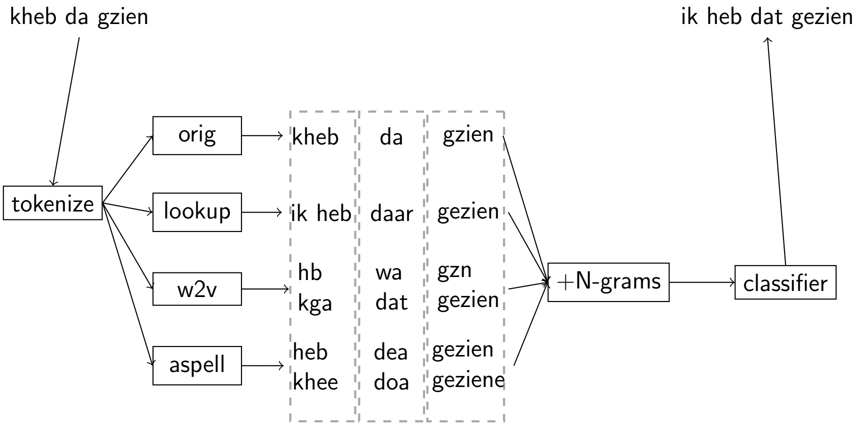
Lexical Normalization for Neural Network Parsing

Rob van der Goot, Gertjan van Noord
University of Groningen
r.van.der.goot@rug.nl

26-01-2018

Last Year (CLIN27)



This Year

- Use normalization to adapt neural network dependency parsers
- Evaluate the effect of normalization versus externally trained word embeddings and character level models
- See if we can exploit top-n candidates
- New treebank to evaluate domain adaptation

New Treebank

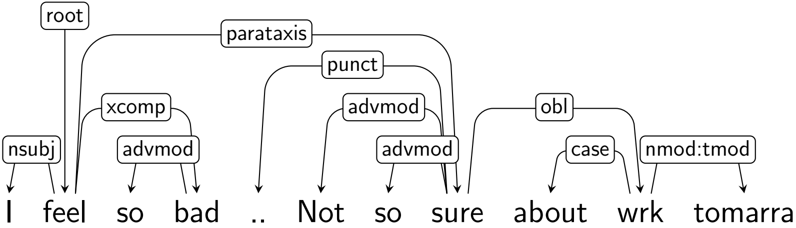
Why?

- Manually corrected train data
- Gold normalization available
- Data should be non-canonical
- UD format

New Treebank

- Pre-filtered to contain non-standard words
- Data from Li and Liu (2015): Owoputi and LexNorm
- 600 Tweets / 10,000 words
- UD2.1 format

New Treebank



New Treebank

Experimental setup:

- Train: English Web Treebank
- Dev: Owoputi
- Test: Lexnorm

Neural Network parser

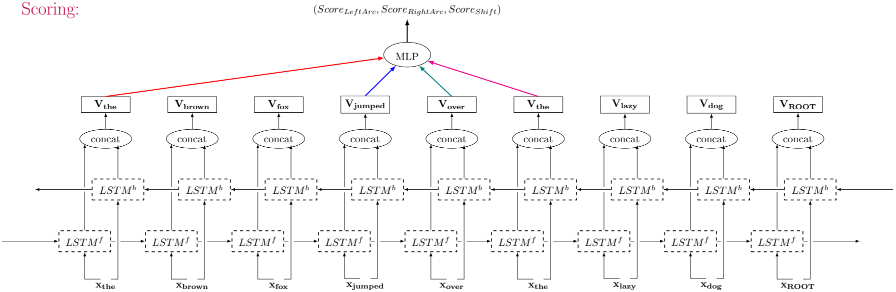
Configuration:

$\mathbf{s_2}$	$\mathbf{s_1}$	$\mathbf{s_0}$	$\mathbf{b_0}$	$\mathbf{b_1}$	$\mathbf{b_2}$	$\mathbf{b_3}$
the	**jumped**	**over**	**the**	lazy	dog	ROOT

fox

brown

Scoring:



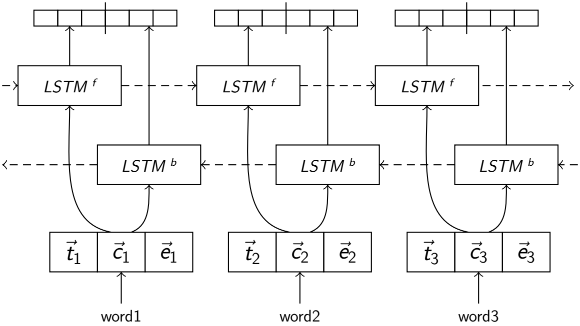
Taken from Kiperwasser and Goldberg (2016)

Neural Network parser

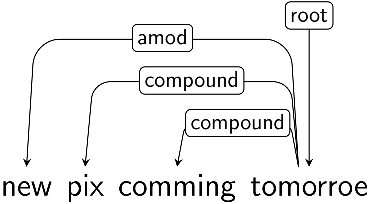
UUparser (de Lhoneux et al., 2017)

- Performs well
- Relatively easy to adapt
- No POS tags
- Characters + external embeddings

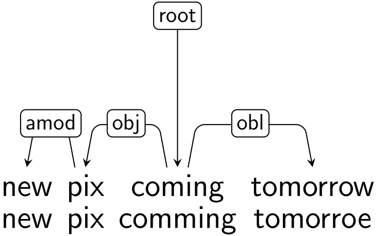
Neural Network parser



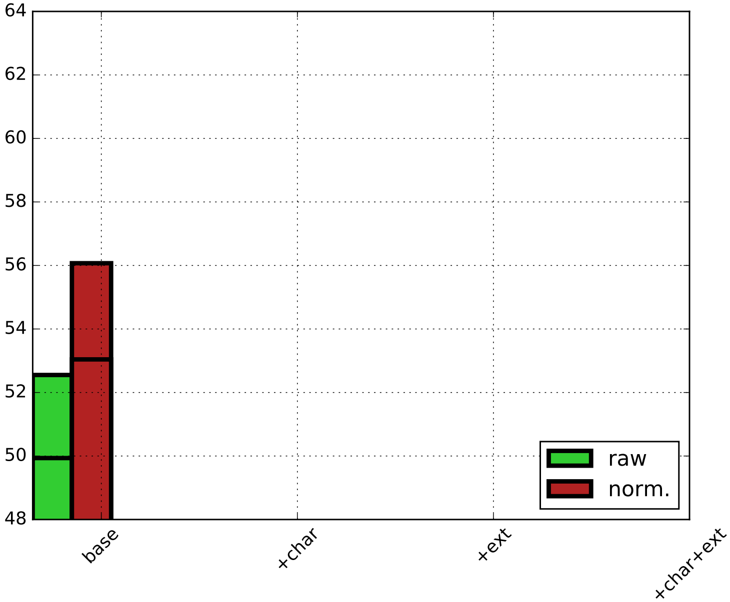
Use Normalization as Pre-processing



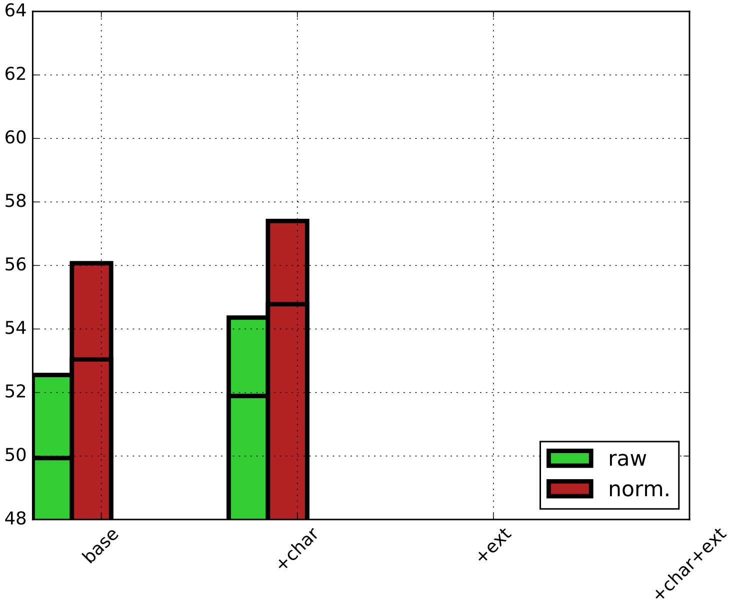
Use Normalization as Pre-processing



