A typological perspective on evaluation for Universal Dependencies

Jimmy Callin
Uppsala University
jimmy.callin@gmail.com

Abstract

Established evaluation metrics assumes that all dependency relations are equally important. Furthermore, when comparing parsing results in a cross-linguistic manner, it is easy to assume equal attachment scores is equivalent to equal performance. We demonstrate why this is not the case, especially when comparing languages with large difference in morphological complexity, and emphasize the necessity for new evaluation metrics that takes these considerations into account. We also present two alternative evaluation metrics motivated by these findings.

1 Introduction

The role of language processing is becoming increasingly multi-lingual. This is reflected in recent efforts into providing dependency parsing frameworks that can reliably be applied on a multitude of languages. One of the currently most ambitious projects in this area is the Universal Dependencies (UD) framework (Nivre, 2015), where the goal is to create a parsing framework with a cross-linguistically consistent grammatical annotation. The purpose of this work is to remove the requirement of language specific components, which has up to this point been a necessity due to inconsistent annotation standards.

Data-driven evaluation metrics have been used as long as the availability of treebanks (see Collins (1999) chap. 4 and Carroll et al. (1998) for surveys on early methods and results). To evaluate statistical parsing models, *Unlabeled Attachment Score* (UAS) has been in use under different names since the early days. Eisner (1997) was first to call it attachment score, which refers to work done by Lin (1998). *Labeled Attachment Score* (LAS) was first introduced in Nivre et al. (2004) to emphasize the importance of correct labeling. UAS is defined as the accuracy of correct attachments in a test collection, while LAS includes the additional constraint of requiring a correct attachment label. These have in their simplicity and intuitiveness served the research area well, but their design relies on a number of assumptions we argue do not hold in the context of UD.

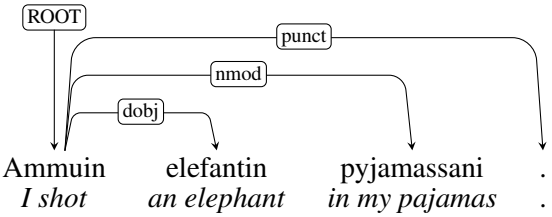

Figure 1: Finnish dependency tree for *I shot an elephant in my pajamas*. Note the edge count of Finnish being 4, while the English edge count is 8.

Firstly, the design of attachment scores assumes that we know next to nothing about the taxonomy and design choices of the parsing framework. This has historically been necessary since there has not been a consistently adapted framework for dependency parsing across many languages. We argue that recent progress in UD has made this assumption invalid. UD has a carefully specified framework that its treebanks have to adapt, and we can exploit this constraint to learn more about the performance of a

model.

Secondly, parsing results are increasingly juxtaposed in a cross-linguistic manner, and this trend will likely continue with the establishment of UD. It is not uncommon to compare the output of e.g. English and Finnish under the assumption that equal evaluation scores is equivalent to equal parsing performance. The reason why this is problematic becomes apparent when studying grammatical morphemes in languages where these may be unbounded (i.e. function words) with languages where they typically are bounded on content words (i.e. affixes).

In figure 1, we have a Finnish sentence with four edges, while the equivalent sentence in English requires a total of eight edges. The problem appears as soon as we introduce a parsing error into the trees: one faulty edge in the Finnish sentence would result in a performance reduction of 25%, while the same error in the English sentence only would reduce the accuracy by 12.5%. Function words are regular in their appearance, and should therefore be relatively easy for a model to parse correctly. Our hypothesis is that languages with a comparatively large number of function words, like English, receive a more or less free performance boost when using classic evaluation metrics.

Based on these intuitions, our aim is to contribute to these two questions:

- How much do languages with a large number of unbounded grammatical morphemes benefit from current evaluation schemes?
- Would focusing on correct classification of content word dependencies be a better evaluation scheme for cross-linguistic parsing performance?

