Multilingual parallel treebanking: a lean and flexible approach

Jonas Kuhn, Michael Jellinghaus

Dept. of Computational Linguistics, Saarland University
Saarbrücken, Germany
{jonask,micha}@coli.uni-sb.de

Abstract

We propose a bootstrapping approach to creating a phrase-level alignment over a sentence-aligned parallel corpus, reporting concrete treebank annotation work performed on a sample of sentence tuples from the Europarl corpus, currently for English, French, German, and Spanish. The manually annotated seed data will be used as the basis for automatically labelling the rest of the corpus. Some preliminary experiments addressing the bootstrapping aspects are presented.

The representation format for syntactic correspondence across parallel text that we propose as the starting point for a process of successive refinement emphasizes correspondences of major constituents that realize semantic arguments or modifiers; language-particular details of morphosyntactic realization are intentionally left largely unlabelled. We believe this format is a good basis for training NLP tools for multilingual application contexts in which consistency across languages is more central than fine-grained details in specific languages (in particular, syntax-based statistical Machine Translation).

1. Background and Motivation

It is well-known from the Machine Translation (MT) literature that although generally in translated texts, there is a systematic cross-linguistic correspondence of phrase structure constituents, the number of exceptions is too high to ignore when working with a fine-grained phrase-structural representation. For dependency representations the consensus across languages is higher (see e.g., (Fox, 2002)). Therefore, "annotation projection" experiments like (Yarowsky et al., 2001; Hwa et al., 2002) typically convert the PennTreebank-style representation from the English parser into a dependency representation before "projecting" it to a different language L, using a statistical word alignment on a parallel corpus. (The data in language L with the projected dependency structure are then used to train a parser for L, with noise-robust techniques.)

A phrase structure representation on the other hand has the advantage that it does not enforce a commitment on what is the head of a group of words: most syntactic theories allow for the possibility of having distinct syntactic and semantic heads, or functional and lexical heads in a constituent. For instance, in a clause with a periphrastic verb form like "I will arrive tomorrow", we may say that the temporal auxiliary *will* is the morphosyntactic or functional head, but the full verb *arrive* is the semantic or lexical head. If we adopt a relatively flat phrase structure representation, making the various heads sisters of each other, the hierarchical structure for clauses with periphrastic verb forms is identical to their cross-linguistic correspondents including complex verb forms, like the French translation "J'arriverai demain" (I arrive.FUT.1sg tomorrow); the only difference lies in the number of preterminal symbols included. (Of course, other more substantial structural divergencies across languages *will* lead to differences in the representation.)

Our longer-term goal is to explore whether better multilingual NLP tools can be obtained with weakly supervised learning techniques based on a "lean" phrase structure representation, which is semantics-driven and designed with the specific cross-linguistic situation in mind. The concrete scenario for applying the scheme in the future is the following: To obtain parallel parsers for a group of languages, a small section of a large multilingual parallel corpus (on the order of 100 translated sentences) is annotated by humans who have reading knowledge of the languages involved. The annotated sentence tuples are used as seed data for bootstrapping phrase correspondence patterns for the entire corpus, building on top of a statistical word alignment. The resulting consensus representation is used to train monolingual parsers that assign tree analyses following the lean, cross-linguistically oriented representation scheme.

The hypothesized advantages of a "lean" phrase-structure consensus representation are:

- the annotation principles can be phrased in a general, mostly language-independent way
- structural ambiguities in a particular language are often resolved in one of the other languages
- the lean format allows for relatively fast annotation
- annotations for additional languages in a multilingual corpus can be added even faster
- for related languages, the annotator need not have full command of all the languages, but can interpolate from the various translations in the other languages.
- the labelling scheme is effective in a minimally supervised bootstrapping approach, providing robust and reliable annotations for a large parallel corpus
- the representation is a good basis for multilingual NLP tools building on parallel corpora, e.g., statistical MT

The last two points in particular require relatively extensive experiments in order to be evaluated. At this point we can only present work in progress towards this goal. We anticipate that certain aspects of the annotation scheme chosen may have to be adjusted to best meet these criteria.

