Mind the Gap: Data Enrichment in Dependency Parsing of Elliptical Constructions

Kira Droganova* **Filip Ginter†** **Jenna Kanerva†** **Daniel Zeman***
*Charles University, Faculty of Mathematics and Physics
†University of Turku, Department of Future Technologies
{droganova,zeman}@ufal.mff.cuni.cz
{figint,jmnybl}@utu.fi

Abstract

In this paper, we focus on parsing rare and non-trivial constructions, in particular ellipsis. We report on several experiments in enrichment of training data for this specific construction, evaluated on five languages: Czech, English, Finnish, Russian and Slovak. These data enrichment methods draw upon self-training and tri-training, combined with a stratified sampling method mimicking the structural complexity of the original treebank. In addition, using these same methods, we also demonstrate small improvements over the CoNLL-17 parsing shared task winning system for four of the five languages, not only restricted to the elliptical constructions.

1 Introduction

Dependency parsing of natural language text may seem like a solved problem, at least for resource-rich languages and domains, where state-of-the-art parsers attack or surpass 90% labeled attachment score (LAS) (Zeman et al., 2017). However, certain syntactic phenomena such as coordination and ellipsis are notoriously hard and even state-of-the-art parsers could benefit from better models of these constructions. Our work focuses on one such construction that combines both coordination and ellipsis: *gapping*, an omission of a repeated predicate which can be understood from context (Coppock, 2001). For example, in *Mary won gold and Peter bronze*, the second instance of the verb is omitted, as the meaning is evident from the context. In dependency parsing this creates a situation where the parent node is missing (omitted verb *won*) while its dependents are still present (*Peter* and *bronze*). In the Universal Dependencies annotation scheme (Nivre et al., 2016) gapping constructions are analyzed by promoting one of the orphaned dependents to the position of its missing parent, and connecting all remaining core arguments to that promoted one with the `orphan` relation (see Figure 1). Therefore the dependency parser must learn to predict relations between words that should not usually be connected. Gapping has been studied extensively in theoretical works (Johnson, 2009, 2014; Lakoff and Ross, 1970; Sag, 1976). However, it received almost no attention in NLP works, neither concerned with parsing nor with corpora creation. Among the recent papers, Kummerfeld and Klein (2017) proposed a one-endpoint-crossing graph parser able to recover a range of null elements and trace types, and Schuster (Schuster et al., 2018) proposed two methods to recover elided predicates in sentences with gapping. The aforementioned lack of corpora that would pay attention to gapping, as well as natural relative rarity of gapping, leads to its underrepresentation in training corpora: they do not provide enough examples for the parser to learn gapping. Therefore we investigate methods of enriching the training data with new material from large raw corpora.

The present work consist of two parts. In the first part, we experiment on enriching data in general, without a specific focus on gapping constructions. This part builds upon self-training and tri-training related work known from the literature, but also develops and tests a stratified approach for selecting a structurally balanced subcorpus. In the second part, we focus on elliptical sentences, comparing general enrichment of training data with enrichment using elliptical sentences artificially constructed by removal of a coordinated element.

2 Data

2.1 Languages and treebanks

For the parsing experiments we selected five treebanks from the Universal Dependencies (UD) col-

Proceedings of the Second Workshop on Universal Dependencies (UDW 2018), pages 47–54
Brussels, Belgium, November 1, 2018.

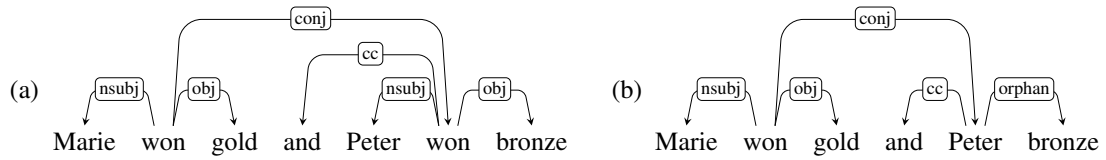

Figure 1: UD representation of a sentence with repeated verb (a), and with an omitted verb in a gapping construction (b).

lection (Nivre et al., 2016). We experiment with the following treebanks: UD_Czech, UD_English, UD_Finnish, UD_Russian-SynTagRus, and UD_Slovak. With the exception of UD_Russian-SynTagRus, all our experiments are based on UD release 2.0. This UD release was used in the CoNLL-17 Shared Task on Multilingual Parsing from Raw Text to Universal Dependencies (Zeman et al., 2017), giving us a point of comparison to the state-of-the-art. For UD_Russian-SynTagRus, we use UD release 2.1, which has a considerably improved annotation of elliptic sentences. For English, which has only a few elliptical sentences in the original treebank, we also utilize in testing a set of elliptical sentences gathered by Schuster et al. (2018).

