A System for Multilingual Dependency Parsing based on Bidirectional LSTM Feature Representations

KyungTae Lim
LATTICE Laboratory
École Normale Supérieure & PSL Univ.
Univ. Sorbonne nouvelle & USPC
Paris, France
KISTI / DaeJeon, Korea
ktlim@ens.fr

Thierry Poibeau
LATTICE Laboratory
École Normale Supérieure & PSL Univ.
Univ. Sorbonne nouvelle & USPC
Paris, France
thierry.poibeau@ens.fr

Abstract

In this paper, we present our multilingual dependency parser developed for the *CoNLL 2017 UD Shared Task* dealing with "Multilingual Parsing from Raw Text to Universal Dependencies"[1]. Our parser extends the monolingual BIST-parser as a multi-source multilingual trainable parser. Thanks to multilingual word embeddings and one hot encodings for languages, our system can use both monolingual and multi-source training. We trained 69 monolingual language models and 13 multilingual models for the shared task. Our multilingual approach making use of different resources yield better results than the monolingual approach for 11 languages. Our system ranked 5^{th} and achieved 70.93 overall LAS score over the 81 test corpora (macro-averaged LAS F1 score).

1 Introduction

Many existing parsers are trainable on monolingual data. Normally such systems take a monolingual corpus in input, along with monolingual word embeddings and possibly monolingual dictionaries as well as other knowledge sources. However for resource-poor languages such as Kurmanji and Buryat[2], there are generally not enough resources to train an efficient parser. One reasonable approach is then to infer knowledge from similar languages (Tiedemann, 2015). Developing tools to process several languages including resource-poor languages has been conducted in many different ways in the past (Heid and Raab, 1989). Thanks to Universal Dependency (Nivre et al., 2016), it is now possible to train a system for several languages from the same set of POS tags. It has also been demonstrated that, with current machine learning approaches, parsing accuracy improves when using multilingual word embeddings (i.e. word embeddings inferred from corpora in different languages) even for resource-rich languages (Ammar et al., 2016a; Guo et al., 2015).

In this paper, we describe the development of a system using either a monolingual or multilingual strategy (depending on the kind of resources available for each language considered) for the CoNLL 2017 shared task (Zeman et al., 2017). For the multilingual model, we assume that learning over words and POS sequences is a first step from which better parsers can then be derived. For this reason, we re-used most of the training algorithms implemented for the BIST-parser since these have proven to be effective when dealing with sequential information even for long sentences, thanks to bidirectional LSTM feature representations (Kiperwasser and Goldberg, 2016).

In addition, our parser can also have recourse to multilingual word embeddings that merge different word vectors in a single vector space in order to get multi-source models. As for multilingual word embeddings, we extend the bilingual word mapping approach (Artetxe et al., 2016) to be able to deal with multilingual data. We have only used this approach based on multilingual word embeddings for two different language groups in this experiment: (*i*) for resource-poor languages for which less than 30 sentences were provided for training such as *surprise languages* and Kazakh, and (*ii*) for another group of 7 resource-rich languages that are all Indo-European languages. This is to show that even the analysis of resource-rich languages can be improved thanks to a multilingual approach.

[1] http://universaldependencies.org/conll17/

[2] http://universaldependencies.org/conll17/surprise.html

Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, pages 63–70,
Vancouver, Canada, August 3-4, 2017. © 2017 Association for Computational Linguistics

