Optimal Constituent Alignment with Edge Covers for Semantic Projection

Sebastian Padó
Computational Linguistics
Saarland University
Saarbrücken, Germany
`pado@coli.uni-sb.de`

Mirella Lapata
School of Informatics
University of Edinburgh
Edinburgh, UK
`mlap@inf.ed.ac.uk`

Abstract

Given a parallel corpus, semantic projection attempts to transfer semantic role annotations from one language to another, typically by exploiting word alignments. In this paper, we present an improved method for obtaining *constituent alignments* between parallel sentences to guide the role projection task. Our extensions are twofold: (a) we model constituent alignment as *minimum weight edge covers* in a bipartite graph, which allows us to find a globally optimal solution efficiently; (b) we propose *tree pruning* as a promising strategy for reducing alignment noise. Experimental results on an English-German parallel corpus demonstrate improvements over state-of-the-art models.

1 Introduction

Recent years have witnessed increased interest in data-driven methods for many natural language processing (NLP) tasks, ranging from part-of-speech tagging, to parsing, and semantic role labelling. The success of these methods is due partly to the availability of large amounts of training data annotated with rich linguistic information. Unfortunately, such resources are largely absent for almost all languages except English. Given the data requirements for supervised learning, and the current paucity of suitable data for many languages, methods for generating annotations (semi-)automatically are becoming increasingly popular.

Annotation projection tackles this problem by leveraging parallel corpora and the high-accuracy tools (e.g., parsers, taggers) available for a few languages. Specifically, through the use of word alignments, annotations are transfered from resource-rich languages onto low density ones. The projection process can be decomposed into three steps: (a) determining the units of projection;

these are typically words but can also be chunks or syntactic constituents; (b) inducing alignments between the projection units and projecting annotations along these alignments; (c) reducing the amount of noise in the projected annotations, often due to errors and omissions in the word alignment.

The degree to which analyses are parallel across languages is crucial for the success of projection approaches. A number of recent studies rely on this notion of parallelism and demonstrate that annotations can be adequately projected for parts of speech (Yarowsky and Ngai, 2001; Hi and Hwa, 2005), chunks (Yarowsky and Ngai, 2001), and dependencies (Hwa et al., 2002).

In previous work (Padó and Lapata, 2005) we considered the annotation projection of semantic roles conveyed by sentential constituents such as AGENT, PATIENT, or INSTRUMENT. Semantic roles exhibit a high degree of parallelism across languages (Boas, 2005) and thus appear amenable to projection. Furthermore, corpora labelled with semantic role information can be used to train shallow semantic parsers (Gildea and Jurafsky, 2002), which could in turn benefit applications in need of broad-coverage semantic analysis. Examples include question answering, information extraction, and notably machine translation.

Our experiments concentrated primarily on the first projection step, i.e., establishing the right level of linguistic analysis for effecting projection. We showed that projection schemes based on constituent alignments significantly outperform schemes that rely exclusively on word alignments. A *local* optimisation strategy was used to find constituent alignments, while relying on a simple filtering technique to handle noise.

The study described here generalises our earlier semantic role projection framework in two important ways. First, we formalise constituent projection as the search for a *minimum weight edge cover* in a weighted bipartite graph. This formalisation

efficiently yields constituent alignments that are *globally* optimal. Second, we propose *tree pruning* as a general noise reduction strategy, which exploits both structural and linguistic information to enable projection. Furthermore, we quantitatively assess the impact of noise on the task by evaluating both on automatic and manual word alignments.

In Section 2, we describe the task of role-semantic projection and the syntax-based framework introduced in Padó and Lapata (2005). Section 3 explains how semantic role projection can be modelled with minimum weight edge covers in bipartite graphs. Section 4 presents our tree pruning strategy. We present our evaluation framework and results in Section 5. A discussion of related and future work concludes the paper.

