Empirically sampling Universal Dependencies

Natalie Schluter
IT University of Copenhagen
natschluter@itu.dk

Željko Agić
IT University of Copenhagen
zeag@itu.dk

Abstract

Universal Dependencies incur a high cost in computation for unbiased system development. We propose a 100% empirically chosen small subset of UD languages for efficient parsing system development. The technique used is based on measurements of model capacity globally. We show that the diversity of the resulting representative language set is superior to the requirements-based procedure.

1 Introduction

The development of natural language parsing systems has historically relied mainly on the central benchmarking dataset the Penn Treebank, as well as, to a lesser extent, a restrictive selection of very well-resourced languages like German and Chinese. This is problematic in that (1) the development of the technology risks being highly biased towards English and other resource-rich languages and their particular annotations for syntax, (2) the technology is inadequately benchmarked against a central group of languages that do not reflect the linguistic diversity required to adequately evaluate parsing systems with respect to language in general, and (3) the development of linguistic resources for unrepresented or poorly represented languages continues to be erroneously regarded as independent of the development of parsing systems, rather than integral to it.

The Universal Dependencies (UD) project (Nivre et al., 2016), has made great strides towards remedying this situation, by providing a single unified syntactic framework and related support for treebank development. The Universal Dependencies 1.4 resource now comprises 64 different treebanks covering 47 different languages (Nivre et al., 2017), and these numbers continue rising (v2.0 is set to add three languages and six treebanks). Parsing scores are now expected to be reported over all (modern) languages if not all treebanks as macroaverages of scores.

With this added diversity in treebanks and languages, and with a growing trend towards more computationally intensive learning algorithms that promise greater accuracy (such as neural networks), feasible parser development should be a rising concern. The availability of computational resources to develop–that is, to train across all interesting parameter/hyper-parameter settings–neural network models for 64 treebanks within a reasonable amount of time will counteract progress and shut out researchers without adequate computational resources. Moreover, there are environmental concerns for the inefficient use of power in the language-exhaustive development of these resources.

In this paper, we provide an entirely empirically motivated sub-sample of nine languages that can can be used to develop monolingual parsing resources. The method uses delexicalised parser performance as a measure of similarity to construct a language similarity network. The network is naturally partitioned into language groups using a standard network clustering algorithm, which does not take the number of clusters as a parameter. The clusters are assumed to be diverse between them but coherent within them, with respect to their individual parser models. Using this technique, the mean and standard deviation of monolingual unlabeled accuracy scores for cluster representatives are found to be close to the true average and standard deviation. Future monolingual parsing systems can extrapolate parser performance over the entire set of languages, using only the set of nine representative languages listed in Table 1, which interestingly excludes English, Chinese, German, and Czech.

Efficient parser development for UD languages. As an efficient alternative to exhaustive parameter search across 47 languages (or 64 treebanks), the

1. Polish
2. Italian
3. Norwegian
4. Old Church Slavonic
5. Sanskrit
6. Coptic
7. Hebrew
8. Indonesian, and
9. Dutch

Table 1: Representative languages for UD parsing resource development.

method we propose for the development of parsing resources is the following:

1. **Development**: develop parsing resources over only the nine languages in Table 1, optimising for average and standard deviation of unlabeled attachment across all languages.

2. **Full testing**: using the parameters discovered in step (1), report final average parsing scores and standard deviation over *all* UD languages.

In Section 3 we will outline the network analytic method for determining these nine representative languages empirically. First we discuss the only preceding approach to sampling UD languages for parser development; the approach is essentially non-empirical.

2 Related work

De Lhoneux and Nivre (2016) presented the first approach to language sampling from UD. They hand-picked a set of representative languages based on the following requirements:

1. **Language family**: include exactly one language from each of 8 coarse-grained language families, and no more than one from each of 15 fine-grained language families,

2. **Morphological diversity**: include at least one isolating, one morphologically rich and one inflecting language,

3. **Treebank size and domain**: ensure varied treebank size and domain,

4. **Non-projectivity**: include one language with a large amount of non-projective trees.

