Massive Choice, Ample Tasks (MaChAmp): A Toolkit for Multi-task Learning in NLP

Rob van der Goot **Ahmet Üstün** **Alan Ramponi** **Ibrahim Sharaf** **Barbara Plank**

IT University of Copenhagen University of Groningen University of Trento
Fondazione the Microsoft Research - University of Trento COSBI Factmata

robv@itu.dk, a.ustun@rug.nl, alan.ramponi@unitn.it
ibrahim.sharaf@factmata.com, bapl@itu.dk

Abstract

Transfer learning, particularly approaches that combine multi-task learning with pre-trained contextualized embeddings and fine-tuning, have advanced the field of Natural Language Processing tremendously in recent years. In this paper we present MaChAmp, a toolkit for easy fine-tuning of contextualized embeddings in multi-task settings. The benefits of MaChAmp are its flexible configuration options, and the support of a variety of natural language processing tasks in a uniform toolkit, from text classification and sequence labeling to dependency parsing, masked language modeling, and text generation.[1]

1 Introduction

Multi-task learning (MTL) (Caruana, 1993, 1997) has developed into a standard repertoire in natural language processing (NLP). It enables neural networks to learn tasks in parallel (Caruana, 1993) while leveraging the benefits of sharing parameters. The shift—or "tsunami" (Manning, 2015)—of deep learning in NLP has facilitated the wide-spread use of MTL since the seminal work by Collobert et al. (2011), which has led to a multi-task learning "wave" (Ruder and Plank, 2018) in NLP. It has since been applied to a wide range of NLP tasks, developing into a viable alternative to classical pipeline approaches. This includes early adoption in Recurrent Neural Network models, e.g. (Lazaridou et al., 2015; Chrupała et al., 2015; Plank et al., 2016; Søgaard and Goldberg, 2016; Hashimoto et al., 2017), to the use of large pre-trained language models with multi-task objectives (Radford et al., 2019; Devlin et al., 2019). MTL comes in many flavors, based on the type of sharing, the weighting of losses, and the design and relations of tasks and layers. In general though, outperforming single-task settings remains a challenge (Martínez Alonso and Plank, 2017; Clark et al., 2019). For an overview of MTL in NLP we refer to Ruder (2017).

As a separate line of research, the idea of language model pre-training and contextual embeddings (Howard and Ruder, 2018; Peters et al., 2018; Devlin et al., 2019) is to pre-train rich representation on large quantities of monolingual or multilingual text data. Taking these representations as a starting point has led to enormous improvements across a wide variety of NLP problems. Related to MTL, recent research effort focuses on fine-tuning contextualized embeddings on a variety of tasks with supervised objectives (Kondratyuk and Straka, 2019; Sanh et al., 2019; Hu et al., 2020).

We introduce MaChAmp, a flexible toolkit for multi-task learning and fine-tuning of NLP problems. The main advantages of MaChAmp are:

- Ease of configuration, especially for dealing with multiple datasets and multi-task setups;
- Support of a wide range of NLP tasks, including a variety of sequence labeling approaches, text classification, dependency parsing, masked language modeling, and text generation (e.g., machine translation);
- Support of the initialization and fine-tuning of any contextualized embeddings from Hugging Face (Wolf et al., 2020).

As a result, the flexibility of MaChAmp supports up-to-date, general-purpose NLP (see Section 2.2). The backbone of MaChAmp is AllenNLP (Gardner et al., 2018), a PyTorch-based (Paszke et al., 2019) Python library containing modules for a variety of deep learning methods and NLP tasks. It is designed to be modular, high-

[1] The code is available at: https://github.com/machamp-nlp/machamp (v0.2), and an instructional video at https://www.youtube.com/watch?v=DauTEdMhUDI.

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, pages 176–197
April 19 - 23, 2021.

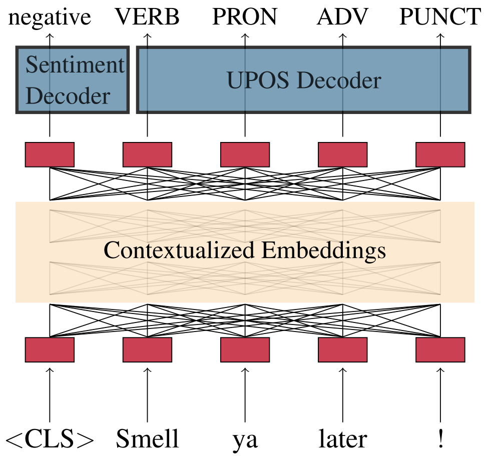

Figure 1: Overview of MACHAMP, when training jointly for sentiment analysis and POS tagging. A shared encoding representation and task-specific decoders are exploited to accomplish both tasks.

level and flexible. It should be noted that contemporary to MACHAMP, `jiant` (Pruksachatkun et al., 2020) was developed, and AllenNLP included multi-task learning as well since release 2.0. MACHAMP distinguishes from the other toolkits by supporting simple configurations, and a variety of multi-task settings.

2 Model

In this section we will discuss the model, its supported tasks, and possible configuration settings.

2.1 Model overview

An overview of the model is shown in Figure 1. MACHAMP takes a pre-trained contextualized model as initial encoder, and fine-tunes its layers by applying an inverse square root learning rate decay with linear warm-up (Howard and Ruder, 2018), according to a given set of downstream tasks. For the task-specific predictions, each task has its own decoder, which is trained for the corresponding task. The model defaults to the embedding-specific tokenizer in Hugging Face (Wolf et al., 2020).[2]

When multiple datasets are used for training, they are first separately split into batches so that each batch only contains instances from one dataset. Batches are then concatenated and shuffled before training. This means that small datasets will be underrepresented, which can be overcome by smoothing the dataset sampling (Section 3.2.2). During decoding, the loss function is only activated for tasks which are present in the current batch. By default, all tasks have an equal weight in the loss function. The loss weight can be tuned (Section 3.2.1).

2.2 Supported task types

We here describe the tasks MACHAMP supports.

SEQ For traditional token-level sequence prediction tasks, like part-of-speech tagging. MACHAMP uses greedy decoding with a softmax output layer on the output of the contextual embeddings.

STRING2STRING An extension to SEQ, which learns a conversion for each input token to its label. Instead of predicting the labels directly, the model can now learn to predict the conversion. This strategy is commonly used for lemmatization (Chrupała, 2006; Kondratyuk and Straka, 2019), where it greatly reduces the label vocabulary. We use the transformation algorithm from UDPipe-Future (Straka, 2018), which was also used by Kondratyuk and Straka (2019).

SEQ_BIO A variant of SEQ which exploits conditional random fields (Lafferty et al., 2001) as decoder, masked to enforce outputs following the BIO tagging scheme.

MULTISEQ An extension to SEQ which supports the prediction of multiple labels per token. Specifically, for some sequence labeling tasks it is unknown beforehand how many labels each token should get. We compute a probability score for each label, employing binary cross-entropy as loss, and outputting all the labels that exceed a certain threshold. The threshold can be set in the dataset configuration file.

DEPENDENCY For dependency parsing, MACHAMP uses the deep biaffine parser (Dozat and Manning, 2017) as implemented by AllenNLP (Gardner et al., 2018), with the Chu-Liu/Edmonds algorithm (Chu, 1965; Edmonds, 1967) for decoding the tree.

MLM For masked language modeling, our implementation follows the original BERT settings (Devlin et al., 2019). The chance that a token is masked is 15%, of which 80% are masked with a `[MASK]` token, 10% with a random token, and 10% are left unchanged. We do not include the next sentence prediction task following Liu et al. (2019), for simplicity and efficiency. We use a cross entropy loss,

[2] This includes both the pre-tokenization (in the traditional sense) and the subword segmentation.

```
smell	VERB
ya	PRON
later	ADV
!	PUNCT
```

(a) Example of a token-level file format (e.g., for POS tagging), where words are in column `word_idx`=0, and a single layer of corresponding annotations is in column `column_idx`=1.

```
smell ya later !	negative
```

(b) Example of a sentence-level file format (e.g., for sentiment classification), where only a sentence is required and is defined in column 0 (i.e., `sent_idxs`=[0]) and a single layer of annotation is in the second column (`column_idx`=1).

Figure 2: Examples of data file formats.

and the language model heads from the defined Hugging Face embeddings (Wolf et al., 2020). It assumes raw text files as input, so no `column_idx` has to be defined (See Section 3.1).

CLASSIFICATION For text classification, it predicts a label for every text instance by using the embedding of the first token, which is commonly a special token (e.g. `[CLS]` or <s>). For tasks which model a relation between multiple sentences (e.g., textual entailment), a special token (e.g. `[SEP]`) is automatically inserted between the sentences to inform the model about the sentence boundaries.

SEQ2SEQ For text generation, MACHAMP employs the sequence to sequence (encoder-decoder) paradigm (Sutskever et al., 2014). We use a recurrent neural network decoder, which suits the auto-regressive nature of the machine translation tasks (Cho et al., 2014) and an attention mechanism to avoid compressing the whole source sentence into a fixed-length vector (Bahdanau et al., 2015).

3 Usage

To use MACHAMP, one needs a configuration file, input data and a command to start the training or prediction. In this section we will describe each of these requirements.

3.1 Data format

MACHAMP supports two types of data formats for annotated data,[3] which correspond to the level of annotation (Section 2.2). For token-level tasks, we will use the term "token-level file format", whereas for sentence-level task, we will use "sentence-level file format".

The token-level file format is similar to the tab-separated CoNLL format (Tjong Kim Sang and De Meulder, 2003). It assumes one token per line (on a column index `word_idx`), with each annotation layer following each token separated by a tab character (each on a column index `column_idx`) (Figure 2a). Token sequences (e.g., sentences) are delimited by an empty line. Comments are lines on top of the sequence (which have a different number of columns with respect to "token lines").[4] It should be noted that for dependency parsing, the format assumes the relation label to be on the `column_idx` and the head index on the following column. Further, we also support the UD format by removing multi-word tokens and empty nodes using the UD-conversion-tools (Agić et al., 2016).

The sentence-level file format (used for text classification and text generation) is similar (Figure 2b), and also supports multiple inputs having the same annotation layers. A list of one or more column indices can be defined (i.e., `sent_idxs`) to enable modeling the relation between any arbitrary number of sentences.

3.2 Configuration

The model requires two configuration files, one that specifies the datasets and tasks, and one for the hyperparameters. For the hyperparameters, a default option is provided (`configs/params.json`, see Section 4).

3.2.1 Dataset configuration

An example of a dataset configuration file is shown in Figure 3. On the first level, the dataset names are specified (i.e., "UD" and "RTE"), which should be unique identifiers. Each of these datasets needs at least a `train_data_path`, a `validation_data_path`, a `word_idx` or `sent_idxs`, and a list of `tasks` (corresponding to the layers of annotation, see Section 3.1).

