Corpus Selection Approaches for Multilingual Parsing from Raw Text to Universal Dependencies

Ryan Hornby¹ and **Clark Taylor**² and **Jungyeul Park**³ ¹BASIS Oro Valley, Oro Valley, AZ

ryanhornby1999@gmail.com

²Department of Computer Science, University of Arizona

cqtboy1988@email.arizona.edu

³Department of Linguistics, University of Arizona

jungyeul@email.arizona.edu

Abstract

This paper describes UALing's approach to the CoNLL 2017 UD Shared Task using corpus selection techniques to reduce training data size. The methodology is simple: We use similarity measures to select a corpus from available training data (even from multiple corpora for surprise languages) and use the resulting corpus to complete the parsing task. The training and parsing is done with the baseline UDPipe system (Straka et al., 2016). While our approach reduces the size of training data significantly, it retains performance within 0.5% of the baseline system. Due to the reduction in training data size, our system performs faster than the naïve, complete corpus method. Specifically, our system runs in less than 10 minutes, ranking it among the fastest entries for this task. Our system is available at https://github. com/CoNLL-UD-2017/UALING.

1 Introduction

Universal Dependencies (UDs) (Nivre et al., 2016) includes corpora from different languages annotated with identical types of labels. This allows for the examination of different theoretical (Schuster and Manning, 2016) and practical applications such as the *CoNLL 2017 UD Shared Task* (Zeman et al., 2017).¹ The specific practical task presented here involves using these corpora in a supervised learning approach in order to achieve the task's goal: Training with the multilingual UD data in order to find dependency relationships not just for these known languages, but also for unknown or little-known language.²

1.1 Theoretical Concepts

Supervised learning occurs when humans encode their judgment into a set of data, which is in turn used to train statistical models with the ultimate goal of using these models to make accurate predictions for previously unseen datasets-which are often too large (and costly) or otherwise unavailable for humans to judge manually. Building these models of human judgment is necessary in cases where explicit rules are too complex to encode, ambiguous, or where such rules are not known; rather than explicitly and programatically encoding rules, supervised learning models "learn" or at least "contain" the rules through models generated from human-judged data. Ideally, the models are used to apply those same rules to the unseen datasets.

The rules contained in human-judged training data are, necessarily, constrained to the domain from which the data derives.³ Unknown domains—such as unknown languages—are difficult to handle, because the rules from another known domain do not necessarily apply and the rules for the target domain are thus not readily known. Creating new corpora for specific domains (which occurs often for biomedical-domain data, for instance) drastically improves accuracy for the given corpus domain (Clegg and Shepherd, 2005). The UD corpora extend training data across

¹http://universaldependencies.org/ conll17

²"Little-known" here implies that the language has little to no human-annotated data—also known as under-resourced languages.

³What constitutes a "domain" may vary across different contexts. Here, the term is used generally to denote sources of data for which there exist significant similarity in grammatical rules, such as entire languages or much smaller topics. In the context of the task, "domain" generally refers to a single language, as our target data is of unknown topics but of a single language.

numerous different language-domains. However, adapting currently known data to new domains (including new languages) is a difficult problem, particularly when human judgment is not available to aid in the adaption.

Similarly, in many cases training data contains rules not relevant or even contradictory to those in unseen data. This occurs both interdomain where the training data contains data from a domain which is not relevant to the unseen data—and intradomain—where training data from a domain is not relevant to other data in the same domain.⁴ This data may not be necessary to building supervised learning models because it does not contain relevant rules; in some circumstances, training data may even introduce rules to the model which contradict rules in the unseen data.⁵ Still other training data contains rules of marginal significance to the model, where such rules apply only to an extremely small segment of unseen data.

Without using supervised learning methods which actively adapt the rules of these data types to incoming unseen data, it is possible to (1) improve algorithmic and model performance by removing contradictory-rule training data, (2) improve algorithmic performance without model performance loss by removing irrelevant-rule training data, and (3) improve algorithmic performance without significant performance loss by removing marginal-significance rule data.

