Linguistic Relevance of Unsupervised Data-Oriented Parsing

Rens Bod
Is Empiricist Language Acquisition Possible?

- Can we acquire language by constructing analogies with previous input?
- Some analogies work, some don’t:

Problems with category matching:

(Pinker 1979)  

\[
\begin{align*}
\text{John likes fish} & : \quad \text{John likes chicken} \\
\text{John eats fish} & : \quad \text{John eats chicken (OK)}
\end{align*}
\]

\[
\begin{align*}
\text{John likes fish} & : \quad \text{John likes chicken} \\
\text{John might fish} & : \quad \text{John might chicken (??)}
\end{align*}
\]

Problems with structural dependencies:

(after Chomsky 1986)  

\[
\begin{align*}
\text{Swimming in the sea is dangerous} & : \quad \text{The sea is dangerous} \\
\text{Swimming in the rivers is dangerous} & : \quad \text{The rivers is dangerous (??)}
\end{align*}
\]
Against and For Analogical Language Acquisition

• Chomsky (1966, 1986), Pinker (1979):

  “The notion of analogy is not wrong, but vacuous” (Chomsky 1986)
  Rules and principles needed for language acquisition (UG)
  Gold (1967) proved that cf-grammar induction is not possible in the limit
  But Horning (1971) proved that probabilistic grammar induction is possible

• In this talk I will demonstrate:

  Correct structure and categories can be learned by probabilistic analogy
  I will show this using a new unsupervised DOP model
  But first I will have to review the original, supervised DOP model
The Supervised Data-Oriented Parsing Model
(following Bod, Scha, Kaplan, Sima’an, Collins, Goodman, Way, Hearne, Zuidema and others)

Given a corpus of sentences annotated with trees…

1. Divide trees into all subtrees – remember them

2. Combine subtrees to produce or parse a new sentence

3. Update corpus with the new sentence

Original DOP model defined on surface phrase-structure trees (Bod 1993)
Simple Illustration of Supervised DOP
(Bod 1993, 1998)

Given an extremely simple corpus consisting of two tree structures:

```
S
  NP
  John
  V
  likes
  NP
  Mary

S
  NP
  Peter
  V
  hates
  NP
  Susan
```
Divide the trees into fragments (subtrees):

NP
Mary

NP
Susan

S
NP
VP
V
hates

S
NP
VP
V
John
likes

S
NP
VP
V
likes
Mary

etc.
Combine fragments to derive trees for new sentences

In DOP, "o" is left-most substitution
A slightly more interesting example

Given following corpus:
She saw the dress with the telescope can be derived in two ways by subtrees from the corpus:

The first derivation would be preferred if we want to maximize analogy with corpus.
How does DOP determine the ‘best’ tree?

- Shortest derivation (of fewest subtrees) maximizes overlaps with previous sentences.

- In case shortest derivation is not unique (which is the typical case) compute most probable tree from among trees proposed by shortest derivations:

  1. Maximizes structural analogy
  2. Maximizes probability

- Probability of a parse tree is computed from probabilities of all subtrees in the corpus

**DOP principle:**

*If you don’t know which subtrees are needed, take them all and let statistics decide*
Probability of...

a subtree $t$:

$$P(t) = \frac{|t|}{\sum_{t' : \text{root}(t')=\text{root}(t)} |t'|}$$

a derivation $d = t_1 \circ \ldots \circ t_n$:

$$P(t_1 \circ \ldots \circ t_n) = \prod_i P(t_i)$$

a parse tree $T$:

$$P(T) = \sum_d \prod_i P(t_{id})$$

$t_{id}$ is the $i$-th subtree in derivation $d$ that produces $T$

NB: in Bod (2006) the subtrees’ relative frequencies are re-estimated by EM
DOP models of this kind are Stochastic Tree-Substitution Grammars (STSGs)

- Formally, DOP is an STSG where the tree-units can be of arbitrary size
- By putting constraints on the tree-units, we can also instantiate:
  - stochastic context-free grammars
  - stochastic lexicalized grammars
  - stochastic tree-insertion grammars
  - stochastic regular grammars
  etc…
How can we extend DOP to language acquisition?

Generalize the DOP principle:

- If you don’t know what kind of structures should be assigned to sentences:
  
  *allow (implicitly) for all possible trees and let linguistic experience decide which is the ‘best’ tree, by most probable shortest derivation.*

- **Unsupervised DOP** *(U-DOP)*
  
  1. Assign all possible unlabeled binary trees to a set of sentences
  2. Convert the binary trees into subtrees (and into a PCFG reduction)
  3. Compute the MPSD for the sentences
How Does U-DOP Operate?

1. Assign all possible binary trees to strings where each root node is labeled $S$ and other nodes labeled $X$, and store them in a parse forest.