2 Related work

Not much work has been done in cross-linguistic evaluation, and papers presenting evaluation scores on several languages simply use previously available metrics without analyzing their shortcomings in such a context. Before the work on UD was initiated, there have been several attempts at automatic normalization of dependency treebanks into a common format for a more robust evaluation (Zeman et al., 2012). In light of this work on cross-linguistically consistent annotation frameworks, Tsarfaty et al. (2011) take a separate approach with cross-framework evaluation, where they suggest an evaluation technique that is robust towards differing annotation criteria.

Since UAS arguably has been more popular of the two attachment scores, finding performance results on subsets of labeled dependency relations is difficult. In cases of extensive error analysis it is possible to find sections devoted to this (Plank, 2011). There has also been work looking at specific constructions, e.g. unbounded dependency evaluation (Nivre et al., 2010), where they argue that some dependency relations are more critical for a parser to get right than others.

Plank et al. (2015) look closer at whether or not manual parsing evaluation correlate with standard dependency metrics, coming to the conclusion that none of today's established metrics are especially well correlated with human quality judgment. One of their main findings is that humans tend to consider content dependencies to be of more importance than function dependencies, which fits well with the assumptions made in this paper.

3 Data

We will be using a subset of the Universal Dependencies treebank 1.2 (Nivre, 2015). To keep them as internally consistent as possible, all treebanks must adhere to the following criteria:

- They have morphological features.
- They have at least 30K tokens.
- They have less than 25% non-projective trees.
- In the case of more than one valid treebank for a language, choose the treebank with manual corrections or largest token count.

A total of 17 treebanks were removed. 5 of these had too few tokens, 4 lacked features, 4 had too many non-projective trees, while 4 treebanks were language duplicates. This leaves us with the 20 languages listed in table 1. Most notably we lost the French and German treebanks.

For measuring the correlation of metrics to manual evaluation, we will be using parts of the human judgment data as provided by Plank et al. (2015). Not all languages in the dataset are from the UD

Treebank	Token size	Non-proj ratio
Arabic	282K	0.01
Bulgarian	156K	0.03
Croatian	87K	0.05
Czech	1503K	0.09
Danish	100K	0.12
English	254K	0.02
Finnish	181K	0.04
Gothic	56K	0.12
Greek	59K	0.21
Hebrew	115K	0.00
Hindi	351K	0.04
Italian	252K	0.01
Norwegian	311K	0.02
Old Church Slavonic	57K	0.13
Persian	151K	0.04
Polish	83K	0.00
Portuguese	212K	0.15
Slovenian	140K	0.11
Spanish	423K	0.02
Swedish	96K	0.01

Table 1: Selected treebanks from the UD 1.2 treebank collection, with their token size and amount of non-projective trees.

Function relations	Content relations
aux auxpass case cc cop det expl mark neg mwe	acl advcl advmod amod appos ccomp compound conj csubj csubjpass dislocated dobj iobj list name nmod nsubj nsubjpass nummod parataxis remnant root vocative xcomp
Other	
list dep foreign reparandum punct goeswith discourse	

Table 2: Classification of content and function relations.

treebanks, thus only English, German, and Spanish are used.

4 Categorizing dependency relations

We motivate our choice of function and content relations based on the specification of universal dependency relations[1], linguistic intuition, and a newly developed statistical method. First, let us define what we consider to be function and content relations:

- A *function relation* is a relation that links a word with a function word.
- A *content relation* is a relation that either has *root* as its head, or links a content word with another content word.

4.1 Categorization by specification

Given the previous definition as well as the specification of universal dependency relations, we can categorize a relation based on how it should occur in UD treebanks. Going through each dependency relation in this manner produces a classification as presented in table 2.

We chose to remove some relations where we cannot make assumptions of its content, labeled *other*. The *foreign* relation has no restrictions on what type of word it should choose as a dependent as long as it is a foreign word. *List* is used in cases where the content cannot be easily analyzed. *Reparandum* and *dep* are neither of semantic nor syntactic nature and we cannot make any assumptions of their content. *Punct* and *discourse* are removed for similar reasons, but also due to particles often being ignored in other evaluation schemes.