For each of the defined tasks, the user is required to define the `task_type` (Section 2.2), and the column index from which to read the relevant labels (i.e., `column_idx`). On top of this template, the following options can be passed on the task level:

[3] The MLM task does not require annotation, thus a raw text file can be provided.

[4] We do not identify comments based on lines starting with a '#', because datasets might have tokens that begin with '#'.

```
{ "UD": {
    "train_data_path": "data/ewt.train",
    "validation_data_path": "data/ewt.dev",
    "word_idx": 1,
    "tasks": {
      "lemma": {
        "task_type": "string2string",
        "column_idx": 2
      },
      "upos": {
        "task_type": "seq",
        "column_idx": 3
  } } }
  "RTE": {
    "train_data_path": "data/RTE.train",
    "validation_data_path": "data/RTE.dev",
    "sent_idxs": [0,1],
    "tasks": {
      "rte": {
        "task_type": "classification",
        "column_idx": 2
} } } }
```

Figure 3: Example dataset configuration file to predict UPOS, lemmas, and textual entailment simultaneously.

Metric For each task type, a commonly used metric is set as default metric. However, one can override the default by specifying a different metric at the task level. Supported metrics are 'acc', 'las', 'micro-f1', 'macro-f1', 'span_f1', 'multi_span_f1', 'bleu' and 'perplexity'.

Loss weight In multi-task settings, not all tasks might be equally important, or some tasks might just be harder to learn, and therefore should gain more weight during training. This can be tuned by setting the `loss_weight` parameter on the task level (by default the value is 1.0 for all tasks).

Dataset embedding Ammar et al. (2016) have shown that embedding which language an instance belongs to can be beneficial for multilingual models. Later work (Stymne et al., 2018; Wagner et al., 2020) has also shown that more fine-grained distinctions on the dataset level[5] can be beneficial when training on multiple datasets within the same language (family). In previous work, this embedding is usually concatenated to the word embedding before the encoding. However, in contextualized embeddings, the word embeddings themselves are commonly used as encoder, hence we concatenate the dataset embeddings in between the encoder and the decoder. This parameter is set on the dataset level with `dataset_embed_idx`, which specifies the column to read the dataset ID from. Setting `dataset_embed_idx` to -1 will use the dataset name as specified in the json file as ID.

[5]These are called treebank embeddings in their work. We will use the more general term "dataset embeddings", which would often roughly correspond to languages and/or domains/genres.

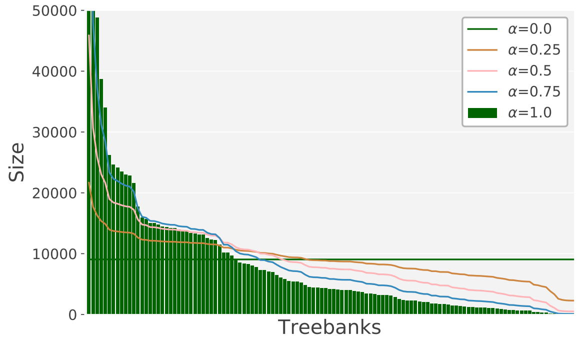

Figure 4: Effect of the sampling parameter α on the training sets of Universal Dependencies 2.6 data.

Max sentences In order to limit the maximum number of sentences that are used during training, `max_sents` is used. This is done before the sampling smoothing (Section 3.2.2), if both are enabled. It should be noted that the specified number will be taken from the top of the dataset.

3.2.2 Hyperparameter configuration

Whereas most of the hyperparameters can simply be changed from the default configuration provided in `configs/params.json`, we would like to highlight two main settings.

Pre-trained embeddings The name/path to pre-trained Hugging Face embeddings[6] can be set in the configuration file at the `transformer_model` key; `transformer_dim` might be adapted accordingly to reflect the embeddings dimension.

Dataset sampling To avoid larger datasets from overwhelming the model, MaChAmp can resample multiple datasets according to a multinomial distribution, similar as used by Conneau and Lample (2019). MaChAmp performs the sampling on the batch level, and shuffles after each epoch (so it can see a larger variety of instances for downsampled datasets). The formula is:

$$\lambda = \frac{1}{p_i} * \frac{p_i^\alpha}{\sum_i p_i^\alpha} \quad (1)$$

where p_i is the probability that a random sample is from dataset i, and α is a hyperparameter that can be set. Setting α=1.0 means using the default sizes,

[6]https://huggingface.co/models

Parameter	Value	Range
Optimizer	Adam	
β_1, β_2	0.9, 0.99	
Dropout	0.2	0.1, 0.2, 0.3
Epochs	20	
Batch size	32	
Learning rate (LR)	1e-4	1e-3, 1e-4, 1e-5
LR scheduler	slanted triangular	
Weight decay	0.01	
Decay factor	0.38	.35, .38, .5
Cut fraction	0.2	.1, .2, .3

Table 1: Final parameter settings, incl. tested ranges.

and α=0.0 results in one average amount of batches for each dataset, similar to Sanh et al. (2019). The effect of different settings of α for the Universal Dependencies 2.6 data is shown in Figure 4. Smoothing can be enabled in the hyperparameters configuration file at the `sampling_smoothing` key.

3.3 Training

Given the setup illustrated in the previous sections, a model can be trained via the following command. It assumes the configuration (Figure 3) is saved in `configs/upos-lemma-rte.json`.

```
python3 train.py --dataset_config \
 configs/upos-lemma-rte.json
```

By default, the model and the logs will be written to `logs/<JSONNAME>/<DATE>`. The name of the directory can be set manually by providing `--name <NAME>`. Further, `--device <ID>` can be used to specify which GPU to use, otherwise the CPU will be used. As a default, `train.py` uses `configs/params.json` for the hyperparameters, but this can be overridden by using `--parameters_config <CONFIG FILE>`.

3.4 Inference

Prediction can be done with:

```
python3 predict.py \
 logs/<NAME>/<DATE>/model.tar.gz \
 <INPUT FILE> <OUTPUT FILE>
```

It requires the path to the best model (serialized during training) stored as `model.tar.gz` in the logs directory as specified above. By default, the data is assumed to be in the same format as the training data (i.e., with the same number of `column_idx` columns), but `--raw_text` can be specified to read a data file containing raw texts with one sentence per line. For models trained

Task	Reference	MaChAmp
EWT2.2	Kondratyuk et al. (2019)	
UPOS*	96.82	**97.07**
Lemma*	97.97	**98.14**
Feats*	97.27	**97.41**
LAS*	89.38	**89.80**
GLUE	Devlin et al. (2019)	
CoLA	**60.5**	53.7
MNLI	**86.7**	83.9
MNLI-mis	**85.9**	82.7
MRPC	**89.3**	87.2
QNLI	**92.7**	90.8
QQP	**72.1**	69.1
RTE	**70.1**	60.0
SST-2	**94.9**	92.5
WMT14	Liu et al. (2020)	
EN-DE	**30.1**	24.7
IWSLT15	Zaheer et al. (2018)	
EN-VI	**29.27**	24.72

Table 2: Scores of single task models on test data for three popular datasets and a variety of tasks. *one joint model. For the GLUE data, BERT-large (English) and tokenized BLEU are used for fair comparison.

on multiple datasets (as "UD" and "RTE" in Figure 3), `--dataset <NAME>` can be used to specify which dataset to use in order to predict all tasks within that dataset.

4 Hyperparameter Tuning

In this section we describe the procedure how we determined robust default parameters for MaChAmp; note that the goal is not to beat the state-of-the-art, but to reach competitive performance for multiple tasks simultaneously.[7]

For the tuning of hyperparameters, we used the GLUE classification datasets (Wang et al., 2018; Warstadt et al., 2019; Socher et al., 2013; Dolan and Brockett, 2005; Cer et al., 2017; Williams et al., 2018; Rajpurkar et al., 2018; Bentivogli et al., 2009; Levesque et al., 2012) and the English Web Treebank (EWT 2.6) (Silveira et al., 2014) with multilingual BERT[8] (mBERT) as embeddings.[9] For each of these setups, we averaged the scores over all datasets/tasks and perform a grid search. The best hyperparameters across all datasets are reported in Table 1 and are the defaults values for MaChAmp.

[7]Compared to MaChAmp v0.1 (van der Goot et al., 2020) we removed parameters with negligible effects (word dropout, layer dropout, adaptive softmax, and layer attention).

[8]`https://github.com/google-research/bert/blob/master/multilingual.md`

[9]We capped the dataset sizes to a maximum of 20,000 sentences for efficiency reasons.

Setup	UD (LAS)	GLUE (Acc)
Single	72.22	**82.38**
All	72.82	80.96
Smoothed	**73.74**	81.87
Dataset embed.*	72.76	–
Sep. decoder*	73.69	–

Table 3: Average results over all development sets. Dataset embeddings and a separate decoder have not been tested in GLUE, because each dataset is annotated for a different task. *includes dataset smoothing.

5 Evaluation

5.1 Single task evaluation

As a starting point, we evaluate single task models to ensure our implementations are competitive with the state-of-the-art. We report scores on dependency parsing (EWT), the GLUE classification tasks, and machine translation (WMT14 DE-EN (Bojar et al., 2014), IWSLT15 EN-VI (Cettolo et al., 2014)) using mBERT as our embeddings.[10] Table 2 reports our results on the test sets compared to previous work. For all UD tasks, we score slightly higher, whereas for GLUE tasks we score consistently lower compared to the references. This is mostly due to differences in fine-tuning strategies, as implementations themselves are highly similar. Scores on the machine translation tasks show the largest drops, indicating that task-specific fine-tuning and pre-processing might be necessary.

5.2 Multi-dataset evaluation

We evaluate the effect of a variety of multi-dataset settings on all GLUE and UD treebanks (v2.7) on the test splits. It should be noted that the UD treebanks all have the same tasks, as opposed to GLUE. First, we jointly train on all datasets (ALL), then we attempt to improve performance on smaller sets by enabling the sampling smoothing (SMOOTHED, Section 3.2.2, we set $\alpha = 0.5$). Furthermore, we attempt to improve the performance by informing the decoder of the dataset through dataset embeddings (DATASET EMBED., Section 3.2.1) or by giving each dataset its own decoder (SEP. DECODER). Results (Table 3) show that multi-task learning is only beneficial for performance when training on the same set of tasks (i.e., UD), dataset smoothing is helpful, dataset embeddings and separate decoders do not improve upon smoothing on average.

[10]For the sake of comparison we use BERT-large for GLUE, and EWT version 2.2.

Model\Size	0	<1k	<10k	>10k
Single	43.5	15.1	57.9	80.1
All	44.5	37.1	66.4	80.3
Smoothed	44.3	**45.4**	67.1	80.3
Dataset embed.*	43.9	36.5	**67.8**	**81.0**
Sep. decoder*	**45.1**	37.7	66.5	80.9

Table 4: Average LAS scores on test splits of UD treebanks grouped by training size (in number of sentences). *includes dataset smoothing.

