The Denoised Web Treebank

Evaluating Dependency Parsing under Noisy Input Conditions

Joachim Daiber and Rob van der Goot
University of Amsterdam, University of Groningen

OVERVIEW

- Novel benchmark for dependency parsing of noisy Web data.
- Our contributions:
  - Treebank
  - Evaluation metric
  - Experiments

MAIN FINDINGS

- Text normalization improves parse quality on noisy content.
- Normalize beyond word level!
- Treebank and evaluation metric: http://jodaiber.de/DenoisedWebTreebank

DATA

- English Tweets randomly selected from 24h time window (07/01/2012).
- Manual language identification to avoid bias towards well-formed sentences.

TREEBANKS FOR NOISY CONTENT

Name	# Trees	OOV	Style	Norm.
EWT [1]	16.6k	28%	C+D	No
Foster [2]	1k	25%	C	No
Foreebank [3]	1k	29%	C	Yes
Tweebank [4]	929	48%	D	No
This work	**500**	**31%**	**D**	**Yes**

REFERENCES

[1] Slav Petrov and Ryan McDonald. Overview of the 2012 shared task on parsing the web. In *SANCL 2012*.

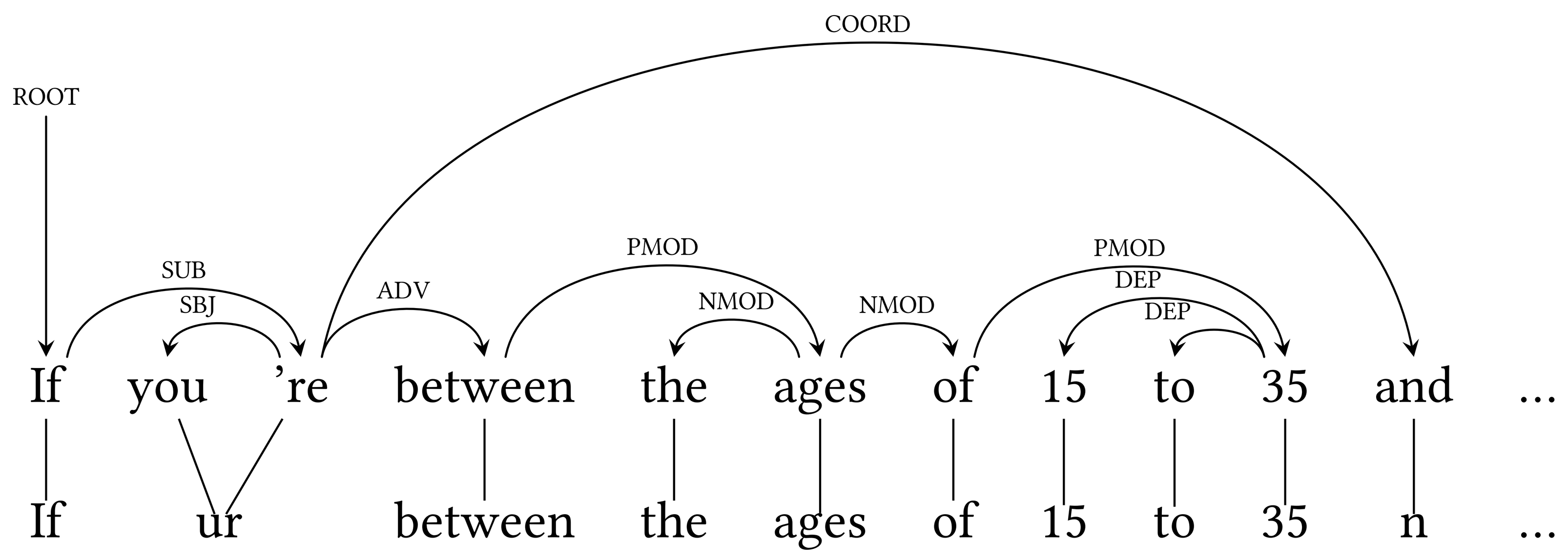
[2] Jennifer Foster et al. From news to comment: Resources and benchmarks for parsing the language of web 2.0. In *IJCNLP 2011*.

[3] Rasoul Kaljahi et al. Foreebank: Syntactic analysis of customer support forums. In *EMNLP 2015*.

[4] Lingpeng Kong et al. A dependency parser for Tweets. In *EMNLP 2014*.

[5] Bo Han and Timothy Baldwin. Lexical normalisation of short text messages: Makn sens a #twitter. In *ACL 2011*.

FORMAT OF THE DATASET



Normalization

- Spelling
- Abbreviations are split (e.g. *cu*)
- Twitter-specific elements
- Zero copulas: Align to empty surface token
- Keeping alignment information

Syntactic annotation

- Syntactic annotation on normalized layer
- Manually annotated POS tags and dependencies (annotated in 2 passes)
- Careful treatment of Twitter-specific items

EVALUATING NOISE-AWARE PARSING

We evaluate:

$D_P = \langle V_P, E_P \rangle \leftarrow$ predicted dependency tree
$D_G = \langle V_G, E_G \rangle \leftarrow$ gold dependency tree
$a_P, a_G \leftarrow$ alignment functions to original text

Aligned precision and recall

- Collect gold and predicted dependencies and the original tokens they align to:

$$M_G = \{\langle a_G(w_i), a_G(w_j) \rangle \mid \langle w_i, r, w_j \rangle \in E_G\}$$
$$M_P = \{\langle a_P(w_i), a_P(w_j) \rangle \mid \langle w_i, r, w_j \rangle \in E_P\}$$

- Calculate gold/predicted overlap:
  - $|M_G \cap M_P|$ true positives
  - $|M_P \setminus M_G|$ false positives
  - $|M_G \setminus M_P|$ false negatives
- Labeled/unlabeled aligned F_1 score:

$$F_1 = 2 \cdot \frac{P \cdot R}{P + R}$$
$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN}$$

APPLICATION: EVALUATING THE EFFECT OF TEXT NORMALIZATION ON PARSING

Normalization method	Unlabeled F_1	Labeled F_1
No normalization	72.41	60.16
+ Twitter syntax rules	76.17*	64.38*
Unsupervised lexical normalization [5]	76.36*	64.80*
Machine translation	76.85*	65.38*
Unsupervised lexical normalization + MT	77.08*	65.57*
Gold normalization, predicted tags	78.20*	68.02*
Gold normalization, gold tags	79.28*	69.85*

* statistically significant against non-normalized baseline at p-value < 0.05.

ACKNOWLEDGEMENTS

Parts of this work were supported through Erasmus Mundus in Language and Communication Technologies (EM LCT). The first author is supported by the EXPERT ITN (EU 7th framework programme). The second author is supported by the Nuance Foundation.