This study is part of the long-term PTOLEMAIOS project on grammar learning from parallel corpora (for an overview of the project agenda see (Kuhn, 2005)). Orginally departing from a fully unsupervised grammar induction approach (Kuhn, 2004), one of the project goals is to explore how much implicit information about the syntax of a language one can exploit from a sentence-aligned parallel corpus.

EN: [$_1$ [$_2$ On behalf of the European People 's Party ,] [$_3$ I] call [$_5$ for a vote [$_6$ in favour of that motion]]]

FR: [$_1$ [$_2$ Au nom du Parti populaire européen ,] [$_3$ je] demande [$_5$ l' adoption [$_6$ de cette résolution]]]

DE: [$_1$ [$_2$ Im Namen der Europäischen Volkspartei] rufe [$_3$ ich] [$_4$ Sie] auf , [$_5$ [$_6$ diesem Entschließungsantrag] zuzustimmen]]

ES: [$_1$ [$_2$ En nombre del Grupo del Partido Popular Europeo ,] solicito [$_5$ la aprobación [$_6$ de la resolución]]]

Figure 1: Example of annotated sentence tuple

For the work we are presently reporting, it is a conscious methodological decision to start out with a preliminary annotation scheme, annotating only relatively few corpus sentences as seed data for bootstrapping and as test data. Based on these data, the usefulness of the scheme is assessed along all dimensions relevant to our project; successive refinements can then be made and the approach can be re-evaluated, etc., following a bootstrapping idea also at the meta-level.

2. Manual Annotation of Seed and Test Data

2.1. Annotation Scheme

We experimented with annotation schemes of various granularities. This paper focuses on the "leanest" scheme which consists of a bracketing for each language and a correspondence relation of the constituents across languages. Neither the constituents nor the embedding or correspondent relations were labelled.[1]

The guiding principle for bracketing is very simple: all and only the units that clearly play the role of a semantic argument or modifier in a larger unit are bracketed. This means that function words, light verbs, "bleeched" PPs like *in spite of* etc. are included with the content-bearing elements, leading to a relatively flat bracketing structure. Referring or quantified expressions that may include adjectives and possessive NPs or PPs are also bracketed as single constituents (e.g., [*the president of France*]), unless the semantic relations reflected by the internal embedding are part of the predication of the sentence. A few more specific annotation rules were specified for cases like coordination and discontinuous constituents.

The correspondence relation is guided by semantic correspondence of the bracketed units; the mapping need not preserve the tree structure. A constituent has at most one correspondent in each of the other languages, but may have no correspondent in a language, since the content of this constituent may be implicit or subsumed by the content of another constituent. "Semantic correspondence" is not restricted to truth-conditional equivalence, but is generalized to situations where two units just serve the same rhetorical function in the original text and the translation.

Figure 1 is an annotation example. Note that index 4 (the audience addressed by the speaker) is realized overtly only in German (*Sie* 'you'); in Spanish, index 3 is realized only in the verbal inflection (which is not annotated).

[1]The intuition is that the bracket spans and the cross-linguistic correspondence relation combined are rich information sources for learning systematic rules. For application contexts requiring category distinctions or role label distinctions, these might be "imported" using an existing analysis tool for English and/or other languages.

2.2. Annotation Work

Annotation is performed with the MMAX2 tool developed by EML Research, Heidelberg, Germany,[2] which was originally designed for monolingual coreference annotation, but can be customized easily. As a preprocessing step, we converted the sentence-aligned parallel corpus into the appropriate XML format, such that the sentence tuples are displayed in a line-by-line format, which has proven highly adequate for the annotation task. So far, 300 sentence tuples (with a length limit of 20 words) have been annotated. With some experience, the average annotation time for such a tuple (i.e., all four languages) is approx. 3 minutes.