This selection of data strives to maximize the amount of elliptical constructions present in the treebanks (Droganova and Zeman, 2017), while also covering different modern languages and providing variation. Decisions are based on the work by Droganova and Zeman (2017) who collected statistics on elliptical constructions that are explicitly marked with orphan relation within the UD treebanks. Relatively high number of elliptical constructions within chosen treebanks is the property of the treebanks rather than the languages.

2.2 Additional material

Automatic parses As an additional data source in our parsing experiments, we use the multilingual raw text collection by Ginter et al. (2017). This collection includes web crawl data for 45 languages automatically parsed using the UDPipe parser (Straka and Straková, 2017) trained on the UD version 2.0 treebanks. For Russian, where we use newer version of the treebank, we reparsed the raw data with UDPipe model trained on the corresponding treebank version to agree with the treebank data in use.

As our goal is to use the web crawled data to enrich the official training data in the parsing experiments, we want to ensure the quality of the automatically parsed data. To achieve this, we apply a method that stands between the standard self-training and tri-training techniques. In self-training, the labeled training data (L) is iteratively enriched with unlabeled data (U) automatically labeled with the same learning system ($L = L+U_l$), whereas in tri-training (Zhou and Li, 2005) there are three different learning systems, A, B and C, and the labeled data for the system A is enriched with instances from U on which the two other systems agree, therefore $L_a = L + (U_b \cap U_c)$. Different variations of these methods have been successfully applied in dependency parsing, for example (McClosky et al., 2006; Søgaard and Rishøj, 2010; Li et al., 2014; Weiss et al., 2015). In this work we use two parsers (A and B) to process the unlabeled crawl data, and then the sentences where these two parsers fully agree are used to enrich the training data for the system A, i.e. $L_a = L + (U_a \cap U_b)$. Therefore the method can be seen as a form of expanded self-training or limited tri-training. A similar technique is successfully used for example by Sagae and Tsujii (2007) in parser domain adaptation and Björkelund et al. (2014) in general parsing.

In our experiments the main parser used in final experiments as well as labeling the crawl data, is the neural graph-based Stanford parser (Dozat et al., 2017), the winning and state-of-the-art system from the CoNLL-17 Shared Task (Zeman et al., 2017). The secondary parser for labeling the crawl data is UDPipe, a neural transition-based parser, as these parses are already provided together with the crawl data. Both of these parsers include their own part-of-speech tagger, which is trained together (but not jointly) with the dependency parser in all our experiments. In the final self-training web crawl datasets we then keep only deduplicated sentences with identical part-of-speech and dependency analyses. All results reported in this paper are measured on gold tokenization, and the parser hyperparameters are those used for these systems in the CoNLL-17

Shared Task.

Artificial treebanks on elliptical constructions For specifically experimenting on elliptical constructions, we additionally include data from the semi-automatically constructed artificial treebanks by Droganova et al. (2018). These treebanks simulate gapping by removing words in particular coordination constructions, providing data for experimenting with the otherwise very rare construction. For English and Finnish the given datasets are manually curated for grammaticality and fluency, whereas for Czech the quality relies on the rules developed for the process. For Russian and Slovak, which are not part of the original artificial treebank release, we create automatically constructed artificial datasets by running the pipeline developed for the Czech language. Size of the artificial data is shown in Table 1.

	Token	Sentence
Czech	50K	2876
English	7.3K	421
Finnish	13K	1000
Russian	87K	5000
Slovak	7.1	564

Table 1: The size of the artificial data

3 Experiments

First, we set out to evaluate the overall quality of the trees in the raw enrichment dataset produced by our self-training variant by parsing and filtering web crawl data. In our baseline experiments we train parsers (Dozat et al., 2017) using purely the new self-training data. From the full self-training dataset we sample datasets comparable to the sizes of the original treebanks to train parsers. These parsers are then evaluated using the original test set of the corresponding treebank. This gives us an overall estimate of the self-training data quality compared to the original treebanks.