2 Cross-lingual Semantic Role projection

Semantic role projection is illustrated in Figure 1 using English and German as the source-target language pair. We assume a FrameNet-style semantic analysis (Fillmore et al., 2003). In this paradigm, the semantics of predicates and their arguments are described in terms of *frames*, conceptual structures which model prototypical situations. The English sentence *Kim promised to be on time* in Figure 1 is an instance of the COMMITMENT frame. In this particular example, the frame introduces two roles, i.e., SPEAKER (*Kim*) and MESSAGE (*to be on time*). Other possible, though unrealised, roles are ADDRESSEE, MESSAGE, and TOPIC. The COMMITMENT frame can be introduced by *promise* and several other verbs and nouns such as *consent* or *threat*.

We also assume that frame-semantic annotations can be obtained reliably through shallow semantic parsing.[1] Following the assignment of semantic roles on the English side, (imperfect) word alignments are used to infer *semantic alignments* between constituents (e.g., *to be on time* is aligned with *pünktlich zu kommen*), and the role labels are transferred from one language to the other. Note that role projection can only take place if the source predicate (here *promised*) is word-aligned to a target predicate (here *versprach*) evoking the same frame; if this is not the case (e.g., in metaphors), projected roles will not be generally appropriate.

We represent the source and target sentences as sets of linguistic units, U_s and U_t, respectively.
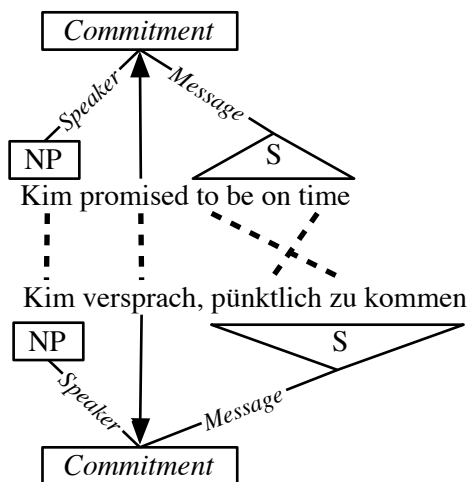
Figure 1: Projection of semantic roles from English to German (word alignments as dotted lines)

The assignment of semantic roles on the source side is a function $role_s : R \to 2^{U_s}$ from roles to sets of source units. Constituent alignments are obtained in two steps. First, a real-valued function $sim : U_s \times U_t \to \mathbb{R}$ estimates pairwise similarities between source and target units. To make our model robust to alignment noise, we use only content words to compute the similarity function. Next, a *decision procedure* uses the similarity function to determine the set of semantically equivalent, i.e., aligned units $A \subseteq U_s \times U_t$. Once A is known, semantic projection reduces to transferring the semantic roles from the source units onto their aligned target counterparts:

$$role_t(r) = \{u_t \mid \exists u_s \in role_s(r) : (u_s, u_t) \in A\}$$

In Padó and Lapata (2005), we evaluated two main parameters within this framework: (a) the choice of linguistic units and (b) methods for computing semantic alignments. Our results revealed that constituent-based models outperformed word-based ones by a wide margin (0.65 Fscore vs. 0.46), thus demonstrating the importance of bracketing in amending errors and omissions in the automatic word alignment. We also compared two simplistic alignment schemes, *backward alignment* and *forward alignment*. The first scheme aligns each target constituent to its most similar source constituent, whereas the second (A_f) aligns each source constituent to its most similar target constituent:

$$A_f = \{(u_s, u_t) \mid u_t = \operatorname*{argmax}_{u'_t \in U_t} sim(u_s, u'_t)\}$$

1162

An example constituent alignment obtained from the forward scheme is shown in Figure 2 (left side). The nodes represent constituents in the source and target language and the edges indicate the resulting alignment. Forward alignment generally outperformed backward alignment (0.65 Fscore vs. 0.45). Both procedures have a time complexity quadratic in the maximal number of sentence nodes: $O(|U_s||U_t|) = O(\max(|U_s|, |U_t|)^2)$.

