Multilingual End-to-end Dependency Parsing with Linguistic typology knowledge

Linguistic Typology

- Linguistic typology is the classification of human languages according to their syntactic, phonological and semantic features.
- Linguistic typology existed as an independent research domain since long but recently it has been used along with Cross-lingual/Multi-lingual NLP to address the issue of data-sparsity in low-resource languages.
- However, all the popular typological databases suffer from a major shortcoming of limited coverage. In fact, values of many important typological features for numerous low-resource languages are missing in these databases. This significantly limits their utility with Cross-lingual/Multi-lingual NLP.

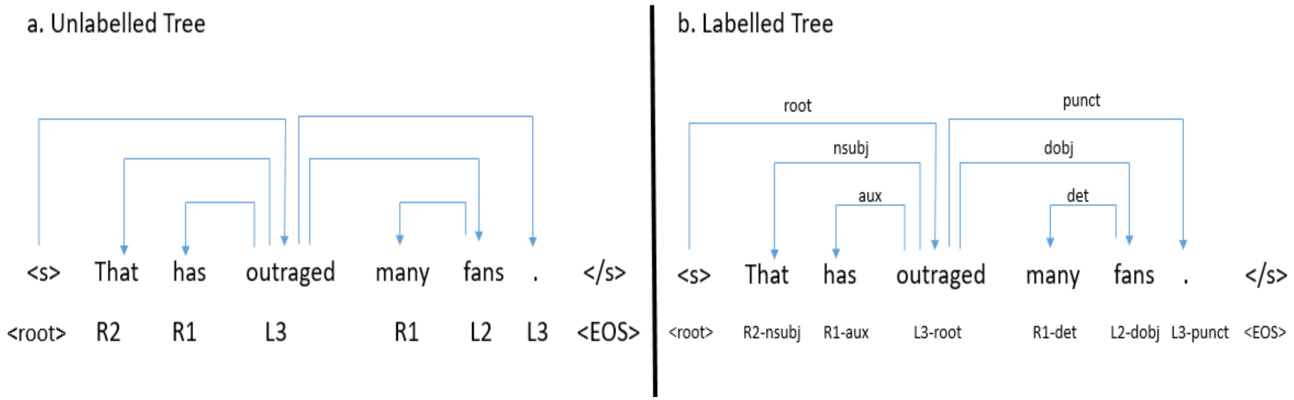
Multitask Learning Framework

- In this work we proposed a Multitasking Model that predicts the missing typology features while utilising the linguistic typology knowledge (both known and predicted) to perform Cross-lingual Dependency Parsing.
- Multitask Learning (MTL) is neural network framework which involves performing of two or more tasks simultaneously leading to knowledge/parameter sharing. These tasks are closely related thus complement each other leading to improved performance on all of them.
- Even in scenarios where we primarily care about a single task, using a closely related task as an auxiliary task for MTL can be useful.
- In this work, we use Linguistic Typology feature-prediction task as auxiliary task for End-to-end Cross-lingual Dependency Parsing.

Major Contribution

In this work, we make following contributions:

1. We evaluated the performance an End-to-end BERT Based Parser which can parse a sentence by directly predicting relative head-position tag for each word within input sentence. This is inspired by the End-to-end Seq2seq Dependency Parser proposed by (Li et al., 2018).
2. We added the auxiliary task of Linguistic typology prediction to our Base End-to-end BERT Based Parser to observe the changes in performances under various experimental settings.