3.1 Tree sampling

Predictably, our automatically selected self-training data is biased towards short, simple sentences where the parsers are more likely to agree. Long sentences are in turn often composed of simple coordinated item lists. To rectify this bias, we employ a sampling method which aims to more closely follow the distribution of the original treebank compared to randomly sampling sentences from the full self-training data. We base the sampling on two features of every tree: the number of tokens, and the number of unique dependency relation types divided by the number of tokens. The latter accounts for tree complexity, as it penalizes trees where the same relation type is repeated too many times, and it specifically allows us to downsample the long coordinated item lists where the ratio drops much lower than average. We of course take into account that a relation type can naturally occur more than once in a sentence, and that it is not ideal to force the ratio close to 1.0. However, as the sampling method tries to mimic the distribution from the original treebank, it should to pick the correct variance while discarding the extremes.

The sampling procedure proceeds as follows: First, we divide the space of the two features, length and complexity, into buckets and estimate from the treebank training data the target distribution, and the expected number of trees to be sampled in each bucket. Then we select from the full self-training dataset the appropriate number of trees into each bucket. Since the web crawl data is heavily skewed, it is not possible to obtain a sufficient number of sampled trees in the exact desired distribution, because many rare length–complexity combinations are heavily underrepresented in the data. We therefore run the sampling procedure in several iterations, until the desired number of trees have been obtained. This results in a distribution closer to, although not necessarily fully matching, the original treebank.

To evaluate the impact of this sampling procedure, we compare it to two baselines. *RandomS* randomly selects the exact same number of sentences as the above-mentioned *Identical* sampling procedure. This results in a dataset which is considerably smaller in terms of tokens, because the web crawl data (on which the two parsers agree) is heavily biased towards short trees. To make sure our evaluation is not affected by simply using less data in terms of tokens, we also provide the *RandomT* baseline, where trees are randomly selected until the same number of tokens is reached as in the *Identical* sample. Here we are able to evaluate the quality of the sampled data, not its bulk.

In Table 2 we see that, as expected, when sampling the same amount of sentences as in the training section of the original treebank, the *RandomS* sampling produces datasets considerably smaller in terms of tokens, whereas *RandomT* results in

Language	Random T	Random S	Identical	TB
Czech	102K/982K	68K/611K	68K/982K	68K/1175K
English	18K/183K	13K/102K	13K/183K	13K/205K
Finnish	17K/144K	12K/92K	12K/144K	12K/163K
Russian	73K/694K	49K/431K	49K/694K	49K/871K
Slovak	11K/83K	8K/58K	8K/83K	8K/81K

Table 2: Training data sizes after each sampling strategy compared to the original treebank training section (TB), sentences/tokens.

datasets considerably larger in terms of trees when the same amount of tokens as in the *RandomS* dataset is sampled. This confirms the assumption that parsers tend to agree on shorter sentences in the web crawl data, introducing the bias towards them. On the other hand, when the same number of sentences is selected as in the *RandomS* sampling and the original treebank, the *Identical* sampling strategy results in dataset much closer to the original treebank in terms of tokens.

Parsing results for the different sampling strategies are shown in Table 3. Except for Slovak, the results follow an intuitively expectable pattern: the sample with the least tokens results in the worst score, and of the two samples with the same number of tokens, the one which follows the treebank distribution receives the better score. Surprisingly, for Slovak the sampling strategy which mimics the treebank distribution receives a score almost 3pp lower than the one with random sampling of the same amount of tokens. A possible explanation is given in the description of the Slovak treebank which mentions that it consists of sentences on which two annotators agreed, and is biased towards short and simple sentences. The data is thus not representative of the language use, possibly causing the effect. Lacking a better explanation for the time being, we also add the *RandomT* sampling dataset into our experiments for Slovak. Overall, the parsing results on the automatically selected data are surprisingly good, lagging only several percent points behind parsers trained on the manually annotated treebanks.