A shortcoming common to both decision procedures is that they are *local*, i.e., they optimise the alignment for each node independently of all other nodes. Consider again Figure 2. Here, the forward procedure creates alignments for all source nodes, but leaves constituents from the target set unaligned (see target node (1)). Moreover, local alignment methods constitute a rather weak model of semantic equivalence since they allow one target node to correspond to any number of source nodes (see target node (3) in Figure 2, which is aligned to three source nodes). In fact, by allowing any alignment between constituents, the local models can disregard important linguistic information, thus potentially leading to suboptimal results. We investigate this possibility by proposing well-understood global optimisation models which suitably constrain the resulting alignments.

Besides matching constituents reliably, poor word alignments are a major stumbling block for achieving accurate projections. Previous research addresses this problem in a post-processing step, by reestimating parameter values (Yarowsky and Ngai, 2001), by applying transformation rules (Hwa et al., 2002), by using manually labelled data (Hi and Hwa, 2005), or by relying on linguistic criteria (Padó and Lapata, 2005). In this paper, we present a novel filtering technique based on tree pruning which removes extraneous constituents in a preprocessing stage, thereby disassociating filtering from the alignment computation.

In the remainder of this paper, we present the details of our global optimisation and filtering techniques. We only consider constituent-based models, since these obtained the best performance in our previous study (Padó and Lapata, 2005).

3   Globally optimal constituent alignment

We model constituent alignment as a *minimum weight bipartite edge cover* problem. A bipartite graph is a graph $G = (V, E)$ whose node set V is partitioned into two nonempty sets V_1 and V_2 in such a way that every edge E joins a node in V_1

to a node in V_2. In a weighted bipartite graph a weight is assigned to each edge. An edge cover is a subgraph of a bipartite graph so that each node is linked to at least one node of the other partition. A minimum weight edge cover is an edge cover with the least possible sum of edge weights.

In our projection application, the two partitions are the sets of source and target sentence constituents, U_s and U_t, respectively. Each source node is connected to all target nodes and each target node to all source nodes; these edges can be thought of as potential constituent alignments. The edge weights, which represent the (dis)similarity between nodes u_s and u_t are set to $1 - sim(u_s, u_t)$.[2] The minimum weight edge cover then represents the alignment with the maximal similarity between source and target constituents. Below, we present details on graph *edge covers* and a more restricted kind, *minimum weight perfect bipartite matchings*. We also discuss their computation.

Edge covers   Given a bipartite graph G, a minimum weight edge cover A_e can be defined as:

$$A_e = \operatorname*{argmin}_{\text{Edge cover } E} \sum_{(u_s, u_t) \in E} 1 - sim(u_s, u_t)$$

An example edge cover is illustrated in Figure 2 (middle). Edge covers are somewhat more constrained compared to the local model described above: all source and target nodes have to take part in some alignment. We argue that this is desirable in modelling constituent alignment, since important linguistic units will not be ignored. As can be seen, edge covers allow one-to-many alignments which are common when translating from one language to another. For example, an English constituent might be *split* into several German constituents or alternatively two English constituents might be *merged* into a single German constituent. In Figure 2, the source nodes (3) and (4) correspond to target node (4). Since each node of either side has to participate in at least one alignment, edge covers cannot account for *insertions* arising when constituents in the source language have no counterpart in their target language, or vice versa, as is the case for *deletions*.

Weighted perfect bipartite matchings   *Perfect bipartite matchings* are a more constrained version of edge covers, in which each node has *exactly* one adjacent edge. This restricts constituent

---

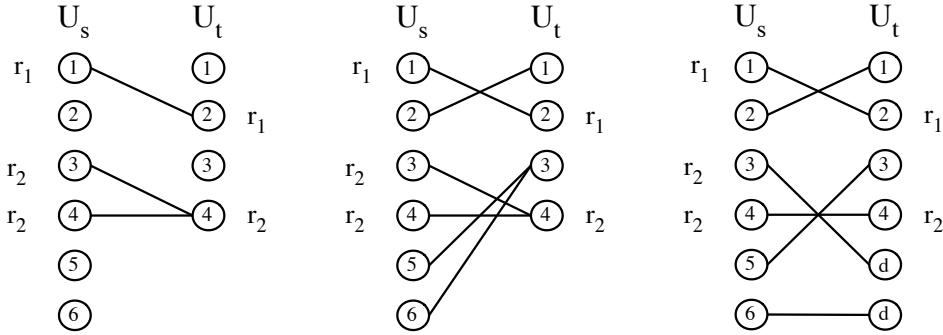[2] The choice of similarity function is discussed in Section 5.