1.2 Resulting Methodology

For this paper, we introduce several methods of automated corpus refinement in order to improve and at best optimize supervised learning by accomplishing the goals enumerated above. Specifically, we propose and evaluate the use of similarity measures to refine the training data set; these similarity metrics "select" training data of the same domain and data which is closest in linguistic rules to the target, unseen data from available corpora. Using only this similar training data ought to remove data containing contradictory or irrelevant rules. Furthermore, similarity metrics provide an opportunity to scale the included data by including only the **most** similar data above a threshold, which also has the potential to remove marginalsignificance training data. This method allows us to drastically reduce corpus size while retaining only the most similar—and, ideally, best—training data. The overall effect on supervised learning performance depends on how well the employed similarity metric matches underlying rule similarity.

To accomplish this, we create a corpus processing pipeline in Java which calculates similarity and selects data. In this implementation, the development data set for monolingual parsing is considered a feature vector $(\S3)$, and similarity $(\S2)$ is calculated between this vector and each sentence in the single monolingual training data of the language. We try to find a *fixed* selection threshold (where sentences above the threshold are kept in the new training dataset) for all languages for monolingual parsing that provides the greatest performance, though performance-per-compression metrics are also valuable in some contexts. The sample data set for surprise languages also constitutes a feature vector $(\S4)$, and the program calculates the similarity between this vector and each sentence in the training data of all other languages. We evaluate various selection thresholds to adapt the under-resourced situation for each surprise language. We use UDPipe 1.1 (Straka et al., 2016) as the baseline system and the UD version 2.0 datasets (Nivre et al., 2017). We are able to reduce the size of training data down to 76.25% of the original in average using the proposed method while retaining UD parsing results are comparable to the baseline system. We can actually increase result accuracies for certain languages by using the resulting compressed training datasets.

2 Similarity

In our methodology, we employ cosine similarity as our similarity metric. The cosine similarity measure is applied to two latent vectors in two datasets. Let $\cos(d_1, d_2)$ be the cosine similarity, which is calculated as follows:

$$\cos(d_1, d_2) = \frac{V_{d_1} \cdot V_{d_2}}{\|V_{d_1}\| \|V_{d_2}\|}$$
(1)

where two feature vectors of V_{d_1} and V_{d_2} are

⁴Both interdomain and intradomain data are considered and dealt with by the method proposed here, though the task focuses on interdomain problems when considering different language domains. Pure corpus compression, as also discussed herein, tends to focus on the intradomain problem.

⁵To illustrate, when considering supervised learning approaches to parts-of-speech annotating, in the domain of formal scientific literature the word "as" might more often be used as a conjunction (as a synonym of "because") while in journalism it might be more often used as an adverb. Models trained on these different domains would likely result in different outcomes when labeling parts-of-speech due to these differences.



Figure 1: Treebank example: tri-gram POS sequences and dependency relationship between POS labels are extracted for features of the similarity measure.

from training and development datasets. Among the 64 languages with training data, 56 provide development data as well. Therefore, we focus on 56 languages for the proposed corpus selection approaches. The entire development data set makes one vector, and then the similarity is calculated between this vector and every sentence in the training data.

Various feature vectors are described in §3 and §4 for the monolingual and cross-lingual corpus selection approaches. For monolingual parsing, we use training and development corpora of the single language set for similarity measurement, extracting the most pertinent training data from the single corpus in order to compress and/or refine it. For cross-lingual parsing when we deal with surprise languages, we use training corpora from all languages, comparing the target language data to all known UD language corpora. This extracts the most similar data from other languages, with the hope that it is also similar in language grammar and structure—and, hence, similar in annotation.

3 Monolingual Corpus Selection Approaches

We use the following features for monolingual parsing:

- 1. tri-gram POS sequences
- 2. dependency relationships between two POS labels

Tri-gram POS sequences represent the tri-gram universal POS labels (Petrov et al., 2012). Dependency relationships represents the part of speech labels of a dependent and a dependee, and their dependency relationships.

