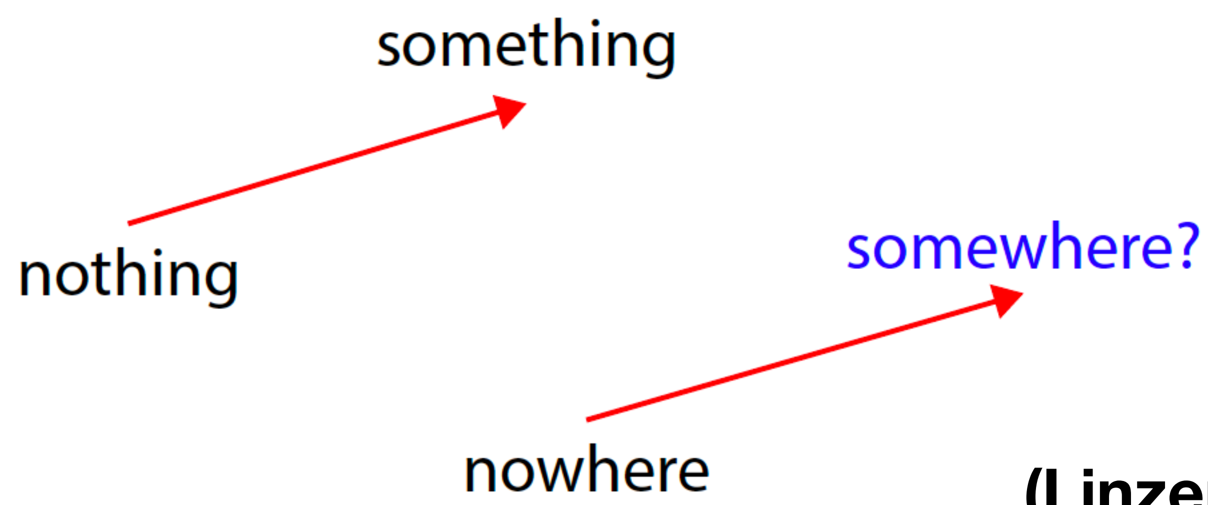
How well do neural NLP systems generalize?

Tal Linzen
Department of Cognitive Science
Johns Hopkins University

Intrinsic evaluation of word embeddings



(Linzen et al. 2016, *SEM)

Issues in evaluating semantic spaces using word analogies

Tal Linzen
LSCP & IJN
École Normale Supérieure
PSL Research University
tal.linzen@ens.fr

(Linzen 2016 , RepEval)

Neural networks are good at language modeling (among other things)

The boys went outside to _____

$$\hat{P}(w_n = w^k \,|\, w_1, \ldots, w_{n-1})$$

Model	Test Perplexity	Number of Params [billions]
Sigmoid-RNN-2048 (Ji et al., 2015a)	68.3	4.1
Interpolated KN 5-gram, 1.1B n-grams (Chelba et al., 2013)	67.6	1.76
Sparse Non-Negative Matrix LM (Shazeer et al., 2015)	52.9	33
RNN-1024 + MaxEnt 9-gram features (Chelba et al., 2013)	51.3	20
LSTM-512-512	54.1	0.82
LSTM-1024-512	48.2	0.82
LSTM-2048-512	43.7	0.83
LSTM-8192-2048 (No Dropout)	37.9	3.3
LSTM-8192-2048 (50% Dropout)	32.2	3.3
2-Layer LSTM-8192-1024 (BIG LSTM)	30.6	1.8
BIG LSTM+CNN Inputs	**30.0**	**1.04**

(Jozefowicz et al., 2016)

Linguistically targeted evaluation

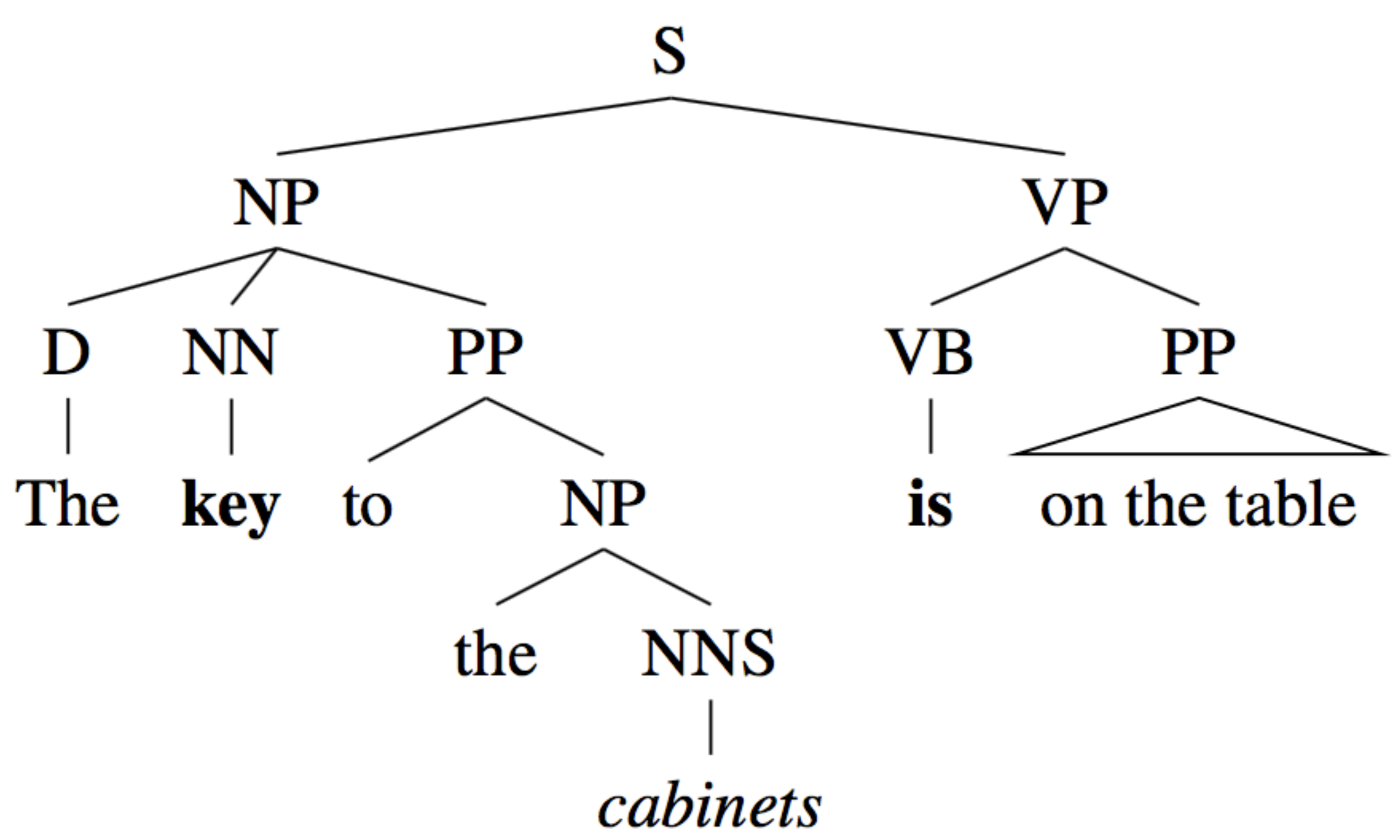
- Average metrics (such as perplexity) are primarily affected by frequent phenomena: those are often very simple
- Effective word prediction on the average case can be due to collocations, semantics, syntax… Is the model capturing all of these?
- How does the model **generalize** to (potentially infrequent) cases that probe a particular linguistic ability?
- Behavioral evaluation of a system as a whole rather than of individual vector representations

Outline

1. Syntactic evaluation of language models
2. Do recurrent neural network language models show human-like syntactic generalization?
3. Syntactic generalization in natural language inference
4. Bonus: measuring compositionality in neural network vector representations

Syntactic evaluation with subject-verb agreement

The **key** to the **cabinets** **is** on the table.