Figure 2: Constituent alignments and role projections resulting from different decision procedures (U_s, U_t: sets of source and target constituents; r_1, r_2: two semantic roles). Left: local forward alignment; middle: edge cover; right: perfect matching with dummy nodes

alignment to a bijective function: each source constituent is linked to exactly one target constituent, and vice versa. Analogously, a *minimum weight perfect bipartite matching* A_m is a minimum weight edge cover obeying the one-to-one constraint:

$$A_m = \underset{\text{Matching } M}{\text{argmin}} \sum_{(u_s, u_t) \in M} 1 - sim(u_s, u_t)$$

An example of a perfect bipartite matching is given in Figure 2 (right), where each node has exactly one adjacent edge. Note that the target side contains two nodes labelled (d), a shorthand for "dummy" node. Since sentence pairs will often differ in length, the resulting graph partitions will have different sizes as well. In such cases, dummy nodes are introduced in the smaller partition to enable perfect matching. Dummy nodes are assigned a similarity of zero with all other nodes. Alignments to dummy nodes (such as for source nodes (3) and (6)) are ignored during projection.

Perfect matchings are more restrictive models of constituent alignment than edge covers. Being bijective, the resulting alignments cannot model splitting or merging operations at all. Insertions and deletions can be modelled only indirectly by aligning nodes in the larger partition to dummy nodes on the other side (see the source side in Figure 2 where nodes (3) and (6) are aligned to (d)). Section 5 assesses if these modelling limitations impact the quality of the resulting alignments.

Algorithms Minimum weight perfect matchings in bipartite graphs can be computed efficiently in cubic time using algorithms for network optimisation (Fredman and Tarjan, 1987; time $O(|U_s|^2 \log |U_s| + |U_s|^2 |U_t|)$) or algorithms for

the equivalent linear assignment problem (Jonker and Volgenant, 1987; time $O(\max(|U_s|, |U_t|)^3)$). Their complexity is a linear factor slower than the quadratic runtime of the local optimisation methods presented in Section 2.

The computation of (general) edge covers has been investigated by Eiter and Mannila (1997) in the context of distance metrics for point sets. They show that edge covers can be reduced to minimum weight perfect matchings of an auxiliary bipartite graph with two partitions of size $|U_s| + |U_t|$. This allows the computation of general minimum weight edge covers in time $O((|U_s| + |U_t|)^3)$.

4 Filtering via Tree Pruning

We introduce two filtering techniques which effectively remove constituents from source and target trees before alignment takes place. Tree pruning as a preprocessing step is more general and more efficient than our original post-processing filter (Padó and Lapata, 2005) which was embedded into the similarity function. Not only does tree pruning not interfere with the similarity function but also reduces the size of the graph, thus speeding up the algorithms discussed in the previous section.

We present two instantiations of tree pruning: word-based filtering, which subsumes our earlier method, and argument-based filtering, which eliminates unlikely argument candidates.

Word-based filtering This technique removes terminal nodes from parse trees according to certain linguistic or alignment-based criteria. We apply two word-based filters in our experiments. The first removes non-content words, i.e., all words which are not adjectives, adverbs, verbs, or nouns, from the source and target sen-
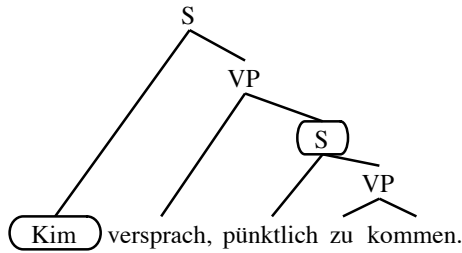
Figure 3: Filtering of unlikely arguments (predicate in boldface, potential arguments in boxes).

tences (Padó and Lapata, 2005). We also use a novel filter which removes all words which remain unaligned in the automatic word alignment. Nonterminal nodes whose terminals are removed by these filters, are also pruned.

