Geometry-Aware Supertagging
with Heterogeneous Dynamic Convolutions

Konstantinos Kogkalidis & Michael Moortgat
Utrecht Institute of Linguistics OTS, Utrecht University

CLIN 32, June 2022, Tilburg

A composition calculus for
vector-based semantic modelling
with a localization for Dutch

NWO 360-89-070, 2017-**2022** (!!!)

Categorial Grammars 101

what are they?

A **family** of syntactic formalisms; each instance consists of:

- a **lexicon**
  a map assigning *categories* to words: (quasi-)logical formulas (or ADTs)
- a small set of **inference rules**
  ways to combine and reduce *expressions* based on their categories

Categorial Grammars 101

Many variations: TLG, ACG, CCG, . . . (*CG)

common points

- **Lexicalized**
  words come packed with their combinatorics
- **Formal**
  proximal to logics, type theory & functional programming
- **Transparent**
  neat syntax-semantics interface

Categorial Grammars 101

Many variations: TLG, ACG, CCG, . . . (*CG)

divergences

different background logics $\implies$

- different linguistic aspects captured
  e.g. surface order, non-local syntax, dependency relations
- different parsing complexity
- different computational semantics
- . . .

Categorial Grammars 101

but! the **parsing pipeline** is always the same

given an input sentence:

1. Assign a category to each word
2. Build the syntactic derivation bottom-up
3. ???
4. Profit

Supertagging: the task

For some input sentence $w_1, \ldots w_n$ find the category assignment $c_1, \ldots c_n$ s.t.

$$argmax_{(c_1, \ldots c_n)} \, p(c_1, \ldots c_n \mid w_1, \ldots w_n)^*$$

Supertagging: the task

For some input sentence $w_1, \ldots w_n$ find the category assignment $c_1, \ldots c_n$ s.t.

$$argmax_{(c_1,\ldots c_n)}\ p(c_1, \ldots c_n \mid w_1, \ldots w_n)^*$$

* *In practice*:
build the best statistical model possible given current technology and available data

Supertagging, traditionally

$p(c_1, \ldots c_n \mid w_1, \ldots w_n) \approx$

- $\prod_i^n (c_i \mid w_i)$
  co-occurrence-based statistical models (90s)
- $\prod_i^n (c_i \mid w_{i-\kappa} \ldots w_{i+\kappa})$
  window-based n-gram models (00s), feed-forward networks (early 10s)
- $\prod_i^n (c_i \mid w_1, \ldots w_n)$
  sequence encoders (mid 10s)
- $\prod_i^n (c_i \mid c_1, \ldots c_{i-1}, w_1, \ldots w_n)$
  seq2seq (late 10s)

NP/N	N/N	N	$\boxed{(NP\backslash S)/NP}$	NP/N	N/N	N
The	Turkish	state	fears	the	Kurdish	resistance

Supertagging, traditionally

$p(c_1, \ldots c_n \mid w_1, \ldots w_n) \approx$

- $\prod_i^n (c_i \mid w_i)$
  co-occurrence-based statistical models (90s)
- $\prod_i^n (c_i \mid w_{i-\kappa} \ldots w_{i+\kappa})$
  window-based n-gram models (00s), feed-forward networks (early 10s)
- $\prod_i^n (c_i \mid w_1, \ldots w_n)$
  sequence encoders (mid 10s)
- $\prod_i^n (c_i \mid c_1, \ldots c_{i-1}, w_1, \ldots w_n)$
  seq2seq (late 10s)

NP/N	N/N	N	$\boxed{(NP\backslash S)/NP}$	NP/N	N/N	N
The	Turkish	state	fears	the	Kurdish	resistance

Supertagging, traditionally

$p(c_1, \ldots c_n \mid w_1, \ldots w_n) \approx$

- $\prod_i^n (c_i \mid w_i)$
  co-occurrence-based statistical models (90s)
- $\prod_i^n (c_i \mid w_{i-\kappa} \ldots w_{i+\kappa})$
  window-based n-gram models (00s), feed-forward networks (early 10s)
- $\prod_i^n (c_i \mid w_1, \ldots w_n)$
  sequence encoders (mid 10s)
- $\prod_i^n (c_i \mid c_1, \ldots c_{i-1}, w_1, \ldots w_n)$
  seq2seq (late 10s)