For analysis purposes, we group the UD treebanks based on training size, and also evaluate UD treebanks which have no training split (zero-shot). For the zero-shot experiments, we select a proxy parser based on word overlap of the first 10 sentences of the target test data and the source training data.[11] Results on the UD data (Table 4) show that multi-task learning is mostly beneficial for medium-sized datasets (<1k and <10k). For these datasets, the combination of smoothing and dataset embeddings are the most promising settings. Perhaps surprisingly, the zero-shot datasets (<1k) have a higher LAS as compared to the small datasets and using a separate decoder based on the proxy treebank is the best setting; this is mainly because for many small datasets there is no other in-language training treebank. For the GLUE tasks (Table 5, Appendix), multi-task learning is only beneficial for the RTE data. This is to be expected, as the tasks are different in this setup, and training data is generally larger. Dataset smoothing here prevents the model from dropping too much in performance, as it outperforms ALL for 7 out of 9 tasks.

6 Conclusion

We introduced MACHAMP, a powerful toolkit for multi-task learning supporting a wide range of NLP tasks. We also provide initial experiments demonstrating the usefulness of some of its options. We learned that multi-task learning is mostly beneficial for setups in which multiple datasets are annotated for the same set of tasks, and that dataset embeddings can still be useful when employing contextualized embeddings. However, the current experiments are just scratching the surface of MACHAMP's capabilities, as a wide variety of tasks and multi-task settings is supported.

[11]Scores on individual sets and proxy treebanks can be found in the Appendix.

Acknowledgments

We would like to thank Anouck Braggaar, Max Müller-Eberstein and Kristian Nørgaard Jensen for testing development versions. Furthermore, we thank Rik van Noord for his participation in the video, and providing an early use-case for MaChAmp (van Noord et al., 2020). This research was supported by an Amazon Research Award, an STSM in the Multi3Generation COST action (CA18231), a visit supported by COSBI, grant 9063-00077B (Danmarks Frie Forskningsfond), and Nvidia corporation for sponsoring Titan GPUs. We thank the NLPL laboratory and the HPC team at ITU for the computational resources used in this work.

References

Anne Abeillé, Lionel Clément, and Alexandra Kinyon. 2000. Building a treebank for French. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*, Athens, Greece. European Language Resources Association (ELRA).

Noëmi Aepli and Simon Clematide. 2018. Parsing approaches for Swiss German. In *Proceedings of the 3rd Swiss Text Analytics Conference (SwissText), Winterthur, Switzerland*.

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*, 4:301–312.

Željko Agić and Nikola Ljubešić. 2015. Universal Dependencies for Croatian (that work for Serbian, too). In *The 5th Workshop on Balto-Slavic Natural Language Processing*, pages 1–8, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

Lars Ahrenberg. 2015. Converting an English-Swedish parallel treebank to Universal Dependencies. In *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, pages 10–19, Uppsala, Sweden. Uppsala University, Uppsala, Sweden.

Linda Alfieri and Fabio Tamburini. 2016. (almost) automatic conversion of the Venice Italian Treebank into the merged Italian Dependency Treebank format. In *CLiC-it/EVALITA*.

Ika Alfina, Indra Budi, and Heru Suhartanto. 2020. Tree rotations for dependency trees: Converting the head-directionality of noun phrases. *Journal of Computer Science*, 16(11):1585–1597.

Héctor Martínez Alonso and Daniel Zeman. 2016. Universal Dependencies for the AnCora treebanks. *Procesamiento del Lenguaje Natural*, 57:91–98.

Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A. Smith. 2016. Many languages, one parser. *Transactions of the Association for Computational Linguistics*, 4:431–444.

Angelina Aquino, Franz de Leon, and Mary Ann Bacolod. 2020. UD_Tagalog-Ugnayan. https://github.com/UniversalDependencies/UD_Tagalog-Ugnayan.

Carolina Coelho Aragon. 2018. Variações estilísticas e sociais no discurso dos falantes akuntsú. *Polifonia*, 25(38.1):90–103.

Maria Jesus Aranzabe, Aitziber Atutxa, Kepa Bengoetxea, Arantza Diaz de Ilarraza, Iakes Goenaga, Koldo Gojenola, and Larraitz Uria. 2015. Automatic conversion of the Basque dependency treebank to universal dependencies. In *Proceedings of the fourteenth international workshop on treebanks an linguistic theories (TLT14)*, pages 233–241.

Masayuki Asahara, Hiroshi Kanayama, Takaaki Tanaka, Yusuke Miyao, Sumire Uematsu, Shinsuke Mori, Yuji Matsumoto, Mai Omura, and Yugo Murawaki. 2018. Universal Dependencies version 2 for Japanese. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Elena Badmaeva and Francis M. Tyers. 2017. Dependency treebank for Buryat. In *Proceedings of the 15th International Workshop on Treebanks and Linguistic Theories (TLT15)*, pages 1–12.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings*, San Diego, CA, USA.

David Bamman and Gregory Crane. 2011. The ancient Greek and Latin dependency treebanks. In *Language technology for cultural heritage*, pages 79–98. Springer.

Verginica Barbu Mititelu, Radu Ion, Radu Simionescu, Elena Irimia, and Cenel-Augusto Perez. 2016. The Romanian treebank annotated according to Universal Dependencies. In *Proceedings of the tenth international conference on natural language processing (hrtal2016)*.

Colin Batchelor. 2019. Universal dependencies for Scottish Gaelic: syntax. In *Proceedings of the Celtic Language Technology Workshop*, pages 7–15, Dublin, Ireland. European Association for Machine Translation.

Shabnam Behzad and Amir Zeldes. 2020. A cross-genre ensemble approach to robust Reddit part of speech tagging. In *Proceedings of the 12th Web as Corpus Workshop*, pages 50–56, Marseille, France. European Language Resources Association.

Eduard Bejček, Eva Hajičová, Jan Hajič, Pavlína Jínová, Václava Kettnerová, Veronika Kolářová, Marie Mikulová, Jiří Mírovskỳ, Anna Nedoluzhko, Jarmila Panevová, et al. 2013. Prague dependency treebank 3.0.

Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The fifth PASCAL recognizing textual entailment challenge. In *Proceedings of the Second Text Analysis Conference, TAC 2009*, Gaithersburg, Maryland, USA.

Yevgeni Berzak, Jessica Kenney, Carolyn Spadine, Jing Xian Wang, Lucia Lam, Keiko Sophie Mori, Sebastian Garza, and Boris Katz. 2016. Universal Dependencies for learner English. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 737–746, Berlin, Germany. Association for Computational Linguistics.

Irshad Bhat, Riyaz A. Bhat, Manish Shrivastava, and Dipti Sharma. 2018. Universal Dependency parsing for Hindi-English code-switching. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 987–998, New Orleans, Louisiana. Association for Computational Linguistics.

Riyaz Ahmad Bhat, Rajesh Bhatt, Annahita Farudi, Prescott Klassen, Bhuvana Narasimhan, Martha Palmer, Owen Rambow, Dipti Misra Sharma, Ashwini Vaidya, Sri Ramagurumurthy Vishnu, et al. 2016. The hindi/urdu treebank project. In *Handbook of Linguistic Annotation*. Springer Press.

Agne Bielinskiene, Loic Boizou, and Jolanta Kovalevskaite. 2016. Lithuanian dependency treebank. In *Human Language Technologies–The Baltic Perspective: Proceedings of the Seventh International Conference Baltic HLT 2016*, volume 289, page 107. IOS Press.

Su Lin Blodgett, Johnny Wei, and Brendan O'Connor. 2018. Twitter Universal Dependency parsing for African-American and mainstream American English. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1415–1425, Melbourne, Australia. Association for Computational Linguistics.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA. Association for Computational Linguistics.

Guillaume Bonfante, Bruno Guillaume, and Guy Perrier. 2018. *Application of Graph Rewriting to Natural Language Processing*. Wiley Online Library.

Emanuel Borges Völker, Maximilian Wendt, Felix Hennig, and Arne Köhn. 2019. HDT-UD: A very large Universal Dependencies treebank for German. In *Proceedings of the Third Workshop on Universal Dependencies (UDW, SyntaxFest 2019)*, pages 46–57, Paris, France. Association for Computational Linguistics.

Cristina Bosco, Felice Dell'Orletta, Simonetta Montemagni, Manuela Sanguinetti, and Maria Simi. 2014. The EVALITA 2014 dependency parsing task. In *EVALITA 2014 Evaluation of NLP and Speech Tools for Italian*, pages 1–8. Pisa University Press.

Gosse Bouma and Gertjan van Noord. 2017. Increasing return on annotation investment: The automatic construction of a Universal Dependency treebank for Dutch. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 19–26, Gothenburg, Sweden. Association for Computational Linguistics.

Anouck Braggaar and Rob van der Goot. 2021. Challenges in annotating and parsing spoken, code-switched, Frisian-Dutch data. In *Proceedings of the Second Workshop on Domain Adaptation for Natural Language Processing*.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen-Schirra, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. TIGER: Linguistic interpretation of a german corpus. *Research on language and computation*, 2(4):597–620.

Bernard Caron, Marine Courtin, Kim Gerdes, and Sylvain Kahane. 2019. A surface-syntactic UD treebank for Naija. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 13–24, Paris, France. Association for Computational Linguistics.

Rich Caruana. 1993. Multitask learning: A knowledge-based source of inductive bias. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 41–48, Amherst, MA, USA.

Rich Caruana. 1997. Multitask learning. In *Learning to learn*, pages 95–133. Springer.

Flavio Massimiliano Cecchini, Marco Passarotti, Paola Marongiu, and Daniel Zeman. 2018. Challenges in converting the index Thomisticus treebank into Universal Dependencies. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 27–36, Brussels, Belgium. Association for Computational Linguistics.

Slavomír Čéplö. 2018. Constituent order in Maltese: A quantitative analysis.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Özlem Çetinoğlu and Çağrı Çöltekin. 2019. Challenges of annotating a code-switching treebank. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 82–90, Paris, France. Association for Computational Linguistics.

Mauro Cettolo, Niehues Jan, Stüker Sebastian, Luisa Bentivogli, Roldano Cattoni, and Marcello Federico. 2014. The IWSLT 2014 evaluation campaign. In *International Workshop on Spoken Language Translation*, Lake Tahoe, CA, USA.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Grzegorz Chrupała, Ákos Kádár, and Afra Alishahi. 2015. Learning language through pictures. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 112–118, Beijing, China. Association for Computational Linguistics.

Grzegorz Chrupała. 2006. Simple data-driven context-sensitive lemmatization. *SEPLN*, 37:121–127.

Yoeng-Jin Chu. 1965. On the shortest arborescence of a directed graph. *Scientia Sinica*, 14:1396–1400.