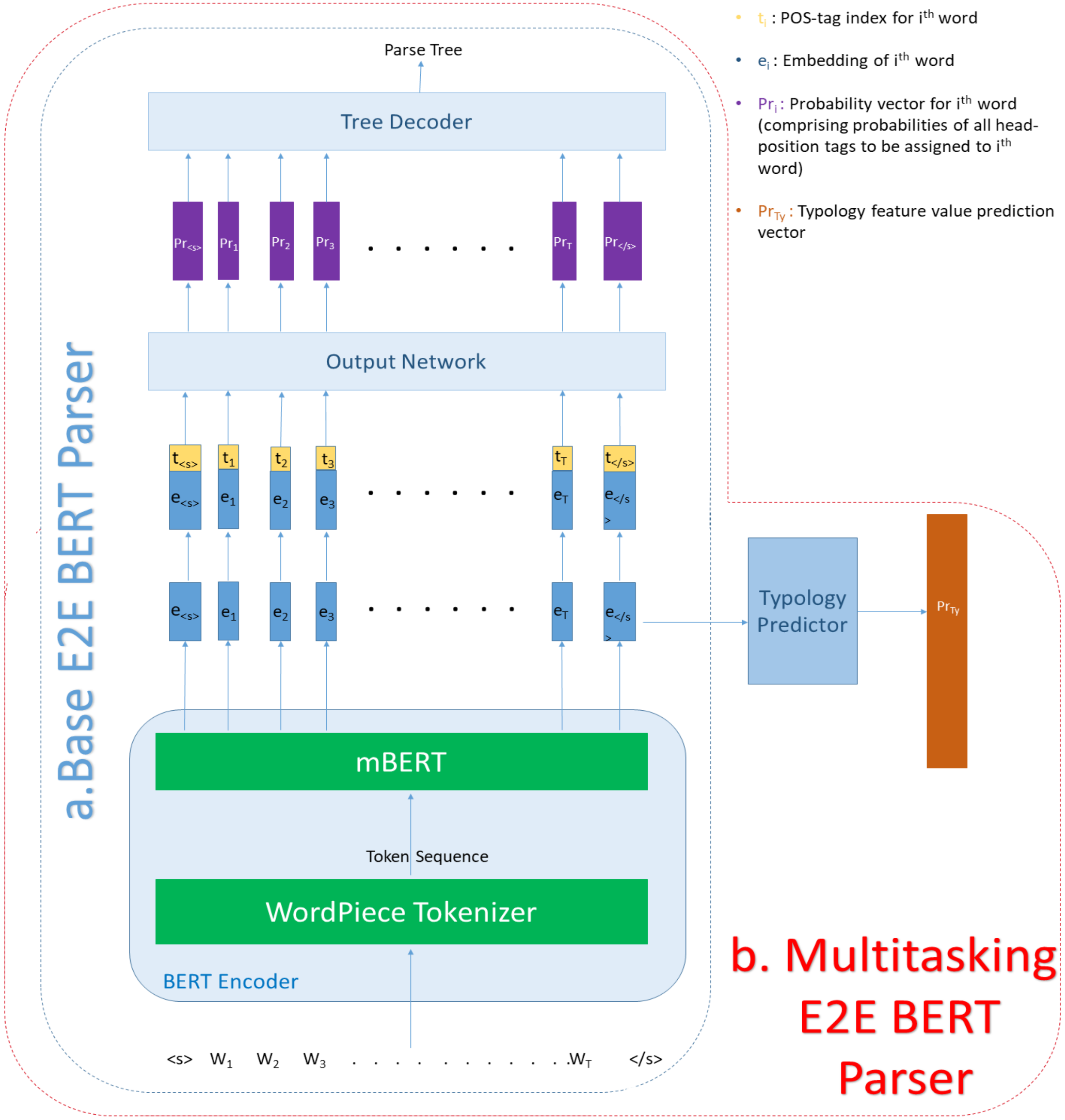


Li, Zuchao, et al. "Seq2seq dependency parsing." *Proceedings of the 27th International Conference on Computational Linguistics*. 2018.

Auxiliary task

Inducing typology knowledge through MTL rather than directly feeding it along with word-embeddings have following advantages.

1. The model can also be applied to low-resource languages for which many typology feature values are unknown/missing.
2. The auxiliary task should help to improve the performance on the main dependency parsing task as well, since it would make the model give special emphasis on the syntactic typology (specially word-order typology) of language being parsed while predicting the dependency relations.



- t_i : POS-tag index for ith word
- e_i : Embedding of ith word
- Pr_i : Probability vector for ith word (comprising probabilities of all head-position tags to be assigned to ith word)
- Pr_{Ty} : Typology feature value prediction vector

mBERT based E2E Dependency Parserers

- The Base End-to-end BERT based Dependency Parser directly predicts the relative head position tag of each word within input sentence as performed by (Li et al., 2018).
- Figure a in the previous slide depicts the architecture of our baseline model. The depicted architecture comprises of three components namely ***BERT Encoder***, ***Output Network*** and ***Tree-decoder*** described in detail in subsequent slides.
- Similarly, Figure b in the previous slide demonstrates the architecture of our proposed Multitasking End-to-end BERT based Dependency Parser. The model is very similar to the Base E2E BERT Parser with one extra component namely the ***Linguistic typology*** predictor which predicts the typology features of language being parsed, described in details in subsequent slides.

