Doubling DOP*

Data Oriented Parsing based on Double-DOP and DOP*

Benno Kruit (10576223)
Sara Veldhoen (10545298)

January 14, 2014

1 Introduction

2 Existing Frameworks: Double-DOP and DOP*

A PTSG consists of the symbolic grammmar, i.e. a set of fragments, and the corresponding weights. In the general case of DOP, all fragments are extracted from all the trees in the treebank. The number of fragments is exponential in the length of the sentences, thus the total number of fragments extracted would be far too large for efficient computation. Later models have therefore restricted the set of fragments in the grammar.

In this section, we outline two approaches to constrain the extraction of fragments: Double-DOPand DOP*. Furthermore, we discuss the similarities and dissimilarities for these two approaches.

2.1 Double-DOP

In the following, we discuss Double-DOP as it was presented in [1]. In this model, no unique fragments are extracted from the dataset: if a construction occurs in one tree only, it is probably not representative for the language. This is carried out by a dynamic algorithm using tree-kernels. It iterates over pairs of trees in the treebank, looking for fragments they have in common. In addition, only the largest shared fragment is stored.

The symbolic grammar that is the output of this algorithmis not guaranteed to derive each tree in the training corpus. Therefore all one-level fragments, consitutuing the set of PCFG-productions, are also added.

After the extraction of the symbolic grammar, the weights are obtained. This is done in a second pass over the treebank, assessing the relative frequencies.

The Double-DOPmodel has its main focus on determining the symbolic grammar. However, it was implemented with different estimators and maximizing objectives. Empirical results show that

2.2 DOP*

In DOP*[2], a rather different approach is taken called held-out estimation. The treebank is split in two parts, the extraction corpus (EC) and a held-out corpus (HC). An initial set of fragments is extracted from the EC, containing all the fragments from its trees. The weights are then determined so as to to maximize the likelihood of HC, under the assumption that this is equivalent to maximizing the joint probability of the *shortest derivations* of the trees in HC. All fragments that do not occur in such a derivation are removed from the symbolic grammar. Note that some trees in HC may not be derivable at all.

Consistency and bias DOP*is claimed to be the first consistent (non-trivial) DOP-estimator, [2] provides a consistency proof. On the other hand DOP* is biased, but Zollmann shows how bias actually arises from generalization: no non-overfitting DOP estimator could be unbiased. Bias is therefore not prohibited but on the contrary a desirable property of an estimator.

In [3] it is argued that there is a problem with the consistency proof given for DOP*, as well as the non-consistency proof for other DOP-estimators by [4]. Zuidema points out that these proofs use a frequency-distribution test, whereas for DOP a weight-distribution test would be more appropriate.

2.3 Comparison

Both DOP* and Double-DOP restrict the symbolic grammar based on some notion of reoccurence of fragments.

In Double-DOP this is evident, it is explicit in the algorithm how (largest) reoccuring fragments are added to the grammar. From a computational point of view, this approach is very intuitive and can be implemented rather efficiently. However, the threshold (two) on the number of reoccurences might seem rather trivial. Indeed, [1] reports experiments varying this threshold that show how performance drops for higher thresholds, with on the other hand a great reduction of the size of the grammar. A proper setting might well depend on the size and nature of the treebank used, and the computational costs one is willing to pay. In short, Double-DOP is computationally attractive but its theoretical foundation is not convincing us.

Theoretically,DOP* is more appealing: we decide on the symbolic grammar by assessing which fragments are actually used in derivations. The main assumption is that shortest derivations

DOP* theoretically more appealing, Double-DOP computationally: lend properties of Double-DOP to implement DOP*

3 A new implementation

4 Results

5 Conclusion

References

[1] F. Sangati and W. Zuidema, "Accurate parsing with compact tree-substitution grammars: Double-dop," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 84–95, Association for Computational Linguistics, 2011.

[2] A. Zollmann and K. Sima'an, "A consistent and efficient estimator for data-oriented parsing," *Journal of Automata, Languages and Combinatorics*, vol. 10, no. 2/3, pp. 367–388, 2005.

[3] W. Zuidema, "Theoretical evaluation of estimation methods for data-oriented parsing," in *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations*, pp. 183–186, Association for Computational Linguistics, 2006.

[4] M. Johnson, "The dop estimation method is biased and inconsistent," *Computational Linguistics*, vol. 28, no. 1, pp. 71–76, 2002.