Jayeol Chun, Na-Rae Han, Jena D. Hwang, and Jinho D. Choi. 2018. Building Universal Dependency treebanks in Korean. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alessandra Teresa Cignarella, Cristina Bosco, and Paolo Rosso. 2019. Presenting TWITTIRÒ-UD: An Italian Twitter treebank in Universal Dependencies. In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*, pages 190–197, Paris, France. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D. Manning, and Quoc V. Le. 2019. BAM! born-again multi-task networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5931–5937, Florence, Italy. Association for Computational Linguistics.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Cagrı Cöltekin. 2015. A grammar-book treebank of Turkish. In *Proceedings of the 14th workshop on Treebanks and Linguistic Theories (TLT 14)*, pages 35–49.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7059–7069, Vancouver, Canada.

Sam Davidson, Dian Yu, and Zhou Yu. 2019. Dependency parsing for spoken dialog systems. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1513–1519, Hong Kong, China. Association for Computational Linguistics.

Mehmet Oguz Derin. 2020. UD_Old_Turkish-Tonqq. `https://github.com/UniversalDependencies/UD_Old_Turkish-Tonqq`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Cheikh Bamba Dione. 2019. Developing Universal Dependencies for Wolof. In *Proceedings of the Third Workshop on Universal Dependencies (UDW, SyntaxFest 2019)*, pages 12–23, Paris, France. Association for Computational Linguistics.

Peter Dirix, Liesbeth Augustinus, Daniel van Niekerk, and Frank Van Eynde. 2017. Universal Dependencies for Afrikaans. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 38–47, Gothenburg, Sweden. Association for Computational Linguistics.

Kaja Dobrovoljc, Tomaž Erjavec, and Simon Krek. 2017. The Universal Dependencies treebank for Slovenian. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*, pages

33–38, Valencia, Spain. Association for Computational Linguistics.

Kaja Dobrovoljc and Joakim Nivre. 2016. The Universal Dependencies treebank of spoken Slovenian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1566–1573, Portorož, Slovenia. European Language Resources Association (ELRA).

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, pages 9–16, Jeju Island, Korea.

Timothy Dozat and Christopher D Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of 5th International Conference on Learning Representations, ICLR 2017, Conference Track Proceedings*, Toulon, France.

Kira Droganova, Olga Lyashevskaya, and Daniel Zeman. 2018. Data conversion and consistency of monolingual corpora: Russian UD treebanks. In *Proceedings of the 17th international workshop on treebanks and linguistic theories (tlt 2018)*, 155, pages 53–66.

Puneet Dwivedi and Guha Easha. 2017. Universal Dependencies for Sanskrit. *International Journal of Advance Research, Ideas and Innovations in Technology*, 3(4).

Hanne Eckhoff, Kristin Bech, Gerlof Bouma, Kristine Eide, Dag Haug, Odd Einar Haugen, and Marius Jøhndal. 2018. The PROIEL treebank family: a standard for early attestations of Indo-European languages. *Language Resources and Evaluation*, 52(1):29–65.

Hanne Martine Eckhoff and Aleksandrs Berdičevskis. 2015. Linguistics vs. digital editions: The Tromsø Old Russian and OCS treebank. *Scripta & e-Scripta*, 14(15):9–25.

Jack Edmonds. 1967. Optimum branchings. *Journal of Research of the national Bureau of Standards B*, 71(4):233–240.

Marhaba Eli, Weinila Mushajiang, Tuergen Yibulayin, Kahaerjiang Abiderexiti, and Yan Liu. 2016. Universal dependencies for Uyghur. In *Proceedings of the Third International Workshop on Worldwide Language Service Infrastructure and Second Workshop on Open Infrastructures and Analysis Frameworks for Human Language Technologies (WLSI/OIAF4HLT2016)*, pages 44–50, Osaka, Japan. The COLING 2016 Organizing Committee.

Marília Fernanda Pereira de Freitas. 2017. A posse em apurinã: Descrição de construções atributivas e predicativas em comparação com outras línguas aruák. *Belém: Programa de Pós-Graduação em Letras, Universidade Federal do Pará (Tese de Doutorado)*.

Marcos Garcia. 2016. Universal dependencies guidelines for the Galician-TreeGal treebank. Technical report, Technical Report, LyS Group, Universidade da Coruna.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

Fabrìcio Ferraz Gerardi. 2020. UD_Tupinamba-TuDeT. `https://github.com/UniversalDependencies/UD_Tupinamba-TuDeT`.

Fabrìcio Ferraz Gerardi. 2021. The structure of Mundurukú.

Memduh Gökırmak and Francis M. Tyers. 2017. A dependency treebank for Kurmanji Kurdish. In *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017)*, pages 64–72, Pisa,Italy. Linköping University Electronic Press.

Xavier Gómez Guinovart. 2017. Recursos integrados da lingua galega para a investigación lingüística. *Gallaecia. Estudos de lingüística portuguesa e galega. Santiago de Compostela: Universidade de Santiago*, pages 1037–1048.

Rob van der Goot and Gertjan van Noord. 2018. Modeling input uncertainty in neural network dependency parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4984–4991, Brussels, Belgium. Association for Computational Linguistics.

Rob van der Goot, Ahmet Üstün, Alan Ramponi, and Barbara Plank. 2020. Massive choice, ample tasks (MaChAmp): A toolkit for multi-task learning in NLP. *arXiv preprint arXiv:2005.14672v2*.

Normunds Gruzitis, Lauma Pretkalnina, Baiba Saulite, Laura Rituma, Gunta Nespore-Berzkalne, Arturs Znotins, and Peteris Paikens. 2018. Creation of a balanced state-of-the-art multilayer corpus for NLU. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Bruno Guillaume, Marie-Catherine de Marneffe, and Guy Perrier. 2019. Conversion et améliorations de corpus du Français annotés en Universal Dependencies. *Traitement Automatique des Langues*, 60(2):71–95.

Jan Hajič, Otakar Smrž, Petr Zemánek, Petr Pajas, Jan Šnaidauf, Emanuel Beška, Jakub Krácmar, and Kamila Hassanová. 2009. Prague Arabic dependency treebank 1.0.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A joint many-task model: Growing a neural network for multiple NLP tasks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1923–1933, Copenhagen, Denmark. Association for Computational Linguistics.

Dag TT Haug and Marius Jøhndal. 2008. Creating a parallel treebank of the old Indo-European Bible translations. In *Proceedings of the second workshop on language technology for cultural heritage data (LaTeCH 2008)*, pages 27–34.

Johannes Heinecke and Francis M. Tyers. 2019. Development of a Universal Dependencies treebank for Welsh. In *Proceedings of the Celtic Language Technology Workshop*, pages 21–31, Dublin, Ireland. European Association for Machine Translation.

Oliver Hellwig, Salvatore Scarlata, Elia Ackermann, and Paul Widmer. 2020. The treebank of Vedic Sanskrit. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5137–5146, Marseille, France. European Language Resources Association.

Barbora Hladká, Jan Hajič, Jirka Hana, Jaroslava Hlaváčová, Jiří Mírovskỳ, and Jan Raab. 2008. The Czech academic corpus 2.0 guide. *The Prague Bulletin of Mathematical Linguistics*, 89(1):41–96.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 4411–4421.

Anton Karl Ingason, Eiríkur Rögnvaldsson, Einar Freyr Sigurosson, and Joel C. Wallenberg. 2020. The Faroese parsed historical corpus. CLARIN-IS, Stofnun Árna Magnússonar.

Olájídé Ishola and Daniel Zeman. 2020. Yorùbá dependency treebank (YTB). In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5178–5186, Marseille, France. European Language Resources Association.

Tomás Jelínek. 2017. FicTree: A manually annotated treebank of Czech fiction. In *ITAT*, pages 181–185.

Anders Johannsen, Héctor Martínez Alonso, and Barbara Plank. 2015. Universal dependencies for Danish. In *International Workshop on Treebanks and Linguistic Theories (TLT14)*, page 157.

Hildur Jónsdóttir and Anton Karl Ingason. 2020. Creating a parallel Icelandic dependency treebank from raw text to Universal Dependencies. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2924–2931, Marseille, France. European Language Resources Association.

Jenna Kanerva. 2020. UD_Finnish-OOD. `https://github.com/UniversalDependencies/UD_Finnish-OOD`.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing universal dependencies universally. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2779–2795, Hong Kong, China. Association for Computational Linguistics.

Kamil Kopacewicz. 2018. UD_Akkadian-PISANDUB. `https://github.com/UniversalDependencies/UD_Akkadian-PISANDUB`.

Natalia Kotsyba, Bohdan Moskalevskyi, Mykhailo Romanenko, Halyna Samoridna, Ivanka Kosovska, Olha Lytvyn, Oksana Orlenko, Hanna Brovko, Bohdana Matushko, Natalia Onyshchuk, Valeriia Pareviazko, Yaroslava Rychyk, Anastasiia Stetsenko, Snizhana Umanets, and Larysa Masenko. 2018. UD_Ukrainian-IU. `https://github.com/UniversalDependencies/UD_Ukrainian-IU`.

Vincent Kríž, Barbora Hladká, and Zdenka Uresova. 2016. Czech legal text treebank 1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2387–2392.

Anne Lacheret-Dujour, Sylvain Kahane, and Paola Pietrandrea. 2019. *Rhapsodie: A prosodic and syntactic treebank for spoken French*, volume 89. John Benjamins Publishing Company.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA.

Angeliki Lazaridou, Nghia The Pham, and Marco Baroni. 2015. Combining language and vision with a multimodal skip-gram model. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 153–163, Denver, Colorado. Association for Computational Linguistics.

John Lee, Herman Leung, and Keying Li. 2017. Towards Universal Dependencies for learner Chinese. In *Proceedings of the NoDaLiDa 2017 Workshop* on Universal Dependencies (UDW 2017)*, pages 67–71, Gothenburg, Sweden. Association for Computational Linguistics.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*, Rome, Italy.

Xiaodong Liu, Kevin Duh, Liyuan Liu, and Jianfeng Gao. 2020. Very deep transformers for neural machine translation. *arXiv preprint arXiv:2008.07772v2*.

Yijia Liu, Yi Zhu, Wanxiang Che, Bing Qin, Nathan Schneider, and Noah A. Smith. 2018. Parsing tweets into Universal Dependencies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 965–975, New Orleans, Louisiana. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Mikko Luukko, Aleksi Sahala, Sam Hardwick, and Krister Lindén. 2020. Akkadian treebank for early neo-assyrian royal inscriptions. In *Proceedings of the 19th International Workshop on Treebanks and Linguistic Theories*, pages 124–134, Düsseldorf, Germany. Association for Computational Linguistics.

Olga Lyashevskaya. 2019. A reusable tagset for the morphologically rich language in change: A case of Middle Russian. In *Proceedings of the International Conference Dialogue 2019*, pages 422–434.

Olga Lyashevskaya, Angelika Peljak-Łapińska, and Daria Petrova. 2017. UD_Belarusian-HSE. https://github.com/UniversalDependencies/UD_Belarusian-HSE.

Olga Lyashevskaya and Dmitry Sichinava. 2017. UD_Lithuanian-HSE. https://github.com/UniversalDependencies/UD_Lithuanian-HSE.