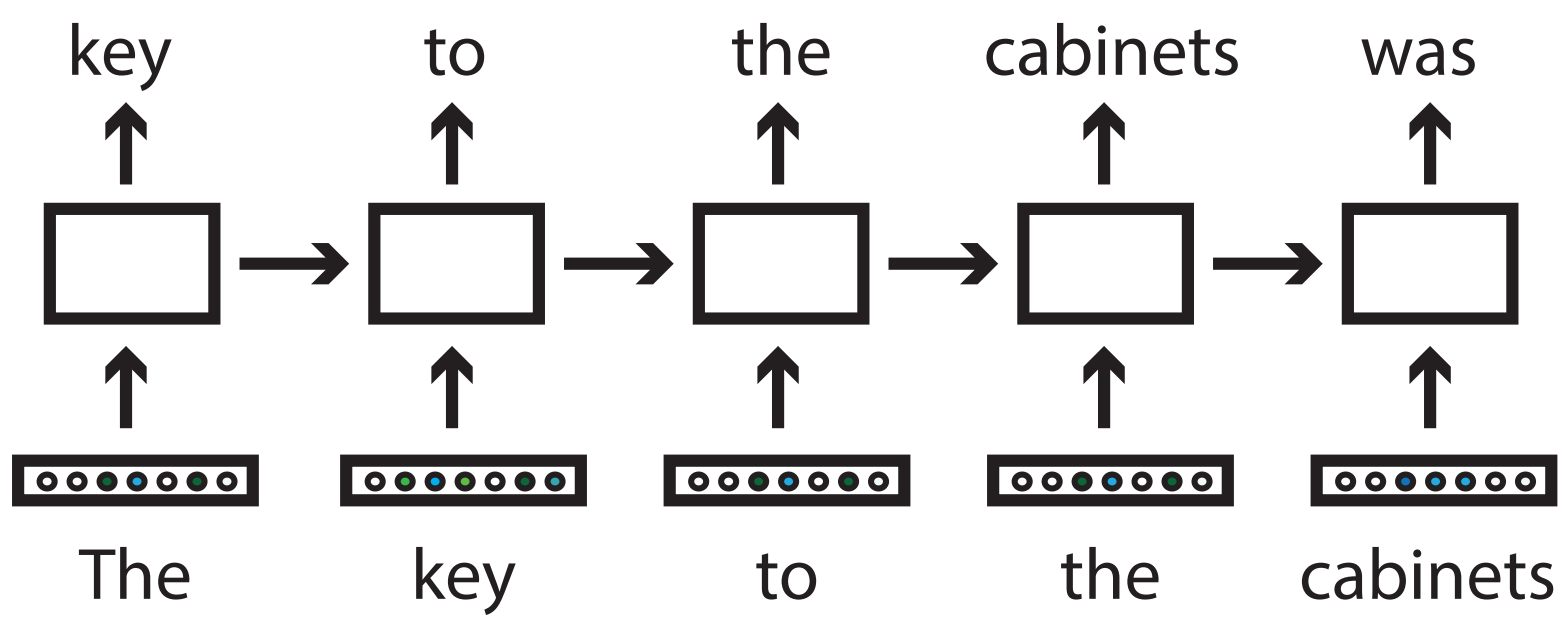


Probing syntactic representations using the number prediction task

- The length of the forewings… **SINGULAR**
- The keys to the cabinets… **PLURAL**

(Bock & Miller, 1991; Elman, 1991)

Evaluating syntactic predictions in a language model



- *The key to the cabinets….* P(*was*) > P(*were*)?

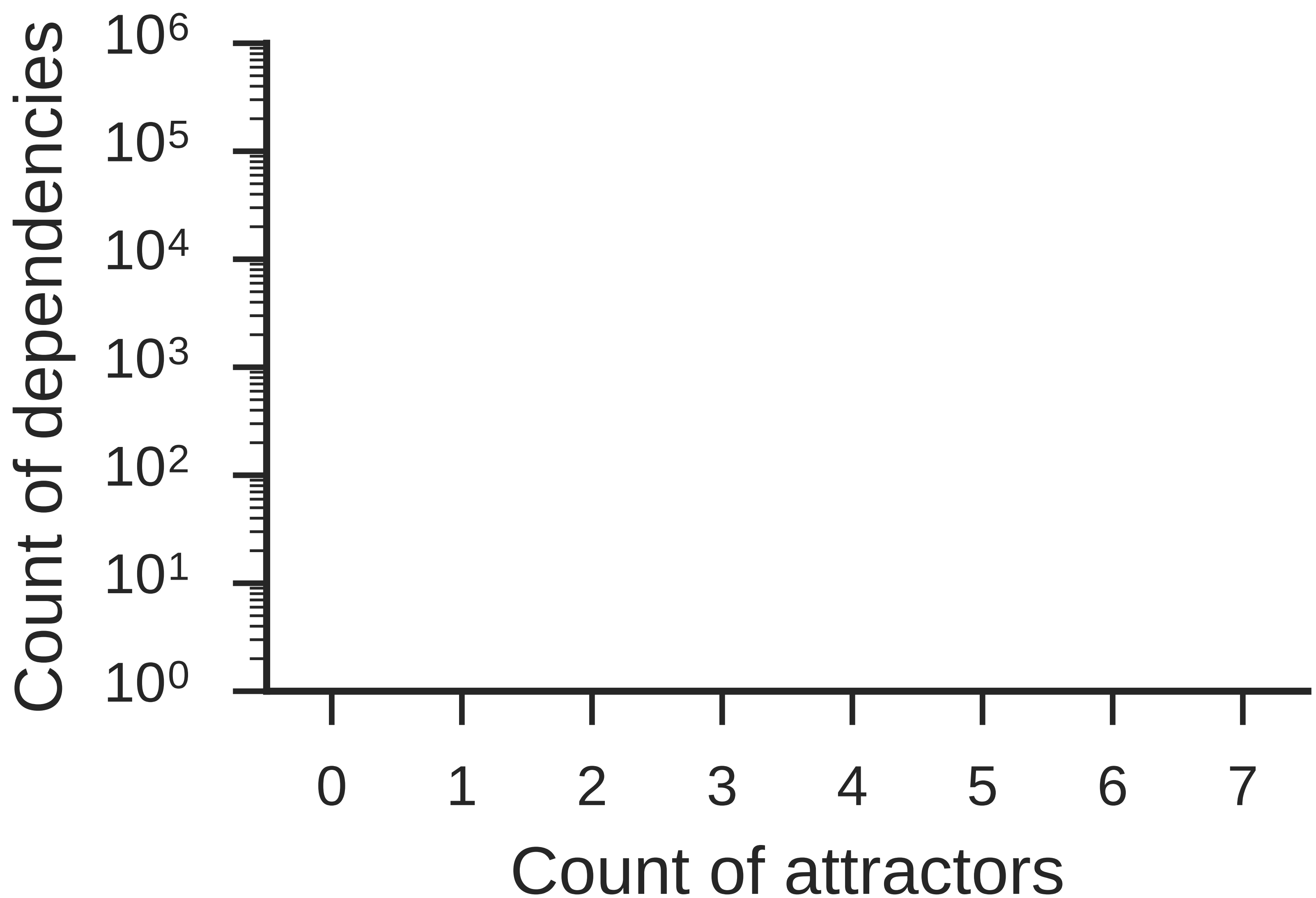
(Linzen, Dupoux & Goldberg, 2016, *TACL*)