Argument filtering   Previous work in shallow semantic parsing has demonstrated that not all nodes in a tree are equally probable as semantic roles for a given predicate (Xue and Palmer, 2004). In fact, assuming a perfect parse, there is a "set of likely arguments", to which almost all semantic roles roles should be assigned to. This set of likely arguments consists of all constituents which are a child of some ancestor of the predicate, provided that (a) they do not dominate the predicate themselves and (b) there is no sentence boundary between a constituent and its predicate. This definition covers long-distance dependencies such as control constructions for verbs, or support constructions for nouns and adjectives, and can be extended slightly to accommodate coordination.

This argument-based filter reduces *target* trees to a set of likely arguments. In the example in Figure 3, all tree nodes are removed except *Kim* and *pünktlich zu kommen.*

5   Evaluation Set-up

Data   For evaluation, we used the parallel corpus[3] from our earlier work (Padó and Lapata, 2005). It consists of 1,000 English-German sentence pairs from the Europarl corpus (Koehn, 2005). The sentences were automatically parsed (using Collin's 1997 parser for English and Dubey's 2005 parser for German), and manually annotated with FrameNet-like semantic roles (see Padó and Lapata 2005 for details.)

Word alignments were computed with the GIZA++ toolkit (Och and Ney, 2003), using the

entire English-German Europarl bitext as training data (20M words). We used the GIZA++ default settings to induce alignments for both directions (source-target, target-source). Following common practise in MT (Koehn et al., 2003), we considered only their intersection (bidirectional alignments are known to exhibit high precision). We also produced manual word alignments for all sentences in our corpus, using the GIZA++ alignments as a starting point and following the Blinker annotation guidelines (Melamed, 1998).

Method and parameter choice   The constituent alignment models we present are *unsupervised* in that they do not require labelled data for inferring correct alignments. Nevertheless, our models have three parameters: (a) the similarity measure for identifying semantically equivalent constituents; (b) the filtering procedure for removing noise in the data (e.g., wrong alignments); and (c) the decision procedure for projection.

We retained the similarity measure introduced in Padó and Lapata (2005) which computes the overlap between a source constituent and its candidate projection, in both directions. Let $y(c_s)$ and $y(c_t)$ denote the yield of a source and target constituent, respectively, and $al(T)$ the union of all word alignments for a token set T:

$$sim(c_s, c_t) = \frac{|y(c_t) \cap al(y(c_s))|}{|y(c_s)|} \frac{|y(c_s) \cap al(y(c_t))|}{|y(c_t)|}$$

We examined three filtering procedures (see Section 4): removing non-aligned words (NA), removing non-content words (NC), and removing unlikely arguments (Arg). These were combined with three decision procedures: local forward alignment (Forward), perfect matching (PerfMatch), and edge cover matching (EdgeCover) (see Section 3). We used Jonker and Volgenant's (1987) solver[4] to compute weighted perfect matchings.

In order to find optimal parameter settings for our models, we split our corpus randomly into a development and test set (both 50% of the data) and examined the parameter space exhaustively on the development set. The performance of the best models was then assessed on the test data. The models had to predict semantic roles for German, using English gold standard roles as input, and were evaluated against German gold standard

---

[3]The corpus can be downloaded from `http://www. coli.uni-saarland.de/~pado/projection/`.

[4]The software is available from `http://www. magiclogic.com/assignment.html`.

	Model	Prec	Rec	F-score
No Filter	WordBL	45.6	44.8	45.1
	Forward	66.0	56.5	60.9
	PerfMatch	**71.7**	**54.7**	**62.1**
	EdgeCover	65.6	57.3	61.2
	UpperBnd	85.0	84.0	84.0

	Model	Prec	Rec	F-score
NC Filter	WordBL	45.6	44.8	45.1
	Forward	64.3	47.8	54.8
	PerfMatch	**73.1**	**56.9**	**64.0**
	EdgeCover	67.5	57.0	61.8
	UpperBnd	85.0	84.0	84.0