De Lhoneux and Nivre (2016) also considered the quality of treebanks and selected those languages that had as few annotation inconsistencies as possible. To ensure comparability, they also only consider treebanks with morphological features. They selected eight languages: Czech, Chinese, Finnish, English, Ancient Greek-PROIEL, Kazakh, Tamil, and Hebrew (cf. Table 2).

Our method differs in that it is entirely empirical, based on delexicalised parsing model similarity. Note that we also control for treebank size and exclude all morphological information.

3 Methodology

Delexicalised and projection-based parser approaches form the state-of-the-art for cross-lingual dependency parsing systems (Rasooli and Collins, 2015). Moreover, as shown by Agić et al. (2016) in upper-bound experiments, languages that are well-known to hold similar syntactic behaviours to one another, given that they come from the same language family, often generate better cross-lingual parsers for one another.

In our approach, we use delexicalised cross-lingual parsing scores to the indicate parser generalisation capacity from one language to another. As such, these parsing scores can be seen as a sort of similarity score between languages. The more similar the POS sequences and associated syntactic structures are between languages, the more similar the optimal parsing model to parse them and the better the resultant delexicalised parsing scores between them. We call this similarity score, (optimal) **model similarity**.

We need a global account of model similarity between UD languages in order to select a naturally small representative subset of UD languages based on maximal coverage of model capacities.

Building the network. We first create a complete weighted directed network $G = (V, E, w)$ to reflect model similarity. Each node in V represents a language from the UD dataset. We make arcs between all ordered pairs of nodes and decorate each arc with a weight as follows.

For a pair of languages L_1 and L_2 in our dataset, the arc (L_1, L_2) is the unlabeled attachment score of the delexicalised parser trained on L_1 and evaluated on L_2. In Section 4, we give the precise parameters of these experiments. The network thus created can be seen to roughly model the flow of model similarity. In order to transform these edge weights into probabilities, which our clustering algorithm requires, we put the set of outgoing weights of a node through soft-max at temperature

language		flow	rank
Cluster 1			
pl	Polish	0.134645	2
sl	Slovenian	0.120378	3
bg	Bulgarian	0.0772124	5
uk	Ukrainian	0.0324838	8
cs	Czech	0.0226545	11
sk	Slovak	0.0105861	17
hr	Croatian	0.00662242	19
de	German	0.00651388	20
ru	Russian	0.00620382	21
el	Greek	0.0039794	23
et	Estonian	0.00267263	24
fi	Finnish	0.000232028	38
lv	Latvian	3.36723e-05	43
Cluster 2			
it	Italian	0.180703	1
ca	Catalan	0.0894462	4
es	Spanish	0.0753139	6
fr	French	0.0598804	7
pt	Portuguese	0.0133104	16
ro	Romanian	0.00190421	26
vi	Vietnamese	0.000169612	39
Cluster 3			
no	Norwegian	0.020558	13
sv	Swedish	0.019848	14
da	Danish	0.00897288	18
en	English	0.00116163	29
Cluster 4			
cu	Old Church Slavonic	0.0242178	10
got	Gothic	0.0212005	12
la	Latin	0.00163673	28
grc	Ancient Greek	0.000146437	40
Cluster 5			
sa	Sanskrit	0.0171218	15
tr	Turkish	0.00405451	22
ta	Tamil	0.00218517	25
hi	Hindi	0.00175879	27
ug	Uyghur	0.00105993	30
eu	Basque	0.000897161	31
kk	Kazakh	0.00084926	32
hu	Hungarian	0.000793513	33
ja	Japanese	0.000640374	35
gl	Galician	6.85458e-05	42
zh	Chinese	4.28534e-06	46
swl	Swedish Sing	5.57449e-07	47
Cluster 6			
cop	Coptic	0.0260177	9
Cluster 7			
he	Hebrew	0.000642038	34
ga	Irish	0.000452355	37
fa	Persian	3.26755e-05	44
ar	Arabic	1.8784e-05	45
Cluster 8			
id	Indonesian	0.000609376	36
Cluster 9			
nl	Dutch	0.000105621	41

Table 2: Language clusters, flow (centrality) and rankings, given temperature $\tau = 0.025$. The most central languages for clusters are highlighted in blue. Red rows are the languages chosen by de Lhoneux and Nivre (2016). And the one purple language, Hebrew, was chosen by both methods.