Most sentences are simple; focus on dependencies with attractors

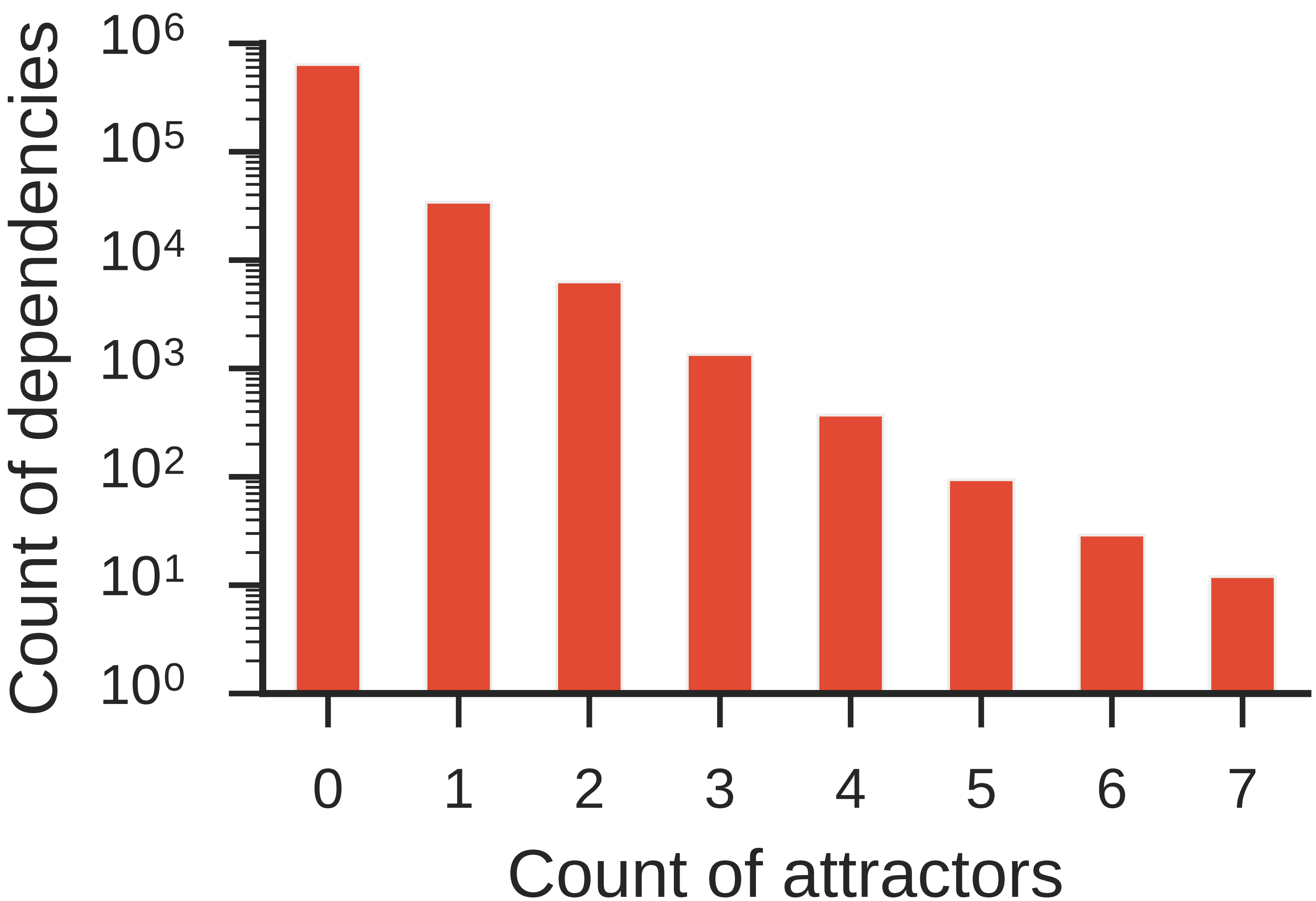
- The **keys are** rusty.
- The **keys** to the cabinet **are** rusty.
- The **ratio** of men to women **is** not clear.
- The **ratio** of men to women and children **is** not clear.
- ~~The **keys** to the cabinets **are** rusty.~~
- ~~The **keys** to the door and the cabinets **are** rusty.~~
- **Evaluation only: the model is still trained on all sentences!**

RNNs' inductive bias favors short dependencies (recency)!
(Ravfogel, Goldberg & Linzen, 2019, *NAACL*)

Averaging over randomly sampled sentences can lead to overly optimistic conclusions

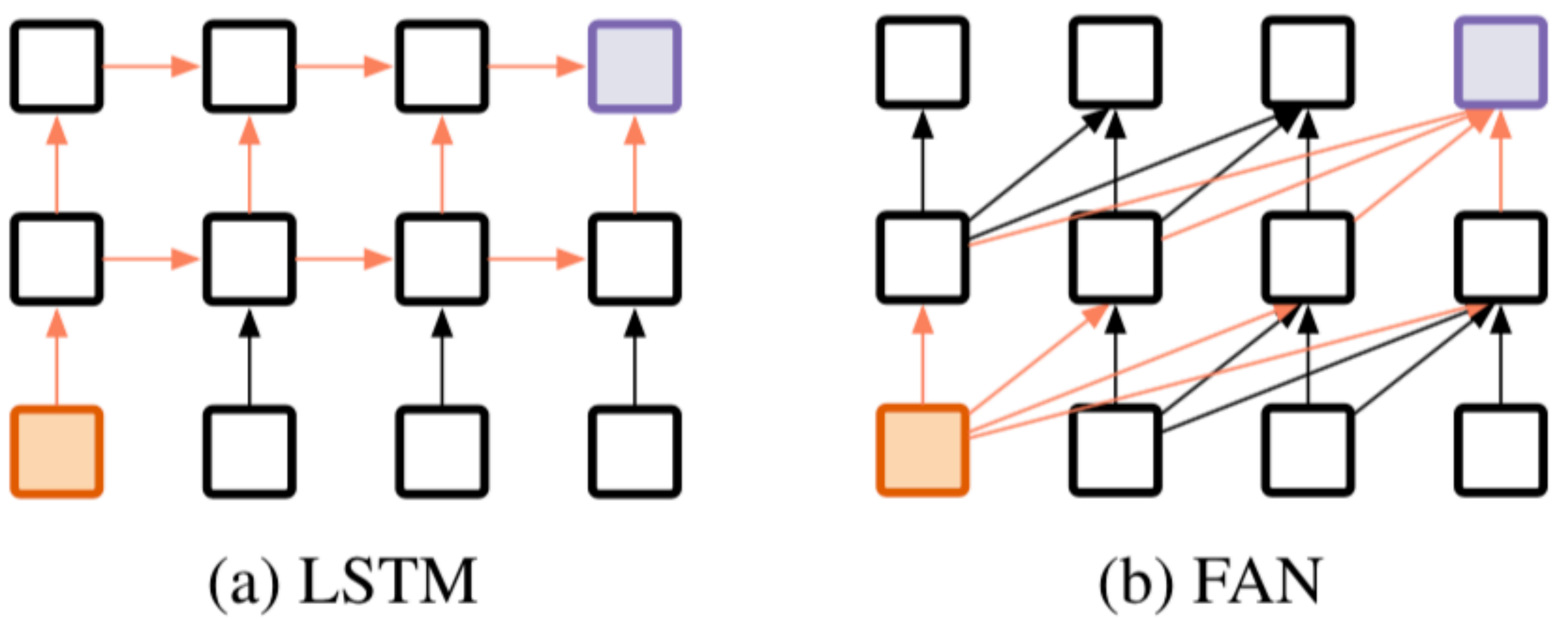


Averaging over randomly sampled sentences can lead to overly optimistic conclusions



Linguistically informed evaluation for model comparison on the long tail

“Transformer”
“Attention is all you need”