Bert Encoder

- It is a BERT based network which takes as input, the entire sentence as sequence of tokens. The model outputs $d-1$ dimensional word-embeddings for all words within the input sentence (where d is a hyperparameter).
- We use WordPiece tokenizer (Wu et al., 2016) to tokenize input sentence and extract embeddings. For each word within input sentence, we use the BERT output corresponding to the first word-piece of it as its embedding, ignoring the rest.
- We add pos-tag information in our parser by appending index of pos-tag of each word, to the encodings outputted by BERT encoder as evident in

Wu, Yonghui, et al. "Google's neural machine translation system: Bridging the gap between human and machine translation." *arXiv preprint arXiv:1609.08144* (2016).

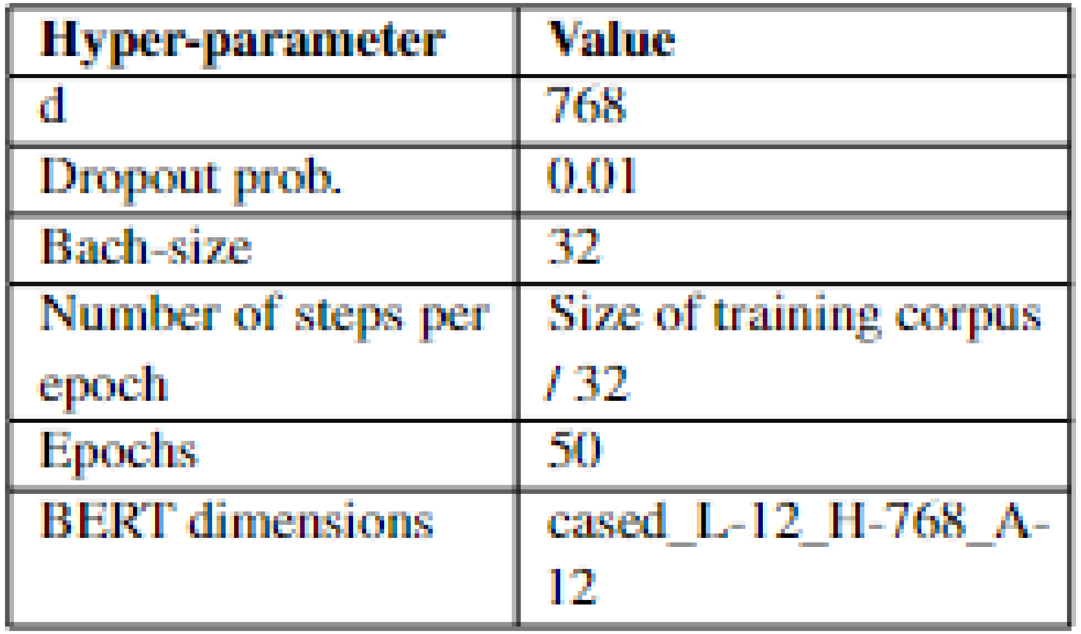
Output Network, Tree-decoder, Typology predictor

- ***Output Network:*** It is a simple feed-forward network with softmax activation function. The network takes-in the embedding matrix from the BERT encoder and outputs the probabilities of all possible relative head position tags to be assigned to each word.
- ***Tree-decoder:*** This component decodes the most probable correct label sequence from Probabilities outputted by Output Network. The correct label sequence would satisfy all the constraints of a dependency parse-tree (outlined by (Li et al., 2018)). We used dynamic programming with beamsearch to efficiently extract the most probable label-sequence.
- ***Linguistic typology predictor:*** It is a simple deep feed forward neural network which takes in the embedding generated by BERT Encoder for token < /s > and outputs probabilities of values of binary syntactic typology features for the language being parsed as 1. Such features are provided by URIEL database

Hyper-parameters

- We trained both BERT Encoder (fine-tuning of pretrained BERT model) and Output Network components of Base E2E BERT Parser model jointly, by optimizing the cross-entropy loss between true relative head-position tags and probabilities outputted by the Output Network.
- On the other hand, Multitasking E2E BERT parser is trained to perform tasks of Prediction of relative head-position tag sequence and Prediction of typology features simultaneously through MTL, by optimizing the total-loss as the sum of cross-entropy loss over true head-position tag-sequence and the binary cross-entropy loss over true typology values.
- The missing typology features pose a problem during training of the Multitasking BERT Parser as there are no true-values for these to optimize loss with. We address this issue through masking technique. We masked the missing typology features and train only on available ones for each source language.
- The next slide outlines hyper-parameters used the training.