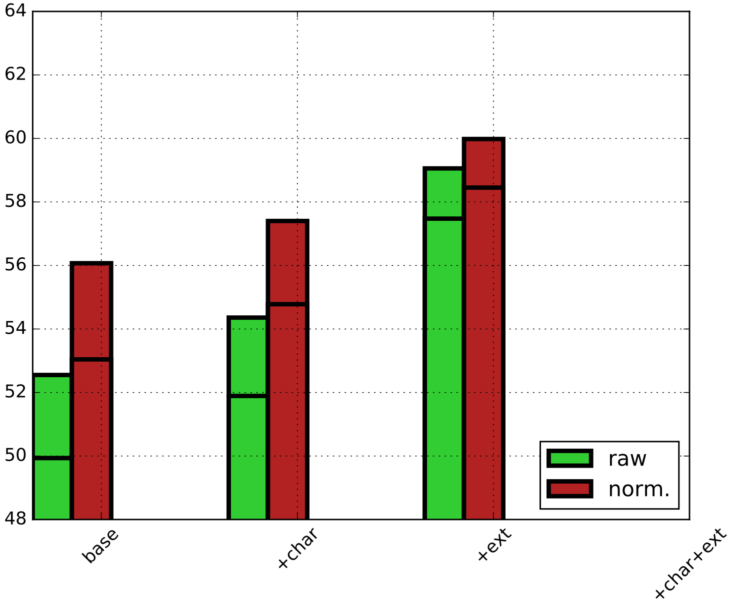
Use Normalization as Pre-processing



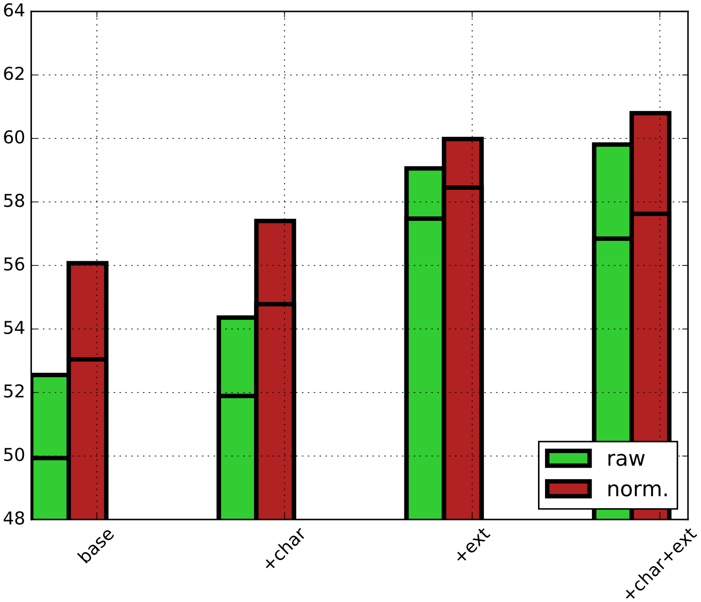
Use Normalization as Pre-processing



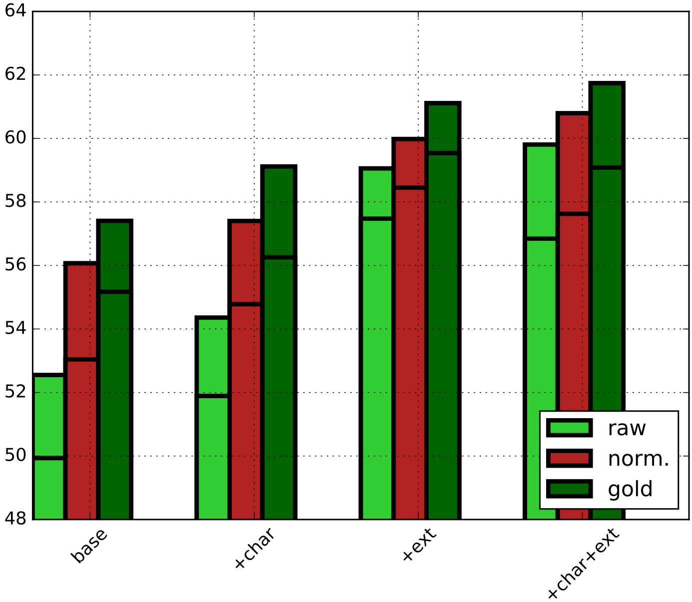
Use Normalization as Pre-processing



Use Normalization as Pre-processing



Use Normalization as Pre-processing



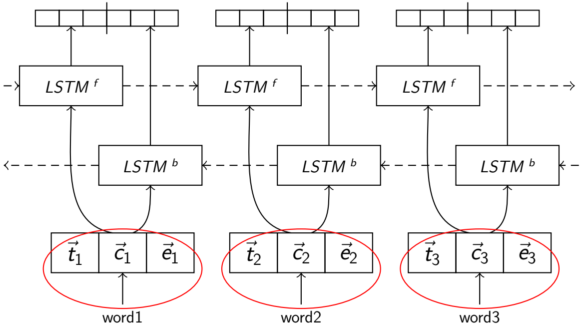
Integrate Normalization

new pix comming tomorroe

Integrate Normalization

new		pix		comming		tomoroe	
new	0.9466	pix	0.7944	coming	0.5684	tomorrow	0.5451