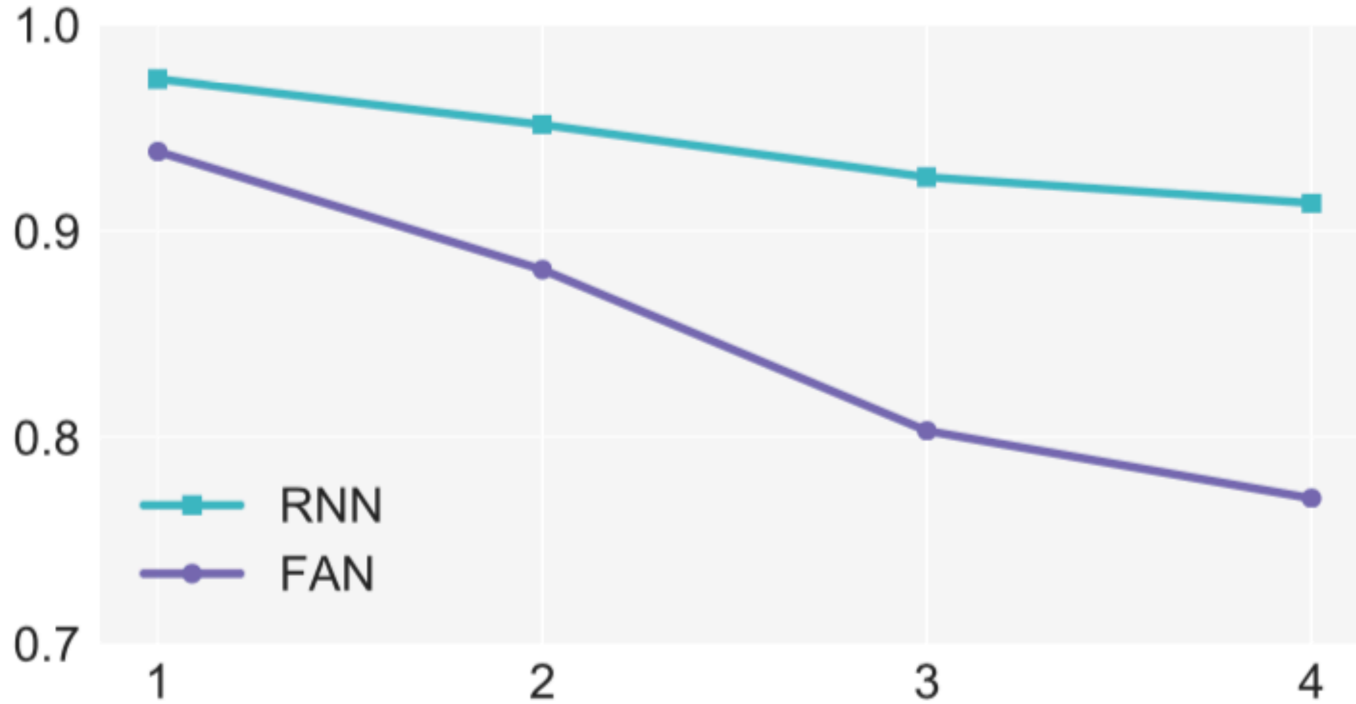


Perplexity: 67.06

Perplexity: 69.14

Tran, Bisazza and Monz (2018), *EMNLP*

Linguistically informed evaluation for model comparison on the long tail



(b) Language model, breakdown by # attractors

FAN shows poorer syntactic performance

Tran, Bisazza and Monz (2018), *EMNLP*

Outline

1. Syntactic evaluation of language models

2. Do recurrent neural network language models show human-like syntactic generalization?

3. Syntactic generalization in natural language inference

4. Bonus: measuring compositionality in neural network vector representations

A controlled syntactic evaluation dataset

- Naturalistic data sets have obvious advantages, but are biased in favor of easy cases, and contain semantic and other confounds

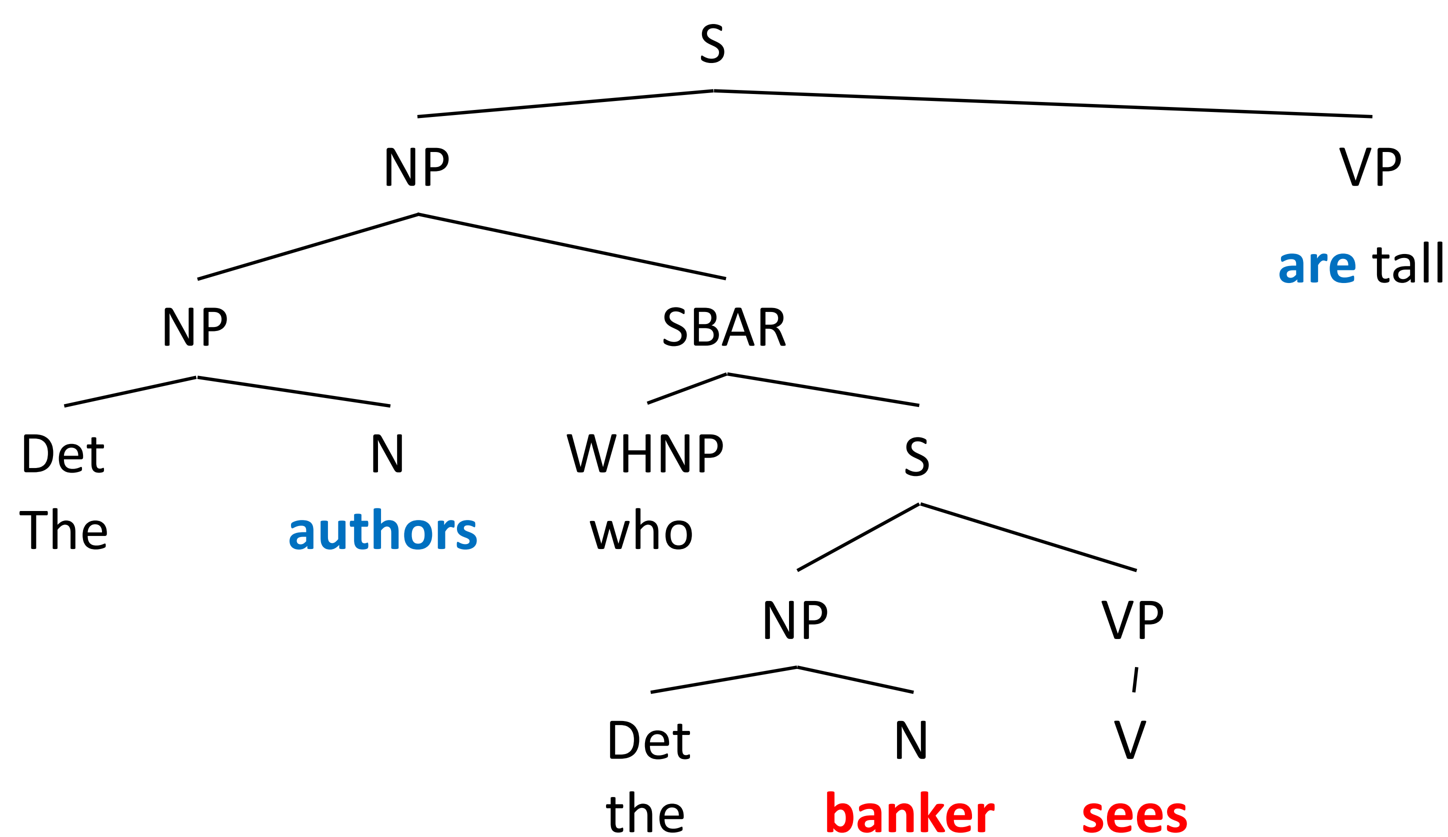
- Not easy to identify the challenging cases that do exist (because of parse errors)

- Counting attractors is a first approximation, but we can do much better by onstructing test sentences ourselves

Agreement across an object relative clause

The **authors** who the **banker** sees **are** tall.