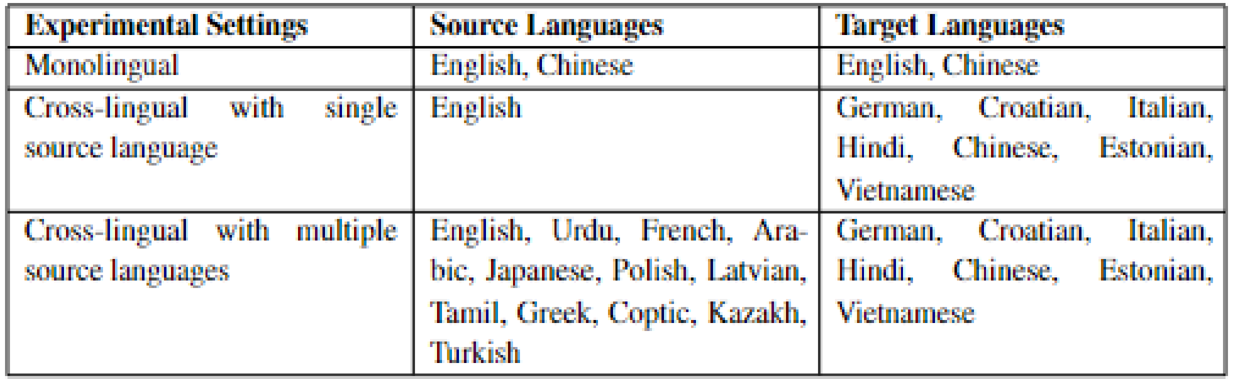
Hyperparameters



Hyper-parameter	Value
d	768
Dropout prob.	0.01
Bach-size	32
Number of steps per epoch	Size of training corpus / 32
Epochs	50
BERT dimensions	cased_L-12_H-768_A-12

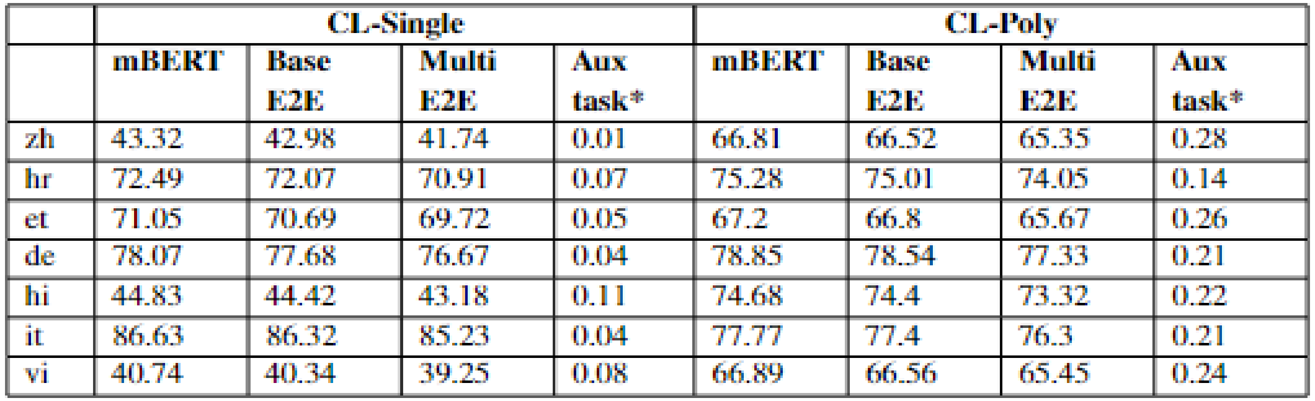
Experimental Setups

- We evaluated the performance of our models in three distinct experimental settings namely ***Monolingual***, ***Cross-lingual with Single source language*** and ***Cross-lingual with Multiple source languages.***
- In ***CL-Single*** settings all the parsers are trained in single source language English, whereas in ***CL-Poly*** settings the parsers are trained on a mixed polyglot corpus of all source languages listed in the table below (with each language equally represented). The training corpus size is always kept constant for experimental accuracy. In both Cross-lingual settings we experimented with *Few-shot* and *Zero-shot* scenerios.