Language	Random T	Random S	Identical	TB
Czech	88.50%	88.18%	**88.77%**	91.20%
English	83.67%	82.86%	**84.18%**	86.94%
Finnish	82.67%	80.69%	**83.01%**	87.89%
Russian	91.28%	90.85%	**91.49%**	93.35%
Slovak	**85.02%**	83.67%	82.35%	86.04%

Table 3: Results of the baseline parsing experiments, using only automatically collected data, reported in terms of LAS%. Random T: random sample, same amount of tokens as in the Random S samples; Random S: random sample, same amount of sentences as in the original treebanks; Identical: identical sample, imitates the distribution of trees in the original treebanks. For comparison, the TB column shows the LAS of a parser trained on the original treebank training data.

Language	TB	+20%	+100%	+200%
Czech	**91.20%**	91.13%	90.98%	90.72%
English	86.94%	87.32%	**87.43%**	87.29%
Finnish	87.89%	87.83%	88.24%	**88.32%**
Russian	93.35%	**93.38%**	93.22%	93.08%
Slovak	86.04%	87.89%	**88.36%**	**88.36%**
Slovak T	86.04%	88.14%	88.57%	**88.77%**

Table 4: Enriching treebank data with identical sample from automatic data, LAS%. TB: original treebank (baseline experiment; the scores are better than reported in the CoNLL-17 Shared Task because we evaluate on gold segmentation while the shared task systems are evaluated on predicted segmentation); +20% – +200%: size of the identical sample used to enrich the treebank data (with respect to the original treebank size). Slovak T: enriching Slovak treebank with random tokens sample instead of identical.

3.2 Enrichment

In this section, we test the overall suitability of the sampled trees as an additional data for parsing. We produce training data composed of the original treebank training section, and a progressively increasing number of sampled trees: 20%, 100%, and 200% (relative to the treebank training data size, i.e. +100% sample doubles the total amount of training data). The parsing results are shown in Table 4. Positively, for all languages except Czech, we can improve the overall parsing accuracy, for Slovak by as much as 2.7pp, which is a rather non-trivial improvement. In general, the smaller the treebank, the larger the benefit. With the exception of Slovak, the improvements are relatively modest, in the less than half-a-percent range. Nevertheless, since our baseline is the winning parser of the CoNLL-17 Shared Task, these constitute improvements over the current state-of-the-art. Based on these experiments,

we can conclude that self-training data extracted from web crawl seem to be suitable material for enriching the training data for parsing, and in next section we continue to test whether the same data and methods can be used to increase occurrences of a rare linguistic construction to make it more learnable for parsers.

3.3 Ellipsis

Our special focus point is that of parsing elliptic constructions. We therefore test whether increasing the number of elliptical sentences in the training data improves the parsing accuracy of these constructions, without sacrificing the overall parsing accuracy. We follow the same data enrichment methods as used above in general domain and proceed to select elliptical sentences (recognized through the `orphan` relation) from the same self-training data automatically produced from web crawl (Section 2.2). We then train parsers using a combination of the ellipsis subset and the original training section for each language. We enrich Czech, Russian and Slovak training data with elliptical sentences, progressively increasing their size by 5%, 10% and 15%. For Finnish, only 5% of elliptical sentences was available in the filtered web crawl data, and for English not a single sentence.

The experiments showed mixed results (Table 5). For Russian and Slovak the accuracy of the dependencies involved in gapping is improved by web crawl enrichment, whereas the results for Czech remained largely the same and Finnish slightly decreased (column *Web crawl*). Unfortunately, for Slovak and Finnish, we cannot draw firm conclusions due to the small number of orphan relations in the test set. For English, even the treebank results are very low: the parser predicts only very few orphan relations (recall 1.71%) and the web crawl data contains no orphans on which the two parsers could agree, thus making it impossible to enrich the data using this method. Clearly, English requires a different strategy, and we will return to it shortly. Positively, none of the languages substantially suffered in terms of overall LAS when adding extra elliptical sentences into the training data. For Slovak, we can even see a significant improvement in overall parsing accuracy, in line with the experiments in Section 3.1. Increasing the proportion of orphan sentences in the training data has the predictable effect of increasing the orphan F-score and decreasing the overall LAS of the parser. These differences are nevertheless only very minor and can only be observed for Czech and Russian which have sufficient number of orphan relation examples in the test set. For Slovak, with 18 examples, we cannot draw any conclusions, and for English and Finnish, there is not a sufficient number of orphan examples in the filtered web crawl data to allow us to vary the proportion.

