Advances in using
Grammars with Latent Annotations
for Discontinuous Parsing

Kilian Gebhardt

Department of Computer Science
Technische Universität Dresden
D-01062 Dresden, Germany
`kilian.gebhardt@tu-dresden.de`

Abstract

We present new experiments that transfer techniques from Probabilistic Context-free Grammars with Latent Annotations (PCFG-LA) to two grammar formalisms for discontinuous parsing: linear context-free rewriting systems and hybrid grammars. In particular, Dirichlet priors during EM training, ensemble models, and a new nonterminal scheme for hybrid grammars are evaluated. We find that our grammars are more accurate than previous approaches based on discontinuous grammar formalisms and early instances of the discriminative models but inferior to recent discriminative parsers.[1]

1 Introduction

Many tasks in natural language processing, such as machine translation, information extraction, and sentiment analysis, benefit from syntactic analysis (Culotta and Sorensen, 2004; Ding and Palmer, 2005; Duric and Song, 2011). Often syntax is represented by means of constituents. Languages with a flexible word order such as German require constituents that are *discontinuous*, i.e., constituents that cover words which do not constitute a continuous interval in the sentence. To this end, generalizations of context-free grammars such as tree adjoining grammars (Joshi et al., 1975) and *linear context-free rewriting systems* (LCFRS, Vijay-Shanker et al., 1987) have been proposed. Although the parsing complexity with these formalisms is polynomial in the length of the input sentence (for a fixed grammar), they are often considered too slow to be practically useful. Also the accuracy of LCFRS-based parsers does not match up to their continuous counterparts.

Instead, a wide range of models that either apply transition systems with a reordering mecha-

nism or are based on a dependency-to-constituency transformation have been proposed in recent years (Hall and Nivre, 2008; Versley, 2014a; Maier, 2015; Fernández-González and Martins, 2015; Coavoux and Crabbé, 2017; Corro et al., 2017; Stanojević and Garrido Alhama, 2017; Coavoux and Cohen, 2019; Fernández-González and Gómez-Rodríguez, 2020). There are two notable exceptions: van Cranenburgh et al. (2016) considers a discontinuous extension of the data-oriented parsing approach. Gebhardt (2018) studies the extension of two grammar formalisms with latent annotations: LCFRS and hybrid grammars (Nederhof and Vogler, 2014), which is a synchronous grammar formalism that couples a string generating grammar (specifically: LCFRS) and a tree generating grammar (specifically: simple definite clause programs; Deransart and Małuszyński, 1989). In particular, Gebhardt (2018) analyses the effect of a generalization of Petrov et al.'s split/merge procedure (2006), which adaptively refines the grammar's nonterminals by automatic splitting and merging. Although this refinement strategy showed vast improvements over the respective unrefined grammar, some choices in the experimental setup of Gebhardt (2018) can be enhanced:

(i) The *expectation maximization algorithm* (EM, Baker, 1979), which is a subroutine of the split/merge procedure, utilizes a likelihood-based objective that is prone to overfitting.

(ii) Ensemble-models obtained by running the split/merge procedure with different random seeds were successfully applied for continuous parsing (Petrov, 2010) but not considered in Gebhardt (2018).

(iii) Gebhardt (2018) supposes that the initial granularity of the grammar's nonterminals matters as the state-refinement procedure does not fully recover them.

---

[1]The implementation is available at `https://github.com/kilian-gebhardt/panda-parser/`.

In this work, we address the above points by (i) comparing the likelihood-based objective to a *Maximum-a-Posteriori* (MAP) objective, (ii) evaluating ensemble models, and (iii) proposing a new nonterminal naming scheme for hybrid grammars. We hypothesize that these steps are complementary in improving the accuracy of the parsing model. Although we do not expect to reach or even surpass the performance of recent discriminative approaches (in particular those utilizing neural nets), we suppose that these experiments foster the understanding of the limits of the different architectures.