τ, to be determined with respect to true parsing score aggregates later.

Our goal is to use the network to determine the language representatives of the UD dataset. To do this, we run the Infomap network clustering algorithm and then extract the most important languages from each cluster.

Clustering the network naturally. We need to now cluster the nodes of the network, given its structure, but without supplying the number of languages as a parameter, in order for the output modular structure the be completely data-driven.

Infomap[1] poses the problem of the clustering of nodes in a weighted directed network as the dual of the problem of minimising the description length of a random walker's movements on a network. Intuitively, the description parts corresponding to various regions of the network may be compressed if the random walker spends longer of periods of time there.

The description of the network (the map equation) to be minimised is

$$L(M) := q_{\curvearrowright} H(Q) + \sum_{i=1}^{m} p_{i\circlearrowright} H(P_i)$$

where $q_{\curvearrowright}$ is the total given probability that the random walker enters some new cluster; $H(Q)$ is entropy of the modular structure of the network; $p_{i\circlearrowright}$ is the probability that some node in cluster i is visited together with the probability of exiting cluster i; and $H(P_i)$ the entropy of the internal network structure in cluster i.

The interested reader is referred to Rosvall et al. (2009) for more details. Infomap outputs three pieces of information that we need here: (1) The number of clusters, (2) the cluster that each node belongs to, and (3) the flow of each node in the network as determined by the random walk traversals. The larger the flow, the more central a node is within the network.

[1] `http://www.mapequation.org/code.html`

Extracting representative languages. For each cluster, the most representative (central) language of the cluster is considered to be the node with the highest flow. In terms of the random walker in the network structure, these are the nodes that are traversed the most within their own clusters, meaning that correspond to languages with highest cluster-wide model similarity. In this sense, they can act as **cluster representatives**.

Calculating parsing score aggregates. In order to fit the modular structure of the network to the true parsing score aggregates we carry out an exhaustive search for optimal temperature within the interval $\tau \in (0,1]$ at increments of 0.005. The value τ is optimal when

$$|\mu - \mu_\tau| + |\sigma - \sigma_\tau| \quad (1)$$

is minimised, where μ and σ are the true macro-average of unlabeled parsing accuracy score mean and standard deviation, and μ_τ and σ_τ are found in the same way except that parsing scores for non-cluster representatives are replaced by that of their unique cluster representatives. This corresponds to a weighted average and standard deviation of scores of cluster representatives based on cluster size.

4 Data preparation

We used UD v1.4 in our experiment. Out of the 64 treebanks it offers, we select the 47 canonical ones for the 47 languages represented in the release.

We filter out all but the following CoNLL-U features from the dataset:[2] ID, UPOSTAG, HEAD, and DEPREL. Note that all our parsers are delexicalised following McDonald et al. (2013), that is, we exclude all lexical information and learn parses over POS sequences. We also filter out all multi-word tokens.

All training data is sub-sampled up to 10k sentences so as to avoid the bias towards the largest training sets.[3] Then, we train our delexicalized models using the graph-based parser MATE with default settings (Bohnet, 2010).

All our parsers assign labels, but here we evaluate for UAS only. While LAS and UAS are the two most highly correlated dependency parsing metrics as per Plank et al. (2015), we find that the latter offers a bit more stability in constructing our similarity network. The aggregates over the 47 UD languages are: average UAS 74.45, and standard deviation UAS 9.4. An optimal language sampling method extrapolates to these aggregates as closely as possible.

5 Method visualisation and discussion

In Figure 1, on the left y-axis, we see the number of clusters generated in the network for varying temperature levels. On the right y-axis, we see the parsing score estimate over cluster representatives for varying temperatures. Equation (1) is minimised when $\tau = 0.025$ and this yields nine separate clusters for our model similarity network. The error for this temperature is 5.05 (with $|\mu - \mu_\tau| = 3.5$ and $|\sigma - \sigma_\tau| = 1.55$) as reported in Table 3.