3.1 Feature extraction

Tri-gram POS sequences are extracted from Universal POS labels of the sentence such as DET NOUN ADJ, NOUN ADJ AUX, etc. (See Figure 1). Uni-gram and bi-gram POS sequences are excluded because we found them to not be distinctive between the languages with Universal POS labels that we examined. We also extract dependency relationships between POS labels for the similarity measure such as NOUN nsubj VERB for commotions ... devenu where commotions/NOUN is dependent on devenu/VERB with nsubj dependency relation.⁶ Figure 2 shows two results by using the different thresholds for tri-gram POS sequences and POS-dep-POS. Using similarity measures to select the subset of the original training data, the proposed method slightly outperforms the results obtained by the original training data set with the similarity threshold $\theta = 0.1$. Actually, it improves the parsing result by 0.01% and 0.15%only using 94% and 77% of the original training datasets for German and Dutch, respectively.

Table 1 shows our entire results of the corpus selection method for monolingual parsing on the dev datasets using label attachment score (LAS) per treebank. We train the full training datasets, and trimmed datasets using similarity of tri-gram POS sequences (pos) and POS-dep-POS (dep). We also train the monolingual parsing models by using results the intersection of two similarity measures (intersection). All results are tested on the dev datasets without 8 languages which do not provide dev datasets.⁷ Table 1 also shows results from the

⁶While current feature selection is based on Universal POS labels, using language-specific POS labels for feature selection is one possible way to extend our approach for the *monolingual* corpus selection.

⁷ We also exclude results of ru_syntagrus from the table because of internal formatting errors that our corpus selection method produced.



(a) Tri-gram POS sequences



(b) POS dep POS

Figure 2: Threshold estimation using Dutch (NL) and German (DE): Similarity thresholds for tri-gram POS sequences and POS-dep-POS. Y-axis shows LAS (left) and the number of sentences (right).

method of length-based corpus selection (length). Since the sentence lengths decay after the peak (of the distribution of the numbers of sentences), our length-based approach to reduce the data set by length is to count the number of sentences before the peak and keep up to that many sentences after the peak.⁸ In Table 1, we also indicate ratios of training datasets. This compression amount uses the scale indicated by the entire (full) and its compressed rates. For example, while grc uses 54.76% () of the full training data set for *length*, its results decreases only by 2.54%. Actually, grc uses only 83.54% (for *intersection*, it outperforms by 0.41%. We improve parsing results for 33 languages on dev data using the proposed corpus selection method by measuring similarity.

3.2 Discussion

Besides features that we presented, we also investigate a length-based approach to select the training data. Instead of using the peak of the distribution of the numbers of sentences as in Table 1, we calculate the simple average numbers of words of the sentences in the dev data set. Then, we obtain the training data set using thresholds with $average \pm scale$ for the number of words, where scale is the number of $\max(|avg - max|, |avg - max|)$ min) words of the sentence in the dev data set. max and min are the maximum and the minimum numbers of words of the sentences. Figure 3 shows results and the number of sentences in the trimmed training datasets using length thresholds. We vary scale multipliers from 1 to 5. Filtering based on sentence length can potentially remove unnecessary size from the training data and possibly remove some inaccuracy, assuming that longer sentences increase entropy and become inherently less predictable. However, as Figure 3 indicated this simple length-based approach cannot keep up with the baseline results. While corpus compression levels compare to the similarity-

⁸This method resulted in some languages having up to 80% of the sentences removed because the peak sentence length was a small number of words. In order to make sure that only the outliers in length are removed we change the algorithm so that the peak value was between ten and twenty words long. This fixes the problem where languages with a low peak length having a large amount of sentences removed.