4.2 Placing dependency relations on a function–content spectrum

Next question to answer is if we can motivate our classification not only on linguistic intuition and the specification of dependency relations, but also from an empirical perspective. We do this by adhering to our previous definition on function and content relations, and what we know of the nature of function words. Since function words are part of closed word classes, meaning new words rarely get introduced into their categories, we can expect the number of distinct word types to be quite small, especially when compared to the word classes shared by typical content words such as *nsubj*.

Assuming this holds, we expect that the probability of a word given a function relation to be zero for all but a few cases, while the probability of a word given a content relation should be much more evenly

[1] http://universaldependencies.github.io/docs/u/dep/index.html

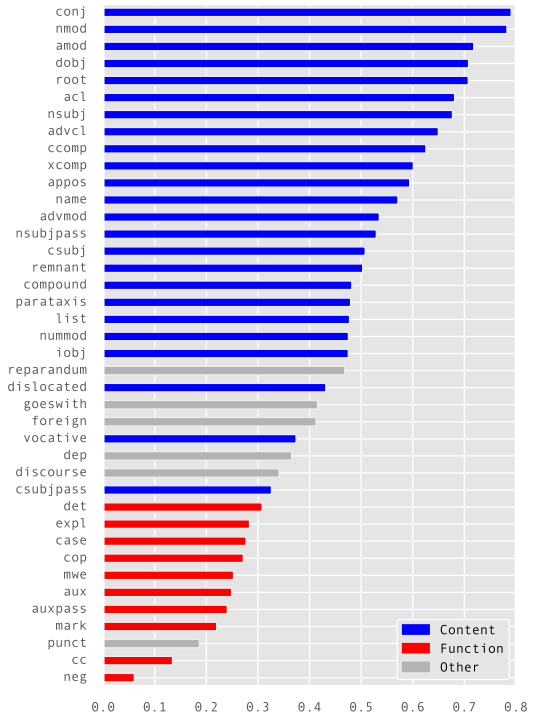

Figure 2: Averaged word dependency entropy for all universal dependency relations, with the manually created categories.

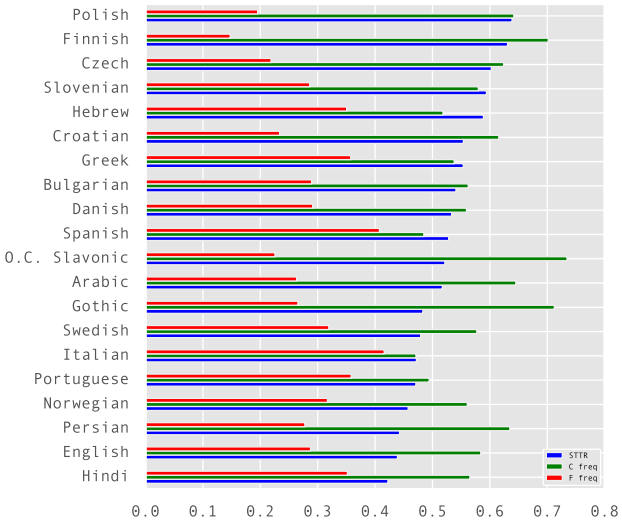

Figure 3: Standardized type/token ratio for chunks of 1000 tokens (blue), with content relation frequency ratio (green) and function relation frequency ratio (red). High STTR score implies high morphological complexity. $R(\text{Content freq}, \text{STTR}) = 0.23$, $R(\text{Function freq}, \text{STTR}) = -0.52$.

spread. This can be quantified by measuring a relation's entropy for its word probabilities. We call this measure *word dependency entropy* (WDE). Calculated for all treebanks we get the averaged WDE.

Here we formally define word dependency entropy. A probability distribution p takes as input a word w conditioned on a treebank $t \in T$, and a dependency relation r. H is the entropy function that takes $p(w|r,t)$ as input and calculates its entropy. The entropy function is normalized by its upper bound $\log n_w$, where n_w is the size of the vocabulary. This keeps the range of the function to $[0,1]$. To calculate the WDE for a set of treebanks, we average WDE for all treebanks $t \in T$. In the case a dependency relation is not present in a treebank, we set its WDE to 0.5 to imply that it is neither a content nor a function relation.

