

Synthetic-Error Augmented Parsing of Swedish as a Second Language: Experiments with Word Order



SPRÅKBANKENTEXT

Arianna Masciolini, Emilie Marie Carreau Francis and Maria Irena Szawerna
Språkbanken Text, Department of Swedish, Multilingualism, Language Technology, University of Gothenburg

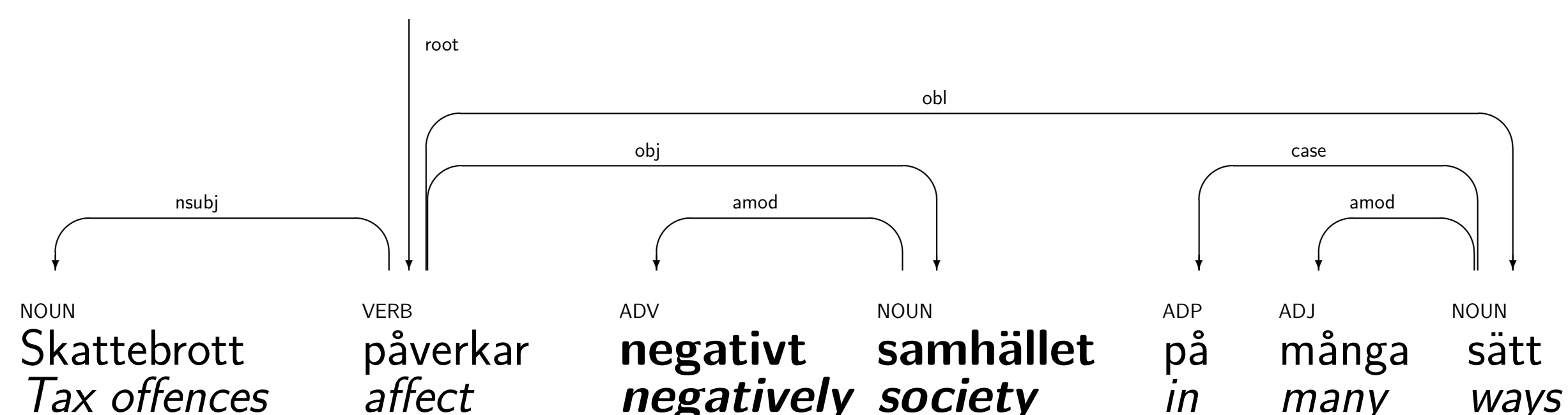
Abstract

Ungrammatical text poses significant challenges for dependency parsers. In this paper, we explore the effectiveness of using synthetic data to improve performance on essays written by learners of Swedish as a second language. Due to their relevance and ease of annotation, we restrict our initial experiments to word order errors. To do that, we build a corrupted version of the standard Swedish Universal Dependencies (UD) treebank Talbanken, mimicking the error patterns and frequency distributions observed in the Swedish Learner Language (SweLL) corpus. We then use the MaChAmp (Massive Choice, Ample tasks) toolkit to train an array of BERT-based dependency parsers, fine-tuning on different combinations of original and corrupted data. We evaluate the resulting models not only on their respective test sets but also, most importantly, on a smaller collection of sentence-correction pairs derived from SweLL. Results show small but significant performance improvements on the target domain, with minimal decline on normative data.

Keywords: Dependency Parsing, Data Augmentation, Second Language Acquisition, L2 Swedish

Background

- ungrammatical text is challenging for automatic annotation tools
- the lack of manually annotated L2 corpora makes it hard to train domain-specific models
- experiments with **synthetically generated errors**, focusing on word order