To get a first impression of inter-annotator agreement in the bracketing task, a subset of 39 sentence tuples was annotated by two people (the two authors of this paper) independently.[3] The table in Figure 2 shows the agreement for each of the four languages, based on unlabelled brackets.[4]

Language	EN	FR	DE	ES
Complete match (% of sentences)	28.2	23.7	48.6	20.5
Precision	89.7	83.9	91.3	87.4
Recall	86.6	84.5	89.3	81.4
F-Score	88.1	84.2	90.3	84.3

Figure 2: Inter-annotator agreement (annotator 1 relative to annotator 2) based on 39 sentence tuples

We believe that a greater degree of agreement can be reached if in a revised annotation scheme, a more concrete and syntactically explicit criterion is used to determine what constituents should be bracketed. It is not surprising to find a certain amount of disagreement in the question whether or not a syntactically embedded element does play the role of a semantic argument or modifier in the larger unit, or whether it is semantically vacuous. The cross-linguistic correspondence links will still be subject to the semantic criterion.

Since the labelling of alignments *across* languages depended so much on what brackets were chosen in the first place, we do not present an anlysis of this part of the annotation task for now. Informal inspection of the annota-

[2]http://mmax.eml-research.de

[3]Agreement with a random bracketing has an f-score of 22.3; so on the (small) basis of the 39 examples, the kappa coefficient for inter-annotator agreement is .84, which indicates high agreement.

[4]The differences across languages is possibly due to the fact that the data were presented in the order German – English – French – Spanish. The semantically based bracketing criterion can be applied most easily for the first language considered in parallel treebanking; for languages added later, there may be an occasional choice as to whether some constituent is bracketed to make it parallel to the other languages or whether it should be considered a mismatch.

tions suggest a very high level of agreement where the same brackets were chosen.

3. Bootstrapping Experiments

To fully assess the practicality of the annotation format for weakly supervised learning in a multilingual NLP context, large-scale bootstrapping experiments will be required. Also, it will be important to evaluate the resulting annotations in a task-oriented way, e.g., in the context of phrase-based statistical MT. We expect that analyzing such experiments will provide crucial feedback on the original choice of the annotation format, and we foresee going through several cycles of revised annotation guidelines before coming up with a format that is most suitable for both (i) easy seed data annotation with high inter-annotator agreement, and (ii) effective bootstrapping of the annotation on large amounts of parallel text.

For the time being, we can report on some initial bootstrapping experiments which have the status of a proof of concept that the envisaged methodology is valid. We cannot present any task-oriented evaluation results yet.

3.1. Study A: Selection of training data using consensus on tree structure

In a first pilot study, we used a subset of our annotated data to train (simplistic) treebank parsers for the four languages, using Bikel's reimplementation (Bikel, 2004) of Collins' parser (Collins, 1999)). The goal of this study was to see whether parsers trained on such a small set of seed data will nevertheless reach some degree of consensus when parsing unseen sentence tuples (and whether the consensus can be considered useful for a bootstrapping approach). In a full bootstrapping scenario, the analyses of sentence tuples with a high degree of cross-parser consensus would be used as training data for the next generation of parsers. What we are focusing on here is the "parse tuple filter" for deciding whether a particular parse tuple from the set of unseen data should be included in the next training data or not.

To evaluate the filter, we applied the parsers and the filter on a set of unseen sentence tuples for which we had a held-out gold-standard annotation. A good filter should eliminate sentences for which no reliable consensus can be found; hence the degree of gold-standard matching in the data that *pass* the filter gives us a good indication of the quality of the filter (ideally only exact matches should pass the filter). The parse tuple filter we used was very simple. Given a tuple from the parallel corpus, we used (a) the highest-scoring parse from each of the four parsers, and (b) a standard GIZA++ statistical word alignment.[5] We projected the word-level alignmed to phrases using simple heuristics (essentially saying that two phrases align if all the contained words are linked by the word alignment). We could then define a family of parse tuple filters by setting thresholds on the number of phrases in the parses for which the alignment condition holds. Filtering was indeed effective: depending on the threshold and the language pair we compared, the data passing the filter had an f-score against the gold standard that was between 4 and 10 percentage points higher than for the larger, unfiltered set (70.1 vs. 60.7 for a particular experiment; the baseline f-score for random bracketing is generally in the low 20's).

