One Tree is not Enough: Cross-lingual Accumulative Structure Transfer for Semantic Indeterminacy

Patrick Ziering
Institute for Natural Language Processing
University of Stuttgart, Germany
Patrick.Ziering
@ims.uni-stuttgart.de

Lonneke van der Plas
Institute of Linguistics
University of Malta, Malta
Lonneke.vanderPlas@um.edu.mt

Abstract

We address the task of parsing semantically indeterminate expressions, for which several correct structures exist that do not lead to differences in meaning. We present a novel non-deterministic structure transfer method that accumulates all structural information based on cross-lingual word distance derived from parallel corpora. Our system's output is a ranked list of trees. To evaluate our system, we adopted common IR metrics. We show that our system outperforms previous cross-lingual structure transfer methods significantly. In addition, we illustrate that tree accumulation can be used to combine partial evidence across languages to form a single structure, thereby making use of sparse parallel data in an optimal way.

1 Introduction

Parsing linguistic expressions (e.g., noun phrases (NPs)) is a fundamental component in many natural language processing (NLP) tasks like machine translation (MT) or information retrieval (IR) and indispensable for understanding the meaning of complex units. For example, while [*natural language*] *processing* means the (machine) processing of natural languages, *natural* [*language processing*] denotes the natural processing of (any) languages.

As previous work has shown, multilingual data can help resolving various kinds of structural ambiguity such as prepositional phrase (PP) attachment (Schwartz et al., 2003; Fossum and Knight, 2008), subject/object distinction (Schwarck et al., 2010) or coordination ellipsis (Bergsma et al., 2011). Parallel sentences have been jointly parsed supported by word alignment features (Smith and Smith, 2004; Burkett and Klein, 2008). Yarowsky and Ngai (2001) project part-of-speech (PoS) tags and basic NP structures across languages. Hwa et al., (2005) use projected tree structures for bootstrapping new non-English parsers. Unsupervised multilingual grammar induction has been performed on parallel corpora (Snyder et al., 2009) and on non-parallel data (Berg-Kirkpatrick and Klein, 2010; Iwata et al., 2010).

In addition to previous work focused on disambiguation, we show that multilingual data can be used to point to semantic indeterminacy. Syntactic structures are usually understood deterministically in that for every structure there exists conditions that can have no other structure. However, previous work in NLP shows that such a deterministic take might not always be suitable. Hindle and Rooth (1993) were the first to discuss the phenomenon of **semantic indeterminacy** in PP attachment, e.g., in the sentence *They mined the roads along the coast*, the PP *along the coast* may be attached to both the verb or the object without changing the meaning. On the NP level, Lauer (1995) observed 12.54% semantically indeterminate three-noun compounds (3NCs) in his dataset, e.g., in 'Most advanced aircraft have *precision navigation systems*', both *precision navigation* and *navigation system* can be bracketed leading to the same meaning. We found more striking evidence from parallel corpora, where the multiple translations found for a given NP reflect large differences in structure. While *tobacco advertising ban* is translated to German as *Werbeverbot für Tabakerzeugnisse* (advertising ban for tobacco products), the Danish equivalent is *forbuddet mod tobaksreklamer* (ban of tobacco advertising). Similarly, *animal welfare standards* is once translated to Dutch as *normen op het gebied van dierenwelzijn* (standards in the field of animal welfare) and to German as *Wohlfahrtsstandards für Tiere* (welfare standards for animals).

Despite the fact that previous work discussed

Proceedings of Recent Advances in Natural Language Processing, pages 739–746,
Hissar, Bulgaria, Sep 7–9 2015.

semantic indeterminacy, to the best of our knowledge, no attempt has been made to include this phenomenon in syntactic analysis. Vadas (2009) argues that in most cases the intended structure is unambiguous[1] and therefore chooses not to include semantic indeterminacy in his NP annotation of the Penn Treebank (Marcus et al., 1993). We believe that it is important to include semantic indeterminacy in NLP, e.g., an anaphora resolver needs to know the structural equivalence for finding all possible nested antecedents, e.g., both *animal welfare* and *welfare standards*.