news	0.0315	selfies	0.0882	comming	0.4314	tomoroe	0.3946
knew	0.0111	pictures	0.0559	combing	0.0002	tomorrow's	0.0191
now	0.0063	photos	0.0449	comping	<0.0001	Tagore	0.0174
newt	0.0045	pic	0.0165	common	<0.0001	tomorrows	0.0173

Integrate Normalization



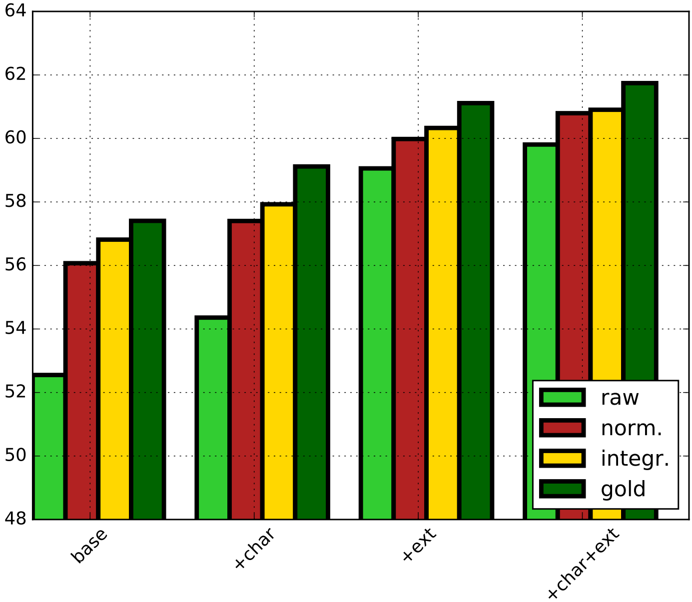
Integrate Normalization

$$\vec{w}_i = \sum_{j=0}^{n} P_{ij} * \vec{n}_{ij}$$

Integrate Normalization

$\vec{w}_1 = (\vec{new} * 0.9466) + (\vec{news} * 0.0315) + (\vec{knew} * 0.0111) + (\vec{now} * 0.0063) + (\vec{newt} * 0.0045)$