2 Model

We consider models based on LCFRS and hybrid grammars. Vijay-Shanker et al. (1987) and Gebhardt et al. (2017) give formal definitions of these formalisms, respectively. *Baseline grammars* are induced from the training data using the induction techniques of Kallmeyer and Maier (2013) and Gebhardt et al. (2017), respectively. LCFRS are either binarized right-to-left (r2ℓ) or head-outward (ho) with vertical and horizontal Markovization set to 1. For hybrid grammars the induction is parametrized such that the LCFRS-component has fanout 2.

The baseline grammars are refined using a variant of Petrov et al.'s split/merge algorithm (2006), which is described in Gebhardt (2018). This algorithm consists of multiple split/merge cycles in each of which the EM algorithm is used to fit the rule weights to the training data. To obtain MAP training, we modify the EM algorithm by including a Dirichlet prior (see Johnson et al., 2007, Sec. 2.3). This is implemented by incrementing each rule count by a non-negative default value in the "expectation" phase.

Gebhardt (2018) reports that LCFRS outperform hybrid grammars but conjectures that this might be an artifact of the choice of nonterminals in the baseline grammar. The nonterminal naming schemes *child* and *strict labeling* for hybrid grammars (originally introduced by Nederhof and Vogler, 2014) name a nonterminal based on the subset of tree nodes U^2 that is generated by the nonterminal. As illustrated in fig. 1, in strict labeling the roots of maximum subtrees formed by U are used in the label, where consecutive sibling nodes are joined. In child labeling, sequences of sibling nodes of length

---

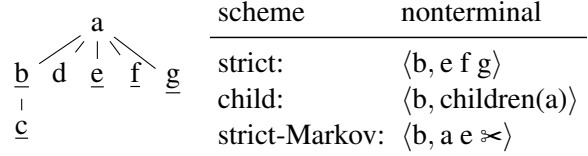|	scheme	nonterminal
	strict:	$\langle b, e\ f\ g\rangle$
	child:	$\langle b, \text{children}(a)\rangle$
	strict-Markov:	$\langle b, a\ e \bowtie\rangle$

Figure 1: A tree, a subset of its nodes (underlined), and the nonterminals created for this subset according to different nonterminal labeling schemes.

> 1 are replaced by the "children(p)" where p is the parent node. Gebhardt (2018) supposes that these strategies either lead to a number of nonterminals that is too small or too large for favorable automatic refinement. We explore a trade-off by *Markovizing* (cf. Klein and Manning, 2003) the strict labeling strategy. Where strict labeling uses sequences $s = n_1 \cdots n_k$ of sibling nodes in nonterminal labels such that $k > 1$, we now use the parent node p of $n_1 \cdots n_k$, the first node n_1, and a cut-off marker $\bowtie$. Each nonterminal label then consists of sequences $s' = p\ n_1 \bowtie$.

After split/merge refinement, the grammars are used to parse unseen sentences. Since exact parsing is NP-hard already for PCFG-LA (see Matsuzaki et al., 2005), we follow the literature and apply an approximate tractable objective called *max-rule-product* (short: mrp, which projects weights from the refined grammar to a sentence-specific coarse one and computes the the most probable derivation of the latter, see Petrov and Klein, 2007).

Petrov (2010) presents experiments with ensemble models, where different PCFG-LA are obtained from the same training data by changing the random seed. During parsing the mrp objective is applied where each rule weight of the coarse grammar is set to the product of the weights that result from projecting with the individual grammars. This ensemble grammar showed significantly improved accuracy over the best single grammar. We instantiate this approach for LCFRS and hybrid grammars using 4 random seeds.

3 Experiments

Data. We present experiments on the German TIGER (Brants et al., 2004) and NEGRA (Skut et al., 1997) corpora. For TIGER, we use the SPMRL split (Seddah et al., 2014, short: SPMRL) and the split by Hall and Nivre (2008, short: HN08) but optimize hyperparameters solely on HN08. For NEGRA we use the split by Dubey and Keller (2003). For evaluation we compute the labelled

---

[2]How these nonterminals/subsets are determined is described in Gebhardt et al. (2017).