This work aims at capturing semantic indeterminacy within a structural analysis. We exploit cross-linguality for this task because structural variation for semantic indeterminacy is visible in particular across languages. In a monolingual approach, we expect less variation, due to conventional language use. As a result, parse forests resulting from monolingual data would therefore be less rich in variation. We transfer syntactic structure from cross-lingual surface variation directly, without inducing grammars or annotating the source language. We coin the term **cross-lingual structure transfer** (CST) for this method.

Our system is inspired by Ziering and Van der Plas (2015), who exploit cross-lingual surface variation for bracketing 3NCs. There are various ways of translating English noun compounds. Germanic languages such as Swedish frequently use closed compounds (i.e., single nouns), whereas Romance languages such as French use open compounds (i.e., lexemes composed of several words). Paraphrased translations (e.g., *human rights abuse* aligned to the partially closed German *Verletzung der Menschenrechte* (abuse of human rights)) can reveal the internal structure of a compound. While Ziering and Van der Plas (2015) follow the deterministic take by producing a single tree output, we gather all structural information and produce a ranked list of plausible trees, where similarly-ranked trees indicate semantic indeterminacy.

Our contributions are as follows: we develop a non-deterministic cross-lingual structure transfer method which is suitable for dealing with semantic indeterminacy. We present two models that differ in granularity. The coarse-grained model restricts to full structures acquired from various languages. The fine-grained model also includes substructures, which makes it more robust against word alignment errors, and points to an intended structure. Inspired by IR metrics, we treat CST as a kind of structure retrieval and propose an evaluation method that measures quality and quantity of retrieved structures. In a case study, we present results on processing 3NCs and 4NCs. Finally, we illustrate how our methods can be used to combine partial evidence across languages to form a single structure, where individual languages fail. This way, we are able to exploit more data from sparse parallel corpora than previous work.

2 Cross-lingual Structure Transfer

Linguistic expressions, such as kNCs, occurring in parallel data have been processed in previous work using cross-lingual aligned word distance:

$$\text{AWD}(c_i, c_j) = \min_{x \in AW_i, y \in AW_j} |pos(x) - pos(y)|$$

where AW_n is the set of aligned content words of a constituent c_n and $pos(\alpha)$ is the position of a word α in an aligned sentence. Inspired by Behaghel's (1909) First Law saying that elements which belong close together intellectually will also be placed close together, the AWD of constituents functions as indicator for the semantic cohesion. For example, the 3NC *human rights violations* being aligned to the Italian *le **violazioni** gravi e sistematiche dei **diritti umani*** indicates that *human rights* (*diritti umani*) has a stronger cohesion than *rights violations* (*violazioni* ... *diritti*), which points to a left-branched structure in English. Ziering and Van der Plas (2015) developed an AWD-based bracketing system applied on English kNCs in a parallel corpus. For each aligned language, they start bottom-up with one constituent per noun. They compare the AWDs between all adjacent constituents and iteratively merge the constituent pair with the smallest AWD until there is only one constituent left. If there is a tie among the possible AWDs, the system does not produce a tree structure. For the final decision, Ziering and Van der Plas (2015) use the majority vote across all aligned languages. If this number is not unique, the system is undecided. The main limitation of this system is that it provides a deterministic result both for each individual language and for the majority vote. As a consequence, the system neither allows several struc-

[1] As example, Vadas (2009) mentions *American President George Bush*, where the intended structure is [*American President*] *George Bush*, because Bush's nationality is not relevant but his political function.

tures for a semantically indeterminate target nor combines partial results from several languages to a final structure. Subsequently, we will refer to Ziering and Van der Plas' (2015) **l**anguage-**i**solated **d**eterministic **s**tructure **t**ransfer as LIDST.