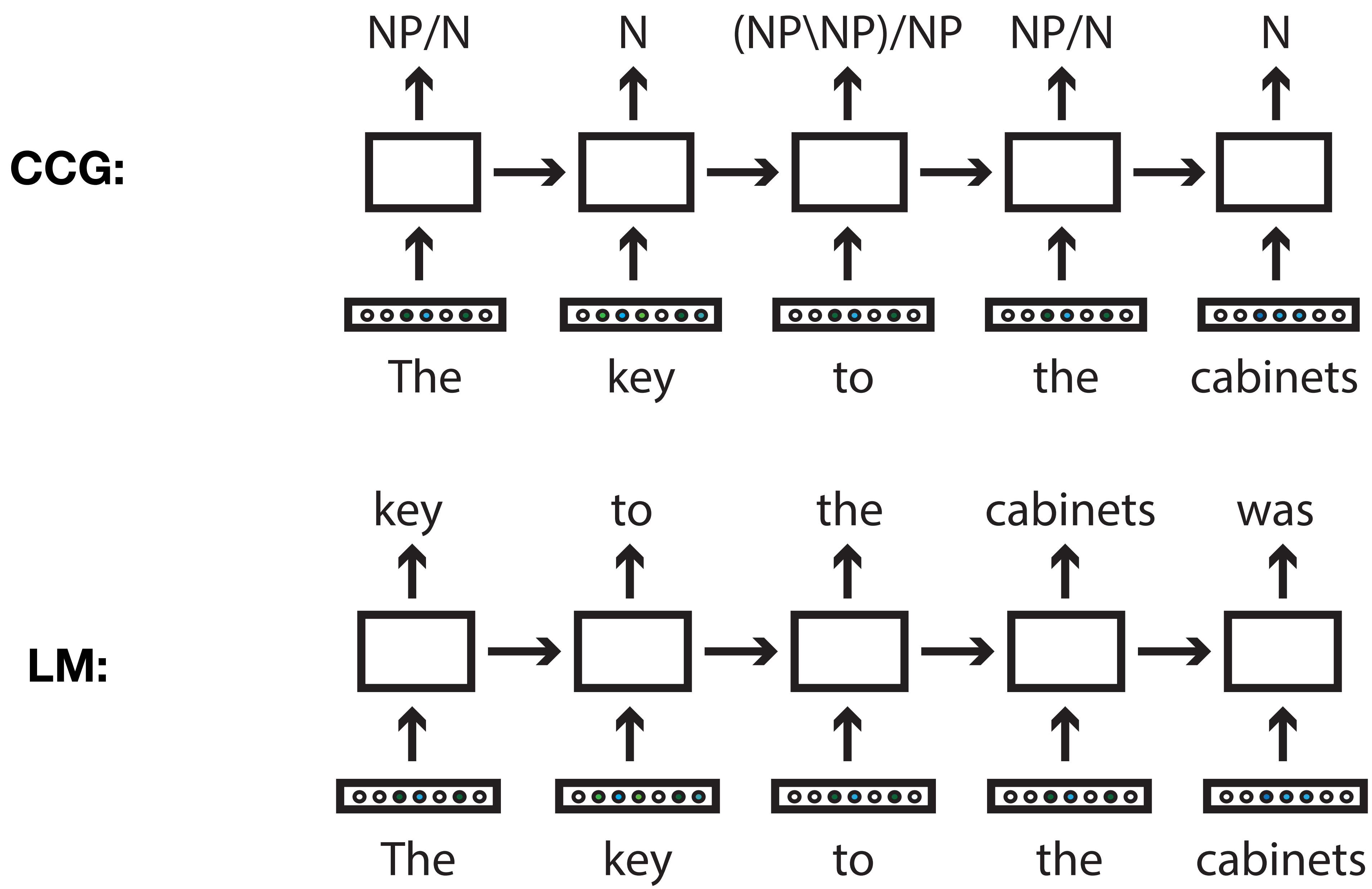
*The **authors** who the **banker** sees **is** tall.



Experimental setup

- Human experiment on Mechanical Turk: which of these two sentences is better?
- Three language models trained on a 90-million word English Wikipedia corpus
  1. Trigram language model
  2. RNN language model: LSTM, 2 layers, 650 units per layer (Gulordava et al. 2018)

3. Multitask CCG/LM

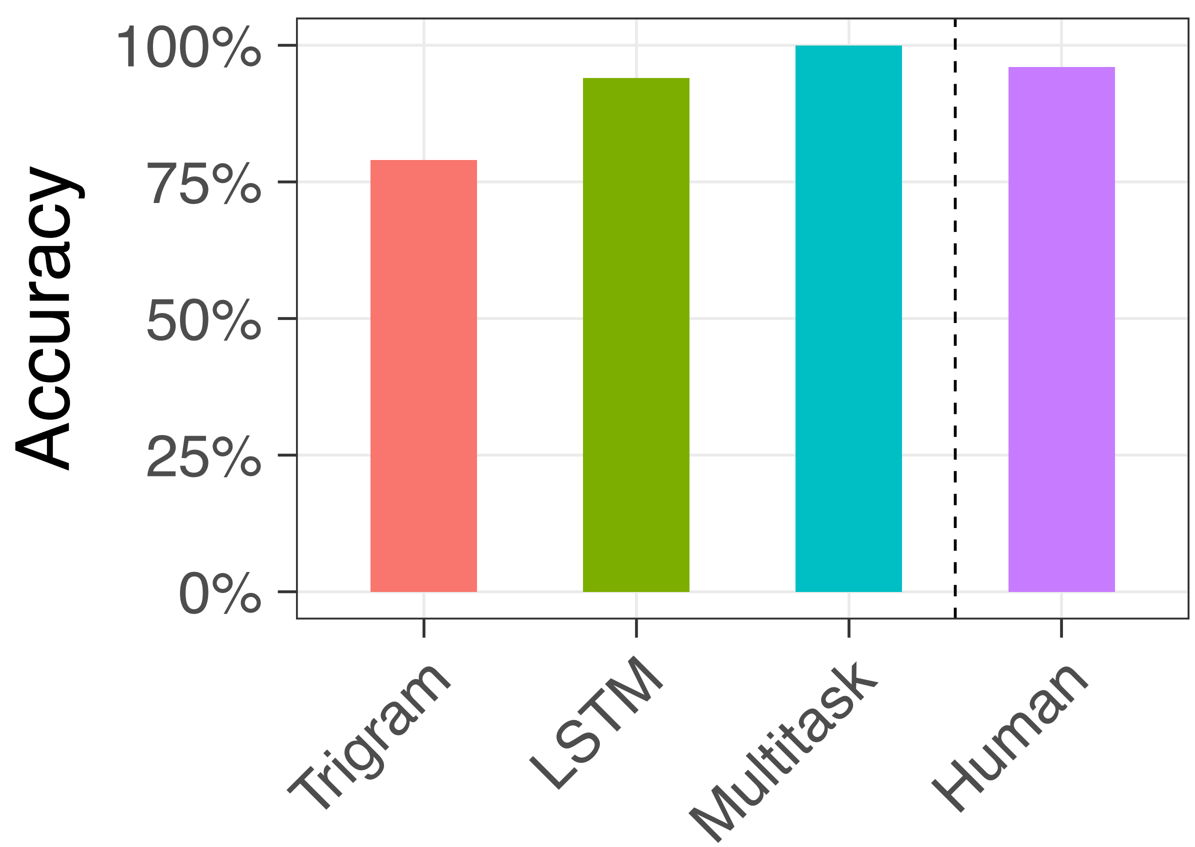


Enguehard, Goldberg & Linzen (2017), *CoNLL*

Agreement in a simple sentence

The **author laughs**.

*The **author laugh**.