Teresa Lynn and Jennifer Foster. 2016. Universal dependencies for irish. In *CLTW*.

Aibek Makazhanov, Aitolkyn Sultangazina, Olzhas Makhambetov, and Zhandos Yessenbayev. 2015. Syntactic annotation of Kazakh: Following the Universal Dependencies guidelines. a report. In *3rd International Conference on Turkic Languages Processing, (TurkLang 2015)*, pages 338–350.

Christopher D Manning. 2015. Computational linguistics and deep learning. *Computational Linguistics*, 41(4):701–707.

Cătălina Mărănduc, Cenel-Augusto Perez, and Radu Simionescu. 2016. Social media-processing Romanian chat and discourse analysis. *Computación y Sistemas*, 20(3):405–414.

Héctor Martínez Alonso and Barbara Plank. 2017. When is multitask learning effective? semantic sequence prediction under varying data conditions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, Valencia, Spain. Association for Computational Linguistics.

Héctor Martínez Alonso, Djamé Seddah, and Benoît Sagot. 2016. From noisy questions to Minecraft texts: Annotation challenges in extreme syntax scenario. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 13–23, Osaka, Japan. The COLING 2016 Organizing Committee.

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. Association for Computational Linguistics.

Maria Mitrofan, Verginica Barbu Mititelu, and Grigorina Mitrofan. 2019. MoNERo: a biomedical gold standard corpus for the Romanian language. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 71–79, Florence, Italy. Association for Computational Linguistics.

Foroushani Mojiri, Hossein Amir, Hamid Aghaei, and Amir Ahmadi. 2020. UD_Soi-AHA. https://github.com/UniversalDependencies/UD_Soi-AHA.

AmirHossein Mojiri Foroushani, Hamid Aghaei, and Amir Ahmadi. 2020a. UD_Khunsari-AHA. https://github.com/UniversalDependencies/UD_Khunsari-AHA.

AmirHossein Mojiri Foroushani, Hamid Aghaei, and Amir Ahmadi. 2020b. UD_Nayini-AHA. https://github.com/UniversalDependencies/UD_Nayini-AHA.

Kadri Muischnek, Kaili Müürisep, Tiina Puolakainen, Eleri Aedmaa, Riin Kirt, and Dage Särg. 2014. Estonian dependency treebank and its annotation scheme. In *Proceedings of 13th workshop on treebanks and linguistic theories (TLT13)*, pages 285–291.

Kadri Muischnek, Kaili Müürisep, and Dage Dage Särg. 2019. CG roots of UD treebank of Estonian web language. In *Proceedings of the NoDaLiDa 2019 Workshop on Constraint Grammar-Methods, Tools and Applications, 30 September 2019, Turku, Finland*, 168, pages 23–26. Linköping University Electronic Press.

Robert Munro. 2020. Human-in-the-loop machine learning. *Sl: O'REILLY MEDIA*.

Phuong-Thai Nguyen, Xuan-Luong Vu, Thi-Minh-Huyen Nguyen, Van-Hiep Nguyen, and Hong-Phuong Le. 2009. Building a large syntactically-annotated corpus of Vietnamese. In *Proceedings of the Third Linguistic Annotation Workshop (LAW III)*, pages 182–185, Suntec, Singapore. Association for Computational Linguistics.

Rik van Noord, Antonio Toral, and Johan Bos. 2020. Character-level representations improve DRS-based semantic parsing Even in the age of BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4587–4603, Online. Association for Computational Linguistics.

Atul Kr. Ojha and Daniel Zeman. 2020. Universal Dependency treebanks for low-resource Indian languages: The case of Bhojpuri. In *Proceedings of the WILDRE5– 5th Workshop on Indian Language Data: Resources and Evaluation*, pages 33–38, Marseille, France. European Language Resources Association (ELRA).

Mai Omura, Yuta Takahashi, and Masayuki Asahara. 2017. Universal dependency for modern Japanese. In *Proceedings of the 7th Conference of Japanese Association for Digital Humanities (JADH2017)*, pages 34–36.

Robert Östling, Carl Börstell, Moa Gärdenfors, and Mats Wirén. 2017. Universal Dependencies for Swedish Sign Language. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 303–308, Gothenburg, Sweden. Association for Computational Linguistics.

Lilja Øvrelid and Petter Hohle. 2016. Universal Dependencies for Norwegian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1579–1585, Portorož, Slovenia. European Language Resources Association (ELRA).

Lilja Øvrelid, Andre Kåsen, Kristin Hagen, Anders Nøklestad, Per Erik Solberg, and Janne Bondi Johannessen. 2018. The LIA treebank of spoken Norwegian dialects. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Martha Palmer, Rajesh Bhatt, Bhuvana Narasimhan, Owen Rambow, Dipti Misra Sharma, and Fei Xia. 2009. Hindi syntax: Annotating dependency, lexical predicate-argument structure, and phrase structure. In *The 7th International Conference on Natural Language Processing*, pages 14–17.

Niko Partanen, Rogier Blokland, KyungTae Lim, Thierry Poibeau, and Michael Rießler. 2018. The first Komi-Zyrian Universal Dependencies treebanks. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 126–132, Brussels, Belgium. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8026–8037. Vancouver, Canada.

Agnieszka Patejuk and Adam Przepiórkowski. 2018. *From Lexical Functional Grammar to Enhanced Universal Dependencies: Linguistically informed treebanks of Polish*. Institute of Computer Science, Polish Academy of Sciences, Warsaw.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Jussi Piitulainen and Hanna Nurmi. 2017. UD_Finnish-FTB. `https://github.com/UniversalDependencies/UD_Finnish-FTB`.

Tommi A Pirinen. 2019. Building minority dependency treebanks, dictionaries and computational grammars at the same time—an experiment in Karelian treebanking. In *Proceedings of the Third Workshop on Universal Dependencies (UDW, SyntaxFest 2019)*, pages 132–136, Paris, France. Association for Computational Linguistics.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 412–418, Berlin, Germany. Association for Computational Linguistics.

Prokopis Prokopidis and Haris Papageorgiou. 2017. Universal Dependencies for Greek. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 102–106, Gothenburg, Sweden. Association for Computational Linguistics.

Yada Pruksachatkun, Phil Yeres, Haokun Liu, Jason Phang, Phu Mon Htut, Alex Wang, Ian Tenney, and Samuel R. Bowman. 2020. jiant: A software toolkit for research on general-purpose text understanding models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics:*

System Demonstrations, pages 109–117, Online. Association for Computational Linguistics.

Sampo Pyysalo, Jenna Kanerva, Anna Missilä, Veronika Laippala, and Filip Ginter. 2015. Universal Dependencies for Finnish. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 163–172, Vilnius, Lithuania. Linköping University Electronic Press, Sweden.

Peng Qi and Koichi Yasuoka. 2019. UD_Chinese-GSDSimp. `https://github.com/UniversalDependencies/UD_Chinese-GSDSimp`.

Alexandre Rademaker, Fabricio Chalub, Livy Real, Cláudia Freitas, Eckhard Bick, and Valeria de Paiva. 2017. Universal Dependencies for Portuguese. In *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017)*, pages 197–206, Pisa,Italy. Linköping University Electronic Press.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Taraka Rama and Sowmya Vajjala. 2017. A Telugu treebank based on a grammar book. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 119–128, Prague, Czech Republic.

Loganathan Ramasamy and Zdeněk Žabokrtský. 2012. Prague dependency style treebank for Tamil. In *Proceedings of Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, pages 1888–1894, İstanbul, Turkey.

Vinit Ravishankar. 2017. A Universal Dependencies treebank for Marathi. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 190–200, Prague, Czech Republic.

Ines Rehbein, Josef Ruppenhofer, and Bich-Ngoc Do. 2019. tweeDe – a Universal Dependencies treebank for German tweets. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 100–108, Paris, France. Association for Computational Linguistics.

Eiríkur Rögnvaldsson, Anton Karl Ingason, Einar Freyr Sigurosson, and Joel Wallenberg. 2012. The Icelandic parsed historical corpus (IcePaHC). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1977–1984, Istanbul, Turkey. European Language Resources Association (ELRA).

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Sebastian Ruder and Barbara Plank. 2018. Strong baselines for neural semi-supervised learning under domain shift. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1044–1054, Melbourne, Australia. Association for Computational Linguistics.

Jack Rueter and Niko Partanen. 2019. Survey of Uralic Universal Dependencies development. In *Workshop on Universal Dependencies*, page 78. The Association for Computational Linguistics.

Jack Rueter, Niko Partanen, and Larisa Ponomareva. 2020. On the questions in developing computational infrastructure for Komi-permyak. In *Proceedings of the Sixth International Workshop on Computational Linguistics of Uralic Languages*, pages 15–25, Wien, Austria. Association for Computational Linguistics.

Jack Rueter and Francis Tyers. 2018. Towards an open-source universal-dependency treebank for Erzya. In *Proceedings of the Fourth International Workshop on Computational Linguistics of Uralic Languages*, pages 106–118.

Jack Michael Rueter. 2018. Mordva. In *The Uralic Languages*. Routledge.

Mohammad Sadegh Rasooli, Pegah Safari, Amirsaeid Moloodi, and Alireza Nourian. 2020. The Persian dependency treebank made universal. *arXiv e-prints*, pages arXiv–2009.

Alessio Salomoni. 2019. UD_German-LIT. `https://github.com/UniversalDependencies/UD_German-LIT`.

Tanja Samardžić, Mirjana Starović, Željko Agić, and Nikola Ljubešić. 2017. Universal Dependencies for Serbian in comparison with Croatian and other Slavic languages. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*, pages 39–44, Valencia, Spain. Association for Computational Linguistics.

Stephanie Samson and Cagrı Cöltekin. 2020. UD_Tagalog-TRG. `https://github.com/UniversalDependencies/UD_Tagalog-TRG`.

Manuela Sanguinetti and Cristina Bosco. 2014. Towards a Universal Stanford Dependencies parallel treebank. In *Proceedings of the 13th Workshop on Treebanks and Linguistic Theories (TLT-13)*. Springer.

Manuela Sanguinetti, Cristina Bosco, Alberto Lavelli, Alessandro Mazzei, Oronzo Antonelli, and Fabio Tamburini. 2018. PoSTWITA-UD: an Italian Twitter treebank in Universal Dependencies. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Victor Sanh, Thomas Wolf, and Sebastian Ruder. 2019. A hierarchical multi-task approach for learning embeddings from semantic tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6949–6956, Honolulu, Hawaii, USA.

Kengatharaiyer Sarveswaran and Gihan Dias. 2020. ThamizhiUDp: A dependency parser for Tamil. *arXiv preprint arXiv:2012.13436*.

Kevin Scannell. 2020. Universal Dependencies for Manx Gaelic. In *Proceedings of the Fourth Workshop on Universal Dependencies (UDW 2020)*, pages 152–157, Barcelona, Spain (Online). Association for Computational Linguistics.