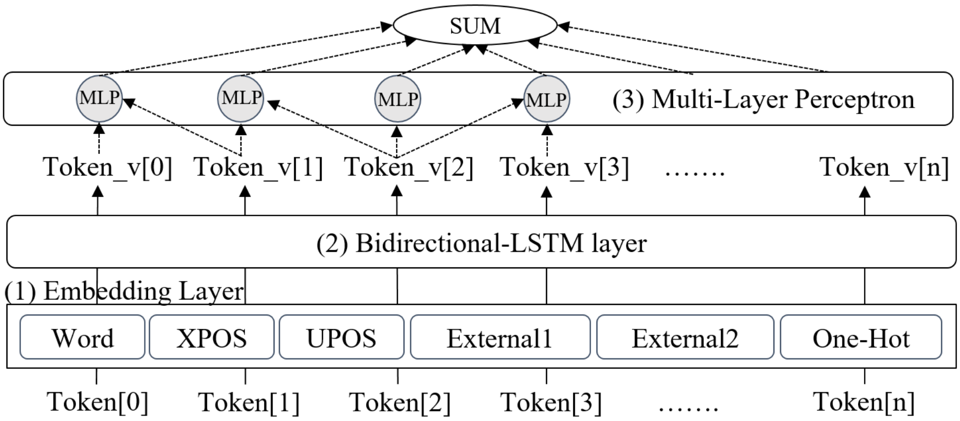

Figure 1: Overall system structure for training language models. **(1) Embedding Layer**: vectorized features that are feeding into Bidirectional LSTM. **(2) Bidirectional-LSTM**: train representation of each token as vector values based on bidirectional LSTM neural network. **(3) Multi-Layer Perceptron**: build candidate of parse trees based on trained(changed) features by bidirectional LSTM layer, and then calculate probabilistic scores for each of candidates. Finally, if it has multiple roots, revise it (section 3) or select the best parse tree.

Although we could theoretically train a single model for all the languages considered in the evaluation based on our multilingual approach, relevant results can only be obtained if one takes into account language similarities and typological information. Moreover, given the limited time and the specific resource environment designed for the shared task, it was hard to get better results using a multilingual approach than using a monolingual approach for resource-rich languages since training new word embeddings requires time. Thus, we processed 69 corpora with monolingual models, and only 13 corpora with our multilingual approach.

In what follows we describe the architecture of our system (section 2), our monolingual (section 3) as well as our multilingual approach (section 4). Finally, we compare the results with the baseline provided by UDPipe1.1 and with the results of other teams (section 5).

2 System Overview

Our system extends the Graph-based parser (Taskar et al., 2005) especially in BIST-parser that works by default with monolingual data. Basically the Graph-based BIST-parser uses bidirectional Long Short Term Memory (LSTM) feature representations thanks to two neural network layers (Kiperwasser and Goldberg, 2016). In order to select the best relation and head for each tokens in a sentence, Kiperwasser and Goldberg link the output of the bidirectional LSTM with the Multi-Layer Perceptron (MLP) thanks to one neural layer. Here we adopt the same feature representation and MLP but different training features and decision models.

In order to adapt the parser to a multilingual approach, we add new parameters and new features to the training algorithm, notably the possibility to use multilingual word embeddings and one hot encoding to encode languages. Finally, the parser can be trained on monolingual and multilingual data depending on the parameters chosen for training. An overview of the overall architecture is given in Figure 1. Details on word embeddings along with the number of dimensions considered are given below.

- ***Word:*** randomly generated word embedding (100)
- **XPOS:** language-specific POS (25)
- **UPOS:** universal POS (25)
- **External embedding1:** pretrained word embedding (100)
- **External embedding2:** pretrained word embedding that is replaced with *Word* (100)
- **one-hot encoding:** one-hot encoding of the language ID (65)

Corpus	Embedding model	Bilingual Dic	Training corpora
Buryat	Buryat-Russian	wiki dump	brx(20), ru
Kurmanji	Kurmanji-English	wiki dump	kmr(20), en
North Sámi	North Sámi-Finnish	wiki dump	sme(20), fi, fi-fbt
Upper Sorbian	Upper Sorbian-Polish	OPUS	hsb(20), pl
Kazakh	Kazakh-Turkish	OPUS	kk(20), tr
Portuguese	7 languages*	Europarl7,WMT11	en, it, fr, es, pt, de, sv
Italian	7 languages*	Europarl7,WMT11	en, it, fr, es, pt, de, sv
Italian_ParTUT	7 languages*	Europarl7,WMT11	en, en_partut, fr_partut, it, it_partut
English_ParTUT	7 languages*	Europarl7,WMT11	en, en_partut, fr_partut, it_partut
French_ParTUT	7 languages*	Europarl7,WMT11	en_partut, fr(2), fr_partut, it_partut
Czech-CLTT	Czech	-	cs, cs_cac, cs_cltt
Galician-TreeGal	Galician	-	ga, ga_treegal
Slovenian-SST	Slovenian	-	sl, sl_sst

Table 1: Languages trained by a multilingual model. **Embedding model:** applied languages that were used for making multilingual word embeddings. **Bilingual Dic:** resources to generate bilingual dictionaries **Training corpora:** Training corpora that were used. **7 languages:** English, Italian, French, Spanish, Portuguese, German, Swedish. **(number):** the number of multiplication to expand the total amount of corpus.