2.1 Full Tree Accumulation Structure Transfer

In the **f**ull tree **a**ccumulation **s**tructure **t**ransfer system (FAST), we consider all possible binary tree structures of an expression. Among those, there are demoted structures for a given language, because they combine constituents that have a stronger semantic cohesion than their subparts. For example, *air* [*traffic control*] is demoted for the Dutch paraphrase *controle van het luchtvaartverkeer* (control of air traffic), because *air traffic* has the strongest cohesion (as being aligned to a closed compound). For a given English expression Ψ, FAST is applied to each aligned language, as shown in Figure 1.

1: $Trees \Leftarrow$ create all binary tree structures
2: **for** t in $Trees$ **do**
3: annotate all nodes N in t with AWD
4: **if** $\exists N[N.\text{AWD} > \text{mother}(N).\text{AWD}]$ **then**
5: $t.invalid \Leftarrow$ TRUE
6: **end if**
7: **end for**
8: **return** $\{t \in Trees \mid \text{not } t.invalid\}$

Figure 1: FAST algorithm

We first create all possible binary trees for Ψ (line 1). The number of possible binary trees increases with the Catalan numbers (Church and Patil, 1982), e.g., 3NCs have two possible trees (i.e., left- or right-branched), 4NCs have five possible trees and kNCs have C_{k-1} possible trees, where C_n is the n-th Catalan number as given in (1).

$$C_n = \frac{(2n)!}{(n+1)! \cdot n!} \quad (1)$$

All tree nodes N_i in these trees are annotated with AWD numbers (line 3) according to (2), i.e., leaf nodes get zero AWD and other nodes are annotated with the AWD between their left and right children's constituent.

$$N_i.AWD = \begin{cases} \text{leaf}(N_i) & \mapsto 0 \\ \text{else} & \mapsto AWD(N_i.\text{L}, N_i.\text{R}) \end{cases} \quad (2)$$

In the next step, all annotated trees are validated (lines 4-6). A tree is **valid**, if its AWD annotation is monotonically decreasing when traversing the tree top down. If there is a node N whose AWD is larger than the AWD of its mother node, the tree is marked as invalid. Finally, we return the set of tree structures which are not marked as invalid (line 8).

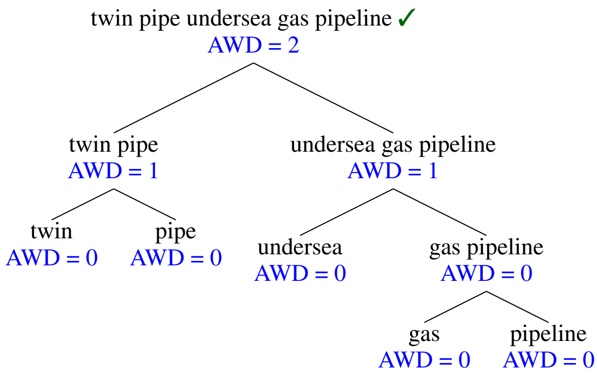

Figure 2: A valid FAST tree structure

Figure 2 shows an example of a valid AWD-annotated tree structure of the 5NC *twin pipe undersea gas pipeline* aligned to the Dutch paraphrase *onderzeese gaspijpleiding met dubbele pijp* (undersea {gas pipeline} with twin pipes).

In the final step, we put all valid trees from all languages into a tree accumulation (TA) and rank them by frequency (i.e., trees being valid in most cases are ranked first). For example, for the semantically indeterminate *air traffic control centres*, FAST assigns the same top rank to the semantically equivalent structures as shown in Table 1.

Rank	Structure	TA
1	[*air traffic*] [*control centres*]	13
1	[[*air traffic*] *control*] *centres*	13
2	[*air* [*traffic control*]] *centres*	10

Table 1: FAST top-ranking for *air traffic control centres*

In addition to a token-based setting, FAST can also be applied on expression types. In this case, we put all valid trees from all aligned languages of all instances of Ψ into the TA.

2.2 Subtree Accumulation Structure Transfer

In some cases, an invalid full tree (✗$_{ft}$) still contains an informative valid[2] subtree[3] (✓$_{st}$) as shown

[2] We use the same validity conditions as for FAST.