E.g., for WSJ sentence *Investors suffered heavy losses*:
Binary Trees are Stored in a Packed Forest

- Number of possible binary trees grows exponentially with string length
  but can be efficiently stored in a packed parse forest in quadratic space

- Elsewhere we show that a PCFG reduction exists which operates directly on shared parse forests:
  each AND-OR node in the forest will receive maximally 8 PCFG rules

- We will not go into the computational details in this talk,
  see Bod (2007, Proc. ACL’07)
2. Convert the parse forests directly into all subtrees (or, actually, into a compact PCFG reduction). For instance,

```
 Investors suffered  heavy losses  suffered heavy
        X                  X          X

 S

 Investors  losses  suffered
          X            X             X

 S

 Investors suffered  heavy losses
          X            X             X
```
3. Compute most probable tree for new string (just as in DOP):
Quantitative Evaluation

- Evaluation by $n$-fold testing using f-score (F1) of unlabeled precision (UP) and unlabeled recall (UR) to test on hand-annotated data:

$$F1 = 2*UP*UR/(UP+UR)$$

- We added an *unsupervised* part-of-speech tagger (Biemann 2006), based on graph clustering

- In evaluating U-DOP, we had to binarize test-set trees in Penn Treebank otherwise the f-scores become meaningless (as in Klein 2005; Klein and Manning 2002, 2004)
Evaluation against Hand-Annotated Data

*Penn Treebank* (Marcus et al. 1993):

50,000 manually analyzed sentences, e.g. from WSJ:
Experiments with U-DOP on entire Penn’s WSJ
(10-fold testing)

<table>
<thead>
<tr>
<th>Max. Subtree Depth</th>
<th>F-score (Wall St Journal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>28.7</td>
</tr>
<tr>
<td>2</td>
<td>40.5</td>
</tr>
<tr>
<td>3</td>
<td>58.9</td>
</tr>
<tr>
<td>4</td>
<td>66.0</td>
</tr>
<tr>
<td>6</td>
<td>68.7</td>
</tr>
<tr>
<td>8</td>
<td>68.8</td>
</tr>
<tr>
<td>10</td>
<td>70.2</td>
</tr>
<tr>
<td>16</td>
<td>70.3</td>
</tr>
</tbody>
</table>

- F-score increases with larger subtrees
- F-score is very low at depth 1 (simple treebank PCFG induction)
U-DOP compared to other models on WSJ-10
(using part-of-speech strings up to 10 tags)

<table>
<thead>
<tr>
<th>Model</th>
<th>f-score on WSJ-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCM</td>
<td>71.9</td>
</tr>
<tr>
<td>DMV</td>
<td>52.1</td>
</tr>
<tr>
<td>DMV+CCM</td>
<td>77.6</td>
</tr>
<tr>
<td>U-DOP’2006</td>
<td>78.5</td>
</tr>
<tr>
<td>U-DOP</td>
<td>82.7</td>
</tr>
</tbody>
</table>

CCM: Klein and Manning (2002)

Similar results are obtained for German and Chinese data (Bod 2006)
Why does U-DOP work?

• Repetition guides “phrase learning”? (following Bybee 2006):

   Investors suffered heavy losses
   Heavy losses were reported

• An example with discontiguous dependencies:

   Swimming in rivers is dangerous
   Swimming with other people is fun

• The next subtree gets a higher frequency by U-DOP than other subtrees:
Repetition alone is insufficient and can be harmful

- Patterns such as IN DT (e.g. *in the*) and DT JJ (e.g. *the nice*) are very frequent, but do not form constituents!

  *In the second sentence …*
  *In the first place …*
  *In the case of …*
  *In the previous section …*
  *In the field of …*
  etc

- ‘*in the*’ is a non-constituent (distituent) in all linguistic theories

- Moreover, ‘*in the*’ is **not** a multi-word unit such as *Es gibt* or *Il y a*.

- Thus U-DOP must rule it out, but does it?
Let’s have a look at the most frequently learned constituents by U-DOP:

<table>
<thead>
<tr>
<th>Rank</th>
<th>Most frequent U-DOP constituents</th>
<th>Most Frequent WSJ10 constituents</th>
<th>Most frequent WSJ10 substring</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DT NN</td>
<td>DT NN</td>
<td>NNP NNP</td>
</tr>
<tr>
<td>2</td>
<td>NNP NNP</td>
<td>NNP NNP</td>
<td>DT NN</td>
</tr>
<tr>
<td>3</td>
<td>DT JJ NN</td>
<td>CD CD</td>
<td>JJ NN</td>
</tr>
<tr>
<td>4</td>
<td>IN DT NN</td>
<td>JJ NNS</td>
<td>IN DT</td>
</tr>
<tr>
<td>5</td>
<td>CD CD</td>
<td>DT JJ NN</td>
<td>NN IN</td>
</tr>
<tr>
<td>6</td>
<td>DT NNS</td>
<td>DT NNS</td>
<td>DT JJ</td>
</tr>
<tr>
<td>7</td>
<td>JJ NNS</td>
<td>JJ NN</td>
<td>JJ NNS</td>
</tr>
<tr>
<td>8</td>
<td>JJ NN</td>
<td>CD NN</td>
<td>NN NN</td>
</tr>
<tr>
<td>9</td>
<td>VBN IN</td>
<td>IN NN</td>
<td>CD CD</td>
</tr>
<tr>
<td>10</td>
<td>VBD NNS</td>
<td>IN DT NN</td>
<td>NN VBZ</td>
</tr>
</tbody>
</table>