Word refers to words taken from the training corpora and used as lexical features with vectorized embeddings in our parser. Both *XPOS* and *UPOS*[3] are used as delexicalized features. The content of *Word* and *POS* is set randomly when the training phase starts. In addition, two external word embeddings are added to the representations of words, one is concatenated with the *Word* vector additionally, and the other is used to replace the *Word* vector. For example, let *Word* be generated randomly with 100 dimensional vector values and *External1* and let *External2* be pretrained word embeddings made from different resources with 100 dimensional vector values. If we just add *External1* to an additional word embedding, then final word embedding could be *Word+External1* (200 dimensions) based on concatenation. However, if we add just *External2* as an additional word embedding, *Word* is deleted because it is replaced with *External2* so that final word embedding could be *External2* (100 dimensions). If both are used, final word embedding could be *External1* + *External2* (*Word* is deleted because of *External2*). Since we have found if we can use two external word embeddings, replacing one word embedding as the *Word* made better results than concatenating two word embeddings based on experiments.

Since our goal is to develop a multilingual parsing model, we took the idea of one-hot encodings from (Ammar et al., 2016a). The idea is to add language one-hot encoding as an additional feature while training multilingual models. It allows the model to directly focus on language specificities. There are 65 hot-encoding dimensions because there are 64 languages in UD 2.0 (Nivre et al., 2017) plus *unknown* languages.

3 Monolingual Model

There were 81 different corpora to be parsed within the CoNLL 2017 shared task. We used a monolingual approach for 69 corpora, and our results are detailed in section 5. As mentioned above, training a monolingual model in our system is very similar to training a BIST-parser model. However, we made two modifications to the original approach.

Multiple roots: The BIST-parser can generate multiple roots for a given sentence. This is not a problem in general but for the shared task we need to provide only one single root per sentence. Not detecting the right root for a sentence leads to major errors so the problem had to be addressed carefully. We chose to develop a simple algorithm: when the parser returns multiple roots, our system revises the overall sentence analysis so as to select one single primary root and change other previous roots as links pointing to the new head node. Choosing the primary root is the result of an

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

empirical process depending on the language considered (i.e. taking into account language-specific word order). For example, the primary root is the first head node in the case of an English sentence and the last one in the case of a Korean sentence. This very simple trick improved the LAS scores by 0.43 overall F1-measure on the development set.

Customizing for UD: Basically, the BIST-parser is not adapted to the Universal Dependency format. Thus several changes had to be made. First, we added both *XPOS* and *UPOS* categories to the parser. Second, if a word in a training sentence did not exist in external word embeddings, we replaced the word as a lemma of the word. Third, we used the external word embeddings provided by the shared task organizers[4] and concatenated them with the original *Word* embedding.

4 Multilingual Model

We processed 13 test corpora with our multilingual model. The F1-measure of our system for these corpora are better than with our monolingual system for resource-poor languages and even for most of the resource-rich languages.

4.1 Surprise Languages and Kazakh

There were four surprise languages provided for evaluation within the CONLL 2017 shared task: Buryat, Kurmanji, North Sámi and Upper Sorbian (all in the Universal Dependency format). Less than 30 sentences were provided for training, and Kazakh also had 30 sentences for training. We divided the training corpus in half: half od the data were set apart for development and never used for training.

Word embeddings. The first step for training multilingual model is finding topologically similar languages. Thus, we selected three languages for each surprise language in order to be able to derive multilingual word embeddings. The choice of languages was based on the *Word Atlas of Language Structures*[5] and on advices from linguists.