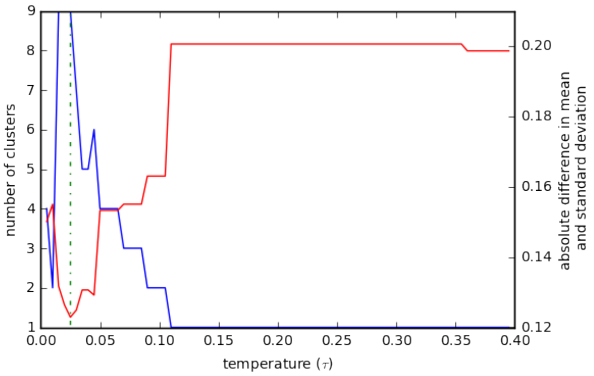

Figure 1: Number of clusters over varying temperatures, with respect to soft-max temperature. Optimal temperature at $\tau = 0.025$ (dotted green line). The number of clusters and error remain unchanged for $\tau > 0.4$.

Visualising the model similarity network. A visualisation of the network for $\tau = 0.025$ is given in Figure 2. We notice that, as expected, many of the clusters follow language family closely, but there are a number of outliers. For instance, Dutch is entirely alone in its cluster and Vietnamese is grouped together with the Romance languages.

Language centrality. In Table 2, we also see the rank of languages in terms of their centrality (flow score) in the network. The centrality score in our case provides an indication of model similarity between parsers trained on the language in ques-

[2] http://universaldependencies.org/format.html

[3] Czech, the largest training set in our UD subset, is 4.5x larger than each of the 12 languages that follow it.

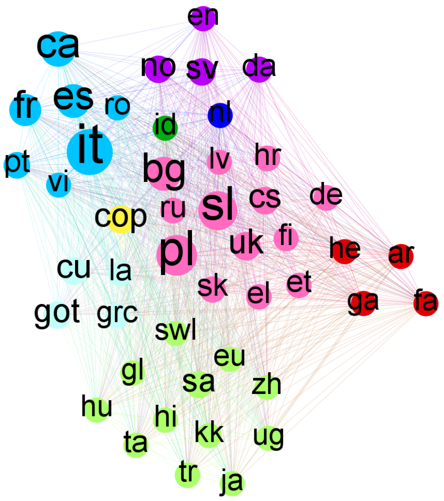

Figure 2: Visualisation of the model similarity network. Node centrality corresponds to node size.

language	cluster size	score
Polish	13	84.91
Italian	7	85.11
Norwegian	4	79.99
Old Church Slav.	4	73.72
Sanskrit	12	66.10
Coptic	1	85.01
Hebrew	4	79.60
Indonesian	1	77.73
Dutch	1	75.05
average	77.96 (`error` = 3.5)	
std	7.85 (`error` = 1.55)	
`total error`	**5.05**	

Table 3: UAS contributions and aggregates of our representative UD languages. The contributions (`cluster size`) * `score` are collected over the 9 sampled languages and normalised over the 47 languages.

language	score
Czech	78.49
Chinese	68.08
Finnish	68.00
English	79.71
Anc. Greek-P.	62.37
Kazakh	69.29
Tamil	71.39
Hebrew	79.60
average	72.12 (`error` = 2.33)
std	6.45 (`error` = 2.95)
`total error`	5.28

Table 4: UAS and aggregates of de Lhoneux and Nivre's (2016) representative UD languages. The score aggregates are calculated over the 8 sampled languages.

tion and those of all other languages in the network. Surprisingly, English is ranked in 29th position, which provides simple empirical evidence that parsing resources developed mainly on and optimised for English risk suboptimal overall performance. Interestingly, other well-studied languages like Chinese and Arabic have considerably low rank both in the entire networks well as in their respective clusters.

