BME-TUW at SR'20
Lexical grammar induction for surface realization

Gábor Recski[1], Ádám Kovács[1,2], Kinga Gémes[1,2], Judit Ács[2], András Kornai[3]

[1]TU Wien
firstname.lastname@tuwien.ac.at

[2]Dept. of Automation and Applied Informatics, Budapest U of Technology
lastname.firstname@aut.bme.hu

[3]SZTAKI Institute of Computer Science
andras@kornai.com

Third Workshop on Multilingual Surface Realisation, 12/12/2020

Summary

Summary

- Rule-based system for word order restoration + DL reinflection

Summary

- Rule-based system for word order restoration + DL reinflection
- Improves the grammar-based approach in Kovács et al. (2019)

Summary

- Rule-based system for word order restoration + DL reinflection
- Improves the grammar-based approach in Kovács et al. (2019)
- Still inferior to DL systems, but opens up new possibilities

Interpreted Regular Tree Grammars

Interpreted Regular Tree Grammars

Interpreted Regular Tree Grammars (IRTGs, Koller and Kuhlmann, 2011) encode the correspondence between operations over a string algebra and an s-graph algebra (Courcelle and Engelfriet, 2012; Koller, 2015).

Interpreted Regular Tree Grammars

Interpreted Regular Tree Grammars (IRTGs, Koller and Kuhlmann, 2011) encode the correspondence between operations over a string algebra and an s-graph algebra (Courcelle and Engelfriet, 2012; Koller, 2015).

```
VERB -> _nsubj(VERB, NOUN)
[string] *(?2, ?1)
[ud] f_dep1(merge(merge(?1,"(r<root> :nsubj d1<dep1>)"),r_dep1(?2)))
```

Read: constructing the subgraph VERB $\xrightarrow{\text{nsubj}}$ NOUN corresponds to concatenation in the order NOUN VERB.

Generating subgraphs

Generating subgraphs

$$He/\texttt{PRON} \xleftarrow{nsubj} enjoy/\texttt{VERB} \xrightarrow{obj} it/\texttt{PRON}.$$

Generating subgraphs

$$He/\texttt{PRON} \xleftarrow{nsubj} enjoy/\texttt{VERB} \xrightarrow{obj} it/\texttt{PRON}.$$

PRON	$\xleftarrow{nsubj}$	VERB
PRON	$\xleftarrow{nsubj}$	enjoy
He	$\xleftarrow{nsubj}$	VERB
He	$\xleftarrow{nsubj}$	enjoy
VERB	$\xrightarrow{obj}$	it
VERB	$\xrightarrow{obj}$	PRON
enjoy	$\xrightarrow{obj}$	PRON
enjoy	$\xrightarrow{obj}$	it

PRON	$\xleftarrow{nsubj}$	VERB	$\xrightarrow{obj}$	PRON
PRON	$\xleftarrow{nsubj}$	VERB	$\xrightarrow{obj}$	it
PRON	$\xleftarrow{nsubj}$	enjoy	$\xrightarrow{obj}$	PRON
He	$\xleftarrow{nsubj}$	VERB	$\xrightarrow{obj}$	PRON
PRON	$\xleftarrow{nsubj}$	enjoy	$\xrightarrow{obj}$	it
He	$\xleftarrow{nsubj}$	VERB	$\xrightarrow{obj}$	it
He	$\xleftarrow{nsubj}$	enjoy	$\xrightarrow{obj}$	PRON
He	$\xleftarrow{nsubj}$	enjoy	$\xrightarrow{obj}$	it

For a head word with N dependents, we enumerate $\sim 3^N$ subgraphs.

Model statistics

Lang	N_{patt}	D_{max}	$\|V\|$	D_{words}	N_{tok}
ar	8.6M	4.8	14K	36.9	224K
en	29.8M	5.0	25K	17.6	352K
es	50.2M	5.5	48K	29.0	827K
fr	37.1M	5.7	34K	24.6	429K
hi	17.2M	5.5	15K	21.1	281K
id	7.0M	5.2	19K	21.8	98K
ja	14.5M	5.6	24K	22.5	160K
ko	8.6M	3.9	119K	12.9	353K
pt	27.2M	5.2	32K	25.7	462K
ru	41.6M	4.7	51K	18.0	946K
zh	14.8M	6.8	20K	24.7	99K