Bilingual Dictionary. There has been many attempts to build multilingual embeddings (Ammar et al., 2016b; Smith et al., 2017). One simple but powerful method is finding a linear transformation matrix from two monolingual embeddings. (Artetxe et al., 2016) propose to do this with pretrained word embeddings and bilingual dictionaries. We tried to follow their approach using monolingual embeddings provided by Facebook research[6] and building bilingual dictionaries. Unfortunately there were not many resources (even with a limited coverage) for building a bilingual dictionary in the case of surprise languages.

For some languages we were able to find bilingual dictionaries from OPUS[7]. When no corpus was available, we used Swadesh lists from Wikipedia dumps. Swadesh lists are composed of 207 bilingual words that are supposed to be *"basic concepts for the purposes of historical-comparative linguistics"*[8]. Finally, we transformed both embeddings in a single vector space.

Table1 shows details about the selected pairs of languages and the different sources used for our dictionaries. From these resources, we trained a multilingual model and after testing with the development set set apart for each pair of candidate languages, we picked up the best candidate for the different surprise languages and for Kazakh.

4.2 Italian and Portuguese

There have been several attempts aiming at training multilingual models for resource-rich languages (Guo et al., 2016; Ammar et al., 2016a). We have tested our multilingual system in a similar way as explained in the previous section for resource-rich languages, except that we of course changed the resources used. We used the multilingual word embeddings for 7 languages presented in Ammar et al.'s paper (average and brown cluster model), and then trained a multilingual model with the training set provided for the 7 languages considered. Although the size of word vectors for multilingual embeddings is almost 10 times smaller than with the monolingual embeddings made by Facebook research, the result (F1-measure) is slightly better than with the monolingual model for Italian and Portuguese, with 0.39 and 0.41 within development sets.

4.3 ParTUT corpora

We assumed that all ParTUT corpora are related with each other. Thus, we used a similar multilingual approach to parse the ParTUT corpora but we used different compositions of corpora for

[4] https://lindat.mff.cuni.cz/repository/xmlui/handle/11234/1-1989

[5] The Word Atlas of Language Structures provides information about different languages in the world (family, latitude and longitude, see http://wals.info).

[6] https://github.com/facebookresearch/fastText/

[7] http://opus.lingfil.uu.se/

[8] https://en.wikipedia.org/wiki/Swadesh_list

Corpus	LAS	UAS	Rank(LAS)	Rank(UAS)	Baseline(LAS)
Overall (81)	70.93	76.75	5	5	68.35
Big treebanks only (55)	75.79	80.55	5	5	73.04
PUD treebanks only (14)	70.77	77.64	4	4	68.33
Small treebanks only (8)	54.78	64.99	4	5	51.80
Surprise language only (4)	36.93	48.66	12	15	37.07
English_PUD	82.38	85.77	2	2	78.95
Russian_PUD	72.03	79.31	2	2	68.31
Spanish	85.22	88.40	2	3	81.47