We took this as a first indication for the potential effectiveness of the bootstrapping set-up and the annotation scheme.[6]

3.2. Study B: A full bootstrapping architecture

We designed and implemented a full bootstrapping architecture based on Charniak's parser (Charniak, 2000; Charniak and Johnson, 2005),[7] as follows (a more detailed description is provided in (Kuhn, in preparation)): The manually annotated seed data are split into a training and development set. Parsers for each of the languages are trained on the (monolingual) parse trees included in the annotations of the training set.[8] In addition, a binary classifier for the alignment of phrase pairs is trained on the link information included in the training set, using the MegaM Maximum Entropy (MaxEnt) kit developed by Hal Daume III at ISI, USC.[9] The features applied in the MaxEnt classifier rely on correspondences of (combinations of) part-of-speech categories and word forms, phrase length, position within the sentence (distortion) and the geometry of links from a GIZA++ statistical word alignment.

The parsers are applied in k-best mode on the development set, giving rise to candidate pairs of trees for each pair of languages. The phrase pair classifier is also applied on the development data in order to determine the highest-scoring phrase alignment for each candidate tree pair.[10] The gold standard trees and phrase alignment for the development data is used to generate training data for a MaxEnt tree pair ranker, which is optimized to put the highest score on the gold standard tree pairs in the training corpus.[11] The features for the tree pair ranker exploit structural information from the parsers, and they also incorporate the scores of the phrase alignments used. They can also include the parser scores; however, in the bootstrapping scenario, versions of the tree pair ranker are trained that are uninformed about the parsing score in one of the languages.

Going beyond the seed data, the k-best parsers and the phrase alignment classifier are applied on unlabelled data to generate tree pair candidates. The tree pair ranker is applied to determine a ranking over these candidates. The filter criterion for including analyses of the unseen data in the next bootstrapping cycle can now be defined based on

[5] GIZA++ by Franz Josef Och is available from http://www.fjoch.com/GIZA++.html

[6] We discontinued development of the particular system underlying the study reported in this section, since the massive use of k-best parsing in the actual bootstrapping steps made it seem advisable to switch to Charniak's parser.

[7] ftp://ftp.cs.brown.edu/pub/nlparser/

[8] The non-terminal nodes from the unlabelled bracketing are named S, NP, PP, ADVP, ADJP or X, using simple heuristics.

[9] http://www.isi.edu/~hdaume/megam/

[10] We apply a greedy search to approximate the best phrase alignment for a tree pair, subject to the constraint that no phrase can be aligned to more than one phrase in the other language.

[11] The actual gold standard trees are not necessarily included in the parses for the development data; if they are missing, the tree with (1) highest (unlabelled) recall and (2) highest precision is picked.

the k-best ranked parses (for two or more languages) and the tree pair ranking for pairs of languages.

	Labelled bracketing			Average crossing	Unlabelled bracketing		
	Recall	Precision	F-Score		Recall	Precision	F-Score
No bootstrapping	62.7	69.7	66.0	.36	68.0	75.5	71.6
Bootstrapping (1 cycle)	64.6	72.7	68.4	.22	69.1	77.7	73.2

Figure 3: Results of some first bootstrapping experiments for a parser of German

At the present stage, we can only present preliminary test results that were based on a very small set of 4600 unlabelled sentence pairs for bootstrapping a German grammar from English-German parallel text. The seed training and development sets each consisted of 94 hand-labelled sentence pairs. No tools at all (such as part-of-speech taggers etc.) were applied to the unlabelled training data; we relied exclusively on the unknown word mechanisms of Charniak's parser (which was certainly not designed for training on less than 200 sentences).