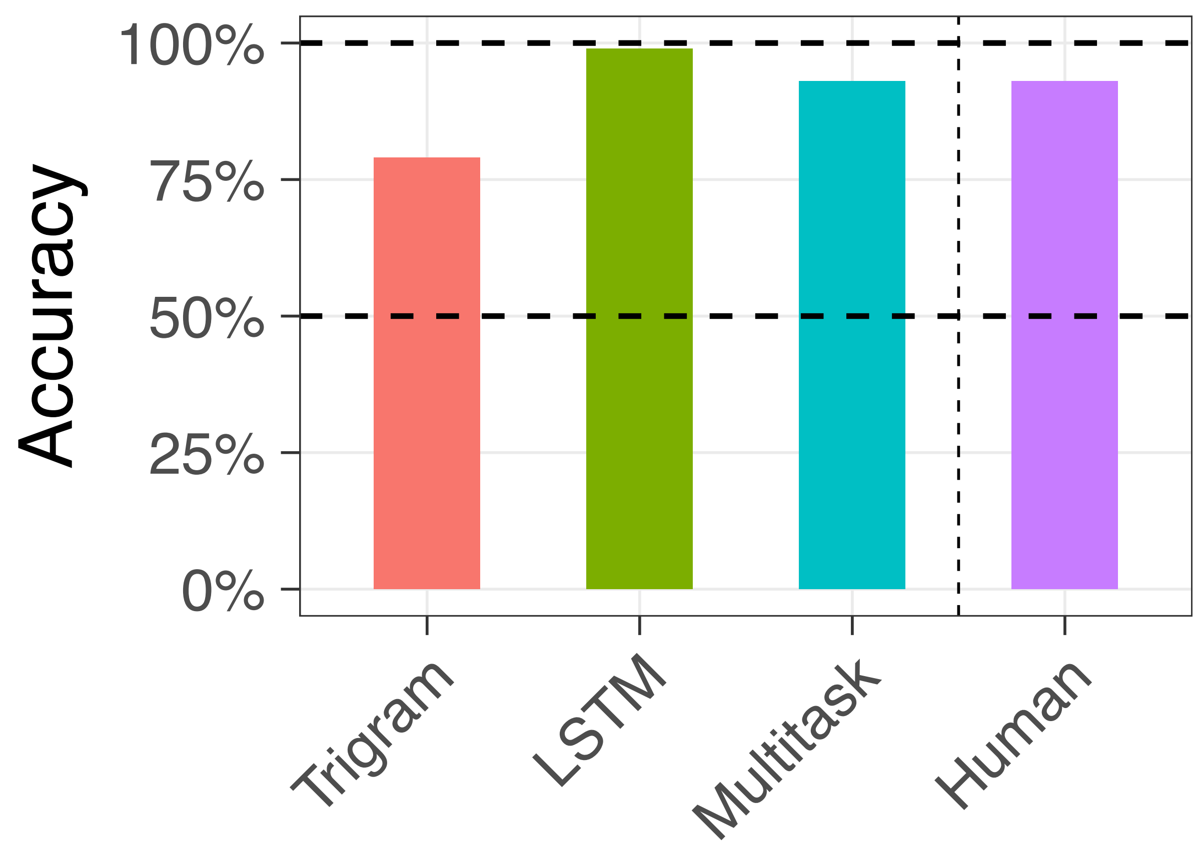


Marvin & Linzen (2018), *EMNLP*

Agreement in a sentential complement

The **mechanics** said the security **guard laughs**.

*The **mechanics** said the security **guard laugh**.



No interference from sentence-initial noun

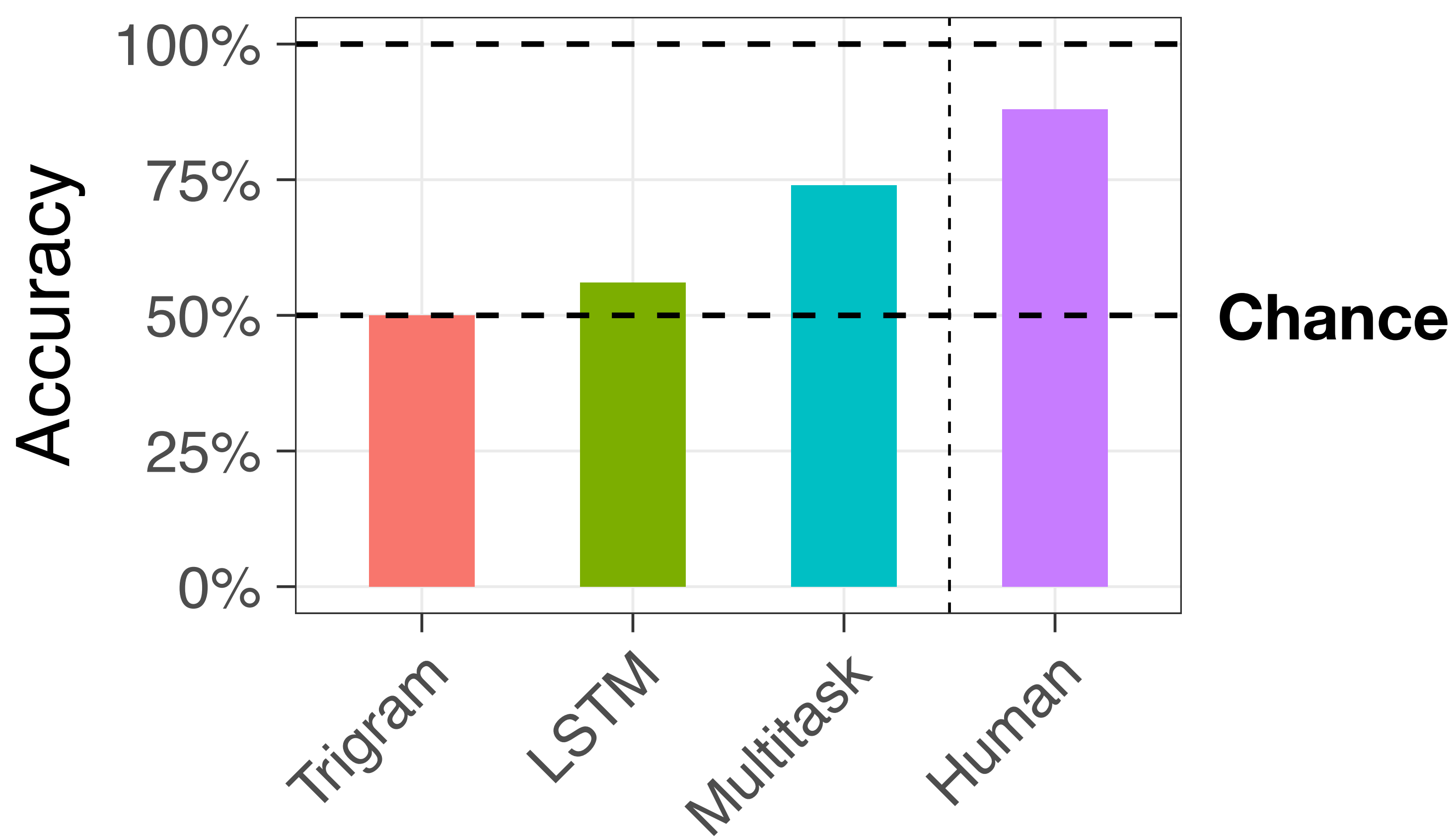
Marvin & Linzen (2018), *EMNLP*

Agreement across a subject relative clause

The **officers** that love the **skater smile**.

*The **officers** that love the **skater smiles**.

Multitask learning with syntax helps!