Djamé Seddah and Marie Candito. 2016. Hard time parsing questions: Building a QuestionBank for French. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2366–2370, Portorož, Slovenia. European Language Resources Association (ELRA).

Mojgan Seraji, Filip Ginter, and Joakim Nivre. 2016. Universal Dependencies for Persian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2361–2365, Portorož, Slovenia. European Language Resources Association (ELRA).

Binyam Ephrem Seyoum, Yusuke Miyao, and Baye Yimam Mekonnen. 2018. Universal Dependencies for Amharic. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Tatiana Shavrina and Olga Shapovalova. 2017. To the methodology of corpus construction for machine learning:"Taiga" syntax tree corpus and parser. In *Proceedings of "CORPORA-2017" International Conference*, pages 78–84.

Mo Shen, Ryan McDonald, Daniel Zeman, and Peng Qi. 2016. UD_Chinese-GSD. `https://github.com/UniversalDependencies/UD_Chinese-GSD`.

Timothy Shopen. 2018. UD_Warlpiri-UFAL. `https://github.com/UniversalDependencies/UD_Lithuanian-HSE`.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Chris Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2897–2904, Reykjavik, Iceland. European Language Resources Association (ELRA).

Kiril Simov, Petya Osenova, Alexander Simov, and Milen Kouylekov. 2005. Design and implementation of the Bulgarian HPSG-based treebank. *Journal of Research on Language and Computation. Special Issue*, pages 495–522.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Anders Søgaard and Yoav Goldberg. 2016. Deep multi-task learning with low level tasks supervised at lower layers. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 231–235, Berlin, Germany. Association for Computational Linguistics.

Achim Stein and Sophie Prévost. 2013. Syntactic annotation of medieval texts. *New methods in historical corpora*, 3:275.

Milan Straka. 2018. UDPipe 2.0 prototype at CoNLL 2018 UD shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207, Brussels, Belgium. Association for Computational Linguistics.

Sara Stymne, Miryam de Lhoneux, Aaron Smith, and Joakim Nivre. 2018. Parser training with heterogeneous treebanks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 619–625, Melbourne, Australia. Association for Computational Linguistics.

Umut Sulubacak, Memduh Gokirmak, Francis Tyers, Çağrı Çöltekin, Joakim Nivre, and Gülşen Eryiğit. 2016. Universal Dependencies for Turkish. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3444–3454, Osaka, Japan. The COLING 2016 Organizing Committee.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, volume 27, pages 3104–3112, Montreal, Canada.

Guillaume Thomas. 2019. Universal Dependencies for Mbyá Guaraní. In *Proceedings of the Third Workshop on Universal Dependencies (UDW, SyntaxFest 2019)*, pages 70–77, Paris, France. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Marsida Toska, Joakim Nivre, and Daniel Zeman. 2020. Universal Dependencies for Albanian. In *Proceedings of the Fourth Workshop on Universal Dependencies (UDW 2020)*, pages 178–188, Barcelona, Spain (Online). Association for Computational Linguistics.

Utku Türk, Furkan Atmaca, Şaziye Betül Özateş, Gözde Berk, Seyyit Talha Bedir, Abdullatif Köksal, Balkiz Öztürk Başaran, Tunga Güngör, and Arzucan Özgür. 2020. Resources for Turkish dependency parsing: Introducing the BOUN treebank and the BoAT annotation tool.

Francis Tyers and Karina Mishchenkova. 2020. Dependency annotation of noun incorporation in polysynthetic languages. In *Proceedings of the Fourth Workshop on Universal Dependencies (UDW 2020)*, pages 195–204, Barcelona, Spain (Online). Association for Computational Linguistics.

Francis Tyers, Mariya Sheyanova, Aleksandra Martynova, Pavel Stepachev, and Konstantin Vinogorodskiy. 2018. Multi-source synthetic treebank creation for improved cross-lingual dependency parsing. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 144–150, Brussels, Belgium. Association for Computational Linguistics.

Francis M Tyers and Vinit Ravishankar. 2018. A prototype dependency treebank for Breton. In *Actes de la Conférence TALN. Volume 1 - Articles longs, articles courts de TALN*, pages 197–204, Rennes, France. ATALA.

Francis M. Tyers and Mariya Sheyanova. 2017. Annotation schemes in North Sámi dependency parsing. In *Proceedings of the Third Workshop on Computational Linguistics for Uralic Languages*, pages 66–75, St. Petersburg, Russia. Association for Computational Linguistics.

Veronika Vincze, Dóra Szauter, Attila Almási, György Móra, Zoltán Alexin, and János Csirik. 2010. Hungarian dependency treebank. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Valentin Vydrin. 2013. Bamana Reference Corpus (BRC). *Procedia-Social and Behavioral Sciences*, 95:75–80.

Joachim Wagner, James Barry, and Jennifer Foster. 2020. Treebank embedding vectors for out-of-domain dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8812–8818, Online. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Hongmin Wang, Jie Yang, and Yue Zhang. 2019. From genesis to creole language: Transfer learning for Singlish Universal Dependencies parsing and pos tagging. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(1):1–29.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Tak-sum Wong, Kim Gerdes, Herman Leung, and John Lee. 2017. Quantitative comparative syntax on the Cantonese-Mandarin parallel dependency treebank. In *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017)*, pages 266–275, Pisa,Italy. Linköping University Electronic Press.

Alina Wróblewska. 2018. Extended and enhanced Polish dependency bank in Universal Dependencies format. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 173–182, Brussels, Belgium. Association for Computational Linguistics.

Mary Yako. 2019. UD_Assyrian-AS. `https://github.com/UniversalDependencies/UD_Assyrian-AS`.

Koichi Yasuoka. 2019. Universal dependencies treebank of the four books in Classical Chinese. In *DADH2019: 10th International Conference of Digital Archives and Digital Humanities*, pages 20–28. Digital Archives and Digital Humanities.

M Yavrumyan, H Khachatrian, A Danielyan, and G Arakelyan. 2017. ArmTDP: Eastern Armenian treebank and dependency parser. In *XI International Conference on Armenian Linguistics, Abstracts. Yerevan*.

Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. 2018. Adaptive methods for nonconvex optimization. In *Advances in Neural Information Processing Systems*, volume 31, pages 9793–9803, Montreal, Canada.

Shorouq Zahra. 2020. Parsing low-resource Levantine Arabic: Annotation projection versus small-sized annotated data.

Amir Zeldes. 2017. The GUM corpus: Creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.

Amir Zeldes and Mitchell Abrams. 2018. The Coptic Universal Dependency treebank. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 192–201, Brussels, Belgium. Association for Computational Linguistics.

Dan Zeman, Anna Nedoluzhko, and Martin Majliš. 2017. UD_Upper_Sorbian-UFAL. https://github.com/UniversalDependencies/UD_Upper_Sorbian-UFAL.

Daniel Zeman. 2017. Slovak dependency treebank in Universal Dependencies. *Journal of Linguistics/Jazykovednỳ casopis*, 68(2):385–395.

Dataset size	RTE 2k	MRPC 4k	CoLa 9k	SST-2 67k	QNLI 105k	QQP 364k	MNLI 393k	MNLI-mis 393k	SNLI 550k
Single	67.1	85.5	74.7	88.4	85.2	90.5	80.2	80.8	88.9
All	69.3	81.6	70.2	88.2	82.3	90.1	79.2	79.7	88.1
Smoothed	72.9	82.8	72.7	87.6	83.1	90.3	78.8	80.1	88.4

Table 5: The scores (accuracy) per dataset on the GLUE tasks (dev) for a variety of multi-task settings (ordered by size, indicated in number of sentences in training data).

Appendix

Multi-dataset evaluation on GLUE tasks Table 5 contains the per-dataset scores for the GLUE tasks for all our tested settings. Only for RTE the performance increases when using multi-task learning. Overall, smoothing helps to overcome some of the performance loss we get when training one model on all datasets simultaneously.

Multi-dataset evaluation on UD treebanks Table 6 (on the next four pages) shows the LAS scores for each treebank (UD2.7) for all of our settings. We pre-processed the data with the UD-conversion tools to remove all language-specific sub-labels, and the multi-word tokens and empty nodes. *However, we calculate the scores against the official files for fair comparison.*[12] We included as many datasets as we could find. In the top part of the table, we include all official UD datasets for which we could get the words (only UD_Arabic-NYUAD and UD_Japanese-BCCWJ are missing), and the last 12 treebanks are taken from other sources, some have undergone some specific pre-processing to pass the evaluation script; details about this process can be found in the repository in `scripts/udExtras`.

[12]This is why the scores for some datasets might seem low compared to previous work, which did either do tokenization or did not take it into account during evaluation. In our case the model is punished for not tokenizing.