NP/N	N/N	N	$\boxed{(NP\backslash S)/NP}$	NP/N	N/N	N
The	Turkish	state	fears	the	Kurdish	resistance

Supertagging, traditionally

$p(c_1, \ldots c_n \mid w_1, \ldots w_n) \approx$

- $\prod_i^n (c_i \mid w_i)$
  co-occurrence-based statistical models (90s)
- $\prod_i^n (c_i \mid w_{i-\kappa} \ldots w_{i+\kappa})$
  window-based n-gram models (00s), feed-forward networks (early 10s)
- $\prod_i^n (c_i \mid w_1, \ldots w_n)$
  sequence encoders (mid 10s)
- $\prod_i^n (c_i \mid c_1, \ldots c_{i-1}, w_1, \ldots w_n)$
  seq2seq (late 10s)

NP/N	N/N	N	$\boxed{(NP\backslash S)/NP}$	NP/N	N/N	N
The	Turkish	state	fears	the	Kurdish	resistance

Supertagging, traditionally

$p(c_1, \ldots c_n \mid w_1, \ldots w_n) \approx$

- $\prod_i^n (c_i \mid w_i)$
  co-occurrence-based statistical models (90s)
- $\prod_i^n (c_i \mid w_{i-\kappa} \ldots w_{i+\kappa})$
  window-based n-gram models (00s), feed-forward networks (early 10s)
- $\prod_i^n (c_i \mid w_1, \ldots w_n)$
  sequence encoders (mid 10s)
- $\prod_i^n (c_i \mid c_1, \ldots c_{i-1}, w_1, \ldots w_n)$
  seq2seq (late 10s)

NP/N	N/N	N	$\boxed{(NP\backslash S)/NP}$	NP/N	N/N	N
The	Turkish	state	fears	the	Kurdish	resistance

Supertagging, traditionally

$p(c_1, \dots c_n \mid w_1, \dots w_n) \approx$

- $\prod_i^n (c_i \mid w_i)$
  co-occurrence-based statistical models (90s)
- $\prod_i^n (c_i \mid w_{i-\kappa} \dots w_{i+\kappa})$
  window-based n-gram models (00s), feed-forward networks (early 10s)
- $\prod_i^n (c_i \mid w_1, \dots w_n)$
  sequence encoders (mid 10s)
- $\prod_i^n (c_i \mid c_1, \dots c_{i-1}, w_1, \dots w_n)$
  seq2seq (late 10s)

in sum

- domain generalization
- wider receptive field
- what about the co-domain?

Intermezzo: the curse(?) of sparsity

The majority of unique categories in common datasets are rare

the "*fix*": ignore rare categories

- small penalty in accuracy
- less so for coverage..
- meta: sparse grammars = bad

the **fix**: decompose categories & build them up during decoding

- ↯ unlimited ~~power~~ generalization
- meta: sparse grammars = ok

Intermezzo: the curse(?) of sparsity

The majority of unique categories in common datasets are rare

the *"fix"*: ignore rare categories

- small penalty in accuracy
- less so for coverage..
- meta: sparse grammars = bad

the **fix**: decompose categories & build them up during decoding

- ϟ unlimited ~~power~~ generalization
- meta: sparse grammars = ok

Intermezzo: the curse(?) of sparsity

The majority of unique categories in common datasets are rare

the "*fix*": ignore rare categories

- small penalty in accuracy
- less so for coverage..
- meta: sparse grammars = bad

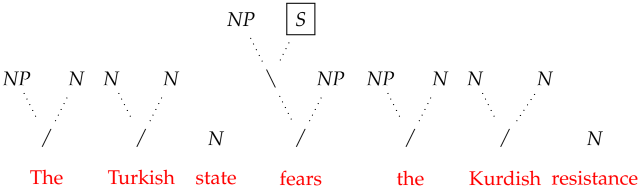
the **fix**: decompose categories & build them up during decoding

