- A. Newell, 1990. *Unified Theories of Cognition*, Harvard University Press.
- F.J. Och, 2003. Minimum Error Rate Training for Statistical Machine Translation, In *Proceedings ACL 2003*.
- A. Poutsma, 2000. Data-Oriented Translation. *Proceedings COLING 2000*, Saarbruecken.
- B. Roark, M. Saraclar and M. Collins, 2006. Discriminative n-gram language modelling. To appear in *Computer Speech and Language*.
- S. Russell and P. Norvig 1995. *Artificial Intelligence*. Prentice Hall.
- J. Saffran, Loman, M. and Robertson, R. 2000. Infant Memory for Musical Experiences. *Cognition*, 77, B16-23.
- R. Scha, 1990. Taaltheorie en Taaltechnologie; Competence en Performance, in Q. de Kort and G. Leerdam (eds), *Computertoepassingen in de Neerlandistiek*, Almere: Landelijke Vereniging van Neerlandici (LVVN-jaarboek).

- R. Scha, R. Bod & K. Sima'an 1999. A Memory-Based Model of Syntactic Analysis: Data-Oriented Parsing. *Journal of Experimental and Theoretical Artificial Intelligence*, 11(3), 409-440.
- R. Schaefer, J. Murre and R. Bod, 2004. Limits to Universality in Segmentation of Simple Melodies, *Proceedings ICMPC'8*, Evanston, IL.
- H. Schaffrath, 1995. The Essen Folksong Collection in the Humdrum Kern Format. D. Huron (ed.). Menlo Park, CA: Center for Computer Assisted Research in the Humanities.
- H. Schuetze 1995. Distributional Part-of-Speech Tagging. *Proceedings EACL 1995*.
- K. Sima'an, 1999. *Learning Efficient Disambiguation*. ILLC Dissertation Series 1999-02, Utrecht University / University of Amsterdam, The Netherlands.
- M. Steedman, R. Hwa, S. Clark, M. Osborne, A. Sarkar, J. Hockenmaier, P. Ruhn, S. Baker and J. Crim, 2003. Example Selection for Bootstrapping Statistical Parsers. *Proceedings of the Joint Conference of Human Language Technologies and the Annual Meeting of the North American Chapter of the ACL 2003*.
- D. Temperley, 2001. *The Cognition of Basic Musical Structures*. Cambridge, The MIT Press.
- M. Tomasello, 2003. *Constructing a Language*. Harvard University Press.
- K. VanLehn, 1998. Analogy Events: How Examples are Used During Problem Solving, *Cognitive Science*, 22(3), 347-388.
- M. Veloso and J. Carbonell, 1993. Derivational Analogy in PRODIGY: Automating Case Acquisition, Storage, and Utilization. *Machine Learning*, 10(3), 249-278.
- M. Wertheimer, 1923. Untersuchungen zur Lehre von der Gestalt. *Psychologische Forschung* 4, 301-350.
- M. van Zaanen, 2000. ABL: Alignment-Based Learning. *Proceedings COLING 2000*, Saarbrücken.
- M. van Zaanen, 2002. *Bootstrapping Structure into Language*. PhD thesis. University of Leeds.
- M. van Zaanen, R. Bod and H. Honing, 2003. A Memory-Based Approach to Meter Induction, *Proceedings ESCOM5*, Hanover.
- A. Zollmann and K. Sima'an 2005. A consistent and efficient estimator for data-oriented parsing. *Journal of Automata, Languages and Combinatorics* 10, 367-388.
- W. Zuidema, 2006. What are the productive units of natural language grammar? A DOP approach to the automatic identification of constructions. *Proceedings CONLL 2006*, New York.