	Model	Prec	Rec	F-score
NA Filter	WordBL	45.6	44.8	45.1
	Forward	74.1	56.1	63.9
	PerfMatch	**73.3**	**62.1**	**67.2**
	EdgeCover	70.5	62.9	66.5
	UpperBnd	85.0	84.0	84.0

	Model	Prec	Rec	F-score
Arg Filter	WordBL	45.6	44.8	45.1
	Forward	**69.9**	**60.7**	**65.0**
	PerfMatch	80.4	48.1	60.2
	EdgeCover	69.6	60.6	64.8
	UpperBnd	85.0	84.0	84.0

Table 1: Model comparison using intersective alignments (development set)

roles. To gauge the extent to which alignment errors are harmful, we present results both on intersective and manual alignments.

Upper bound and baseline In Padó and Lapata (2005), we assessed the feasibility of semantic role projection by measuring how well annotators agreed on identifying roles and their spans. We obtained an inter-annotator agreement of 0.84 (F-score), which can serve as an upper bound for the projection task. As a baseline, we use a simple word-based model (WordBL) from the same study. The units of this model are words, and the span of a projected role is the union of all target terminals aligned to a terminal of the source role.

6 Results

Development set Our results on the development set are summarised in Table 1. We show how performance varies for each model according to different filtering procedures when automatically produced word alignments are used. No filtering is applied to the baseline model (WordBL).

Without filtering, local and global models yield comparable performance. Models based on perfect bipartite matchings (PerfMatch) and edge covers (EdgeCover) obtain slight F-score improvements over the forward alignment model (Forward). It is worth noticing that PerfMatch yields a significantly higher precision (using a χ^2 test, $p < 0.01$) than Forward and EdgeCover. This indicates that, even without filtering, PerfMatch delivers rather accurate projections, however with low recall.

Model performance seems to increase with tree pruning. When non-aligned words are removed (Table 1, NA Filter), PerfMatch and EdgeCover reach an F-score of 67.2 and 66.5, respectively. This is an increase of approximately 3% over the local Forward model. Although the latter model yields high precision (74.1%), its recall is significantly lower than PerfMatch and EdgeCover ($p < 0.01$). This demonstrates the usefulness of filtering for the more constrained global models which as discussed in Section 3 can only represent a limited set of alignment possibilities.

The non-content words filter (NC filter) yields smaller improvements. In fact, for the Forward model, results are worse than applying no filtering at all. We conjecture that NC is an overly aggressive filter which removes projection-critical words. This is supported by the relatively low recall values. In comparison to NA, recall drops by 8.3% for Forward and by almost 6% for PerfMatch and EdgeCover. Nevertheless, both PerfMatch and EdgeCover outperform the local Forward model. PerfMatch is the best performing model reaching an F-score of 64.0%.

We now consider how the models behave when the argument-based filter is applied (Arg, Table 1, bottom). As can be seen, the local model benefits most from this filter, whereas PerfMatch is worst affected; it obtains its highest precision (80.4%) as well as its lowest recall (48.1%). This is somewhat expected since the filter removes the majority of nodes in the target partition causing a proliferation of dummy nodes. The resulting edge covers are relatively "unnatural", thus counterbalancing the advantages of global optimisation.

To summarise, we find on the development set that PerfMatch in the NA Filter condition obtains the best performance (F-score 67.2%), followed closely by EdgeCover (F-score 66.5%) in the same

	Model	Prec	Rec	F-score
Intersective	WordBL	45.7	45.0	43.3
	Forward (Arg)	72.4	63.2	67.5
	PerfMatch (NA)	**75.7**	**63.7**	**69.2**
	EdgeCover (NA)	73.0	64.9	68.7
	UpperBnd	85.0	84.0	84.0

	Model	Prec	Rec	F-score
Manual	WordBL	62.1	60.7	61.4
	Forward (Arg)	72.2	68.6	70.4
	PerfMatch (NA)	**75.7**	**67.5**	**71.4**
	EdgeCover (NA)	71.9	69.3	70.6
	UpperBnd	85.0	84.0	84.0

Table 2: Model comparison using intersective and manual alignments (test set)

condition. In general, PerfMatch seems less sensitive to the type of filtering used; it yields best results in three out of four filtering conditions (see boldface figures in Table 1). Our results further indicate that Arg boosts the performance of the local model by guiding it towards linguistically appropriate alignments.[5]

A comparative analysis of the output of PerfMatch and EdgeCover revealed that the two models make similar errors (85% overlap). Disagreements, however, arise with regard to misparses. Consider as an example the sentence pair:

The Charter is [$_{NP}$ an opportunity to bring the EU closer to the people.]