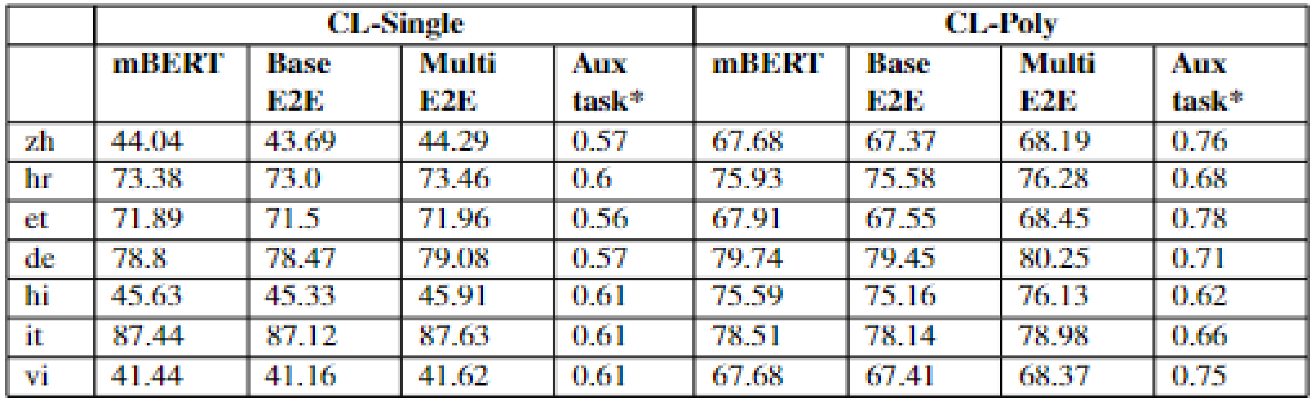
Experimental Settings	**Source Languages**	**Target Languages**
Monolingual	English, Chinese	English, Chinese
Cross-lingual with single source language	English	German, Croatian, Italian, Hindi, Chinese, Estonian, Vietnamese
Cross-lingual with multiple source languages	English, Urdu, French, Arabic, Japanese, Polish, Latvian, Tamil, Greek, Coptic, Kazakh, Turkish	German, Croatian, Italian, Hindi, Chinese, Estonian, Vietnamese

Results



	CL-Single				CL-Poly			
	mBERT	Base E2E	Multi E2E	Aux task*	mBERT	Base E2E	Multi E2E	Aux task*
zh	43.32	42.98	41.74	0.01	66.81	66.52	65.35	0.28
hr	72.49	72.07	70.91	0.07	75.28	75.01	74.05	0.14
et	71.05	70.69	69.72	0.05	67.2	66.8	65.67	0.26
de	78.07	77.68	76.67	0.04	78.85	78.54	77.33	0.21
hi	44.83	44.42	43.18	0.11	74.68	74.4	73.32	0.22
it	86.63	86.32	85.23	0.04	77.77	77.4	76.3	0.21
vi	40.74	40.34	39.25	0.08	66.89	66.56	65.45	0.24

Table 6: Unlabeled Attachment Scores (UAS) achieved in both Cross-lingual settings under *Zero-shot* scenario. *F1 values achieved on the auxiliary task of linguistic typology prediction (excluding missing values)



	CL-Single				CL-Poly			
	mBERT	Base E2E	Multi E2E	Aux task*	mBERT	Base E2E	Multi E2E	Aux task*
zh	44.04	43.69	44.29	0.57	67.68	67.37	68.19	0.76
hr	73.38	73.0	73.46	0.6	75.93	75.58	76.28	0.68
et	71.89	71.5	71.96	0.56	67.91	67.55	68.45	0.78
de	78.8	78.47	79.08	0.57	79.74	79.45	80.25	0.71
hi	45.63	45.33	45.91	0.61	75.59	75.16	76.13	0.62
it	87.44	87.12	87.63	0.61	78.51	78.14	78.98	0.66
vi	41.44	41.16	41.62	0.61	67.68	67.41	68.37	0.75

Table 7: Unlabeled Attachment Scores (UAS) achieved in both Cross-lingual settings under *Few-shot* scenario. *F1 values achieved on the auxiliary task of linguistic typology prediction (excluding missing values)

Key Inferences

- In *CL-Single* setup under both Few-shot and Zero-shot scenarios, all the evaluated mBERT based cross-lingual models (baseline and proposed models) perform better on target languages which are genealogically or geographically closer to the source language English.
- On the other hand, in CL-Poly setup, the evaluated models show almost uniform performance across all target languages in both Few-shot and Zero-shot scenarios.
- Overall, Cross-lingual transfering ability of an mBERT based multilingual dependency parser, to a distinct and unseen target language increases significantly due to polygot training.
- In Monolingual settings, the auxiliary task of predicting linguistic typology features does lead to improvement in parsing performance indeed
- In Cross-lingual settings, the auxiliary task does not help the model to improve the cross-lingual transfer parsing in an unseen language (which are not the part of training corpus). However, the task does enable the model to better learn to distinctively parse in each of the languages on which it is trained, even if the training corpus consists of only few sentence in the language.

Thank You