	full	length	pos	dep	intersection
grc	49.79	47.24	48.67	49.49	50.20
grc_proiel	65.87	57.95	65.66	66.40	66.69
ar	66.99	51.59	65.56	65.87	67.14
eu	67.48	52.35	66.94	67.86	67.32
bg	83.55	82.30	82.86	83.19	82.98
ca	81.80	75.23	81.26	81.05	81.59
zh	66.20	51.45	66.30	66.45	66.35
hr	73.87	72.64	73.41	74.23	74.33
cs_cac	79.89	73.24	79.81	79.48	79.74
cs_cltt	67.93	55.10	68.00	68.27	68.00
cs	79.42	76.48	79.42	79.03	79.42
da	72.68	59.66	73.29	73.78	73.83
nl_lassysmall	73.05	58.06	73.57	74.12	72.88
nl	72.31	60.26	72.46	73.00	71.56
en	77.67	76.50	77.72	77.78	77.16
en_lines	73.77	55.38	73.39	74.35	73.68
en_partut	75.66	56.99	74.30	74.97	74.30
et	60.11	44.93	59.87	60.46	59.75
fi fi_ftb	74.99	73.30	73.90	74.76	73.81
n_no fr	75.17 8 3.29 1		73.49	74.04	73.18
fr_sequoia	81.73	80.75	83.54	83.57	83.15
-	74.95	42.68	74.31	74.95	74.31
gl de	70.80	66.15	70.81	70.98	70.33
got	66.84	60.09	66.65	67.90	66.76
el	75.41	65.71	75.08	75.08	75.08
he	75.46	73.90	75.18	74.35	75.49
hi	85.85	77.60	85.62	85.49	85.49
hu	67.23	60.76	66.54	66.37	66.54
id	71.88	70.05	71.67	71.65	71.14
it	83.21	79.19	81.48	83.08	82.57
it_partut	77.32	65.10	77.24	78.22	77.24
ja	92.42	82.50	92.12	91.99	92.06
ko	51.05	51.05	51.26	51.78	50.53
la_ittb	69.43	58.16	69.48	68.71	68.54
la_proiel	65.58	65.58	65.14	65.51	64.58
lv	63.02	48.34	61.97	62.78	62.87
no₋bokmaal	82.34	79.81	81.06	81.59	81.40
no_nynorsk	80.20	79.93	79.95	80.07	79.72
cu	71.26	64.04	71.14	71.90	70.63
fa	75.93	50.92	75.73	75.88	75.29
pl	78.81	71.11	78.71	78.96	77.77
pt_br	83.44	80.95	83.28	82.87	83.61
pt	81.87	80.74	81.87	80.74	81.87
ro	77.70	69.25	77.14	77.74	76.72
ru	54.21	50.80	54.63	53.77	53.94
sk	75.10	65.66	73.61	75.06	73.69
sl	79.72	69.10	79.19	79.73	79.20
es_ancora	80.35	79.65	80.54	80.57	81.02
es	81.57	79.64	80.72	81.29	81.51
sv_lines	73.43	69.37	73.92	73.40	73.19
sv	72.88	68.01	71.97	72.92	72.47
tr	53.91	45.92	51.95	54.19	51.99
ur	75.23	66.55	74.60	75.36	74.20
vi	54.16	44.44	53.82	53.38	54.10
average	72.6545	64.9818	72.2545	71.3636	71.0727

Table 1: Monolingual corpus selection results on dev datasets. The numerical entries are LAS, and the bar indicates corpus compression amount. Length-based is trimmed based on the length of sentences in training data. Tri-gram POS sequences and POS-relation-POS are trimmed based on similarities between full training and dev data. *intersection* applies two feature extraction together (POS sequences and POS-relation-POS trimming). Threshold is fixed for all languages (0.1). We also indicate ratios of trimming of training datasets alongside parsing results.



Figure 3: Threshold estimation for Dutch (nl) and German (de): Length scale thresholds based on the average number of words in the dev data set. Y-axis shows LAS (left) and the number of sentences (right).