- ↯ unlimited ~~power~~ generalization
- meta: sparse grammars = ok

Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

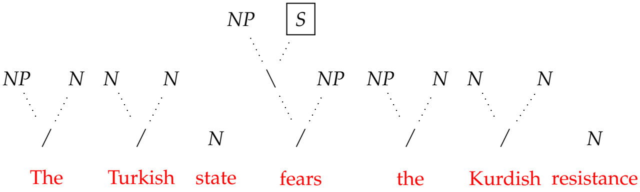
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

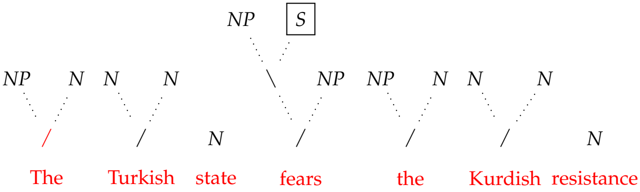
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

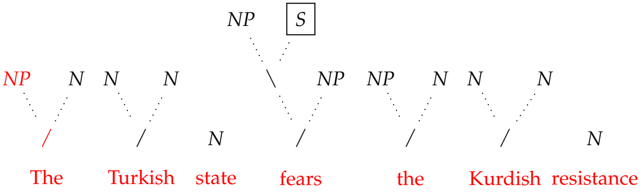
- $\prod_i^m(\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m(\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

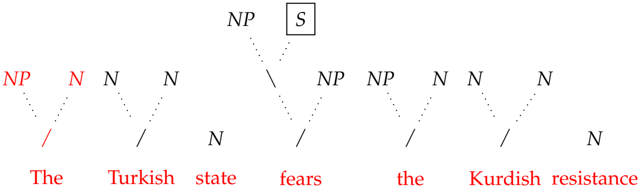
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

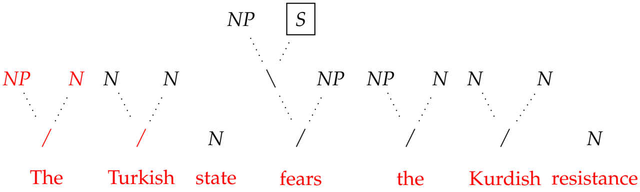
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

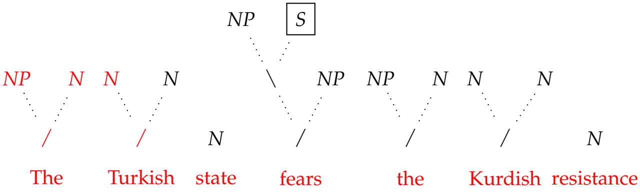
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

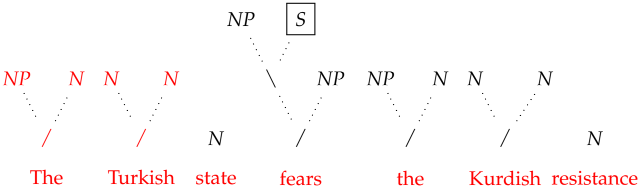
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

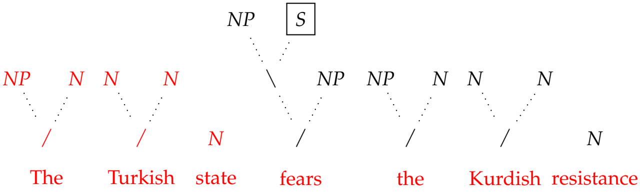
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

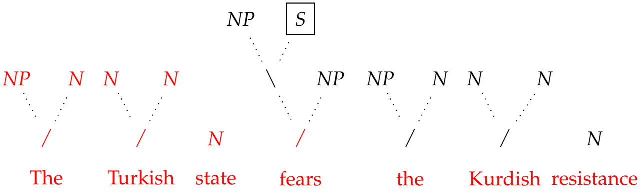
- $\prod_i^m(\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m(\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

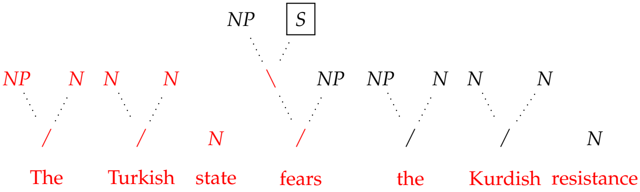
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