grammar	TIGER HN08				NEGRA			
	NTs	coverage dev. set	parse fails	F1 (≤ 40)	NTs	coverage dev. set	parse fails	F1 (≤ 40)
LCFRS$_{ho}$	767	86.9%	4	68.29	716	88.2%	4	68.88
LCFRS$_{r2\ell}$	817	84.7%	4	70.36	787	82.7%	8	70.07
hybrid child	288	92.5%	11	63.19	279	93.5%	2	62.06
hybrid strict-Markov	1,783	81.9%	108	72.18	1,623	81.1%	42	71.34
hybrid strict	32,281	50.0 %	166	69.90	20,766	53.9%	34	68.82

Table 1: Statistics for the baseline grammars induced from NEGRA and TIGER (after likelihood-based training). For the dev. set we report: *coverage*, i.e., the percentage of gold parse trees that can be derived with the grammar; *parse fails*, i.e., the number of sentences for which the grammar cannot derive any parse tree; labeled F1 for the parse tree with the most probable derivation.

grammar	HN08				NEGRA			
	F1	*(ens.)*	Disc. F1	*(ens.)*	F1	*(ens.)*	Disc. F1	*(ens.)*
Likelihood-based training								
LCFRS$_{ho}$	78.85 ±0.34	79.91	32.57 ±1.91	33.81	80.14 ±0.54	81.35	39.57 ±2.00	41.09
LCFRS$_{r2\ell}$	79.25 ±0.15	79.93	33.82 ±0.21	34.19	78.74 ±0.43	80.25	38.01 ±1.64	39.49
hybrid child	77.82 ±0.24	78.52	30.64 ±0.28	31.15	80.38 ±0.33	82.06	40.73 ±0.86	**44.52**
hybrid strict-Markov	79.66 ±0.23	80.55	45.56 ±0.55	47.65	77.51 ±0.32	78.17	39.27 ±1.96	42.18
MAP training (default count: 1.0)								
LCFRS$_{ho}$	79.03 ±0.42	79.83	30.98 ±0.53	33.00	80.88 ±0.23	82.27	40.40 ±2.27	41.96
LCFRS$_{r2\ell}$	80.01 ±0.16	81.02	34.15 ±0.22	35.70	79.38 ±0.79	80.79	40.92 ±2.48	43.04
hybrid child	78.56 ±0.23	79.36	32.15 ±0.61	33.36	81.75 ±0.26	**83.33**	40.31 ±0.68	43.06
hybrid strict-Markov	80.54 ±0.16	**81.75**	47.32 ±0.96	**49.82**	77.62 ±0.12	78.60	40.23 ±0.89	42.52

Table 2: (Average) F1-scores on the dev. set (length ≤ 40) after training for 4 s/m cycles at 50% merge rate (HN08) and 6 s/m cycles at 80% merge rate (NEGRA). Columns labeled *(ens.)* show scores for the ensemble models.

F1 and labelled discontinuous F1 using *discodop*[3] (van Cranenburgh et al., 2016) with the included `proper.prm` parameter file. In the tables we list average F1 scores and standard deviation over 4 grammars with different random seeds, except for ensemble experiments where we execute just one run combining those 4 grammars.

Properties of baseline grammars. We induce baseline grammars from the training data and display their properties in table 1. We see that Markovizing the strict labels effectively reduces the number of nonterminals (NTs) and leads to the most accurate baseline grammars. Still, this approach comes at the cost of reduced coverage in comparison to the child labeling hybrid grammar and the LCFRS. On NEGRA we see an increase in parse failures also in comparison to strict labeling. This is due to the addition of vertical context

(i.e., the parent node) that is not present in strict labeling.

Training objective. We use the split/merge algorithm to refine the baseline grammars using both a likelihood-based and a MAP-based objective with default count 1.0 during EM training[4]. Differences between both training modes are displayed in table 2 for HN08 and NEGRA. MAP training in many settings improves the accuracy. In particular, using non-ensemble grammars the average F1 score on all constituents always improves. The average F1

---

[4]Next to the default count the training has other hyperparameters, e.g., the rate of splits that is merged or the number of s/m cycles. In early experiments we found default count values around 1.0 to work best. Also, an increase of the merge rate from 50% to e.g. 80% or 90% often does not harm the accuracy and allows for smaller grammars or the execution of additional split/merge cycles. We used such an optimized setting for NEGRA (and for TIGER on the test set). An exhaustive grid search that optimizes these parameters for all considered grammars and corpora is computationally expensive and beyond the scope of this article.