Figure 4: Threshold estimation for surprise languages: we fix 0.3 for tri-gram POS sequences and we vary between 0.7 and 0.9 for POS-dep-POS. Y-axis shows LAS.

based approaches, parsing results drop significantly. The empirical reasons that naive lengthbased approaches do not work well may be worth further consideration, but as a general matter the length metric is overly simplistic and may omit significant amounts of pertinent training data; similarity metrics, by contrast, attempt to retain the most pertinent data.

4 Cross-lingual Similarities for Surprise Language Parsing

We use the same similarity measures to identify the training data of the surprise language. Since surprise languages are provided without training data, we select the training datasets from training datasets of *all* languages by calculating similarities with sample datasets of surprise languages. Figure 4 shows threshold estimation for surprise languages. We fix the similarity threshold at 0.3 for tri-gram POS sequences because larger thresholds provide too little training data, and the smaller thresholds do not compress adequately. Thus, we tune based on resulting corpus size.

We vary similarity threshold between 0.7 and 0.9 for POS-dep-POS. 0.3 for tri-gram POS sequences and 0.7 for POS-dep-POS both result in a size of about 25% of the training data set for monolingual corpus selection.

5 Results

For the submitted official results through TIRA (Potthast et al., 2014), we use the *intersection* model for all languages. Since we focus on the corpus selection, we do not perform additional preprocessing and we use the provided training datasets as they are. We fix 0.1 both for trigram POS sequences and POS-dep-POS because it gives the best results for dev datasets for mono-lingual training. We fix 0.3 for tri-gram POS sequences and we use the thresholds described in Table 2 for POS-dep-POS to select training datasets from all languages for surprise language parsing. We provide the basic parsing model for PUD treebanks, for example, we use cs parsing model for

bx	r	hsb	kmr	sme
0.	8	0.9	0.8	0.7

Table 2: POS-dep-POS thresholds for surprise languages

lang	BASELINE UDPipe 1.1	UALING
All treebanks	68.35	65.24
Big treebanks only	73.04	69.59
PUD treebanks only	68.33	64.29
Small treebanks only	51.80	52.27
Surprise languages only	37.07	34.57

Table 3: Summary of LAS results

cs_pud treebank without any modification. We obtain 65.24% LAS F_1 score for the submitted model where we position 22nd. Table 3 and 4 show the summaries of LAS and results per treebank, respectively.

6 Discussion and Conclusion

In this paper, we introduced the idea of refining the training datasets to UD parsing and cross-lingual parsing to select training datasets from the same language and other languages, respectively. While our approach reduced the size of training data significantly, we retained performance within 0.5% of the baseline system. Additionally, corpus refinement methods can also be of utmost importance in trimming the size of training data for algorithmically intense algorithms or large scale system deployment runtime performance. A total runtime on entire treebanks is only around 10min with a default setting, which is fast enough; additional optimization may improve this. The size of parsing models is smaller than the baseline because we use only the subset of the entire training datasets. Even though we don't use any external data, our final results are competitive to the baseline system even with smaller datasets. The current results presented here show promise, and there exists potential for further refinement by, for instance, using different similarity metrics. Exploring different similarity metrics may further enhance performance for other NLP tasks as well as UD parsing.

References

Andrew B. Clegg and Adrian J. Shepherd. 2005. Evaluating and Integrating Treebank Parsers on a Biomedical Corpus. In *Proceedings of the ACL Workshop on Software*. Association for Computational Linguistics, Ann Arbor, Michigan, pages 14– 33. http://www.aclweb.org/anthology/W05-1102.