dataset	citation	proxy	size	self	conc.	+smoothing conc.	+smoothing sepDec	+smoothing dataEmb
af_afribooms	(Dirix et al., 2017)	—	33,894	86.7	85.9	86.6	**87.0**	85.9
aii_as	(Yako, 2019)	et_ewt	0	**9.7**	3.5	3.9	5.1	3.4
ajp_madar	(Zahra, 2020)	ar_padt	0	31.2	**33.8**	33.1	33.2	31.2
akk_pisandub	(Kopacewicz, 2018)	et_edt	0	3.0	4.3	**4.7**	3.6	3.3
akk_riao	(Luukko et al., 2020)	et_edt	0	4.0	**8.2**	7.6	7.3	8.1
am_att	(Seyoum et al., 2018)	et_ewt	0	**1.8**	0.8	0.8	0.5	0.8
apu_ufpa	(Freitas, 2017)	fi_ftb	0	6.1	13.3	13.1	8.1	**13.4**
aqz_tudet	(Aragon, 2018)	cs_pdt	0	6.7	9.6	9.6	9.2	**14.7**
ar_padt	(Hajič et al., 2009)	—	191,869	31.5	31.4	31.3	31.4	**31.5**
ar_pud	(McDonald et al., 2013)	ar_padt	0	62.8	64.5	63.9	64.0	**64.7**
be_hse	(Lyashevskaya et al., 2017)	—	249,897	81.0	83.6	83.1	81.8	**83.8**
bg_btb	(Simov et al., 2005)	—	124,336	92.5	92.7	92.5	92.7	**92.7**
bho_bhtb	(Ojha and Zeman, 2020)	hi_hdtb	0	**37.7**	36.1	36.2	36.5	36.3
bm_crb	(Vydrin, 2013)	qhe_hiencs	0	**8.8**	6.5	6.1	6.7	6.3
br_keb	(Tyers and Ravishankar, 2018)	fr_gsd	0	**54.9**	32.0	31.3	33.2	32.4
bxr_bdt	(Badmaeva and Tyers, 2017)	—	153	11.6	23.9	**29.0**	21.5	24.0
ca_ancora	(Alonso and Zeman, 2016)	—	416,659	92.1	**92.2**	91.8	91.9	92.2
ckt_hse	(Tyers and Mishchenkova, 2020)	ru_syntagrus	0	8.1	**15.3**	15.3	13.7	14.5
cop_scriptorium	(Zeldes and Abrams, 2018)	—	12,926	0.8	0.8	0.7	0.7	**0.9**
cs_cac	(Hladká et al., 2008)	—	471,594	91.2	**92.2**	91.0	90.8	92.0
cs_cltt	(Kríž et al., 2016)	—	27,752	83.9	89.6	88.7	87.7	**89.6**
cs_fictree	(Jelínek, 2017)	—	133,137	91.5	93.0	92.3	92.4	**93.3**
cs_pdt	(Bejček et al., 2013)	—	1,171,190	92.7	92.8	91.2	91.1	**92.8**
cs_pud	(McDonald et al., 2013)	cs_pdt	0	87.7	**88.4**	88.0	88.1	88.2
cu_proiel	(Haug and Jøhndal, 2008)	—	37,432	65.1	67.1	**68.0**	67.6	67.3
cy_ccg	(Heinecke and Tyers, 2019)	—	15,706	74.5	73.9	**76.2**	76.0	73.8
da_ddt	(Johannsen et al., 2015)	—	80,378	86.7	86.1	86.5	**86.8**	86.0
de_gsd	(Brants et al., 2004)	—	259,194	81.7	79.9	81.5	**82.0**	80.8
de_hdt	(Borges Völker et al., 2019)	—	2,753,627	**96.7**	96.6	90.0	94.8	96.6
de_lit	(Salomoni, 2019)	de_hdt	0	76.9	78.9	**79.8**	77.8	78.4
de_pud	(McDonald et al., 2013)	de_hdt	0	78.5	81.2	**82.3**	78.8	80.6
el_gdt	(Prokopidis and Papageorgiou, 2017)	—	41,212	86.9	89.0	88.9	88.8	**89.0**
en_ewt	(Silveira et al., 2014)	—	202,141	**87.6**	85.6	85.4	86.7	86.0
en_gum	(Zeldes, 2017)	—	81,861	**89.0**	87.3	87.3	88.9	88.1
en_lines	(Ahrenberg, 2015)	—	57,372	87.4	86.8	86.9	**88.0**	87.2
en_partut	(Sanguinetti and Bosco, 2014)	—	43,477	89.7	89.3	89.5	**90.7**	89.8
en_pronouns	(Munro, 2020)	en_ewt	0	81.8	85.5	86.8	84.9	**87.2**
en_pud	(McDonald et al., 2013)	en_ewt	0	89.3	87.8	87.7	**89.7**	89.1
es_ancora	(Alonso and Zeman, 2016)	—	443,086	**90.8**	89.0	88.7	90.5	90.4
es_gsd	(McDonald et al., 2013)	—	375,149	85.6	81.7	81.6	**85.8**	85.0
es_pud	(McDonald et al., 2013)	es_gsd	0	79.4	78.6	78.7	**79.7**	79.5
et_edt	(Muischnek et al., 2014)	—	344,646	86.7	86.7	85.5	85.5	**86.8**
et_ewt	(Muischnek et al., 2019)	—	34,287	74.6	82.4	81.6	80.9	**82.4**
eu_bdt	(Aranzabe et al., 2015)	—	72,974	**83.3**	82.3	82.4	82.4	82.3
fa_perdt	(Sadegh Rasooli et al., 2020)	—	445,587	**89.2**	88.9	84.2	87.8	89.2
fa_seraji	(Seraji et al., 2016)	—	119,945	**87.2**	81.8	84.8	86.9	86.4
fi_ftb	(Piitulainen and Nurmi, 2017)	—	127,359	**89.1**	80.4	80.1	88.6	88.8
fi_ood	(Kanerva, 2020)	fi_tdt	0	77.6	69.5	69.1	77.5	**78.1**
fi_pud	(McDonald et al., 2013)	fi_tdt	0	90.4	86.6	86.0	**90.5**	90.4
fi_tdt	(Pyysalo et al., 2015)	—	162,617	89.1	83.2	82.7	**89.5**	89.5

						+smoothing		
dataset	citation	proxy	size	self	conc.	conc.	sepDec	dataEmb
fo_farpahc	(Ingason et al., 2020)	—	23,089	80.9	87.0	86.5	85.4	**87.1**
fo_oft	(Tyers et al., 2018)	fo_farpahc	0	49.8	62.1	62.2	61.6	**62.7**
fr_fqb	(Seddah and Candito, 2016)	fr_gsd	0	84.9	84.6	84.6	**85.2**	85.2
fr_gsd	(Guillaume et al., 2019)	—	344,975	**88.6**	86.0	85.3	88.5	88.2
fr_partut	(Sanguinetti and Bosco, 2014)	—	23,322	87.0	81.7	82.7	**87.7**	82.7
fr_pud	(McDonald et al., 2013)	fr_gsd	0	85.3	83.9	84.1	85.4	**85.5**
fr_sequoia	(Bonfante et al., 2018)	—	49,157	88.4	85.9	87.1	**89.6**	87.4
fr_spoken	(Lacheret-Dujour et al., 2019)	—	14,921	77.5	81.9	**83.1**	82.3	81.8
fro_srcmf	(Stein and Prévost, 2013)	—	136,020	**88.5**	87.6	87.3	87.4	87.6
ga_idt	(Lynn and Foster, 2016)	—	95,860	77.8	**78.1**	78.1	77.9	78.1
gd_arcosg	(Batchelor, 2019)	—	37,817	72.2	72.8	**73.7**	73.7	72.8
gl_ctg	(Gómez Guinovart, 2017)	—	71,928	**66.3**	65.6	65.4	66.0	65.5
gl_treegal	(Garcia, 2016)	—	14,158	65.9	56.7	63.5	**68.4**	58.5
got_proiel	(Haug and Jøhndal, 2008)	—	35,024	75.4	79.0	**79.7**	77.8	78.9
grc_perseus	(Bamman and Crane, 2011)	—	159,895	59.6	63.3	62.4	62.2	**63.4**
grc_proiel	(Eckhoff et al., 2018)	—	187,033	71.7	74.8	74.0	73.3	**74.9**
gsw_uzh	(Aepli and Clematide, 2018)	de_hdt	0	27.8	36.7	**37.1**	35.1	36.9
gun_thomas	(Thomas, 2019)	it_isdt	0	7.7	10.5	**11.1**	9.2	10.9
gv_cadhan	(Scannell, 2020)	en_singpar	0	2.9	12.2	**13.4**	6.3	12.5
he_htb	(McDonald et al., 2013)	—	98,348	**36.3**	36.0	35.9	36.1	36.2
hi_hdtb	(Palmer et al., 2009)	—	281,057	**92.0**	91.8	91.6	91.8	91.9
hi_pud	(McDonald et al., 2013)	hi_hdtb	0	59.6	**59.8**	59.5	59.6	59.7
hr_set	(Agić and Ljubešić, 2015)	—	152,857	89.1	89.5	88.9	89.7	**90.0**
hsb_ufal	(Zeman et al., 2017)	—	460	10.5	59.8	**65.9**	60.1	59.8
hu_szeged	(Vincze et al., 2010)	—	20,166	82.6	83.9	**85.1**	84.8	84.0
hy_armtdp	(Yavrumyan et al., 2017)	—	41,837	75.0	76.8	**77.3**	76.6	76.2
id_csui	(Alfina et al., 2020)	—	17,904	77.1	74.8	76.9	**79.2**	75.1
id_gsd	(McDonald et al., 2013)	—	97,531	**79.9**	79.7	79.3	79.5	79.9
id_pud	(McDonald et al., 2013)	id_gsd	0	59.6	**63.1**	62.9	61.0	63.1
is_icepahc	(Rögnvaldsson et al., 2012)	—	704,716	**83.5**	83.4	80.3	80.0	83.4
is_pud	(Jónsdóttir and Ingason, 2020)	is_icepahc	0	57.9	**59.3**	59.0	58.7	59.3
it_isdt	(Bosco et al., 2014)	—	257,616	81.1	81.0	80.8	81.0	**81.4**
it_partut	(Sanguinetti and Bosco, 2014)	—	45,477	79.2	80.0	80.1	**80.7**	80.3
it_postwita	(Sanguinetti et al., 2018)	—	95,308	74.0	**74.9**	74.8	74.8	74.7
it_pud	(McDonald et al., 2013)	it_isdt	0	80.1	80.3	80.3	**80.6**	80.6
it_twittiro	(Cignarella et al., 2019)	—	22,656	72.6	**77.3**	77.1	76.5	76.6
it_vit	(Alfieri and Tamburini, 2016)	—	208,795	78.6	78.0	77.6	**78.9**	78.8
ja_gsd	(Asahara et al., 2018)	—	167,482	**93.1**	92.7	92.4	92.4	92.6
ja_modern	(Omura et al., 2017)	ja_gsd	0	51.8	52.9	53.8	**53.8**	52.9
ja_pud	(McDonald et al., 2013)	ja_gsd	0	94.3	**94.3**	94.1	94.2	94.2
kfm_aha	(Mojiri Foroushani et al., 2020a)	fa_perdt	0	17.6	16.7	18.5	18.9	**22.1**
kk_ktb	(Makazhanov et al., 2015)	—	511	21.6	56.7	**59.1**	53.0	56.5
kmr_mg	(Gökırmak and Tyers, 2017)	—	242	12.0	15.8	**36.0**	28.4	16.1
ko_gsd	(Chun et al., 2018)	—	56,687	**85.6**	73.7	77.7	85.0	82.5
ko_kaist	(Chun et al., 2018)	—	296,446	**87.6**	85.0	80.3	86.2	87.1
ko_pud	(McDonald et al., 2013)	ko_kaist	0	47.7	46.1	43.6	48.2	**48.9**
koi_uh	(Rueter et al., 2020)	ru_syntagrus	0	12.2	19.1	**19.4**	18.0	18.4
kpv_ikdp	(Partanen et al., 2018)	ru_syntagrus	0	19.5	22.1	**22.2**	21.1	21.8
kpv_lattice	(Partanen et al., 2018)	ru_syntagrus	0	8.2	11.3	**11.7**	10.5	11.6
krl_kkpp	(Pirinen, 2019)	fi_tdt	0	45.9	42.1	44.9	46.0	**46.4**