[3] A subtree st of a full tree ft is a tree consisting of a node in ft and all of its descendants.

in Figure 3 for the 4NC *church development aid projects* being aligned to the Italian ***progetti ecclesiastici** di **aiuti** allo **sviluppo*** (lit.: projects ecclesiastical of aid to development).

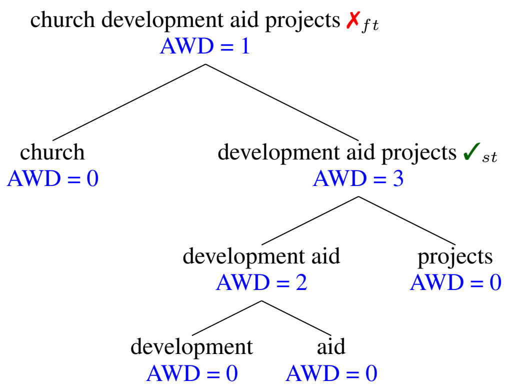

Figure 3: Invalid FAST full tree with valid subtree

The Italian translation does not provide any valid full tree, because the smallest AWD is between c_1, *church* (ecclesiastici), and c_4, *projects* (progetti). Thus, the AWD-annotation of the root node is always 1, which is smaller than any annotations below.

For exploiting as much evidence as possible from sparse parallel data, the **s**ubtree **a**ccumulation **s**tructure **t**ransfer system (SAST) takes into account all valid subtrees from both valid and invalid full trees. After gathering all valid subtrees among all full trees for an expression Ψ in all aligned languages $l \in L$, each subtree gets a subtree score (sts) according to (3), where $freq(st)$ is the number of aligned languages, $|L|$, multiplied by the Δ-th Catalan number, where Δ is the difference in the number of leaf nodes between ft and st.

$$\begin{aligned} sts(st) &= \frac{freq(st.\text{valid})}{freq(st)} \\ &= \frac{freq(st.\text{valid})}{|L| \cdot C_\Delta} \end{aligned} \tag{3}$$

A full tree gets a full tree score (fts), which is the product[4] of all its subtree scores (4).

$$fts(ft) = \prod_{st \in ft} sts(st) \tag{4}$$

In the last step, we rank all full trees according to their fts (i.e., the tree that has the highest fts is ranked first).

[4]While the product performs better in our setup, the sum would be an alternative for cases where no language provides any valid full tree (i.e., the largest subtree).

In contrast to FAST, SAST produces a more fine-grained scoring by exploiting more data. While this approach is more robust to word alignment errors, it also points to an intended structure, e.g., *air traffic control centres* gets a single top-ranked structure as shown in Table 2.

Rank	Structure	fts
1	[[*air traffic*] [*control centres*]]	1.66
2	[[[*air traffic*] *control*] *centres*]	1.35

Table 2: SAST top-ranking for *air traffic control centres*

For our initial example, Figure 4 shows two full tree structures for *church development aid projects* annotated with fts and sts information in SAST applied on a language ensemble including German, French and Italian. While FAST would give both trees the same rank (not shown), SAST exploits the higher prominence of the valid subtree in Figure 3 and thus ranks the tree in Figure 4.1 highest.

In analogy with FAST, SAST can also be applied type-based. In this case, we sum up all full tree scores from all instances of Ψ and rank the structures according to this sum.

3 Experiments

3.1 Dataset

We extracted 3NCs and 4NCs from the initial version (basic dataset) of the Europarl[5] compound database[6] (Ziering and van der Plas, 2014), compiled from the OPUS[7] corpus (Tiedemann, 2012). The database contains 10 European languages in three language families: Germanic (English, Danish, Dutch, German and Swedish), Romance (French, Italian, Portuguese and Spanish) and Hellenic (Greek). The kNCs are extracted using PoS patterns conforming a sequence of k adjacent nouns. The dataset contains 24,848 3NC tokens (16,565 types) and 1468 4NC tokens (1257 types).