The features we used in the classifiers were rather simple (for instance, we completely left out the parser scores in the tree pair ranker). The filter criterion for the bootstrapping data was also simplistic. For retraining the parser, those sentence pairs were used for which both trees in the top-ranked tree pair were included among the three top-ranked parses[12] from the parsers trained on the seed data (which was run in 50-best mode). This resulted in 567 additional training instances for the German parser.

The labelled bracketing F-score of the German parser on a test set of 50 unseen sentences without using bootstrapping data was 66.0. With the additional bootstrapping data, an F-score of 68.4 was reached, i.e., we observe a noticable increase. More details are shown in figure 3. We take these results as additional indication that the bootstrapping approach is promising.

4. Discussion

We proposed a bootstrapping approach to creating a phrase-level alignment over a sentence-aligned parallel corpus: only a small set of seed data is manually annotated; the rest of the corpus is covered using minimally supervised learning techniques. This approach goes along with specific criteria for the annotation scheme. We consider our annotation scheme as work in progress—the bootstrapping approach comes with the considerable advantage that changes in the annotation scheme can be made even at an advanced stage of project development, since most steps in the process are automatic. The set of manually labelled seed data is small enough to do a complete re-annotation if necessary.

While it is too early to report on conclusive results on our approach, we believe that our initial experiments are promising and indicate that the approach is worth further exploration as an alternative to existing work on syntactic analysis of parallel corpora: "annotation projection" work on the one hand (Yarowsky et al., 2001; Hwa et al., 2002), and work applying (inversion) transduction grammars on the other hand (e.g., (Wu, 1997; Melamed, 2003)). Contrary to the former type of approach, we do not assume a high-quality grammar/parser for English (or another language) as given; grammars for all languages are bootstrapped in parallel. This makes the approach more flexible and may lead to emerging consensus representations across the languages (which can be orthogonal to fixed choices in the parser used in an "annotation projection" approach).

There are many parallels between our approach and work based on transduction grammars for multitexts. However, by using a conventional monolingual parser and assuming a structurally unconstrained correspondence relation between tree nodes, we can split up the learning problem in a different way, relying on existing tools for comparatively simple subproblems (the parsers and the MaxEnt classifiers/rankers). Moreover, most steps can easily be parallelized. Larger-scale experiments will have to show whether the search heuristics that our approach forces us to adopt are indeed empirically unproblematic.

Acknowledgement

This work was supported by the *Deutsche Forschungsgemeinschaft* (DFG; German Research Foundation) in the Emmy Noether project PTOLEMAIOS on grammar learning from parallel corpora, awarded to Jonas Kuhn.

5. References

Daniel M. Bikel. 2004. Intricacies of Collins' parsing model. *Computational Linguistics*, 30(4):479–511.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of ACL'05*, pages 173–180, Ann Arbor, Michigan.

Eugene Charniak. 2000. A maximum entropy-inspired parser. In *Proceedings of NAACL 2000*, pages 132–139, Seattle, Washington.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, Univ. of Pennsylvania.

Heidi J. Fox. 2002. Phrasal cohesion and statistical machine translation. In *Proceedings of EMNLP 2002*, pages 304–311.

Rebecca Hwa, Philip Resnik, and Amy Weinberg. 2002. Breaking the resource bottleneck for multilingual parsing. In *Proceedings of LREC*.

Jonas Kuhn. 2004. Experiments in parallel-text based grammar induction. In *Proceedings of ACL 2004*.

Jonas Kuhn. 2005. An architecture for parallel corpus-based grammar learning. In B. Fisseni, H.-C. Schmitz, B. Schröder, and P. Wagner, editors, *Beiträge zur GLDV-Tagung 2005 in Bonn*, pages 132–144. Peter Lang.

Jonas Kuhn. in preparation. Bootstrapping phrase alignments on a parallel corpus. Ms., Saarland University.

I. Dan Melamed. 2003. Multitext grammars and synchronous parsers. In *Proceedings of NAACL/HLT*.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–403.

D. Yarowsky, G. Ngai, and R. Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *HLT 2001*, pages 161–168.

[12] We also experimented (briefly) with a smaller and larger number; using parses up to rank three gave us the best results.