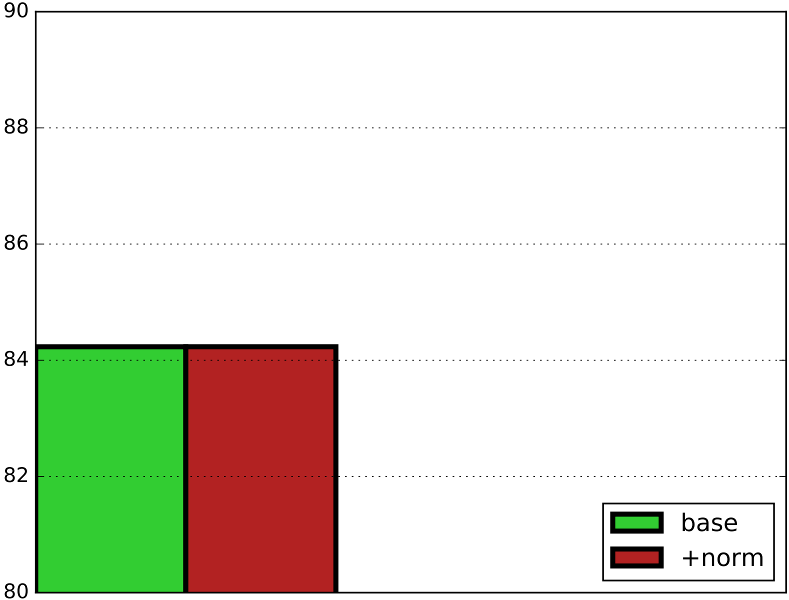
Integrate Normalization



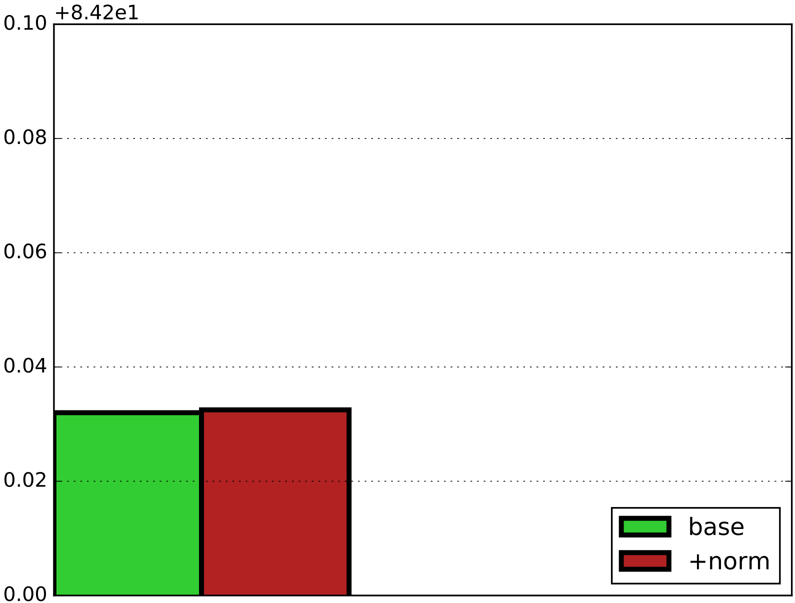
Integrate Normalization

But what about in-domain performance?

Integrate Normalization



Integrate Normalization



Integrate Normalization

Test data:

Model	UAS	LAS
raw	70.47	60.16
normalization-		
direct	71.03*	61.83*
integrated	71.15	62.30*
gold	71.45	63.16*

Table: *indicates statistical significance compared to previous entry.

Integrate Normalization

Conclusions:

- Normalization is still helpful on top of character and external embeddings
- Integrating normalization leads to a small but consistent/significant improvement
- Performance +-60% from using gold normalization
- New dataset will be made available, provides a nice benchmark for domain adaptation

Next CLIN

- Effect of different categories of normalization replacements
- Get closer to gold normalization

Bibliography

Miryam de Lhoneux, Yan Shao, Ali Basirat, Eliyahu Kiperwasser, Sara Stymne, Yoav Goldberg, and Joakim Nivre. From raw text to universal dependencies - look, no tags! In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 207–217, Vancouver, Canada, August 2017. Association for Computational Linguistics.

Eliyahu Kiperwasser and Yoav Goldberg. Simple and accurate dependency parsing using bidirectional LSTM feature representations. *TACL*, 4:313–327, 2016.

Chen Li and Yang Liu. Joint POS tagging and text normalization for informal text. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1263–1269, 2015.

Integrate Normalization

- Foster: not noisy, constituency
- Denoised Web Treebank: no train
- Tweebank: no train
- Foreebank: not noisy