3.2 Gold Standard

We use the 3NC test set[8] created by Ziering and Van der Plas (2015), which contains 278 left- or right-branched and 120 semantically indeterminate 3NC tokens. For keeping the ratio of 3NCs

[5]statmt.org/europarl
[6]ims.unistuttgart.de/data/NCDatabase.html
[7]opus.lingfil.uu.se
[8]ims.uni-stuttgart.de/data/AWDB.data.tgz

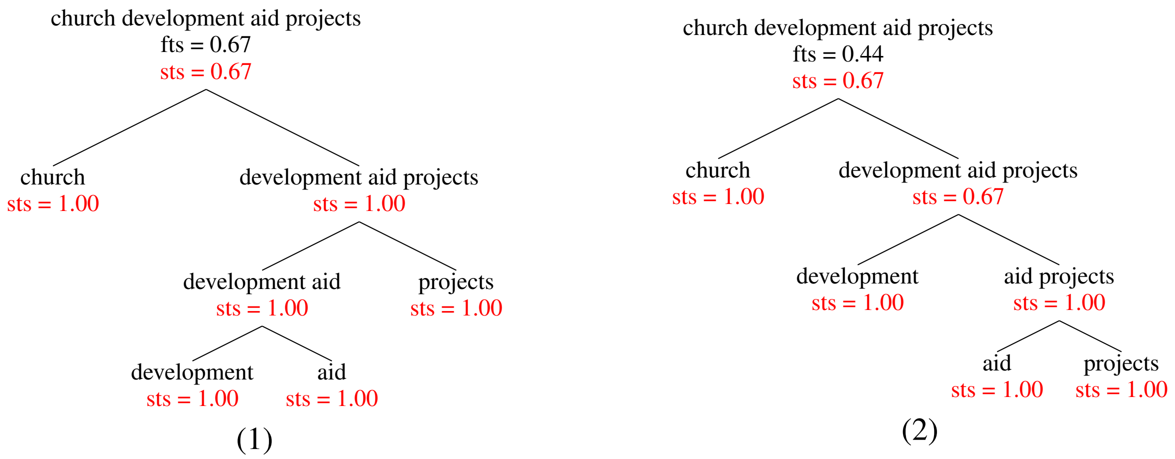

Figure 4: SAST trees for *church development aid projects* on first and second position

to 4NCs as reflected in the token numbers of our dataset, we decided on a random set of 50 4NC samples to be labeled by two trained independent annotators. We adopted the annotation guidelines described in Vadas (2009) and use the following labels for annotating 4NCs: 1, ..., 5 (referring to the five possible 4NC structures), EXTRACTION (for extraction errors, i.e., incomplete NCs or fragments of incomplete constituents as in *climate change target* ***cannot***), UNDECIDED[i; $\ldots$; j] (for cases in which the context cannot help to disambiguate between the distinctive structures i, $\ldots$, j), FLAT (for expressions showing no internal structure (e.g., *John A. Smith*)) and SEMANTIC INDETERMINACY[i; $\ldots$; j] (for expressions with the equivalent structures i; $\ldots$; j). For addressing semantic indeterminacy, we take the union of single structure labels and semantic indeterminacy labels from both annotators to a test set comprising 33 4NC tokens and discard 17 tokens, which have been tagged as extraction error.

Structure pattern	Frequency									
	13	6	5	2		1				
A [B [C D]]		*					*	*	*	
A [[B C] D]				*				*	*	*
[A B] [C D]	*		*			*	*	*		
[A [B C]] D						*				*
[[A B] C] D	*				*	*				

Table 3: Frequency distribution of structures in the 4NC test set

Table 3 shows the frequency distribution of 4NC structures in the test set, where structures are represented as structure patterns[9]. Analogously to the majority class LEFT for 3NCs, the structure combination having the two left-most nouns grouped as a constituent is annotated most often.