[3]https://github.com/andreasvc/disco-dop

	NEGRA			SPMRL		HN08
	F1 (≤ 40)	F1	Disc. F1	F1	Disc. F1	F1 (≤ 40)
this work						
hybrid grammar (average of 4 grammars)	81.1	80.3	40.7	76.7	38.9	80.7
hybrid grammar (ensemble)	82.5	81.7	43.5	77.7	40.7	81.6
chart-based approaches						
Kallmeyer and Maier (2013) ○	75.8†	-	-	-	-	-
van Cranenburgh et al. (2016) ○	76.8	-	-	-	-	78.2
Versley (2016) ○	-	-	-	79.5	-	82.9
Gebhardt (2018) ○	-	-	-	75.1	-	79.3
Corro (2020) (without pre-trained word embeddings)	-	**86.2**	54.1	**85.5**	53.8	-
Corro (2020) (with BERT, Devlin et al., 2019)	-	**91.6**	66.1	**90.0**	**62.1**	-
dependency-to-constituency conversions						
Hall and Nivre (2008)	-	-	-	-	-	79.9
Fernández-González and Martins (2015)	81.1	80.5	-	-	-	85.5
Corro et al. (2017)	-	-	-	81.6	-	-
Fernández-González and Gómez-Rodríguez (2020)	-	86.1	59.9	86.3	60.7	-
transition systems						
Versley (2014a,b)	-	-	-	-	-	74.2
Maier (2015)	77.0†	-	19.8†	74.7	18.8	79.5
Maier and Lichte (2016)	-	-	-	76.5	16.3	80.0
Coavoux and Crabbé (2017)	82.8	82.2	50.0	81.6	49.2	85.1
Stanojević and Garrido Alhama (2017)	83.4	82.9	-	81.6	-	85.3
Coavoux and Cohen (2019)	-	83.2	56.3	82.5	55.9	-
Coavoux et al. (2019)	-	83.2	54.6	82.7	55.9	-

Table 3: Test results with gold POS tags (systems/scores with ⬤ use predicted tags; †: sentence length ≤ 30 and no discounting of root notes in F1-score; ○: grammar-based model; neural scoring component).

score on only discontinuous constituents does not adhere to this trend in two cases. Note however that the prediction of discontinuous constituents seems to be comparably unstable (cf. the high variance). This indicates that a larger sample size is needed to reliably judge the influence of MAP training on the discontinuous F1. Also in case of ensemble models there are two grammars where the ensemble model trained with the MAP objective is less accurate.

Ensemble models. Comparing the (Disc.) F1-scores of the ensemble model with the averages of the individual grammars, we always see an improvement, which in many cases is also well above the standard deviation.

Selection of grammars. For HN08 we obtain the best results with hybrid grammars with the Markovized strict labeling, which also outperform LCFRS in contrast to the experiments by Gebhardt (2018). In experiments with NEGRA we see that the child nonterminal scheme is more accurate than the Markovized strict one. This might be explained by the smaller corpus size which may lead to sparsity

problems if nonterminal granularity is higher. The hybrid grammar with child labeling scores better than the LCFRS.

External comparison. Test set results are given in table 3. For NEGRA we apply the child labeling scheme and train for 7 s/m cycles using a merge rate of 80% and the MAP objective. For TIGER we apply the Markovized strict labeling scheme and train for 5 s/m cycles using a merge rate of 90% and the MAP objective. The comparison with results from the literature indicates that the ensemble of hybrid grammars performs better than other grammar-based approaches except for the pseudo-projective approach by Versley (2016). They are also more accurate than early dependency-to-constituency and transition-based approaches[5]. However, recent models, in particular those utilizing neural nets, are far more accurate than the hybrid grammars.

---

[5]Interestingly, the model by Fernández-González and Martins (2015) is superior on TIGER but inferior on NEGRA (which has a smaller training set).