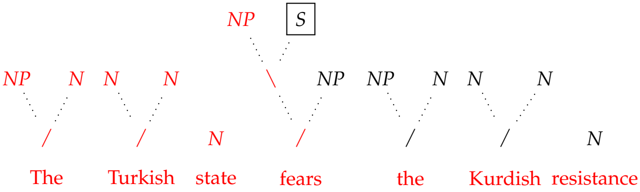
- $\prod_i^m(\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m(\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

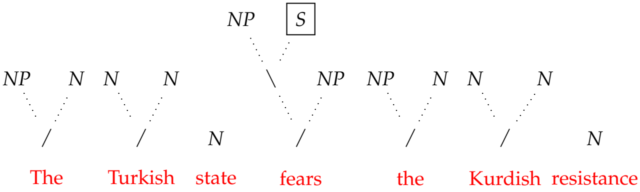
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \dots \sigma_m \mid w_1, \dots w_n) \approx$

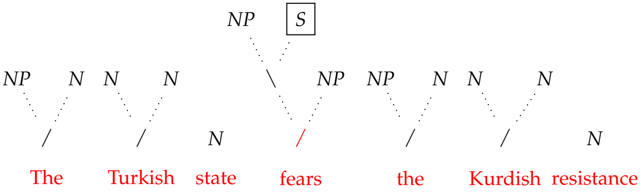
- $\prod_i^m (\sigma_i \mid \sigma_1, \dots \sigma_{i-1}, w_1, \dots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \text{anc}(\sigma_i), w_1, \dots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

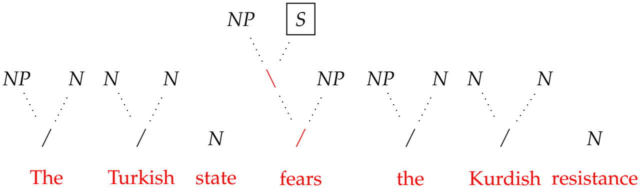
- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \text{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \dots \sigma_m \mid w_1, \dots w_n) \approx$

- $\prod_i^m (\sigma_i \mid \sigma_1, \dots \sigma_{i-1}, w_1, \dots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \dots w_n)$
  tree-recursive (Prange et. al 2020)



Supertagging, constructively

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx$

- $\prod_i^m (\sigma_i \mid \sigma_1, \ldots \sigma_{i-1}, w_1, \ldots w_n)$
  sequential constructive (w/ Moortgat & Deoskar, 2019)
- $\prod_i^m (\sigma_i \mid \mathrm{anc}(\sigma_i), w_1, \ldots w_n)$
  tree-recursive (Prange et. al 2020)

in sum

output structure	sequence-like	tree-like
context	☺ global	☹ local
complexity	☹ quadratic	☺ constant
treeness	☹ implicit, learned	☺ explicit, captured
sequenceness	☹ misaligned	☹ ignored

A fresh perspective

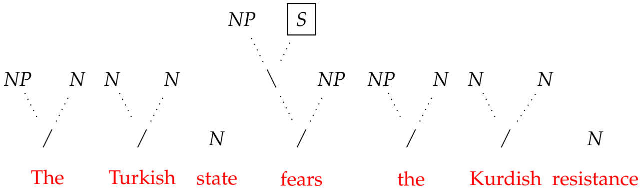
neither sequence nor tree but sequence of trees

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx \prod_i^m (\sigma_i \mid \sigma_j : \text{depth}(\sigma_j) < \text{depth}(\sigma_i), w_1, \ldots w_n)$

A fresh perspective

neither sequence nor tree but sequence of trees

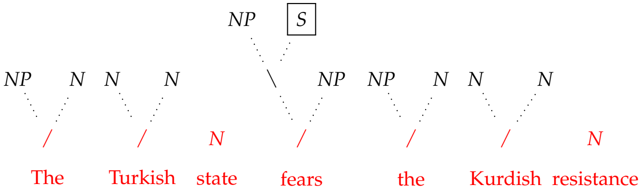
$p(\sigma_1, \dots \sigma_m \mid w_1, \dots w_n) \approx \prod_i^m (\sigma_i \mid \sigma_j : \mathrm{depth}(\sigma_j) < \mathrm{depth}(\sigma_i), w_1, \dots w_n)$