Table 2: Official experiment results with rank. (number): number of corpora

Corpus	LATTICE	Baseline
Arabic_PUD	47.13	43.14
Arabic	68.54	65.3
Bulgarian	85.6	83.64
Catalan	86.83	85.39
Czech-CAC	84.77	82.46
Czech_PUD	80.86	79.8
Czech	83.68	82.87
Old_Church_Slavonic	60.81	62.76
Danish	76.47	73.38
Danish_PUD	71.45	66.53
German	75.09	69.11
Greek	81.13	79.26
English-LinES	76.17	72.94
English_PUD	82.38	78.95
English	78.91	75.84
Spanish-AnCora	86.87	83.78
Spanish_PUD	79.87	77.65
Spanish	85.22	81.47
Estonian	62.93	58.79
Basque	72.13	69.15
Persian	82.63	79.24
Finnish-FTB	79.44	74.03
Finnish_PUD	80.82	78.65
Finnish	77.11	73.75
French_PUD	76.55	73.63
French-Sequoia	83.7	79.98
French	82.83	80.75
Irish	64.39	61.52
Galician	80.68	77.31
Gothic	60.55	59.81
Ancient_Greek-PROIEL	60.58	65.22
Ancient_Greek	51.5	56.04
Hebrew	61.24	57.23
Hindi_PUD	50.94	50.85
Hindi	86.99	86.77
Croatian	80.96	77.18
Hungarian	68.49	64.3
Indonesian	76.6	74.61
Italian_PUD	86.49	83.7
Japanese_PUD	77.41	76.28
Japanese	73.98	72.21
Korean	72.35	59.09
Latin-ITTB	74.33	76.98
Latin-PROIEL	55.04	57.54
Latin	51.95	43.77
Latvian	64.49	59.95
Dutch-LassySmall	75.67	78.15
Dutch	70.6	68.9
Norwegian-Bokmaal	85.55	83.27
Norwegian-Nynorsk	84.57	81.56
Polish	85.94	78.78
Portuguese-BR	88.56	85.36
Portuguese_PUD	76.56	73.96
Romanian	81.93	79.88
Russian_PUD	72.03	68.31
Russian-SynTagRus	87.9	86.76
Russian	78.42	74.03
Slovak	79.23	72.75
Slovenian	84.52	81.15
Swedish-LinES	78.15	74.29
Swedish_PUD	73.4	70.62
Swedish	81.07	76.73
Turkish_PUD	34.82	34.53
Turkish	58.89	53.19
Uyghur	34.94	34.18
Ukrainian	63.63	60.76
Urdu	79.26	76.69
Vietnamese	39.87	37.47
Chinese	61.94	57.4
Average	**73.13**	**70.45**

Table 3: Official experiment results processed by monolingual models.

Corpus	LATTICE-Multi	LATTICE-Mono	Baseline
Buryat	27.08	19.7	31.5
Kurmanji	41.71	37.59	32.35
North Sámi	28.39	25.89	30.6
Upper Sorbian	50.54	41.23	53.83
Kazakh	22.11	19.98	24.51
Italian	87.75	87.98	85.28
Portuguese	84.08	84.08	82.11
English-ParTUT	80.45	77.62	73.64
French-ParTUT	83.26	80.66	77.38
Italian-ParTUT	84.09	80.36	-
Czech-CLTT	75.45	74.85	71.64
Galician-TreeGal	68.01	67.75	65.82
Slovenian-SST	49.94	48.06	46.45

Table 4: Official experiment results processed by multilingual models.

training, such as French_ParTUT with en_partut, fr_partut, it_partut and doubled-fr corpus. Finally, the best training compositions are listed in Table1.

4.4 Czech-CLTT, Galician-TreeGal, Slovenian-SST

These three corpora have a small number of training sentences. We thus chose to train them together but with different language hot-encoding values.

5 Experimental Results

Because we wanted to focus on the dependency parsing task, we used automatically annotated corpora for testing and also trained all models with the annotated corpora provided by UDPipe (Straka et al., 2016).

As described in section 4, we used different word embeddings and training corpora for multilingual models. As for monolingual models, we simply trained the system with monolingual embeddings (see details in section 3).

Overall results. Table 2, 3 and 4 show the official results (except for it_ParTUT), using the F1-measure computed by the TIRA platform (Potthast et al., 2014) for the CoNLL 2017 Shared task[9]. Our system achieved 70.93 F1 (LAS) on the overall 81 test sets and ranked 5^{th} out of 33 teams. The average gap between the baseline obtained with UDPipe1.1 (Straka et al., 2016) and our system is 2.58 LAS in our favor. Our system shows better results in avoiding over-fitting issues. Performance gaps are narrowed when considering only PUD test sets (for example, our system ranked second best for processing English_PUD and Russian_PUD), which is encouraging for practical applications.

Multilingual model. Table 4 shows the results obtained when using the multilingual models on the small treebank dataset (fr_partut, ga, gl_treegal, kk, la, sl_sst, ug, uk). We ranked 4^{th}, with 54.78 LAS score on this group of languages. However, in terms of extremely resource-poor languages (surprise languages), we have ranked only 12^{th}, with 36.93 LAS score. This is slightly lower than the UDPipe1.1 baseline model: we assume this is the result of using half of the corpus for training surprise languages (section 4). If we compare monolingual models of surprise languages with multilingual ones, we see an improvement between 2.5 and 9.31 percent. The same kind of improvement can be observed for the ParTUT group. In this case, the multilingual approach improves performance by almost 3 points.