This produces the following mathematical functions:

$$\text{WDE}(r,t) = \frac{H(p(w|r,t))}{\log n_w}$$

$$\text{Averaged WDE}(r,T) = \frac{1}{n_T} \sum_{t \in T} \text{WDE}(r,t)$$

The averaged WDE for UD 1.2 is presented in figure 2, along with our manual categorization of dependency relations. Ignoring the *other* relations, we find that the WDE gives an intuitive ordering of the dependency relations, while empirically supporting our choice of content and function relations.

4.3 Finding correlation with external measurements

We would expect the ratio of function words in a given language's treebank to correlate with its degree of synthesis. Measuring degree of synthesis is not a trivial task, and there have been several proposed algorithms for this. Despite having obvious drawbacks, an often used indirect measurement of degree of synthesis is the type/token ratio (Kettunen, 2014). This assumes that synthetic languages, with their morphologically rich systems, will have fewer tokens per word than analytic languages such as English or Hindi. This is not par-

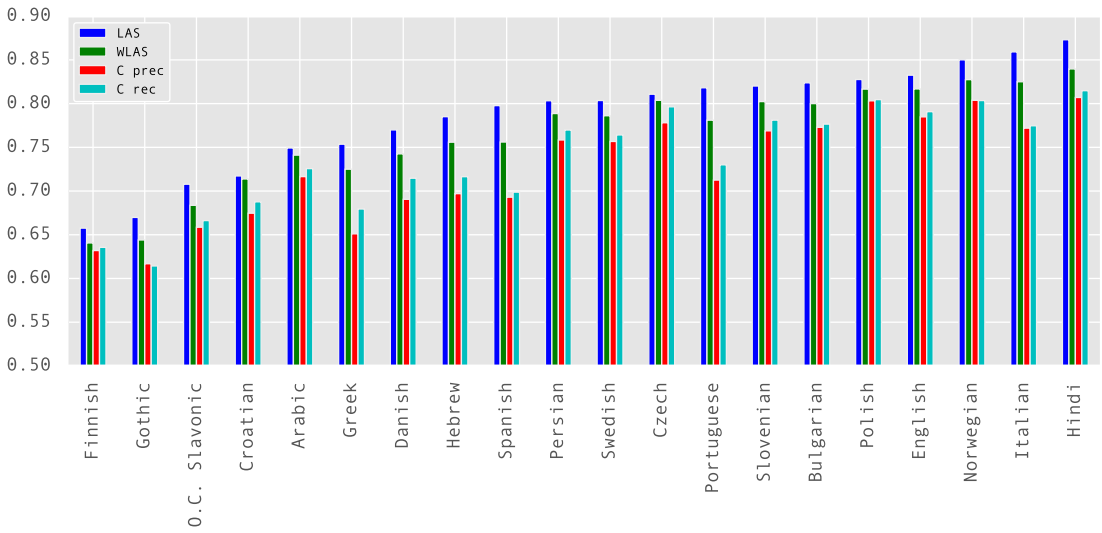

Figure 4: Overall LAS, WLAS, precision and recall for content dependencies. Sorted by LAS.

ticularly robust when comparing across corpora of different sizes. As such, we will be using the *standardized* type/token ratio (STTR), which calculates average TTR in chunks of 1000 tokens[2]. Figure 3 shows that there is a weak correlation between languages' frequency ratio of content relations and their STTR, while the negative correlation against ratio of function relations is much stronger. Some languages for content relations are clear outliers such as Old Church Slavonic, Arabic, Gothic, Persian, English, and Hindi. It is not clear why the ratio correlation is stronger for function relations than for content relations. Whether this is a result of a bad degree of synthesis measurement, a bad classification of content dependency relations, or a combination of both is up for discussion.