4 Discussion and conclusions

The experiments provide further evidence that the split/merge method is applicable and effective beyond PCFG. The use of priors and ensembles of grammars is mostly beneficial and complementary. From the performance differences between child labeling and Markovized strict labeling, we can surmise that the initial nonterminal granularity matters as the split/merge method cannot fully recover important splits or at least during parsing the mrp objective relies on guidance by the baseline nonterminal structure. More generally, the performance differences between the considered grammars indicate that a careful choice of the grammar formalism and the extraction algorithm is not redundant despite split/merge refinement.

Interestingly, the pseudo-projective approach by Versley (2016) outperforms our strictly discontinuous one. He uses a linguistically motivated (de)projectivization strategy[6] that seems to address the sparsity of discontinuous constituents in the data very well. Hence, we may conjecture that true discontinuous grammar formalisms that make available a large number of discontinuous productions (based on scarce evidence) may rarely benefit from the additional expressiveness. To substantiate this claim certainly a controlled experiment is necessary as differences may as well be artefacts of the handling of lexical and fall-back rules by Versley (2016). However, a similar observation was made concerning (discontinuous) tree substitution grammars (van Cranenburgh et al., 2016). Also a recent study by Corro (2020) finds that a very restricted mode of discontinuity, which can be simulated by a series of continuous combinations, is more accurate than more expressive modes.

The research on discontinuous parsing with latent variable grammars may also be extended by considering spectral algorithms (cf. Cohen, 2017, for an overview). In particular, Louis and Cohen (2015) use latently annotated LCFRS obtained by spectral algorithms to parse the topical structure of forum threads. Yet, the application of spectral algorithms for *discontinuous syntactic* parsing has not been investigated.

Acknowledgements

---

[6]Boyd (2007) found that discontinuous trees that actually occur in treebanks can be (de)projectivized without losses.

References

James K. Baker. 1979. Trainable grammars for speech recognition. In *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, pages 547–550.

Adriane Boyd. 2007. Discontinuity revisited: An improved conversion to context-free representations. In *Proceedings of the Linguistic Annotation Workshop*, pages 41–44, Prague, Czech Republic.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen-Schirra, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. Tiger: Linguistic interpretation of a german corpus. *Research on Language and Computation*, 2(4):597–620.

Maximin Coavoux and Shay B. Cohen. 2019. Discontinuous constituency parsing with a stack-free transition system and a dynamic oracle. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 204–217, Minneapolis, Minnesota.

Maximin Coavoux and Benoît Crabbé. 2017. Incremental discontinuous phrase structure parsing with the gap transition. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1259–1270, Valencia, Spain.

Maximin Coavoux, Benoît Crabbé, and Shay B. Cohen. 2019. Unlexicalized transition-based discontinuous constituency parsing. *Transactions of the Association for Computational Linguistics*, 7:73–89.

Shay Cohen. 2017. Latent-variable PCFGs: Background and applications. In *Proceedings of the 15th Meeting on the Mathematics of Language*, pages 47–58, London, UK.

Caio Corro. 2020. Span-based discontinuous constituency parsing: a family of exact chart-based algorithms with time complexities from $\mathcal{O}(n^6)$ down to $\mathcal{O}(n^3)$.

Caio Corro, Joseph Le Roux, and Mathieu Lacroix. 2017. Efficient discontinuous phrase-structure parsing via the generalized maximum spanning arborescence. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1644–1654, Copenhagen, Denmark.

Andreas van Cranenburgh, Remko Scha, and Rens Bod. 2016. Data-oriented parsing with discontinuous constituents and function tags. *Journal of Language Modelling*, 4(1):57–111.

Aron Culotta and Jeffrey Sorensen. 2004. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA.

Pierre Deransart and Jan Małuszyński. 1989. A grammatical view of logic programming. In *Programming Languages Implementation and Logic Programming*, pages 219–251, Berlin, Heidelberg.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota.

Yuan Ding and Martha Palmer. 2005. Machine translation using probabilistic synchronous dependency insertion grammars. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 541–548, Stroudsburg, PA, USA.

Amit Dubey and Frank Keller. 2003. Probabilistic parsing for German using sister-head dependencies. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 96–103, Sapporo, Japan.