- Joakim Nivre, Željko Agić, Lars Ahrenberg, et al. 2017. Universal dependencies 2.0 CoNLL 2017 shared task development and test data. LIN-DAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University. http://hdl.handle.net/11234/1-2184.
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A Multilingual Treebank Collection. In Luis von Ahn, editor, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), Portorož, Slovenia.
- Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A Universal Part-of-Speech Tagset. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012). European Language Resources Association (ELRA), Istanbul, Turkey, pages 2089–2096.
- Martin Potthast, Tim Gollub, Francisco Rangel, Paolo Rosso, Efstathios Stamatatos, and Benno Stein. 2014. Improving the Reproducibility of PAN's Shared Tasks:. In Evangelos Kanoulas, Mihai Lupu, Paul Clough, Mark Sanderson, Mark Hall, Allan Hanbury, and Elaine Toms, editors, *Information Access Evaluation. Multilinguality, Multimodality, and Interaction: Proceedings of the 5th International Conference of the CLEF Initiative (CLEF 2014)*, Springer International Publishing, Sheffield, UK, pages 268–299. https://doi.org/10.1007/978-3-319-11382-1_22.
- Sebastian Schuster and Christopher D. Manning. 2016. Enhanced English Universal Dependencies: An Improved Representation for Natural Language Under-

standing Tasks. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), Paris, France.

- Milan Straka, Jan Hajic, and Jana Straková. 2016. UD-Pipe: Trainable Pipeline for Processing CoNLL-U Files Performing Tokenization, Morphological Analysis, POS Tagging and Parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), Paris, France.
- 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 Gökırmak, Anna Nedoluzhko, Silvie Cinková, Jan Hajič jr., 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, Hěctor Martínez Alonso, 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 Strnadova, 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. Association for Computational Linguistics.

	BASELINE UDPipe 1.1	UALING		BASELINE UDPipe 1.1	UALING
ar	65.30	61.86	hsb	53.83	46.78
ar₋pud	43.14	42.57	hu	64.30	64.18
bg	83.64	82.43	id	74.61	72.59
bxr	31.50	19.15	it	85.28	84.01
ca	85.39	81.93	it_pud	83.70	82.31
cs	82.87	78.04	ja	72.21	71.75
cs_cac	82.46	79.59	ja_pud	76.28	75.61
cs_cltt	71.64	71.86	kk	24.51	24.75
cs_pud	79.80	76.75	kmr	32.35	40.57
cu	62.76	61.35	ko	59.09	52.61
da	73.38	72.61	la	43.77	44.62
de	69.11	66.00	la_ittb	76.98	73.29
de_pud	66.53	63.65	la_proiel	57.54	54.53
el	79.26	78.43	lv	59.95	59.73
en	75.84	72.84	nl	68.90	66.14
en_lines	72.94	72.55	nl_lassysmall	78.15	72.99
en_partut	73.64	73.59	no_bokmaal	83.27	80.56
en_pud	78.95	76.42	no_nynorsk	81.56	79.16
es	81.47	78.73	pl	78.78	78.24
es_ancora	83.78	80.90	pt	82.11	36.40
es_pud	77.65	76.31	pt_br	85.36	82.58
et	58.79	58.85	pt_pud	73.96	35.76
eu	69.15	66.77	ro	79.88	76.68
fa	79.24	76.23	ru	74.03	73.56
fi	73.75	73.06	ru_pud	68.31	67.64
fi_ftb	74.03	72.60	ru_syntagrus	86.76	52.22
fi_pud	78.65	77.22	sk	72.75	72.95
fr	80.75	77.30	sl	81.15	79.29
fr_partut	77.38	78.29	sl_sst	46.45	46.09
fr_pud	73.63	72.03	sme	30.60	31.77
fr_sequoia	79.98	79.12	SV	76.73	75.32
ga	61.52	62.20	sv_lines	74.29	72.63
gl	77.31	74.02	sv_pud	70.62	69.43
gl_treegal	65.82	66.12	tr	53.19	50.69
got	59.81	57.62	tr_pud	34.53	33.55
grc	56.04	52.56	ug	34.18	34.97
grc_proiel	65.22	62.07	uk	60.76	61.12
he	57.23	55.79	ur	76.69	74.92
hi	86.77	85.56	vi	37.47	35.98
hi_pud	50.85	50.83	zh	57.40	55.85
hr	77.18	74.59			
			avg	68.35	65.24

Table 4: LAS results per treebank. We highlight the score where we can improve the results compared to the baseline system.