5 Experimental setup

Based on the previous findings, we propose two alternative metrics to the LAS. The first metric is based on our manual classification of content and function dependencies, while the latter is exploiting the weights outputted by the WDE. For testing the evaluation metrics, we train MaltParser 1.7 using Nivre Arc-Eager with default settings on each treebank's training data and parse the included test data (Nivre et al., 2006).

Performance of content relations In this metric, we look at the precision and recall for all content dependency relations, ignoring any relation that is not a part of this class. Since not all dependency relations are involved, the precision and recall can differ and thus become interesting to analyze separately. We call these *content precision* and *content recall*.

Weighting relations by their WDE We weight each dependency relation by its averaged WDE as presented in figure 2. This will increase the importance of content relations, while the function relations provide less to the overall score. We call this the *Weighted Labeled Attachment Score* (WLAS). Using WLAS has the additional interesting property of also being easily calculated and deployable to non-UD frameworks.

6 Evaluation

Figure 4 lists the parsing results for all languages, with LAS, WLAS, and content precision and recall. We can tell that LAS is consistently providing higher scores for each output compared to WLAS, while the content precision and recall scores are substantially lower. There are small differences for precision and recall for all except the worst performing

[2]Introduced by Mike Scott in `http://lexically.net/downloads/version6/HTML/index.html?type_token_ratio_proc.htm`

languages where, given the higher recall, the parsing model seems to have a bias towards content dependencies.

Table 3 lists the Pearson correlation coefficient between treebanks for WLAS, LAS, content precision and recall, function precision and recall, and the frequency ratio of content and function relations in the treebanks. The correlation between the various suggested measurements, as well as with LAS, are quite strong. The content frequency ratio has a strong negative correlation with function frequency ratio, and negative across all the other measurements as well. Function frequency ratio has more or less strong positive correlations with all metrics.

Figure 5 shows what happens with the variance when cumulatively adding languages in a top-scoring fashion for each measurement. For content precision and recall as well as WLAS, the variance is lower for high performing languages, but takes off for content precision and recall when you get past the first eleven treebanks. WLAS keeps an even score for less well-performing languages compared with LAS until the end where they join content performance in a variance of about 0.0035.

Table 4 lists Spearman correlations between manual evaluation of two parser models, where the evaluators given parsed sentences in each case chose which of two parser models they consider to provide the best output. Content precision and recall are in all cases except one inferior to LAS and WLAS, where the two latter are indistinguishable.

Table 5 lists the ratio of correct parent dependency relations given a faulty relation of either function or content class, labeled or unlabeled. The parent of a token with a faulty relation is defined as the parent of the system output relation, and *not* as the parent of the gold relation. A lower ratio means that the dependency class is more prone to cascading errors. The results show that function dependencies have more cascading errors than content dependencies.

7 Discussion

Looking at function frequency and its precision in table 3, they have a correlation of 0.71. This suggests that the larger rate of function words in a language, the easier it is to parse its function words.

	LAS	C prec	C rec	F prec	F rec	C freq	F freq
WLAS	0.98	0.95	0.97	0.85	0.85	-0.58	**0.42**
LAS		0.91	0.92	0.92	0.92	-0.67	**0.54**
C prec			0.99	0.68	0.70	-0.34	**0.17**
C rec				0.72	0.72	**-0.39**	0.20
F prec					0.98	-0.77	**0.71**
F rec						-0.77	**0.71**
C freq							-0.86

Table 3: Pearson correlation matrix between treebanks for content and function frequency ratio (*C freq* and *F freq*), content precision and recall (*C prec* and *C rec*), function precision and recall (*C prec* and *C rec*), LAS, and WLAS. Boldfaced figures are mentioned in the discussion.