6 Conclusion

In this paper, we have described our system for multilingual dependency parsing that has been tested over the 81 Universal Dependency corpora provided for the CoNLL 2017 shared task. Our parser mainly extends the monolingual BIST-parser as a multi-source trainable parser. We proposed three main contributions: (*1*) the integration of multilingual word embeddings and one hot encodings for the different languages, which means our system can work using monolingual models as well as on multilingual ones. (*2*) a simple but

[9] http://universaldependencies.org/conll17/results.html

effective way to solve the multiple roots problem of the original BIST parser and (*3*) an original approach for the elaboration of multilingual dictionaries for resource-poor languages and the projection of monolingual word embeddings in a single vector space. Our system ranked 5^{th} and achieved 70.93 overall F1-measure over the 81 test corpora provided for evaluation. We are confident there is room for improvement since this system is only preliminary and lots of components could be optimized. A better account of language typology could also help the process and show the benefit of linguistic knowledge in this kind of environment.

7 Acknowledgements

Kyung Tae Lim is supported by the ANR ERA-NET ATLANTIS project. This work is also supported by the LAKME project through a grant from Paris Sciences et Lettres within the framework of the IDEX (Initiatives d'Excellence) PSL reference ANR-10-IDEX-0001-02.

References

Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A. Smith. 2016a. One parser, many languages. *CoRR* abs/1602.01595. http://arxiv.org/abs/1602.01595.

Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A Smith. 2016b. Massively multilingual word embeddings. *arXiv preprint arXiv:1602.01925* .

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Jiang Guo, Wanxiang Che, David Yarowsky, Haifeng Wang, and Ting Liu. 2015. Cross-lingual dependency parsing based on distributed representations. In *Proceedings of ACL*.

Jiang Guo, Wanxiang Che, David Yarowsky, Haifeng Wang, and Ting Liu. 2016. A representation learning framework for multi-source transfer parsing. In *AAAI*. pages 2734–2740.

Ulrich Heid and Sybille Raab. 1989. Collocations in multilingual generation. In *Proceedings of the fourth conference on European chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 130–136.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional LSTM feature representations. *TACL* 4:313–327. https://transacl.org/ojs/index.php/tacl/article/view/885.

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)*. European Language Resources Association, Portorož, Slovenia, pages 1659–1666.

Joakim Nivre et al. 2017. Universal Dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, Prague, `http://hdl.handle.net/11234/1-1983`. http://hdl.handle.net/11234/1-1983.

Martin Potthast, Tim Gollub, Francisco Rangel, Paolo Rosso, Efstathios Stamatatos, and Benno Stein. 2014. Improving the reproducibility of PAN's shared tasks: Plagiarism detection, author identification, and author profiling. In Evangelos Kanoulas, Mihai Lupu, Paul Clough, Mark Sanderson, Mark Hall, Allan Hanbury, and Elaine Toms, editors, *Information Access Evaluation meets Multilinguality, Multimodality, and Visualization. 5th International Conference of the CLEF Initiative (CLEF 14)*. Springer, Berlin Heidelberg New York, pages 268–299. https://doi.org/10.1007/978-3-319-11382-1_22.

Samuel L. Smith, David H. P. Turban, Steven Hamblin, and Nils Y. Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *CoRR* abs/1702.03859. http://arxiv.org/abs/1702.03859.

Milan Straka, Jan Hajič, and Jana Straková. 2016. UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association, Portorož, Slovenia.

Ben Taskar, Vassil Chatalbashev, Daphne Koller, and Carlos Guestrin. 2005. Learning structured prediction models: A large margin approach. In *Proceedings of the 22nd international conference on Machine learning*. ACM, pages 896–903.

Jörg Tiedemann. 2015. Cross-lingual dependency parsing with universal dependencies and predicted pos labels. *Depling 2015* page 340.

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.