Die Charta ist [$_{NP}$ eine Chance], [$_S$ die EU den Bürgern näherzubringen.]

An ideal algorithm would align the English NP to both the German NP and S. EdgeCover, which can model one-to-many-relationships, acts "confidently" and aligns the NP to the German S to maximise the overlap similarity, incurring both a precision and a recall error. PerfMatch, on the other hand, cannot handle one-to-many relationships, acts "cautiously" and aligns the English NP to a dummy node, leading to a recall error. Thus, even though EdgeCover's analysis is partly right, it will come out worse than PerfMatch, given the current dataset and evaluation method.

Test set We now examine whether our results carry over to the test data. Table 2 shows the

performance of the best models (Forward (Arg), PerfMatch (NA), and EdgeCover (NA)) on automatic (Intersective) and manual (Manual) alignments.[6] All models perform significantly better than the baseline but significantly worse than the upper bound (both in terms of precision and recall, $p < 0.01$). PerfMatch and EdgeCover yield better F-scores than the Forward model. In fact, PerfMatch yields a significantly better precision than Forward ($p < 0.01$).

Relatively small performance gains are observed when manual alignments are used. The F-score increases by 2.9% for Forward, 2.2% for PerfMatch, and 1.9% for EdgeCover. Also note that this better performance is primarily due to a significant increase in recall ($p < 0.01$), but not precision. This is an encouraging result indicating that our filters and graph-based algorithms eliminate alignment noise to a large extent. Analysis of the models' output revealed that the remaining errors are mostly due to incorrect parses (none of the parsers employed in this work were trained on the Europarl corpus) but also to modelling deficiencies. Recall from Section 3 that our global models cannot currently capture one-to-zero correspondences, i.e., deletions and insertions.

7 Related work

Previous work has primarily focused on the projection of grammatical (Yarowsky and Ngai, 2001) and syntactic information (Hwa et al., 2002). An exception is Fung and Chen (2004), who also attempt to induce FrameNet-style annotations in Chinese. Their method maps English FrameNet entries to concepts listed in HowNet[7], an on-line ontology for Chinese, without using parallel texts.

The present work extends our earlier projection framework (Padó and Lapata, 2005) by proposing global methods for automatic constituent alignment. Although our models are evaluated on the semantic role projection task, we believe they also show promise in the context of statistical machine translation. Especially for systems that use syntactic information to enhance translation quality. For example, Xia and McCord (2004) exploit constituent alignment for rearranging sentences in the source language so as to make their word or-

---

[5]Experiments using different filter combinations did not lead to performance gains over individual filters and are not reported here due to lack of space.

[6]Our results on the test set are slightly higher in comparison to the development set. The fluctuation reflects natural randomness in the partitioning of our corpus.

[7]See http://www.keenage.com/zhiwang/e_zhiwang.html.

der similar to that of the target language. They learn tree reordering rules by aligning constituents heuristically using a naive local optimisation procedure analogous to forward alignment. A similar approach is described in Collins et al. (2005); however, the rules are manually specified and the constituent alignment step reduces to inspection of the source-target sentence pairs. The global optimisation models presented in this paper could be easily employed for the reordering task common to both approaches.

Other approaches treat rewrite rules not as a preprocessing step (e.g., for reordering source strings), but as a part of the translation model itself (Gildea, 2003; Gildea, 2004). Constituent alignments are learnt by estimating the probability of tree transformations, such as node deletions, insertions, and reorderings. These models have a greater expressive power than our edge cover models; however, this implies that approximations are often used to make computation feasible.