For all languages, we also experiment with the artificial elliptic sentence dataset of Droganova et al. (2018), described earlier in Section 2.2. For Czech, English and Finnish, the dataset contains semi-automatically produced, and in the case of English and Finnish, also manually validated instances of elliptic sentences. For Slovak and Russian, we replicate the procedure of Droganova et al., sans the manual validation, obtaining artificial orphan datasets for all the five languages under study. Subsequently, we train parsers using a combination of sentences from the artificial treebank and the original training set. The results of this experiments are in Table 5, column *Artificial*. Compared to web crawl, the artificial data results in a lower performance on orphans for Czech, Slovak and Russian, and higher for Finnish, but once again keeping in mind the small size of Finnish and Slovak test set, it is difficult to come to a firm conclusion. Clearly, though, the web crawl data does not perform substantially worse than the artificial data, even though it is gathered fully automatically. A very substantial improvement is achieved on English, where the web crawl data fails to deliver even a single orphan example, whereas the artificial data gains recall of 9.62%.

This offers us an opportunity to once again try to obtain orphan examples for English from the web crawl data, since this time we can train the parsers on the combination of the original treebank and the artificial data, hopefully resulting in parsers which are in fact able to predict at least some orphan relations, which in turn can result in new elliptic sentences from the web crawl data. As seen from Table 5, the artificial data increases the orphan F-score from 3.36% to 17.18% relative to training only on the treebank, and we are therefore able to obtain a parser which is at least by the order of magnitude comparable to the other four languages in parsing accuracy of elliptic constructions. We observe no loss in terms of the over-

Language	All	Treebank				Web crawl +5/+10/+15%				Artificial			
		LAS	O Pre	O Rec	O F	LAS	O Pre	O Rec	O F	LAS	O Pre	O Rec	O F
Czech	418	91.20%	54.84%	56.94%	**55.87%**	**91.22%** 91.11% 91.06%	48.96% 49.80% 50.19%	61.72% 60.05% 62.68%	54.60% 54.45% 55.74%	91.15%	51.79%	58.85%	55.10%
English	2+466	86.94%	100.00%	1.71%	3.36%	—	—	—	—	**86.95%**	80.36%	9.62%	**17.18%**
Finnish	43	87.89%	66.67%	32.56%	43.75%	87.76%	48.15%	30.23%	37.14%	**88.04%**	54.76%	53.49%	**54.12%**
Russian	138	93.35%	44.57%	29.71%	35.65%	**93.50%** 93.41% 93.42%	42.86% 38.26% 40.69%	39.13% 41.30% 42.75%	40.91% 39.72% **41.70%**	93.20%	33.14%	40.58%	36.48%
Slovak	18	86.04%	60.00%	16.67%	26.09%	**87.90%** 87.76% 87.80%	36.36% 33.33% 30.77%	22.22% 16.67% 22.22%	**27.59%** 22.22% 25.81%	87.80%	37.50%	16.67%	23.08%

Table 5: Enriching treebank data with elliptical sentences. All: number of `orphan` labels in the test data; Treebank: original treebank (baseline experiment); Web crawl: Enriching the original treebank with the elliptical sentences extracted from the automatically parsed web crawl data; Artificial: Enriching the original treebank with the artificial ellipsis treebank; LAS, %: overall parsing accuracy; O Prec (orphan precision): number of correct `orphan` nodes divided by the number of all predicted `orphan` nodes; O Rec (orphan recall): number of correct `orphan` nodes divided by the number of gold-standard `orphan` nodes; O F (Orphan F-score): F-measure restricted to the nodes that are labeled as `orphan` : 2PR / (P+R). For English, the orphan P/R/F scores are evaluated on a dataset of the two orphan relations in the original test section, combined with 466 English elliptic sentences of Schuster et al. (2018). The extra sentences are not used in the LAS column, so as to preserve comparability of overall LAS scores across the various runs.

all LAS, demonstrating that it is in fact possible to achieve a substantial improvement in parsing of a rare, non-trivial construction without sacrificing the overall performance.