3.3 Structure Retrieval

Our system's output is a ranked list of tree structures. Inspired by IR models, we treat CST as a kind of structure retrieval and measure how well a ranking fits to the set of gold trees. Therefore, we adapt the R-Precision score (Buckley and Voorhees, 2000) as given in (5):

$$\text{R-Prec}(k\text{NC}) = \frac{|\text{top-R}(sys\ trees) \cap gold\ trees|}{|\text{top-R}(sys\ trees)|} \quad (5)$$

where R is the number of gold trees and top-R($sys\ trees$) refers to the R highest-ranked system trees. For trees having the same rank, we choose a random order. If there are less than R system trees, the ranking is randomly complemented. Observing that this random process lead to unstable numbers, we apply it 1000 times and take the average of the resulting scores. The mean R-Precision takes the macro average of the R-Precision scores as given in (6)

$$\text{MRP} = \frac{\sum_{\Psi \in \Omega} \text{R-Prec}(\Psi)}{|\Omega|} \quad (6)$$

where Ω is the set of all expressions. In addition, we measure precision at k ($P@k$) and recall at k ($R@k$) as given in (7) and (8). We present the macro average for $P@k$ as $MP@k$ and for $R@k$ as $MR@k$. Macro F1 at k is the harmonic mean of $MP@k$ and $MR@k$. Since semantically indeterminate kNCs have about two gold trees, we evaluate the systems for $1 \leq k \leq 2$.

[9]Structure patterns are generalized structures such as [A B] [C D] for [*air traffic*] [*control centres*].

System	MRP	MP@1	MR@1	MF1@1	MP@2	MR@2	MF1@2
FAST	**70.0%**	**72.7%**	**47.5%**	**57.5%**	60.6%	74.2%	66.7%
SAST	69.5%	69.7%	44.4 %	54.2%	**63.6%**	**78.8%**	**70.4%**
LIDST	54.5%‡	69.7%	44.4%	54.2%	47.0% ‡	59.1% ‡	52.4%‡
LINDST	62.9%‡	69.7%	44.4%	54.2%	54.5% †	66.7% †	60.0 %†
UPPER	86.0%	96.7%	67.2%	79.3%	70.0%	87.8%	77.9%
FREQ	60.1%	63.6%	38.4%	47.9%	56.1%	65.2%	60.3%
CHANCE	32.0%	39.4%	23.7%	29.6%	33.3 %	42.4 %	37.3%

Table 4: Results on CST of 4NCs; ‡ means significantly outperformed by FAST and SAST; † means significantly outperformed by FAST or SAST

$$P@k = \frac{|\text{top-}k(sys\ trees) \cap gold\ trees|}{|\text{top-}k(sys\ trees)|} \tag{7}$$

$$R@k = \frac{|\text{top-}k(sys\ trees) \cap gold\ trees|}{|gold\ trees|} \tag{8}$$

3.4 Models in Comparison

We compare FAST and SAST against LIDST. While this system uses the majority vote as deterministic output, we add a further system by ranking all trees by vote frequency and evaluate this ranking as the **l**anguage-**i**solated **n**on-**d**eterministic **s**tructure **t**ransfer, LINDST. As baselines, we use the random baseline, CHANCE, that creates an arbitrary tree ranking, and the frequency baseline, FREQ, that creates a tree ranking according to the structure pattern frequencies in the test set (i.e., the tree with the most frequent structure pattern is ranked first), e.g., [A B] [C D] is most often annotated as shown in Table 3. To calculate an upper bound, one of the authors provided an additional annotation of the 4NC test set, UPPER. Since Ziering and Van der Plas (2015) showed that CST on kNCs works best in a type-based setting, we evaluate all models on types.

3.5 Results and Discussion

Table 5 shows the results of the mean R-Precision (MRP) on the test set of 3NCs and 4NCs. All CST systems outperform the baselines. Moreover, FAST and SAST outperform LIDST and LINDST, but differences are small.

Because the annotations suggest that 4NCs contain more semantically indeterminate structures, we expect to find larger differences between deterministic and non-deterministic CST when evaluating on 4NCs separately.