Adnan Duric and Fei Song. 2011. Feature selection for sentiment analysis based on content and syntax models. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA 2.011)*, pages 96–103, Portland, Oregon.

Daniel Fernández-González and André F. T. Martins. 2015. Parsing as reduction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1523–1533, Beijing, China.

Daniel Fernández-González and Carlos Gómez-Rodríguez. 2020. Discontinuous constituent parsing with pointer networks.

Kilian Gebhardt. 2018. Generic refinement of expressive grammar formalisms with an application to discontinuous constituent parsing. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3049–3063, Santa Fe, New Mexico, USA.

Kilian Gebhardt, Mark-Jan Nederhof, and Heiko Vogler. 2017. Hybrid grammars for parsing of discontinuous phrase structures and non-projective dependency structures. *Computational Linguistics*, 43(3):465–520.

Johan Hall and Joakim Nivre. 2008. Parsing discontinuous phrase structure with grammatical functions. In *Advances in Natural Language Processing*, pages 169–180, Berlin, Heidelberg.

Mark Johnson, Thomas Griffiths, and Sharon Goldwater. 2007. Bayesian inference for PCFGs via Markov chain Monte Carlo. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 139–146, Rochester, New York.

Aravind K. Joshi, Leon S. Levy, and Masako Takahashi. 1975. Tree adjunct grammars. *J. Comput. Syst. Sci.*, 10(1):136–163.

Laura Kallmeyer and Wolfgang Maier. 2013. Data-driven parsing using probabilistic linear context-free rewriting systems. *Computational Linguistics*, 39(1):87–119.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, pages 423–430, Sapporo, Japan.

Annie Louis and Shay B. Cohen. 2015. Conversation trees: A grammar model for topic structure in forums. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1543–1553, Lisbon, Portugal.

Wolfgang Maier. 2015. Discontinuous incremental shift-reduce parsing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1202–1212, Beijing, China.

Wolfgang Maier and Timm Lichte. 2016. Discontinuous parsing with continuous trees. In *Proceedings of the Workshop on Discontinuous Structures in Natural Language Processing*, pages 47–57, San Diego, California.

Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2005. Probabilistic CFG with latent annotations. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 75–82, Ann Arbor, Michigan.

Mark-Jan Nederhof and Heiko Vogler. 2014. Hybrid grammars for discontinuous parsing. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1370–1381, Dublin, Ireland.

Slav Petrov. 2010. Products of random latent variable grammars. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27, Los Angeles, California.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 433–440, Sydney, Australia.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411, Rochester, New York.

Djamé Seddah, Sandra Kübler, and Reut Tsarfaty. 2014. Introducing the SPMRL 2014 shared task on parsing morphologically-rich languages. In *Proceedings of the First Joint Workshop on Statistical Parsing of Morphologically Rich Languages and Syntactic Analysis of Non-Canonical Languages*, pages 103–109, Dublin, Ireland.

Wojciech Skut, Brigitte Krenn, Thorsten Brants, and Hans Uszkoreit. 1997. An annotation scheme for free word order languages. In *Fifth Conference on Applied Natural Language Processing*, pages 88–95, Washington, DC, USA.

Miloš Stanojević and Raquel Garrido Alhama. 2017. Neural discontinuous constituency parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1666–1676, Copenhagen, Denmark.

Yannick Versley. 2014a. Experiments with easy-first nonprojective constituent parsing. In *Proceedings of the First Joint Workshop on Statistical Parsing of Morphologically Rich Languages and Syntactic Analysis of Non-Canonical Languages*, pages 39–53, Dublin, Ireland.

Yannick Versley. 2014b. Incorporating semi-supervised features into discontinuous easy-first constituent parsing.

Yannick Versley. 2016. Discontinuity (re)^2-visited: A minimalist approach to pseudoprojective constituent parsing. In *Proceedings of the Workshop on Discontinuous Structures in Natural Language Processing*, pages 58–69, San Diego, California.

Krishnamurti Vijay-Shanker, David J. Weir, and Aravind K. Joshi. 1987. Characterizing structural descriptions produced by various grammatical formalisms. In *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, pages 104–111, Stanford, California, USA.