8 Conclusions

In this paper, we have proposed a novel method for obtaining constituent alignments between parallel sentences and have shown that it is useful for semantic role projection. A key aspect of our approach is the formalisation of constituent alignment as the search for a minimum weight edge cover in a bipartite graph. This formalisation provides efficient mechanisms for aligning constituents and yields results superior to heuristic approaches. Furthermore, we have shown that tree-based noise filtering techniques are essential for good performance.

Our approach rests on the assumption that constituent alignment can be determined solely from the lexical similarity between constituents. Although this allows us to model constituent alignments efficiently as edge covers, it falls short of modelling translational divergences such as substitutions or insertions/deletions. In future work, we will investigate minimal tree edit distance (Bille, 2005) and related formalisms which are defined on tree structures and can therefore model divergences explicitly. However, it is an open question whether cross-linguistic syntactic analyses are similar enough to allow for structure-driven computation of alignments.

References

P. Bille. 2005. A survey on tree edit distance and related problems. *Theoretical Computer Science*, 337(1-3):217–239.

H. C. Boas. 2005. Semantic frames as interlingual representations for multilingual lexical databases. *International Journal of Lexicography*, 18(4):445–478.

X. Carreras, L. Màrquez, eds. 2005. *Proceedings of the CoNLL shared task: Semantic role labelling*, Boston, MA, 2005.

M. Collins, P. Koehn, I. Kučerová. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd ACL*, 531–540, Ann Arbor, MI.

M. Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the ACL/EACL*, 16–23, Madrid, Spain.

A. Dubey. 2005. What to do when lexicalization fails: parsing German with suffix analysis and smoothing. In *Proceedings of the 43rd ACL*, 314–321, Ann Arbor, MI.

T. Eiter, H. Mannila. 1997. Distance measures for point sets and their computation. *Acta Informatica*, 34(2):109–133.

C. J. Fillmore, C. R. Johnson, M. R. Petruck. 2003. Background to FrameNet. *International Journal of Lexicography*, 16:235–250.

M. L. Fredman, R. E. Tarjan. 1987. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM*, 34(3):596–615.

P. Fung, B. Chen. 2004. BiFrameNet: Bilingual frame semantics resources construction by cross-lingual induction. In *Proceedings of the 20th COLING*, 931–935, Geneva, Switzerland.

D. Gildea, D. Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

D. Gildea. 2003. Loosely tree-based alignment for machine translation. In *Proceedings of the 41st ACL*, 80–87, Sapporo, Japan.

D. Gildea. 2004. Dependencies vs. constituents for tree-based alignment. In *Proceedings of the EMNLP*, 214–221, Barcelona, Spain.

C. Hi, R. Hwa. 2005. A backoff model for bootstrapping resources for non-english languages. In *Proceedings of the HLT/EMNLP*, 851–858, Vancouver, BC.

R. Hwa, P. Resnik, A. Weinberg, O. Kolak. 2002. Evaluation of translational correspondence using annotation projection. In *Proceedings of the 40th ACL*, 392–399, Philadelphia, PA.

R. Jonker, T. Volgenant. 1987. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38:325–340.

P. Koehn, F. J. Och, D. Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the HLT/NAACL*, 127–133, Edmonton, AL.

P. Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the MT Summit X*, Phuket, Thailand.

I. D. Melamed. 1998. Manual annotation of translational equivalence: The Blinker project. Technical Report IRCS TR #98-07, IRCS, University of Pennsylvania, 1998.

F. J. Och, H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–52.

S. Padó, M. Lapata. 2005. Cross-lingual projection of role-semantic information. In *Proceedings of the HLT/EMNLP*, 859–866, Vancouver, BC.

F. Xia, M. McCord. 2004. Improving a statistical MT system with automatically learned rewrite patterns. In *Proceedings of the 20th COLING*, 508–514, Geneva, Switzerland.

N. Xue, M. Palmer. 2004. Calibrating features for semantic role labeling. In *Proceedings of the EMNLP*, 88–94, Barcelona, Spain.

D. Yarowsky, G. Ngai. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the HLT*, 161–168, San Diego, CA.