System	MRP
FAST	93.7%
SAST	**94.0%**
LIDST	92.6%
LINDST	92.0%
FREQ	84.6%
CHANCE	62.5%

Table 5: MRP results on CST of 3NC/4NCs

Table 4 shows the results on CST of 4NCs. For the mean R-Precision, FAST and SAST significantly[10] outperform LIDST and LINDST. Precision and Recall at 1 are similar for all CST methods, i.e., the top position of the systems' rankings hardly differ. For Precision and Recall at 2, FAST and SAST significantly outperform deterministic CST. Furthermore, SAST outperforms the non-deterministic LINDST significantly in MRP and Precision/Recall at 2. Beside the benefit of a non-deterministic approach for dealing with semantic indeterminacy, the global perspective of FAST and SAST makes the process more robust to word alignment errors: while the monolingually deterministic approaches merge adjacent constituent pairs on each tree level in isolation, FAST and SAST validate trees according to AWD annotations across all levels of the tree. This way, unwanted trees are demoted.

As an example, Table 6 shows the different rankings for the semantically indeterminate expression *harmful business tax regimes*, which has the two gold structures *harmful* [*business* [*tax regimes*]] and *harmful* [[*business tax*] *regimes*]. While FAST ranks both correct structures on first position (rows 1-2) and the false structure [*harmful business*] [*tax regimes*] on the second position,

[10]Approximate randomization test (Yeh, 2000), $p < 5\%$

the deterministic LIDST has decided for the false structure and the non-deterministic LINDST has at least one correct tree among the top 2 structures.

Structure Pattern	FAST	LIDST	LINDST
A [B [C D]]	1	–	2
A [[B C] D]	1	–	–
[A B] [C D]	2	1	1

Table 6: Ranking for *harmful business tax regimes*

4 Tree Accumulation for Deterministic Structure Transfer

Beside the non-deterministic structure transfer motivated by semantic indeterminacy, accumulative CST also represents a way for combining partial structure evidence from several languages into a deterministic output, where each individual language cannot provide a single structure.

For example, the determinate 4NC *energy efficiency action plan* has only one gold structure: [*energy efficiency*] [*action plan*]. A Spanish translation is ***plan** de **acción** de **eficiencia energética*** (plan of action of efficiency energy$_{ADJ}$). Since AWD(*energy efficiency*, *action*) equals AWD(*action*, *plan*), Spanish provides two possible structures: [[*energy efficiency*] *action*] *plan* and [*energy efficiency*] [*action plan*]. A German translation is ***Aktionsplan** zur **Effizienz** von **Energie*** (action plan {for the} efficiency of energy). According to German, AWD(*energy*, *efficiency*) equals AWD(*efficiency*, *action plan*). This leads to the two structures *energy* [*efficiency* [*action plan*]] and [*energy efficiency*] [*action plan*]. Since no language provides a single structure, LIDST cannot produce a deterministic output. In contrast, using tree accumulation we can combine the fact that the Spanish translation groups *energy* and *efficiency* closest together with the fact that the German equivalent puts *action* and *plan* into a closed compound. This results in the top-ranked structure: [*energy efficiency*] [*action plan*].

In an alternative scenario, the determinate 4NC *air transport industry representatives* having the gold structure [[*air transport*] *industry*] *representatives* is translated to Dutch as *vertegenwoordigers van de luchtvervoersector* (representatives of the air transport sector) and to Italian as *rappresentanti del settore del trasporto aereo* (representatives of the sector of the transport air$_{ADJ}$). Since the closed Dutch compound *luchtvervoersector* (air transport sector) hides the internal structure and the Italian paraphrase leads to AWD(*air transport*, *industry*) being equal to AWD(*industry*, *representatives*), both individual languages cannot be used for producing a single structure. However, the Dutch translation provides the information that *representatives* has to be separated from the rest and the Italian translation provides evidence for *air transport* having the strongest semantic cohesion. Accumulating all valid trees from Dutch and Italian, we get the single top-ranked structure: [[*air transport*] *industry*] *representatives*.

5 Conclusion

We have addressed semantic indeterminacy in NPs, a phenomenon often discussed, but usually discarded in previous work. We presented two models of cross-lingual structure transfer that output a ranked list of possible tree structures accumulated from parallel data. Having observed that structural variation for semantic indeterminacy is encountered in particular across languages, we applied our cross-lingual tree ranking for capturing semantically equivalent structures. To be able to evaluate our systems, we use common IR metrics. In an experiment on 3NCs and 4NCs, we showed that our methods outperform previous work significantly. Finally, we showed how tree accumulation can be used for combining partial structure evidence from various languages to form a deterministic structure output.

