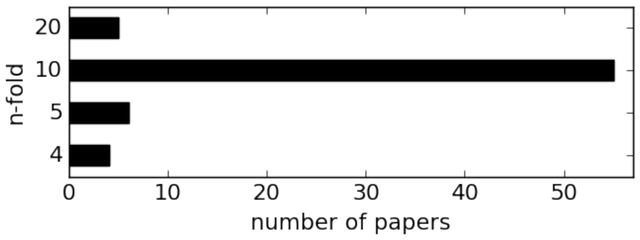
How (not) to train a dependency parser: The curious case of jackknifing part-of-speech taggers

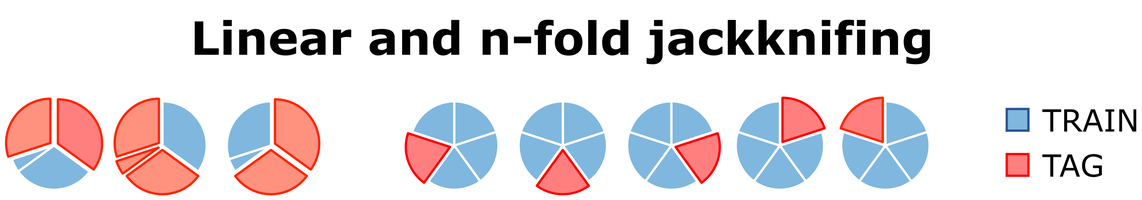
Željko Agić <zeag@itu.dk> and **Natalie Schluter** <natschluter@itu.dk>
Department of Computer Science, IT University of Copenhagen, Denmark

"The traditions in literature"



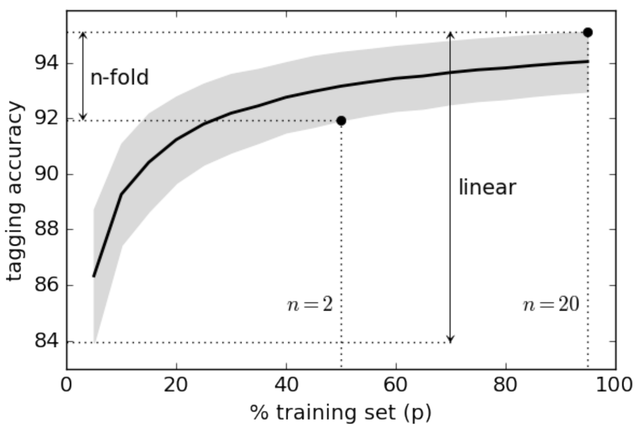
Tenfold jackknifing (n=10) of treebank part-of-speech tags is used indiscriminately as makeshift adaptation in dependency parsing.

Linear and n-fold jackknifing



linear: p=35%

n-fold: n=5



Linear jackknifing permits exploring training sets with less than 50% of treebank data:

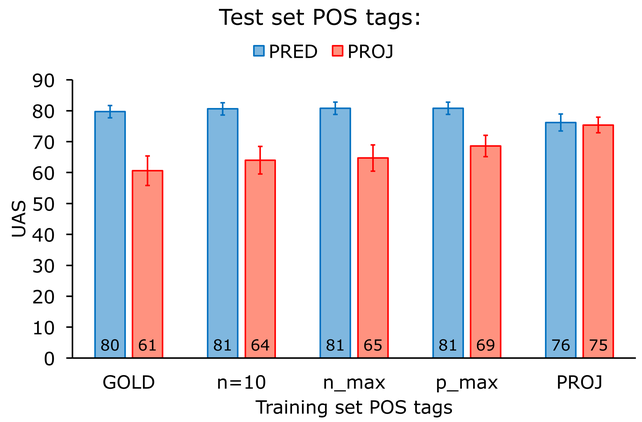
n=2 equals p=50%
p<50% is inaccessible to n

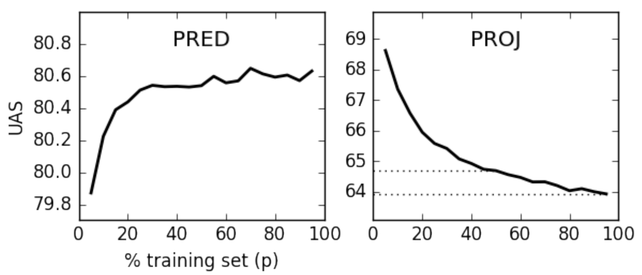
Adaptation thus suffers especially in low-resource scenarios.

Experiment setup

Data. 26 languages, overlap between UD v1.2 and WTC parallel corpus
Tagger and parsers. TnT tagger, Mate: graph-based, and Yara: transition-based
Tagging quality. GOLD, PRED: direct supervision (94.1%), PROJ: cross-lingual (71.7%)
Other. Results averaged over 5 randomized runs for all experiments

Monolingual parsing





Observations

Median p_max = **75**% ; **5**%
Median n_max = **11** ; **2**

Fixing n=10 is suboptimal, **-0.2** and **-4.6** UAS to p_max.

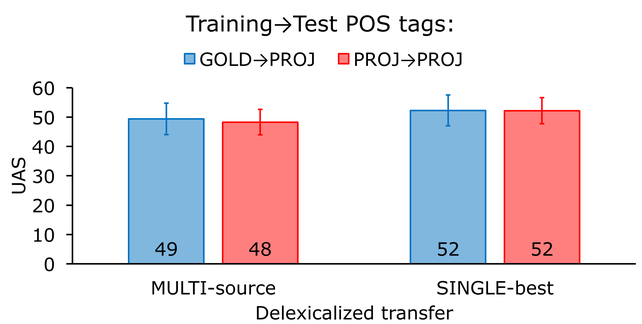
GOLD training provides the best parser for 0 languages.

PROJ→PROJ is by far the best low-resource option.

Best choice for # of 26 languages

	PRED	PROJ
GOLD	0	0
n=10	**9**	0
n_max	**18**	0
p_max	**21**	0
PROJ	0	**26**

Delexicalized transfer



Observations

The unadapted GOLD→PROJ parsers perform better.

+1.1 ; **+0.1** UAS with Mate
+2.1 ; **+0.7** UAS with Yara

Best choice for 14-17/26 langs.