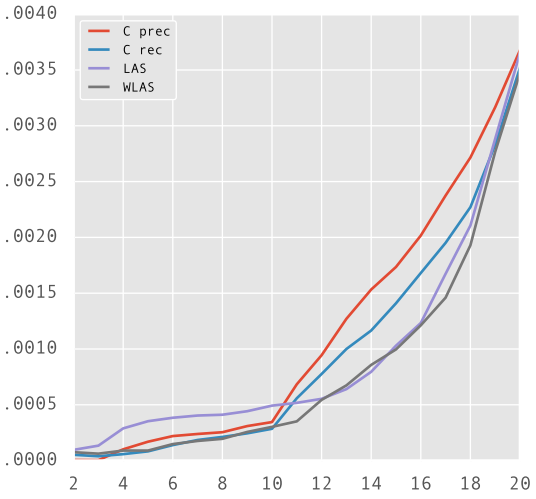

Figure 5: Cumulative variance when adding languages in a top-scoring order.

What is interesting is that this does not hold when looking at the content frequency ratio and its precision or recall, where one might expect that there is a strong positive correlation which would indicate that a high degree of content relations makes it easier to parse these classes. Instead, we find a weak negative correlation of -0.39 for recall. Furthermore, given languages' LAS scores, the more function words there is in a language, the better it performs. Even when looking at the relation between the amount of function words in a language with how well it does on content words, the correlation is weakly positive. We take this to mean that our choice of parser, while not explicitly tuned for any particular language, still

	C precision	C recall	LAS	WLAS
English	0.51	**0.58**	0.55	0.55
German	0.37	0.44	**0.46**	**0.46**
Spanish	0.45	0.46	**0.49**	0.48

Table 4: Correlations of LAS, WLAS, content precision and recall against human judgment data.

	C lab.	C unlab.	F lab.	F unlab.
Arabic	0.51	0.56	0.40	0.49
Bulgarian	0.55	0.55	0.51	0.56
Croatian	0.47	0.46	0.32	0.27
Czech	0.49	0.46	0.37	0.35
Danish	0.41	0.40	0.23	0.25
English	0.48	0.44	0.33	0.36
Finnish	0.39	0.38	0.25	0.29
Gothic	0.37	0.33	0.31	0.37
Greek	0.44	0.43	0.31	0.33
Hebrew	0.51	0.54	0.33	0.47
Hindi	0.61	0.51	0.38	0.31
Italian	0.58	0.56	0.44	0.48
Norwegian	0.55	0.51	0.36	0.38
O.C. Slavonic	0.41	0.38	0.32	0.36
Persian	0.47	0.45	0.31	0.41
Polish	0.59	0.52	0.38	0.44
Portuguese	0.47	0.46	0.31	0.32
Slovenian	0.41	0.38	0.30	0.33
Spanish	0.46	0.48	0.30	0.34
Swedish	0.47	0.45	0.36	0.39

Table 5: In the case of incorrect dependency relation of either function or content class, how frequent is it to have a correct (system output) parent relation? The lower the ratio, the more common are cascading errors.

benefits from a context with a high rate of grammatical function words. These findings support our hypothesis, that languages with a high degree of function dependency relations has an unfair advantage when comparing attachment scores across languages.

Regarding the evaluation scores for different evaluation metrics, as presented in figure 4, it is difficult to tell if any metric is better than the other. One might expect that the scoring difference of LAS and WLAS, or LAS and content performance, would correlate with its STTR score, since analytic languages like Hebrew and English have more to lose on decreasing the importance of function words. Unfortunately, this correlation is rather weak. Assuming that the STTR scoring is reliable, we believe this has to do with what we described above: languages with a high rate of function words provide a better context for content words for parsers.

Going back to table 3, a better measurement than LAS would be expected to have a weaker correlation with the function frequency ratio, showing that the importance of the amount of function words in a language decrease. While WLAS has a somewhat weaker correlation as LAS, content performance is even more so.