In future work, we will further investigate the nature of semantic indeterminacy and try to model this phenomenon using distributional semantics. Along with this paper, we publish[11] our 4NC test set, which can be used as training and test data for supervised learners.

Acknowledgments

We thank the anonymous reviewers for their helpful feedback. This research was funded by the German Research Foundation (Collaborative Research Centre 732, Project D11).

References

Otto Behaghel. 1909. Beziehungen zwischen Umfang und Reihenfolge von Satzgliedern. *Indogermanische Forschungen*.

[11] ims.uni-stuttgart.de/data/4NC.TestSet.tgz

Taylor Berg-Kirkpatrick and Dan Klein. 2010. Phylogenetic Grammar Induction. In *ACL 2010*.

Shane Bergsma, David Yarowsky, and Kenneth Church. 2011. Using Large Monolingual and Bilingual Corpora to Improve Coordination Disambiguation. In *ACL-HLT 2011*.

Chris Buckley and Ellen M. Voorhees. 2000. Evaluating Evaluation Measure Stability. In *SIGIR 2000*.

David Burkett and Dan Klein. 2008. Two Languages are Better than One (for Syntactic Parsing). In *EMNLP 2008*.

Kenneth Church and Ramesh Patil. 1982. Coping with Syntactic Ambiguity or How to Put the Block in the Box on the Table. *Computational Linguistics*.

Victoria Fossum and Kevin Knight. 2008. Using bilingual Chinese-English word alignments to resolve PPattachment ambiguity in English. In *AMTA Student Workshop 2008*.

Donald Hindle and Mats Rooth. 1993. Structural Ambiguity and Lexical Relations. *Computational Linguistics*.

Rebecca Hwa, Philip Resnik, Amy Weinberg, Clara Cabezas, and Okan Kolak. 2005. Bootstrapping Parsers via Syntactic Projection Across Parallel Texts. *Natural Language Engineering*.

Tomoharu Iwata, Daichi Mochihashi, and Hiroshi Sawada. 2010. Learning Common Grammar from Multilingual Corpus. In *ACL 2010*.

Mark Lauer. 1995. *Designing Statistical Language Learners: Experiments on Noun Compounds*. Ph.D. thesis, Macquarie University.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*.

Florian Schwarck, Alexander Fraser, and Hinrich Schütze. 2010. Bitext-Based Resolution of German Subject-Object Ambiguities. In *NAACL-HLT 2010*.

Lee Schwartz, Takako Aikawa, and Chris Quirk. 2003. Disambiguation of English PP Attachment using Multilingual Aligned Data. In *MT Summit IX*.

David A. Smith and Noah A. Smith. 2004. Bilingual Parsing with Factored Estimation: Using English to Parse Korean. In *EMNLP 2004*.

Benjamin Snyder, Tahira Naseem, and Regina Barzilay. 2009. Unsupervised Multilingual Grammar Induction. In *ACL-IJCNLP 2009*.

Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. In *LREC 2012*.

David Vadas. 2009. *Statistical Parsing of Noun Phrase Structure*. Ph.D. thesis.

David Yarowsky and Grace Ngai. 2001. Inducing Multilingual POS Taggers and NP Bracketers via Robust Projection Across Aligned Corpora. In *NAACL 2001*.

Alexander Yeh. 2000. More Accurate Tests for the Statistical Significance of Result Differences. In *COLING 2000*.

Patrick Ziering and Lonneke van der Plas. 2014. What good are 'Nominalkomposita' for 'noun compounds': Multilingual Extraction and Structure Analysis of Nominal Compositions using Linguistic Restrictors. In *COLING 2014*.

Patrick Ziering and Lonneke van der Plas. 2015. From a Distance: Using Cross-lingual Word Alignments for Noun Compound Bracketing. In *IWCS 2015*.