Models

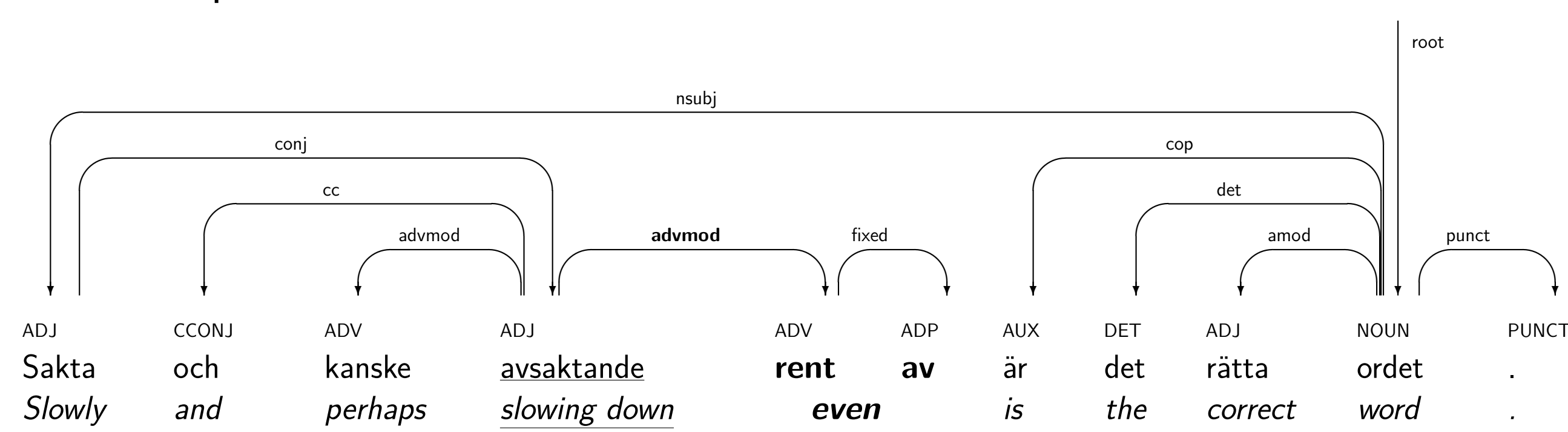
- fine tuning a **Swedish BERT** model with the **MaChAmp toolkit**
- experiments with various data configurations:
 - **baseline** trained on the standard Talbanken
 - corrupted data with normative Swedish to mimic the **real-world error frequency**
 - **50-50 ratio** of corrupted and normative sentences
 - experiments with **sequential training**: the baseline further fine-tuned for 10 or 20 extra epochs on purely synthetic error data

Name	% Normative	% Errors
baseline	100	0
mix15	85	15
mix50	50	50
seq10 (step 1)	100	0
seq10 (step 2)	0	100
seq20 (step 1)	100	0
seq20 (step 2)	0	100

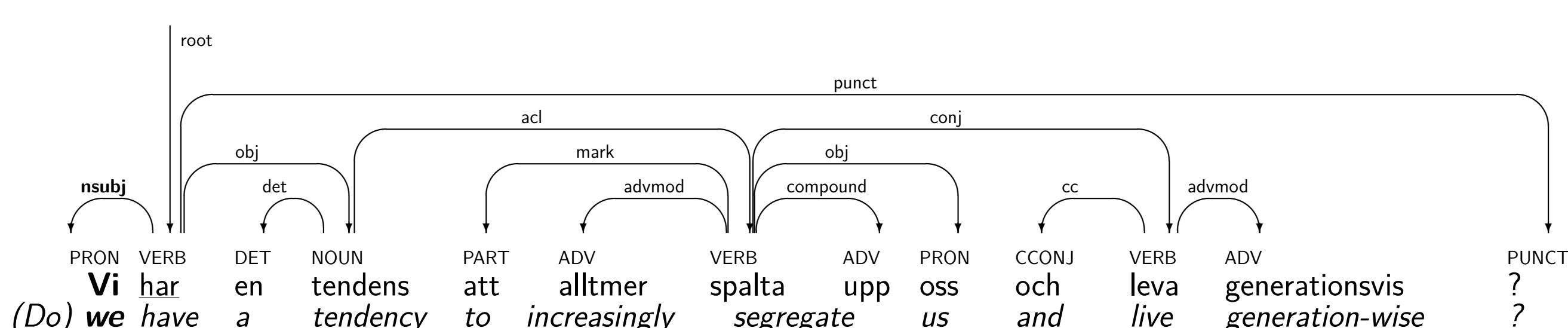
Data

	Description	Train	Dev	Test
SweLL	L2 Swedish	-	-	69
Talbanken	standard Swedish	4303	504	1219
Corrupted	synthetic errors	4303	504	1219