Using the web data self-training filtering procedure with two parsers trained on the treebank+artificial data, we can now repeat the experiment with enriching parser training data with orphan relations, results of which are shown in Table 6. We test the following models:

- original UD_English v.2.0 treebank;
- original UD_English v.2.0 treebank combined with the artificial sentences;
- original UD_English v.2.0 treebank combined with the artificial sentences and web crawl dataset; size progressively increased by 5%, 10% and 15%. Here we use the original UD_English v.2.0 treebank extended with the artificial sentences to train the models (Section 2.2) that produce the web crawl data for English.

Model	LAS	O Precision	O Recall	O F-score
Treebank	86.94%	100%	1.71%	3.36%
Artificial	86.95%	80.36%	9.62%	17.18%
Art.+Web 5%	86.72%	86.11%	19.87%	32.29%
Art.+Web 10%	86.68%	78.36%	22.44%	34.88%
Art.+Web 15%	**87.07%**	84.38%	23.08%	**36.24%**

Table 6: Enriching the English treebank data with elliptical sentences. LAS, %: overall parsing accuracy; O Precision (orphan precision): number of correct `orphan` labels divided by the number of all predicted `orphan` nodes; O Recall (orphan recall): number of correct `orphan` labels divided by the number of gold-standard `orphan` nodes; O F-score (orphan F-score): F-measure restricted to the nodes that are labeled as `orphan` : 2PR / (P+R). For English, the orphan P/R/F scores are evaluated on a dataset of the two orphan relations in the original test set, combined with 466 English elliptic sentences of Schuster et al. (2018). The extra sentences are not used in the LAS column, so as to preserve comparability of overall LAS scores across the various runs. This is necessary since elliptic sentences are typically syntactically more complex and would therefore skew overall parser performance evaluation.

The best orphan F-score of 36%, more than ten times higher compared to using the original treebank, is obtained by enriching the training data with 15% elliptic sentences from the artificial and filtered web data. The orphan F-score of 36% is on par with the other languages and, positively, the overall LAS of the parser remains essentially unchanged — the parser does not sacrifice anything in order to gain the improvement on orphan relations. These English results therefore not only explore the influence of the number of elliptical sentences on the parsing accuracy, but also test a method applicable in the case where the treebank does not contain almost any elliptical constructions and results in parsers that only generate the relation very rarely.

4 Conclusions

We have explored several methods of enriching training data for dependency parsers, with a specific focus on rare phenomena such as ellipsis (gapping). This focused enrichment leads to mixed results. On one hand, for several languages we did not obtain a significant improvement of the parsing accuracy of ellipsis, possibly in part owing to the small number of testing examples. On the other hand, though, we have demonstrated that for English ellipsis parsing accuracy can be improved from single digit numbers to performance on par with the other languages. We have also validated the method of constructing artificial elliptical examples as a mean to enrich parser training data. Additionally, we have shown that useful training data can be obtained using web crawl data and a self-training or tri-training style method, even though the two parsers in question differ substantially in their overall performance.

Finally, we have shown that this parser training data enrichment can lead to improvements of general parser accuracy, improving upon the state of the art for all but one language. The improvement was especially notable for Slovak. Czech was the only treebank not benefiting from this additional data, likely owing to the fact that is is an already very large, and homogenous treebank. As part of these experiments, we have introduced and demonstrated the effectiveness of a stratified sampling method which corrects for the skewed distribution of sentences selected in the web filtering experiments.

Acknowledgments

The work was partially supported by the grant 15-10472S of the Czech Science Foundation (GAČR), the GA UK grant 794417, Academy of Finland, and Nokia Foundation. Computational resources were provided by CSC - IT Center for Science, Finland.

References

Anders Björkelund, Özlem Çetinoğlu, Agnieszka Faleńska, Richárd Farkas, Thomas Müller, Wolfgang Seeker, and Zsolt Szántó. 2014. Self-training for Swedish Dependency Parsing – Initial Results and Analysis. In *Proceedings of the Fifth Swedish Language Technology Conference (SLTC 2014)*.