We ran the measurements on the human judgment data with the results given in table 4. Unfortunately, none of the metrics seem to improve upon the LAS score, which was the top scoring metric reported in the original paper. Only content recall sees some improvements over LAS for English, but other than that the results are either worse or equal to those of LAS. WLAS has overall very small changes compared to LAS, which is somewhat surprising given that the original paper commented on content relations being considered more important than function relations by the manual evaluators. This could possibly be explained by function relations overall performing quite well, and whenever there are erroneous function relations they are cascaded from faulty content relations. While this hypothesis is supported by table 5, showing how erroneous function relations in all languages are more commonly having faulty parents than incorrect content relations, the differences are not large enough to indicate that this is the only reason. This might also just be an effect of function dependencies being further down in the tree than content dependencies,

Another approach is to look at the variance of the metrics. If our initial hypothesis holds, this should mean that some of the differences found between languages before evens out, and thereby lowering the variance when compared to LAS. Figure 5 reveals that this does not hold when looking at all treebanks, but by cumulatively adding treebanks it is possible to study the effect as the performance decreases. Indeed, the variance is lower among high-performing languages for WLAS and and content performance, and could potentially be explained by that differences among top-scoring languages are much less random due to a poor parsing model and

it is first among these that the choice of metric starts to matter.

That brings us to the choice of parser model and its effect on the results. In the name of consistent treebanks, we chose to remove treebanks with a large number of non-projective trees. Another motivation was that our parser can not handle these types of trees especially well, and the results are thus unreliable when comparing across languages. Is is quite possible, and even probable, that there are many similar factors playing a role in the discussed results. For instance, the size of the treebank has definitely an effect. In future work, we should reproduce these results with alternative models and look for any similarities or differences that might strengthen our claims or explain some of the peculiarities we have seen in this work.

8 Conclusion

In this paper we have presented experiments that suggest that languages with many function dependency relations are easier to parse than languages with richer morphology, given current parser models. We have motivated the necessity for new evaluation metrics that take these considerations into account from a typological perspective, while also referring to research that motivates this from human judgment standards. We suggested two new evaluation metrics that raise the importance of content dependency relations with some initial experiments indicating their usefulness.

9 Acknowledgements

We thank Jörg Tiedemann for providing his parsing output on UD 1.0 for initial experiment development.

References

John Carroll, Ted Briscoe, and Antonio Sanfilippo. 1998. Parser evaluation: a survey and a new proposal. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, pages 447–454.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania.

Jason Eisner. 1997. An empirical comparison of probability models for dependency grammar. Technical report, University of Pennsylvania.

Kimmo Kettunen. 2014. Can Type-Token Ratio be Used to Show Morphological Complexity of Languages? *Journal of Quantitative Linguistics*, 21(3):223–245.

Dekang Lin. 1998. A dependency-based method for evaluating broad-coverage parsers. *Natural Language Engineering*, 4(02):97–114.

Joakim Nivre. 2015. Towards a Universal Grammar for Natural Language Processing. In *Computational Linguistics and Intelligent Text Processing*, pages 3–16. Springer, editions.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2004. Memory-Based Dependency Parsing. In *Proceedings of the Eighth Conference on Computational Natural Language Learning*, volumes s. 49-56. Association for Computational Linguistics.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2006. Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of LREC*, volume 6, pages 2216–2219.

Joakim Nivre, Laura Rimell, Ryan McDonald, and Carlos Gomez-Rodriguez. 2010. Evaluation of dependency parsers on unbounded dependencies. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 833–841. Association for Computational Linguistics.

Barbara Plank. 2011. *Domain adaptation for parsing*. PhD thesis, University of Groningen.

Barbara Plank, Héctor Martínez Alonso, Željko Agić, Danijela Merkler, and Anders Søgaard. 2015. Do dependency parsing metrics correlate with human judgments? In *Eighteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics.

Reut Tsarfaty, Joakim Nivre, and Evelina Andersson. 2011. Evaluating dependency parsing: robust and heuristics-free cross-annotation evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 385–396. Association for Computational Linguistics.

Daniel Zeman, David Marecek, Martin Popel, Loganathan Ramasamy, Jan Stepánek, Zdenek Zabokrtsky, and Jan Hajic. 2012. HamleDT: To Parse or Not to Parse? In *LREC*, pages 2735–2741.