Learning Natural Language Syntax from Large Corpora

J. GEERTZEN AND M. VAN ZAANEN
Tilburg University
{J.Geertzen@uvt.nl, M.M.vanZaanen@uvt.nl}

ABSTRACT

This research focuses on the learning of syntactic information of natural language. One successful approach to learning of syntax searches for substitutable parts in a plain text corpus and marks them as possible constituents. Alignment-Based Learning (ABL) is a system that implements this approach and extends it with a probabilistic disambiguation phase. Currently, the main problem with ABL is that finding the possible constituents is relatively slow, which restricts the system to small corpora.
The aim of this research is to speed-up the search for possible constituents, which will allow handling of large corpora. We expect that large corpora will improve accuracy of the learning system, since not only more possible constituents can be found, but also more precise statistics can be applied in the disambiguation phase. We introduce suffix trees and show their merits in this task. Preliminary results show that although this approach seems to yield lower recall, the learning process itself is significantly faster.

1 Introduction

Recently, there is a growing interest in the (unsupervised) learning of natural language syntax. This is not only because the current generation of computers are powerful enough to handle the amount of data and processing needed to search for syntactic structure, but also because applications of syntax learning systems seem to emerge.

- Data mining [1],
- Spoken dialog systems [2],

⋮

Inducing syntax from a corpus of plain text sentences can be done using different approaches. Here, we will focus on **Alignment-Based Learning** (ABL) [5].

2 Alignment-Based Learning

ABL consists of two main phases:

Alignment Learning Sentences are compared in pairs, searching for parts of the sentences that are equal and parts that are unequal in both sentences. The unequal parts of the sentences can be substituted by each other and the result is again a valid sentence. Based on this, the unequal parts of the sentences are considered possible constituents, called hypotheses.

```
Ze [ziet]_X Jan [op]_Y het [leuke feest]_Z
Ze [spreekt]_X Jan [in]_Y het [geheim]_Z
```

In these two sentences, the unequal parts are bracketed and the unequal parts that can be substituted receive the same constituent type.

Selection Learning For each sentence in the corpus, all hypotheses are stored in a tree-like data structure. Using statistics, the most likely hypotheses are selected.

The main problem of ABL is:

Scalability The current implementation of ABL depends on **edit distance** [4]. However, the computation of the edit distance only makes sense in pairwise comparison. To compare a corpus of N sentences, this would imply an upper bound of N^2 edit distance computations to compare each sentence with each other. Furthermore, the computation of the edit distance of two sentences of length m and n takes $O(n*m)$. Because of this, ABL is currently not feasible of learning from big corpora ($>$ 100K sentences) [6].

3 Aligning sentences

To find substitutable parts in a pairwise comparison of two sentences, we need to **align** these sentences such that the unequal parts can be easily identified.

```
Ze ziet     Jan op het leuke feest
Ze spreekt  Jan in het geheim
```

3.1 Alignment using edit distance

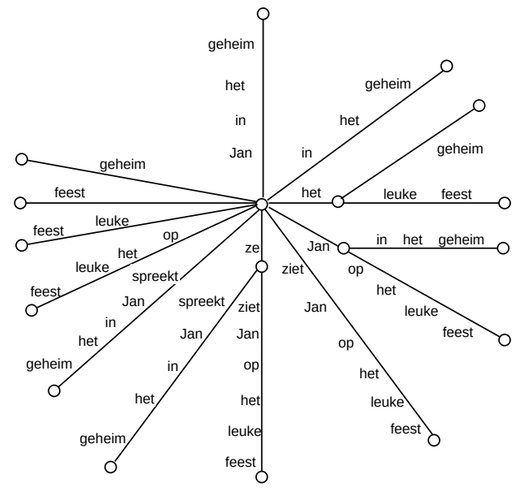
One way of aligning two sentences is to calculate the **edit distance** of two strings. The value of the edit distance depends on the edit operations (insertion, deletion and substitution) that are needed to transform one string into another. The idea is to find the sequence of operations that lead to the lowest edit distance between two sentences. This results in a possible alignments, such as:

	MAT	*SUB*	*MAT*	*DEL*	*INS*	*MAT*	*SUB*	*DEL*
sentence 1	Ze	ziet	Jan	op	-	het	leuke	feest
sentence 2	Ze	spreekt	Jan	-	in	het	geheim	-

Since, `Ze, Jan, het` match, the rests of the sentences are hypotheses: $\{$`ziet, spreekt`$\}_1$, $\{$`op, in`$\}_2$, and $\{$`leuke feest, geheim`$\}_3$.

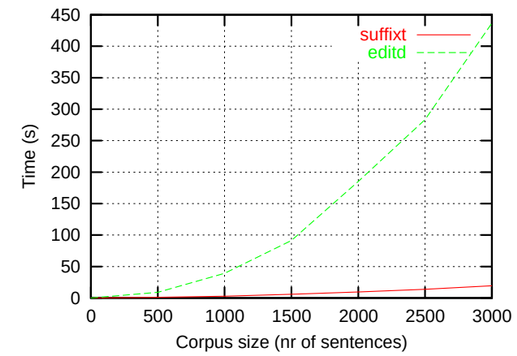
3.2 Alignment using suffix trees

A **suffix tree** is a data structure that contains all the suffixes of a string. In this context, we build a suffix tree containing the entire corpus. This suffix tree can be build efficiently (linearly in the number of words in the corpus) [3].



To find hypotheses, a prefix and a suffix tree is generated. (A prefix tree is a suffix tree of the reversed corpus.) Opening brackets are inserted on each node of depth 1 (directly below the root node). Closing brackets are inserted similarly, based on the prefix tree.

4 Execution times



5 Recall

To evaluate the alignment phase on its own, we select only those hypotheses that are also present in the correct treebank. This results in 100% precision and shows the upper bound of the recall.

	edit distance		suffix tree	
ATIS	48.45	66.09%	23.04	87.79%
OVIS	94.22	58.86%	47.59	90.37%

The percentages indicate how many hypotheses have been removed to reach 100% precision.

6 Conclusions

- The main problem of ABL is that the alignment learning phase is too slow and does not scale very well. The actual problem is the computation of the edit distance.
- Alignment learning using suffix trees allows fast learning in a near linear rate, which allows for the handling of large corpora.
- The results alignment learning using suffix trees are encouraging.

References

[1] Adriaans, (2001), Semantic Induction with EMILE, *Workshop Information extraction in molecular biology*, Enschede, the Netherlands.

[2] Starkie, (2002), Inferring attribute grammars with structured data for natural language processing, *Proceedings of the International Colloquium on Grammatical Inference (ICGI)*, Amsterdam, the Netherlands.

[3] Ukkonen, (1995), On-line construction of suffix trees *Algorithmica* **14(3)**, 249–260.

[4] Wagner and Fisher, (1974), The string-to-string correction problem, *Journal of the Association for Computing Machinery (ACM)* **21(1)**, 168–173.

[5] van Zaanen, (2002), *Bootstrapping structure into language: Alignment-Based Learning*. PhD thesis, University of Leeds, UK.

[6] van Zaanen, (2003), Theoretical and Practical Experiences with Alignment-Based Learning, *Proceedings of the Australasian Language Technology Workshop (ALTW2003)*, Melbourne, Australia, to be published.