Elizabeth Coppock. 2001. Gapping: In defense of deletion. In *Proceedings of the Chicago Linguistics Society*, volume 37, pages 133–148.

Timothy Dozat, Peng Qi, and Christopher D Manning. 2017. Stanford's Graph-based Neural Dependency Parser at the CoNLL 2017 Shared Task. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30.

Kira Droganova and Daniel Zeman. 2017. Elliptic Constructions: Spotting Patterns in UD Treebanks. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, 135, pages 48–57.

Kira Droganova, Daniel Zeman, Jenna Kanerva, and Filip Ginter. 2018. Parse Me if You Can: Artificial Treebanks for Parsing Experiments on Elliptical Constructions. *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*.

Filip Ginter, Jan Hajič, Juhani Luotolahti, Milan Straka, and Daniel Zeman. 2017. CoNLL 2017 Shared Task - Automatically Annotated Raw Texts and Word Embeddings. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Kyle Johnson. 2009. Gapping is not (VP) ellipsis. *Linguistic Inquiry*, 40(2):289–328.

Kyle Johnson. 2014. Gapping.

Jonathan K Kummerfeld and Dan Klein. 2017. Parsing with Traces: An $O(n^4)$ Algorithm and a Structural Representation. *arXiv preprint arXiv:1707.04221*.

George Lakoff and John Robert Ross. 1970. Gapping and the order of constituents. *Progress in linguistics: A collection of papers*, 43:249.

Zhenghua Li, Min Zhang, and Wenliang Chen. 2014. Ambiguity-aware ensemble training for semi-supervised dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 457–467.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Effective self-training for parsing. In *Proceedings of the main conference on human language technology conference of the North American Chapter of the Association of Computational Linguistics*, pages 152–159. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the 10th International Conference on Language Resources and*

Evaluation (LREC 2016), pages 1659–1666, Portorož, Slovenia.

Ivan Sag. 1976. *Deletion and Logical Form*. MIT. PhD dissertation.

Kenji Sagae and Junichi Tsujii. 2007. Dependency parsing and domain adaptation with LR models and parser ensembles. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Sebastian Schuster, Joakim Nivre, and Christopher D. Manning. 2018. Sentences with Gapping: Parsing and Reconstructing Elided Predicates. In *Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2018)*.

Anders Søgaard and Christian Rishøj. 2010. Semi-supervised dependency parsing using generalized tri-training. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1065–1073. Association for Computational Linguistics.

Milan Straka and Jana Straková. 2017. Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.

David Weiss, Chris Alberti, Michael Collins, and Slav Petrov. 2015. Structured Training for Neural Network Transition-Based Parsing. In *Proceedings of ACL 2015*, pages 323–333.

Daniel Zeman, Martin Popel, Milan Straka, Jan Hajič, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gökrmak, Anna Nedoluzhko, Silvie Cinková, jr. Jan Hajič, Jaroslava Hlaváčová, Václava Kettnerová, Zdeňka Urešová, Jenna Kanerva, Stina Ojala, Anna Missilä, Christopher Manning, Sebastian Schuster, Siva Reddy, Dima Taji, Nizar Habash, Herman Leung, Marie-Catherine de Marneffe, Manuela Sanguinetti, Maria Simi, Hiroshi Kanayama, Valeria de Paiva, Kira Droganova, Héctor Martínez Alonso, Çağr Çöltekin, Umut Sulubacak, Hans Uszkoreit, Vivien Macketanz, Aljoscha Burchardt, Kim Harris, Katrin Marheinecke, Georg Rehm, Tolga Kayadelen, Mohammed Attia, Ali Elkahky, Zhuoran Yu, Emily Pitler, Saran Lertpradit, Michael Mandl, Jesse Kirchner, Hector Fernandez Alcalde, Jana Strnadová, Esha Banerjee, Ruli Manurung, Antonio Stella, Atsuko Shimada, Sookyoung Kwak, Gustavo Mendonça, Tatiana Lando, Rattima Nitisaroj, and Josie Li. 2017. CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Stroudsburg, PA, USA. Charles University, Association for Computational Linguistics.

Zhi-Hua Zhou and Ming Li. 2005. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on knowledge and Data Engineering*, 17(11):1529–1541.