A fresh perspective

neither sequence nor tree but sequence of trees

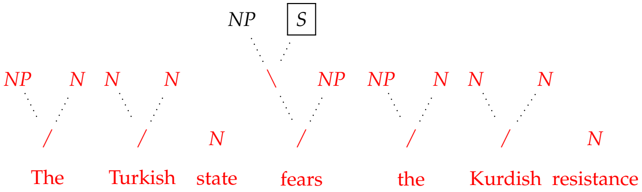
$p(\sigma_1, \dots \sigma_m \mid w_1, \dots w_n) \approx \prod_i^m (\sigma_i \mid \sigma_j : \mathrm{depth}(\sigma_j) < \mathrm{depth}(\sigma_i), w_1, \dots w_n)$



A fresh perspective

neither sequence nor tree but sequence of trees

$p(\sigma_1, \ldots \sigma_m \mid w_1, \ldots w_n) \approx \prod_i^m (\sigma_i \mid \sigma_j : \mathrm{depth}(\sigma_j) < \mathrm{depth}(\sigma_i), w_1, \ldots w_n)$



Implementation: dynamic graph convolutions

1 decoding step per tree depth; 3 message-passing rounds per step

- *contextualize: states* $\rightarrow$ *states*
  universal transformer encoder w/ relative weights
  (many-to-many, update states with neighborhood context)
- *predict: state* $\rightarrow$ *nodes*
  token classification w/ dynamic tree embeddings
  (one-to-many, predict fringe nodes from current state)
- *feedback: nodes* $\rightarrow$ *state*
  heterogeneous graph attention
  (many-to-one, update state with last predicted nodes)

Table with numbers

model	overall	frequent	uncommon	rare	unseen
	accuracy (%)				
CCGbank *(Combinatory Categorial Grammar, en)*					
Sequential RNN	95.10	95.48	65.76	26.02	0.00
Tree Recursive	96.09	96.44	68.10	$\boxed{37.40}$	3.03
Attentive Convolutions	96.25	$\boxed{96.64}$	71.04	–	–
this work	$\boxed{96.29}$	96.61	$\boxed{72.06}$	34.45	$\boxed{4.55}$
CCGrebank *(ditto, improved version)*					
Sequential RNN	94.44	94.93	66.90	27.41	1.23
Tree Recursive	94.70	95.11	68.86	36.76	$\boxed{4.94}$
this work	$\boxed{95.07}$	$\boxed{95.45}$	$\boxed{71.40}$	$\boxed{37.19}$	3.70
TLGBank *(Lambek calculus & control modalities, fr)*					
ELMo LSTM	93.20	95.10	75.19	25.85	–
this work	$\boxed{95.93}$	$\boxed{96.40}$	$\boxed{81.48}$	$\boxed{55.37}$	$\boxed{7.26}$
Æthel *(van Benthem calculus & dependency modalities, nl)*					
Sequential Transformer	83.67	84.55	64.70	50.58	$\boxed{24.55}$
this work	$\boxed{93.67}$	$\boxed{94.72}$	$\boxed{73.45}$	$\boxed{53.83}$	15.78

What of it

model

- ☺ global context
- ☺ constant decoding
- ☺ input/output alignment
- ☺ explicit tree structures

sparsity.. a *friend*?

- more rare cats $\implies$ better acquisition of rare cats
- cascading effect on performance

todo

- beam search: open problem
- parser integration

What of it

model

- ☺ global context
- ☺ constant decoding
- ☺ input/output alignment
- ☺ explicit tree structures

sparsity.. a *friend*?

- more rare cats $\implies$ better acquisition of rare cats
- cascading effect on performance

todo

- beam search: open problem
- parser integration

What of it

model

- ☺ global context
- ☺ constant decoding
- ☺ input/output alignment
- ☺ explicit tree structures

sparsity.. a *friend*?

- more rare cats $\implies$ better acquisition of rare cats
- cascading effect on performance

todo

- beam search: open problem
- parser integration

thanks!

arXiv (includes: mathy equations, more and bigger tables!)
abs/2203.12235

github (includes: mostly working code! be the first to star!)
konstantinosKokos/dynamic-graph-supertagging

Boycott EMNLP'22