Generating subgraphs

Generating subgraphs

- For each UD graph, we generate a separate IRTG

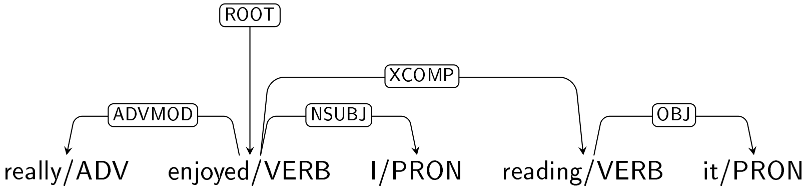
Generating subgraphs

- For each UD graph, we generate a separate IRTG
- For each subgraph, we add the most frequent rule

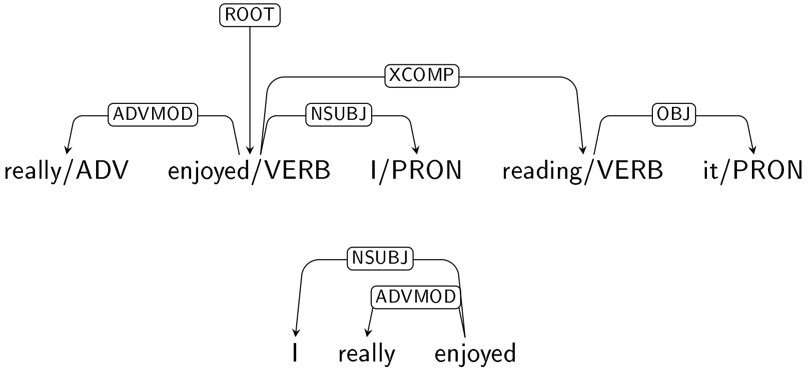
Generating subgraphs

- For each UD graph, we generate a separate IRTG
- For each subgraph, we add the most frequent rule
- Identical rule weights $\rightarrow$ grammars favor shorter derivations with more specific rules

Generating subgraphs



Generating subgraphs

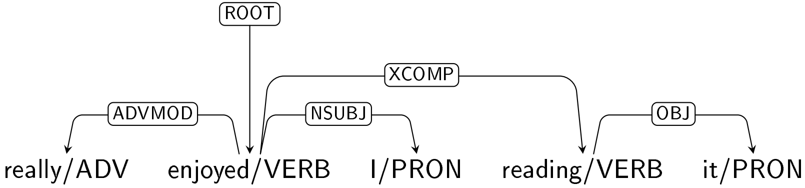


Hierarchical SR

Cut UD graphs along edges between clauses: `acl`, `advcl`, `ccomp`, `xcomp`, `conj`.

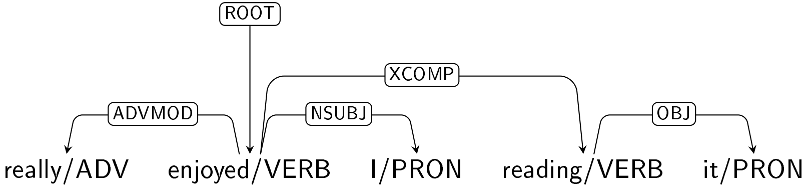
Hierarchical SR

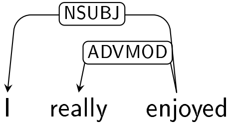
Cut UD graphs along edges between clauses: `acl`, `advcl`, `ccomp`, `xcomp`, `conj`.



Hierarchical SR

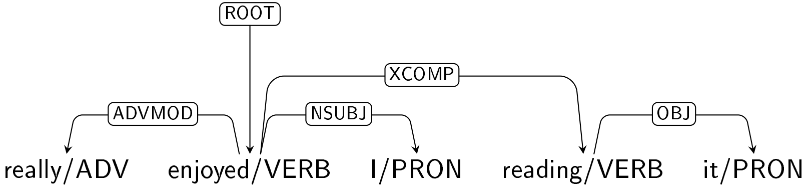
Cut UD graphs along edges between clauses: `acl`, `advcl`, `ccomp`, `xcomp`, `conj`.

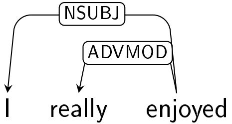


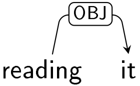


Hierarchical SR

Cut UD graphs along edges between clauses: `acl`, `advcl`, `ccomp`, `xcomp`, `conj`.