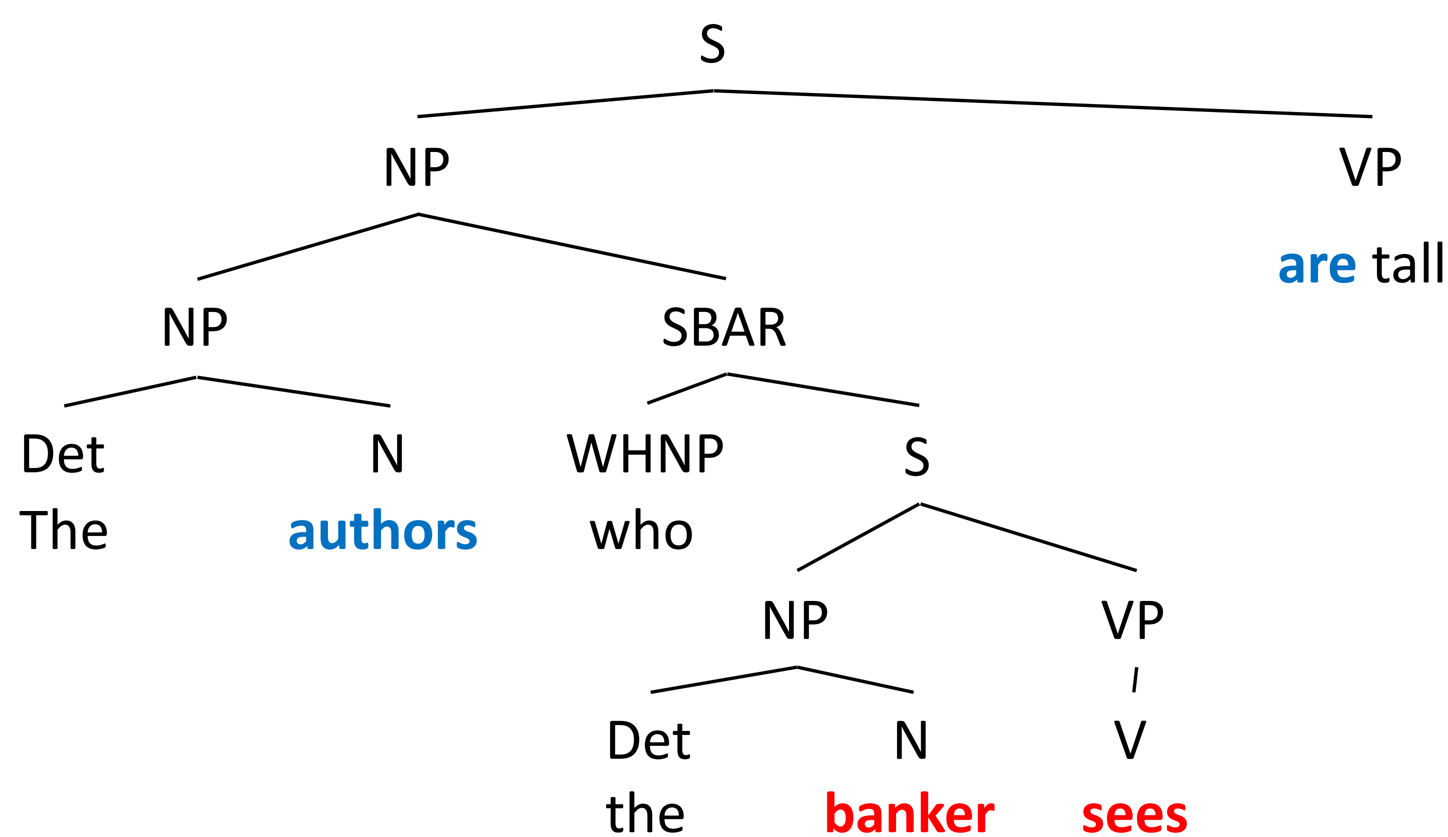


Marvin & Linzen (2018), *EMNLP*

Agreement across an object relative clause

The **authors** who the **banker** sees **are** tall.

*The **authors** who the **banker** sees **is** tall.

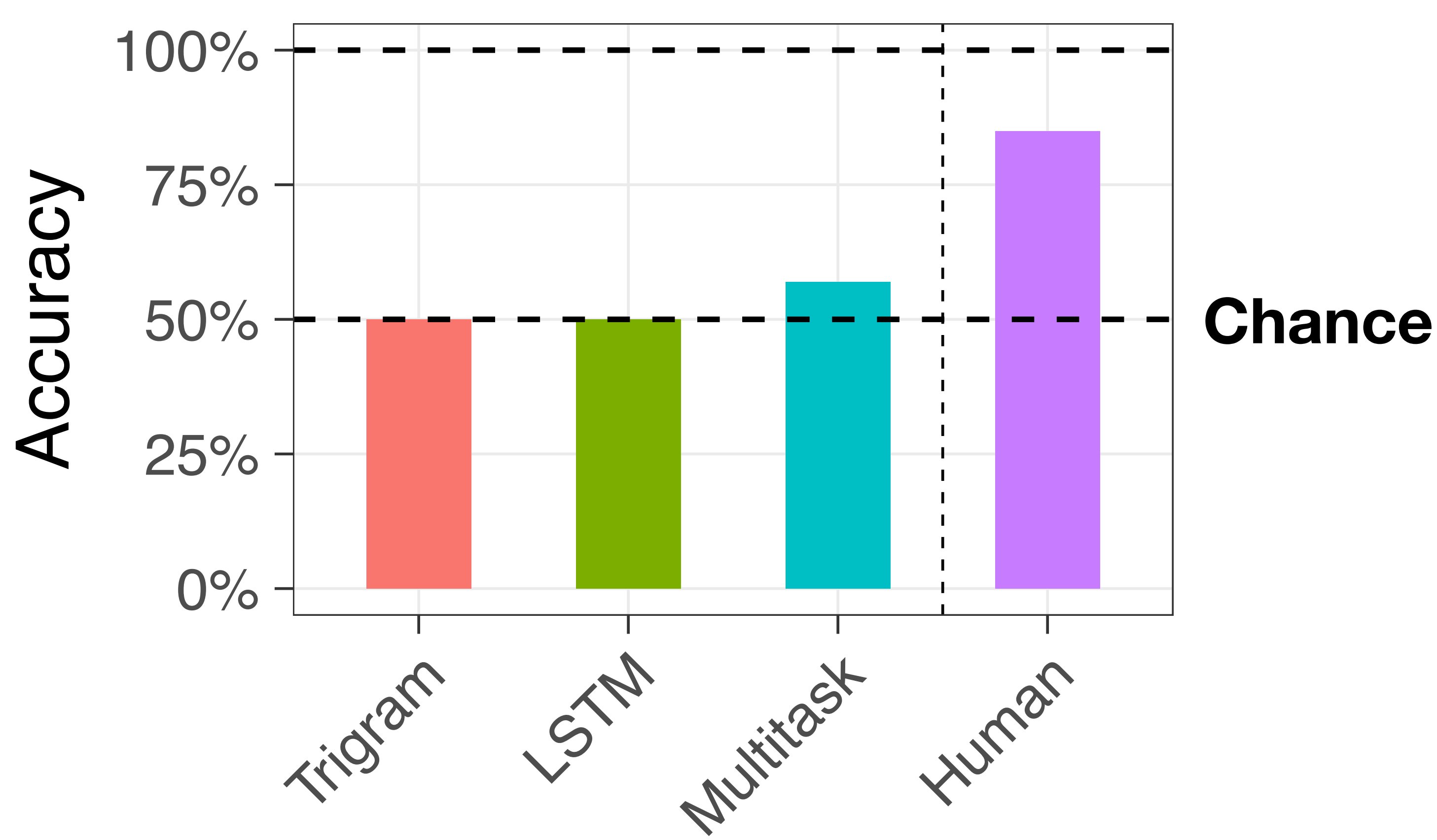


Agreement across an object relative clause

The **authors** who the **banker** sees **are** tall.

*The **authors** who the **banker** sees **is** tall.