						+smoothing		
dataset	citation	proxy	size	self	conc.	conc.	sepDec	dataEmb
la_ittb	(Cecchini et al., 2018)	—	390,785	90.5	**91.0**	89.5	89.8	91.0
la_llct	(Cecchini et al., 2018)	—	194,143	**94.6**	94.6	94.2	94.5	94.5
la_perseus	(Bamman and Crane, 2011)	—	18,184	63.3	68.4	69.1	**69.4**	68.3
la_proiel	(Haug and Jøhndal, 2008)	—	172,133	79.9	**81.6**	80.1	80.1	81.6
lt_alksnis	(Bielinskiene et al., 2016)	—	47,641	78.0	78.1	**78.3**	78.3	78.2
lt_hse	(Lyashevskaya and Sichinava, 2017)	—	3,210	47.8	63.7	64.2	**68.5**	64.3
lv_lvtb	(Gruzitis et al., 2018)	—	167,594	**86.8**	86.6	86.3	86.2	86.8
lzh_kyoto	(Yasuoka, 2019)	—	185,211	79.7	**79.8**	75.9	75.6	79.7
mdf_jr	(Rueter, 2018)	ru_syntagrus	0	16.8	17.7	17.5	**18.2**	17.8
mr_ufal	(Ravishankar, 2017)	—	2,730	50.3	65.9	**67.1**	64.6	64.6
mt_mudt	(Čéplö, 2018)	—	22,880	75.5	76.2	**78.9**	78.1	76.2
myu_tudet	(Gerardi, 2021)	ro_nonstandard	0	16.1	15.4	**17.4**	14.0	14.4
myv_jr	(Rueter and Tyers, 2018)	be_hse	0	**20.1**	18.9	19.1	18.6	18.6
nl_alpino	(Bouma and van Noord, 2017)	—	185,883	90.9	91.4	91.4	91.1	**91.5**
nl_lassysmall	(Bouma and van Noord, 2017)	—	75,080	89.4	91.0	91.0	90.7	**91.2**
no_bokmaal	(Øvrelid and Hohle, 2016)	—	243,886	92.2	**92.6**	92.2	92.3	92.5
no_nynorsk	(Øvrelid and Hohle, 2016)	—	245,330	91.8	92.1	92.0	91.9	**92.2**
no_nynorsklia	(Øvrelid et al., 2018)	—	35,207	74.1	75.6	**76.0**	75.4	75.8
nyq_aha	(Mojiri Foroushani et al., 2020b)	fa_perdt	0	30.8	29.1	37.2	34.2	**38.9**
olo_kkpp	(Pirinen, 2019)	—	144	8.4	40.4	**44.7**	26.3	43.1
orv_rnc	(Lyashevskaya, 2019)	—	10,156	58.3	70.6	**71.6**	69.6	70.5
orv_torot	(Eckhoff and Berdičevskis, 2015)	—	118,630	63.9	65.1	64.6	64.4	**65.4**
otk_tonqq	(Derin, 2020)	et_ewt	0	7.7	11.8	5.9	**11.9**	7.1
pcm_nsc	(Caron et al., 2019)	—	111,843	90.0	90.2	89.9	89.5	**90.2**
pl_lfg	(Patejuk and Przepiórkowski, 2018)	—	104,750	95.7	93.7	93.6	95.7	**95.8**
pl_pdb	(Wróblewska, 2018)	—	279,596	89.4	88.8	88.2	89.3	**89.7**
pl_pud	(Wróblewska, 2018)	pl_pdb	0	91.2	91.0	90.5	91.0	**91.4**
pt_bosque	(Rademaker et al., 2017)	—	191,406	**78.2**	74.1	73.8	78.1	77.1
pt_gsd	(McDonald et al., 2013)	—	238,714	**83.0**	80.8	80.6	82.7	82.7
pt_pud	(McDonald et al., 2013)	pt_gsd	0	68.5	**69.6**	69.3	68.8	68.8
qtd_sagt	(Çetinoğlu and Çöltekin, 2019)	—	4,761	46.4	58.0	**60.9**	59.9	57.7
ro_nonstandard	(Mărănduc et al., 2016)	—	532,881	86.8	87.0	86.0	85.7	**87.1**
ro_rrt	(Barbu Mititelu et al., 2016)	—	185,113	88.3	**88.6**	88.3	88.2	88.5
ro_simonero	(Mitrofan et al., 2019)	—	116,857	**91.3**	91.0	91.2	91.0	91.0
ru_gsd	(McDonald et al., 2013)	—	74,906	87.4	88.9	89.2	89.2	**89.7**
ru_pud	(McDonald et al., 2013)	ru_syntagrus	0	86.8	88.5	**89.0**	86.9	87.4
ru_syntagrus	(Droganova et al., 2018)	—	870,479	**93.7**	93.0	88.9	92.0	93.5
ru_taiga	(Shavrina and Shapovalova, 2017)	—	43,557	77.9	78.7	79.6	**81.0**	80.1
sa_ufal	(Dwivedi and Easha, 2017)	hi_hdtb	0	14.2	15.5	16.2	14.4	**16.5**
sa_vedic	(Hellwig et al., 2020)	—	17,445	54.9	57.9	**60.0**	57.5	57.8
sk_snk	(Zeman, 2017)	—	80,575	92.3	**94.3**	93.7	93.1	94.2
sl_ssj	(Dobrovoljc et al., 2017)	—	112,530	**93.4**	93.2	93.1	93.0	93.0
sl_sst	(Dobrovoljc and Nivre, 2016)	—	19,473	69.4	73.6	**74.7**	73.9	73.5
sme_giella	(Tyers and Sheyanova, 2017)	—	16,835	61.3	65.3	**68.5**	64.5	65.5
sms_giellagas	(Rueter and Partanen, 2019)	id_gsd	0	7.8	**14.9**	14.6	11.7	14.8
soj_aha	(Mojiri et al., 2020)	fa_perdt	0	27.9	37.6	27.3	32.1	**39.4**
sq_tsa	(Toska et al., 2020)	ga_idt	0	52.1	62.8	**64.0**	51.2	62.6
sr_set	(Samardžić et al., 2017)	—	74,259	91.9	91.4	91.9	92.4	**92.5**
sv_lines	(Ahrenberg, 2015)	—	55,451	86.5	**88.3**	88.1	88.2	88.2
sv_pud	(McDonald et al., 2013)	sv_lines	0	83.8	**86.9**	86.9	85.8	86.7

dataset	citation	proxy	size	self	conc.	+smoothing conc.	+smoothing sepDec	+smoothing dataEmb
sv_talbanken	(McDonald et al., 2013)	—	66,645	89.1	89.8	89.7	**90.1**	89.7
swl_sslc	(Östling et al., 2017)	—	644	26.2	26.1	**37.7**	29.4	26.8
ta_mwtt	(Sarveswaran and Dias, 2020)	ta_ttb	0	65.4	**70.0**	66.1	67.1	69.9
ta_ttb	(Ramasamy and Žabokrtský, 2012)	—	5,734	40.8	44.7	44.7	**44.9**	44.3
te_mtg	(Rama and Vajjala, 2017)	—	5,082	82.8	84.2	84.5	**85.7**	84.7
th_pud	(McDonald et al., 2013)	en_ewt	0	**28.2**	25.7	25.4	22.2	26.2
tl_trg	(Samson and Cöltekin, 2020)	en_singpar	0	**34.8**	32.9	29.9	25.0	32.4
tl_ugnayan	(Aquino et al., 2020)	en_singpar	0	**28.4**	24.9	25.0	19.3	27.4
tpn_tudet	(Gerardi, 2020)	cs_pdt	0	**9.7**	5.1	4.2	6.5	3.2
tr_boun	(Türk et al., 2020)	—	97,257	69.6	68.8	67.1	69.9	**70.0**
tr_gb	(Cöltekin, 2015)	tr_boun	0	66.3	64.8	64.1	66.1	**66.6**
tr_imst	(Sulubacak et al., 2016)	—	36,822	62.5	59.1	61.2	**64.2**	63.8
tr_pud	(McDonald et al., 2013)	tr_boun	0	61.4	60.7	59.3	61.2	**61.6**
ug_udt	(Eli et al., 2016)	—	19,262	48.5	**50.3**	50.1	49.7	50.2
uk_iu	(Kotsyba et al., 2018)	—	92,355	88.0	90.2	89.7	89.6	**90.3**
ur_udtb	(Bhat et al., 2016)	—	108,690	81.6	82.4	82.3	82.2	**82.8**
vi_vtb	(Nguyen et al., 2009)	—	20,285	**66.1**	65.3	65.3	65.7	65.4
wbp_ufal	(Shopen, 2018)	id_gsd	0	5.5	6.8	**8.7**	7.6	8.0
wo_wtb	(Dione, 2019)	—	22,817	67.6	68.5	**72.6**	71.4	68.4
yo_ytb	(Ishola and Zeman, 2020)	ga_idt	0	16.0	17.2	14.4	12.7	**18.1**
yue_hk	(Wong et al., 2017)	zh_gsd	0	31.8	32.4	32.5	31.7	**32.7**
zh_cfl	(Lee et al., 2017)	zh_gsdsimp	0	47.4	**48.1**	47.6	46.9	47.9
zh_gsd	(Shen et al., 2016)	—	98,616	**85.9**	84.2	84.4	84.3	84.0
zh_gsdsimp	(Qi and Yasuoka, 2019)	—	98,616	**85.8**	84.1	84.5	84.3	84.2
zh_hk	(Wong et al., 2017)	zh_gsd	0	52.1	**53.7**	53.5	52.9	53.6
zh_pud	(McDonald et al., 2013)	zh_gsd	0	62.1	62.2	62.0	61.7	**62.3**
de_tweede	(Rehbein et al., 2019)	—	5,752	68.2	76.9	77.6	**79.6**	77.7
en_aae	(Blodgett et al., 2018)	en_ewt	0	51.5	55.1	55.9	**56.5**	56.1
en_convbank	(Davidson et al., 2019)	—	5,057	69.1	71.4	70.4	71.2	**71.9**
en_esl	(Berzak et al., 2016)	—	78,541	92.0	91.4	91.3	**92.1**	91.7
en_gumreddit	(Behzad and Zeldes, 2020)	—	10,831	75.9	84.9	84.8	**86.5**	85.5
en_monoise	(van der Goot and van Noord, 2018)	en_ewt	0	55.6	64.7	64.5	62.4	**64.7**
en_singpar	(Wang et al., 2019)	—	27,368	80.3	79.0	78.5	**82.2**	79.4
en_tweebank2	(Liu et al., 2018)	—	24,753	80.5	81.7	82.4	**82.6**	81.6
fr_extremeugc	(Martínez Alonso et al., 2016)	fr_gsd	0	56.2	55.7	56.6	**58.0**	54.4
fr_ftb	(Abeillé et al., 2000)	—	442,228	**83.1**	82.2	81.6	82.9	82.8
qfn_fame	(Braggaar and van der Goot, 2021)	nl_alpino	0	**54.0**	43.2	42.6	43.8	43.4
qhe_hiencs	(Bhat et al., 2018)	—	20,203	62.8	62.4	**65.5**	64.0	62.0

Table 6: LAS scores from official conll2018 script on test splits of all UD datasets we could obtain, averaged over 3 random seeds. Size refers to number of sentences in the training split. Results for single dataset trained models, and our 4 multi-task strategies. The last 12 rows contain datasets that are either available without words on the official Universal Dependencies website or are not officialy submitted.