- Constituents like IN DT and DT JJ occur indeed very frequently as substrings in the WSJ10, but not among U-DOP’s induced constituents. Why is this?

- First note that the substring DT NN occurs even more frequently than the substring IN DT.

- U-DOP's sum-of-products probability model will then favor a covering subtree for IN DT NN which consists of a division into IN X and DT NN rather than into IN DT and X NN! As a consequence IN DT will not be assigned a constituent in the most probable tree (or MPSD).
U-DOP can learn discontiguous constructions and agreement phenomena thanks to use of discontiguous subtrees

The observation of:

Swimming in the rivers is dangerous
Swimming together is fun

yields correct agreement structure between swimming and is

Next, shortest derivation enforces subject-verb agreement between Swimming and is for new sentences.

Same counts for (discontiguous) constructions:

Show me the nearest airport to Leipzig.
BA carried more people than cargo in 2005.
What is this scratch doing on the table?
Don’t take him by surprise.

But can U-DOP also learn other, more sophisticated linguistic phenomena, e.g. auxiliary fronting?
(U-)DOP and Auxiliary Fronting

Auxiliary fronting is widely studied (Crain 1991, MacWhinney 2005, Clark and Eyraud 2006), and is seen as a major challenge to empiricist models of language learning.

The problem of auxiliary fronting

\[\text{The man is hungry}\]
\[\text{Is the man hungry?}\]
\[\text{The man who is eating is hungry}\]
\[\text{Is the man who is eating hungry?}\]

Language learners never produce the incorrect fronting:

\[*\text{Is the man who eating is hungry?}\]

How does U-DOP account for this phenomenon?

Similar to Clark and Eyraud (2006), we show that learning of auxiliary fronting can proceed with just two sentences:

\[\text{The man who is eating is hungry}\]
\[\text{Is the boy hungry?}\]
The following structures could be induced for these two sentences using combined Penn treebank sentences and NANC corpus sentences:

![Diagram](attachment:diagram.png)

(a)          (b)

Figure 1

But to make our argument fully controlled, we showed in Bod (2007) that these structures could also be learned by just a few sentences. For instance, (a) can be learned from just the following sentences:

*The man who is eating mumbled*
*The man is hungry*
*The man mumbled*
*The man is eating*
Next, to produce the correct interrogative, *Is the man who is eating hungry*, (U-)DOP needs to combine only **two** subtrees from the acquired structures in 1 (no probability calculation is necessary):

Instead, to produce sentence with incorrect auxiliary fronting *Is the man who eating is hungry?* we need to combine at least **four** subtrees from figure 1 (and there are longer derivations as well):
Shortest derivation overrules incorrect form

- Shortest derivation overrules incorrect aux-fronting (once the structures have been induced by U-DOP), provided that we allow for arbitrarily large subtrees.

- Our result supports Clark and Eyraud (2006):

  We do not have to observe the complex aux-fronting in the data to learn it!

  But we showed additionally that U-DOP can learn constructions of any size, which may be hard for simple PCFGs used by Clark and Eyraud.
Can U-DOP also learn Syntactic Categories and Semantics?

Use same U-DOP principle as before:

If you don’t know which categories should be assigned to tree-nodes, initially assign \( n \) “abstract” categories from a set \( \{ C_1 \ldots C_n \} \) and let statistics decide which categorizations result in most probable trees.

Analogously, assign all possible predicate-argument structures, as a simplified semantics, to each tree-node -- e.g. \( \text{swim(in(rivers), dangerous)} \) -- and let statistics decide.

Conclusions

• DOP can be generalized to Unsupervised DOP by using same principle:

  *If you don’t know which structures should be assigned to sentences, take them all and let statistics decide (i.e. the MPSD)*

• Complex phenomena can be learned by U-DOP provided that we use subtrees of arbitrary size

  - Obtains state-of-the-art unsupervised parsing results
  - Learns constructions of arbitrary complexity
  - Learns many linguistic phenomena: agreement, fronting, ‘movement’

• U-DOP is one of the richest learning models: It does not limit any kind of dependency, neither structurally nor sequentially, while other models do.