- manually annotated **SweLL** subset with word order errors only
- reference **Talbanken** treebank
- automatically **corrupted** version of Talbanken:
 - **S-Adv** (misplaced adverbial) errors obtained by swapping adverbial subtrees with their syntactic heads
 - **S-FinV** (misplaced finite verb) error obtained by swapping subject subtrees with their syntactic heads
 - other types of errors (**S-WO**) approximated by swapping pairs of random subsequent tokens



S-Adv



S-FinV

Results and Conclusions

	Talbanken LAS	Talbanken UAS	Corrupted LAS	Corrupted UAS	SweLL LAS	SweLL UAS
baseline	92.42	94.30	80.20	83.29	88.28	91.16
mix15	92.23	94.05	87.96	90.50	87.63	90.60
mix50	91.54	93.58	89.59	92.00	89.86	92.93
seq10	92.20	94.06	90.47	92.75	90.05	92.84
seq20	92.53	94.32	90.95	93.08	89.02	92.00

- synthetic word order errors in training, especially with sequential training, have a **small positive effect** on actual learner sentences
- larger improvement** on (in-domain) corrupted sentences
- minimal decline** on standard language

	LAS	UAS
baseline	82.80	86.02
mix15	84.41	89.25
mix50	87.10	90.32
seq10	87.10	89.78
seq20	86.02	89.78

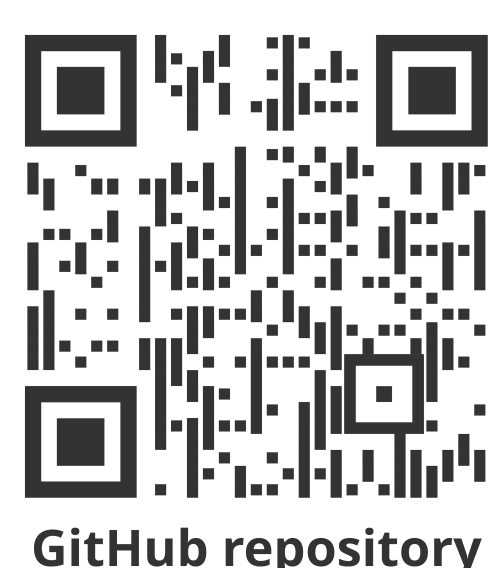
- targeted evaluation** isolating erroneous segments from SweLL sentences shows a **wider performance gap** between baseline and specialized models

Future Work

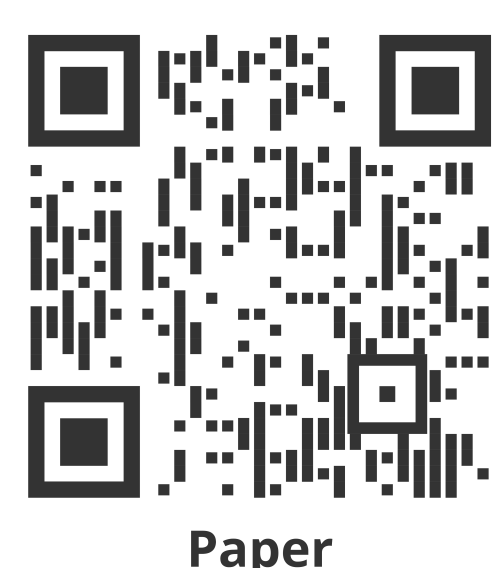
- improving the **corruption pipeline**:
 - more realistic S-WO errors
 - support for more error types
- corrupting texts closer to the **L2 domain**, such as coursebook materials
- rerunning the evaluation on a **larger test set** (possibly including other error types) and/or repeat experiments and compute **multi-run averages**

Acknowledgements

This work is supported by the Swedish national research infrastructure, funded jointly by the Swedish Research Council and the 10 participating partner institutions.



GitHub repository



Paper