Parsing or DOP (Bod 1998). We will generalize the supervised version of DOP to unsupervised parsing. The key idea of our approach is to initially assign all possible unlabeled binary trees to a set of given sentences, and to next use counts of all subtrees from (a large random subset of) these binary trees to compute the most probable parse trees. To the best of our knowledge, such a model has never been tried out. We will refer to this unsupervised DOP model as *U-DOP*, while the supervised DOP model (which uses hand-annotated trees) will be referred to as *S-DOP*. Moreover, we will continue to refer to the general approach simply as *DOP*.

U-DOP is not just an engineering approach to unsupervised learning but can also be motivated from a cognitive perspective (Bod 2006): if we don't have a clue which trees should be assigned to sentences in the initial stages of language acquisition, we can just as well assume that initially all trees are possible. Only those (sub)trees that partake in computing the most probable parse trees for new sentences are actually "learned". We have argued in Bod (2006) that such an integration of unsupervised and supervised methods results in an integrated model for language learning and language use.

In the following we will first explain how U-DOP works, and how it can be approximated by a PCFG-reduction technique. Next, in section 3 we discuss a number of experiments with U-DOP and compare it to previous models on English (WSJ), German (NEGRA) and Chinese (CTB) data. To the best of our knowledge, this is the first paper which bootstraps structure for WSJ sentences up to 40 words obtaining roughly the same accuracy as a binarized *supervised* PCFG. This is remarkable since unsupervised models are clearly at a disavantage compared to supervised models which can literally reuse manually annotated data.

2 Unsupervised data-oriented parsing

At a general level, U-DOP consists of the following three steps:

1. Assign all possible binary trees to a set of sentences
2. Convert the binary trees into a PCFG-reduction of DOP
3. Compute the most probable parse tree for each sentence

Note that in unsupervised parsing we do not need to split the data into a training and a test set. In this paper, we will present results both on entire corpora and on 90-10 splits of such corpora so as to make our results comparable to a *supervised* PCFG using the treebank grammars of the same data ("*S-PCFG*").

In the following we will first describe each of the three steps given above where we initially focus on inducing trees for p-o-s strings for the WSJ10 (we will deal with other corpora and the much larger WSJ40 in section 3). As shown by Klein and Manning (2002, 2004), the extension to inducing trees for words instead of p-o-s tags is rather straightforward since there exist several unsupervised part-of-speech taggers with high accuracy, which can be combined with unsupervised parsing (see e.g. Schütze 1996; Clark 2000).

Step 1: Assign all binary trees to p-o-s strings from the WSJ10

The WSJ10 contains 7422 sentences ≤ 10 words after removing empty elements and punctuation. We assigned all possible binary trees to the corresponding part-of-speech sequences of these sentences, where each root node is labeled *S* and each internal node is labeled *X*. As an example, consider the p-o-s string NNS VBD JJ NNS, which may correspond for instance to the sentence *Investors suffered heavy losses*. This string has a total of five binary trees shown in figure 1 -- where for readability we add words as well.

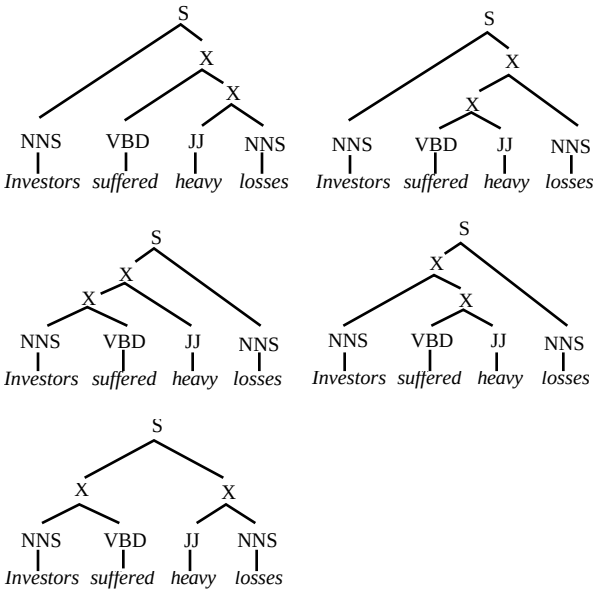

Figure 1. All binary trees for NNS VBD JJ NNS (*Investors suffered heavy losses*)