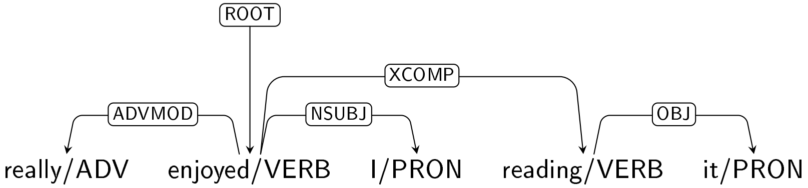


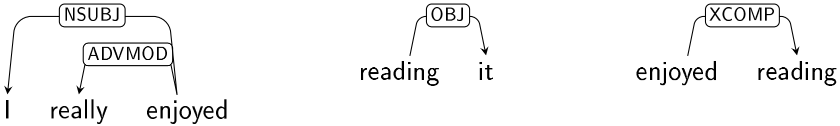




Hierarchical SR

Cut UD graphs along edges between clauses: `acl`, `advcl`, `ccomp`, `xcomp`, `conj`.





Hierarchical SR

In a sample of 500 English sentences, we run 1794 iterations of the core method, and observe recursion depths up to 6.

Hierarchical SR

In a sample of 500 English sentences, we run 1794 iterations of the core method, and observe recursion depths up to 6.

Perhaps had we not gone into this restaurant believing Zahav was going to be golden as its name suggests (and as the many golden reviews seem to attest), we would have enjoyed a decent little expensive experience.

Hierarchical SR

In a sample of 500 English sentences, we run 1794 iterations of the core method, and observe recursion depths up to 6.

Perhaps had we not gone into this restaurant believing Zahav was going to be golden as its name suggests (and as the many golden reviews seem to attest), we would have enjoyed a decent little expensive experience.

gone $\xrightarrow{advcl}$ believing $\xrightarrow{ccomp}$ going $\xrightarrow{xcomp}$ golden $\xrightarrow{advcl}$ suggests $\xrightarrow{conj}$ seem $\xrightarrow{advcl}$ attest

Evaluation

Team	Meaning				Readability			
	ewt		wiki		ewt		wiki	
	Ave.	Ave. z	Ave.	Ave. z	Ave.	Ave. z	Ave.	Ave. z
HUMAN					75.7	0.417	87.4	0.592
IMS	92.7	0.534	92.3	0.475	73.9	0.374	82.1	0.383
ADAPT	90.7	0.476	91.6	0.441	72.5	0.320	81.5	0.373
Concordia	87.0	0.332	88.7	0.275	70.2	0.270	79.6	0.401
BME 2020	79.3	0.086	81.8	-0.050	58.2	-0.152	60.8	-0.299
BME 2019	77.4	0.024	82.4	-0.074	56.7	-0.208	64.4	-0.181

Evaluation

Data	Meaning				Readability			
	BME 2020		BME 2019		BME 2020		BME 2019	
	Ave.	Ave. z	Ave.	Ave. z	Ave.	Ave. z	Ave.	Ave. z
en_ewt	79.3	0.086	77.4	0.024	58.2	-0.152	56.7	-0.208
en_wiki	81.8	-0.050	82.4	-0.074	60.8	-0.299	64.4	**-0.181**
ru_syn	81.2	-0.166	81.3	-0.177	69.7	-0.166	67.3	-0.230
ru_wiki	78.2	**-0.079**	68.2	-0.493	63.2	**0.050**	37.7	-0.781
es_ancora	70.2	-0.276	70.6	-0.271	66.4	-0.401	67.1	-0.378
es_wiki	69.8	**-0.170**	55.5	-0.726	77.2	**0.015**	62.2	-0.628

Plans

Plans

- Use 'unlimited' silver standard UD data

Plans

- Use ‘unlimited’ silver standard UD data
- Learn rule weights

Plans

- Use ‘unlimited’ silver standard UD data
- Learn rule weights
- Qualitative analysis of performance gap

Software

All components of our system are free and open source:

Component	URL	License
Word order restoration	github.com/adaamko/surface_realization	MIT
Reinflection	github.com/juditacs/deep-morphology	MIT
IRTG generation	github.com/recski/tuw-nlp	MIT
IRTG parsing	github.com/coli-saar/alto	Apache 2.0

Thank you!

Courcelle, Bruno and Joost Engelfriet (2012). *Graph structure and monadic second-order logic*. Cambridge University Press.

Koller, Alexander (2015). "Semantic construction with graph grammars". In: *Proceedings of the 14th International Conference on Computational Semantics (IWCS)*. London.

Koller, Alexander and Marco Kuhlmann (2011). "A generalized view on parsing and translation". In: *Proceedings of the 12th International Conference on Parsing Technologies (IWPT)*. Dublin.

Kovács, Ádám, Evelin Ács, Judit Ács, András Kornai, and Gábor Recski (2019). "BME-UW at SRST-2019: Surface realization with Interpreted Regular Tree Grammars". In: *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*. Hong Kong, China: Association for Computational Linguistics, pp. 35–40. DOI: 10.18653/v1/D19-6304.