In Table 2 we have also highlighted the representative languages chosen by de Lhoneux and Nivre (2016). We see that according to our empirical model, the languages they chose reflect neither the centrality nor the diversity intended.

Comparing extrapolations. The error for de Lhoneux and Nivre's (2016) representative set is given in Table 4. We see that total error is lower in the parsing model similarity method we describe here. However, because of the combined optimisation of mean and standard deviation, our sample over-estimates general performance, while de Lhoneux and Nivre (2016)'s sample underestimates the reliability of the parser to achieve the mean performance.

6 Concluding remarks

We have shown the first 100% empirical method for determining a small representative sample of UD languages for parser development, and have proposed an associated methodology. In particular, for the Universal Dependencies v1.4, we given a specific subset of nine languages on which parsing systems can be developed efficiently.

The language clusters presented here have many similarities with well-studied language family distinctions, but also many differences. These clusters could provide an interesting technology-motivated study of syntactic similarity between languages.

References

Željko Agić, Anders Johannsen, Barbara Plank, Héctor Martínez Alonso, Natalie Schluter, and Anders Søgaard. 2016. Multilingual projection for parsing truly low-resource languages. *Transactions of the Association for Computational Linguistics*, 4:303–312.

Bernd Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computa-*

tional Linguistics (Coling 2010), pages 89–97, Beijing, China, August. Coling 2010 Organizing Committee.

Miryam de Lhoneux and Joakim Nivre. 2016. Ud treebank sampling for comparative parser evaluation. In *Proceedings of SLT 2016*.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria, August. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, and et al. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*.

Joakim Nivre, Željko Agić, Lars Ahrenberg, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Eckhard Bick, Cristina Bosco, Gosse Bouma, Sam Bowman, Marie Candito, Gülşen Cebirolu Eryiit, Giuseppe G. A. Celano, Fabricio Chalub, Jinho Choi, Çar Çöltekin, Miriam Connor, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Marhaba Eli, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Cláudia Freitas, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökrmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Nizar Habash, Jan Hajič, Linh Hà M, Dag Haug, Barbora Hladká, Petter Hohle, Radu Ion, Elena Irimia, Anders Johannsen, Fredrik Jørgensen, Hüner Kaşkara, Hiroshi Kanayama, Jenna Kanerva, Natalia Kotsyba, Simon Krek, Veronika Laippala, Phng Lê Hng, Alessandro Lenci, Nikola Ljubešić, Olga Lyashevskaya, Teresa Lynn, Aibek Makazhanov, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Amir More, Shunsuke Mori, Bohdan Moskalevskyi, Kadri Muischnek, Nina Mustafina, Kaili Müürisep, Lng Nguyn Th, Huyn Nguyn Th Minh, Vitaly Nikolaev, Hanna Nurmi, Stina Ojala, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Guy Perrier, Slav Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Lauma Pretkalnia, Prokopis Prokopidis, Tiina Puolakainen, Sampo Pyysalo, Alexandre Rademaker, Loganathan Ramasamy, Livy Real, Laura Rituma, Rudolf Rosa, Shadi Saleh, Manuela Sanguinetti, Baiba Saulīte, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Lena Shakurova, Mo Shen, Dmitry Sichinava, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Dima Taji, Takaaki Tanaka, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Larraitz Uria, Gertjan van Noord, Viktor Varga, Veronika Vincze, Jonathan North Washington, Zdeněk Žabokrtský, Amir Zeldes, Daniel Zeman, and Hanzhi Zhu. 2017. Universal dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University in Prague.

Barbara Plank, Héctor Martínez Alonso, Željko Agić, Danijela Merkler, and Anders Søgaard. 2015. Do dependency parsing metrics correlate with human judgments? In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 315–320, Beijing, China, July. Association for Computational Linguistics.

Mohammad Sadegh Rasooli and Michael Collins. 2015. Density-driven cross-lingual transfer of dependency parsers. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 328–338, Lisbon, Portugal, September. Association for Computational Linguistics.

Martin Rosvall, Daniel Axelsson, and Carl T. Bergstrom. 2009. The map equation. *Eur. Phys. J. Special Topics*, 178.