Multitask learning with syntax barely helps…

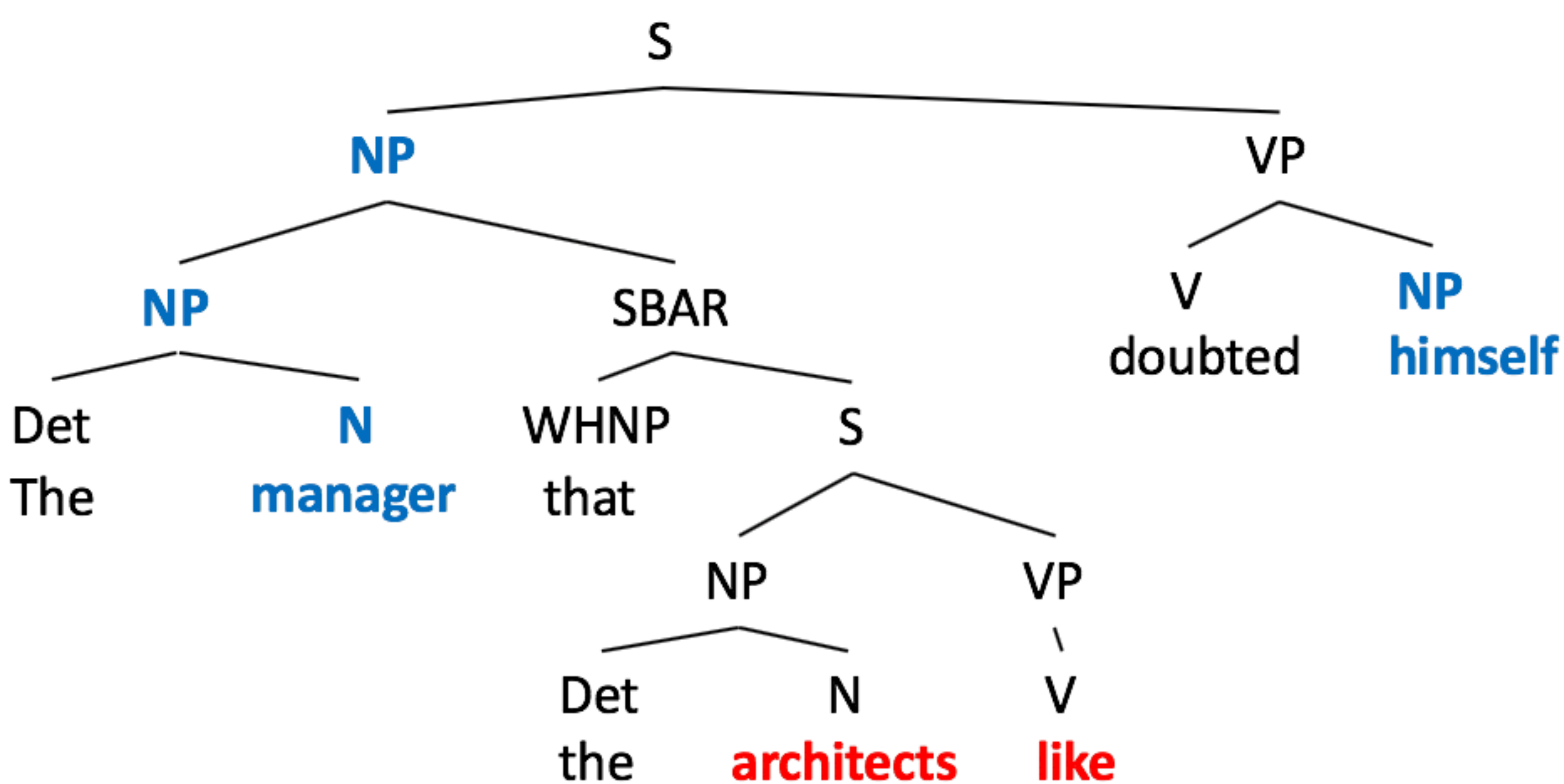


Marvin & Linzen (2018), *EMNLP*

Reflexive anaphora across an object relative clause

The **manager** that the **architects** like doubted **himself**.

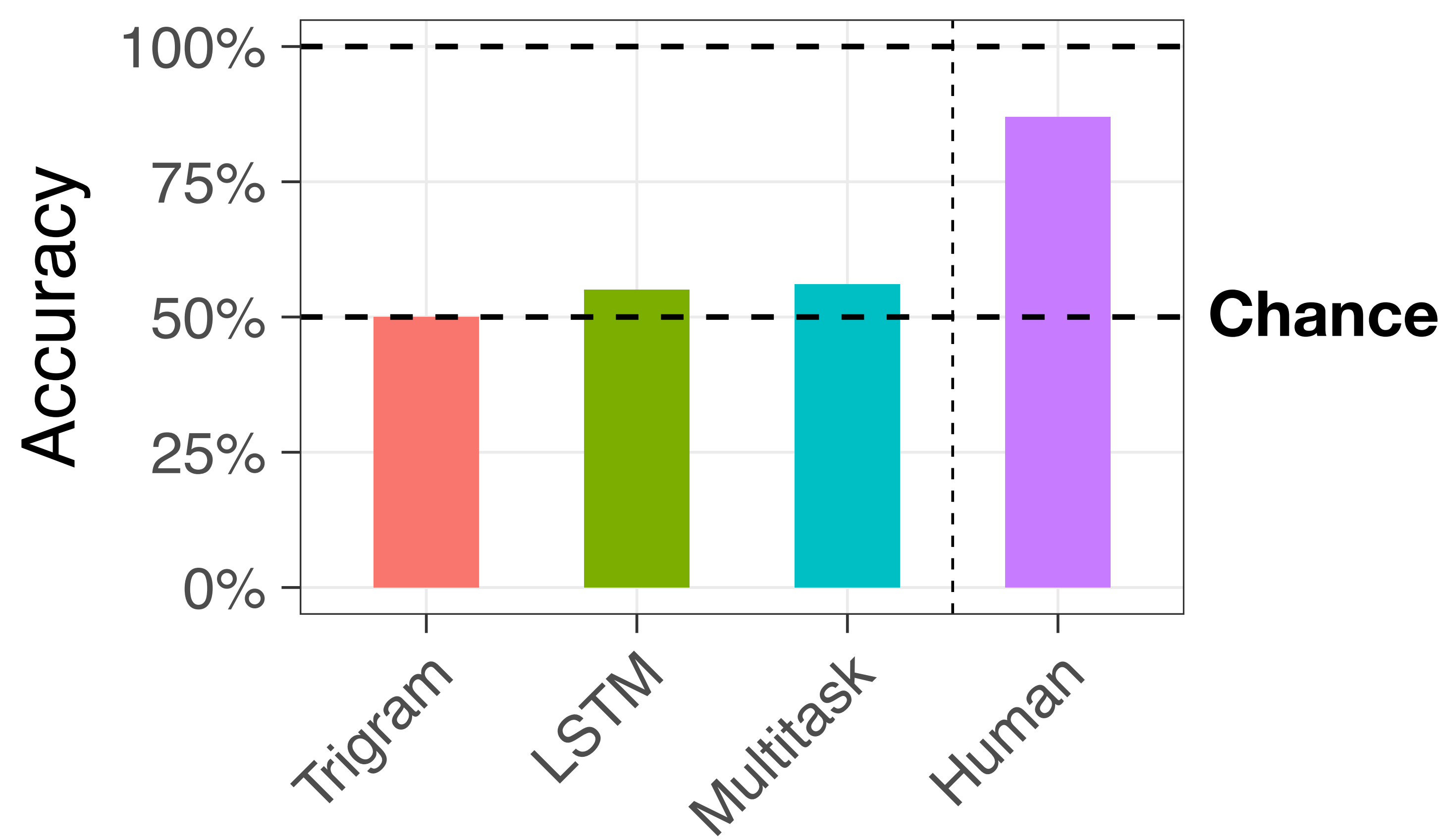
*The **manager** that the **architects** like doubted **themselves**.



Reflexive anaphora across an object relative clause

The **manager** that the **architects** like doubted **himself**.

*The **manager** that the **architects** like doubted **themselves**.



Interim discussion

- Linguistically informed evaluation: sample the evaluation sentences based on syntactic complexity rather than have training and test set come from the same distribution
- Attractors significantly increase error rates
- Still, the RNN overcomes its recency bias: error rate is much less than 100%
- When tested on challenging constructed sentences, the RNN's accuracy approaches 50%
- Explicit syntactic training reduces errors, but not completely (Enguehard, Goldberg & Linzen, 2017)

Outline

1. Syntactic evaluation of language models
2. Do recurrent neural network language models show human-like syntactic generalization?
3. Syntactic generalization in natural language inference
4. Bonus: measuring compositionality in neural network vector representations

Natural language inference

A soccer game with multiple males playing.

Some men are playing a sport.

Entailment

A man inspects the uniform of a figure.

The man is sleeping.

Contradiction

(Dagan et al., 2006; Bowman et al., 2015)

NLI: evaluation

- Trained and tested on datasets such as SNLI and MNLI (Bowman et al. 2016, Williams et al. 2018)
- MNLI: workers generate sentences that follow from or contradict a prompt sentence
- Neural models perform well on MNLI (BERT: 84%)
- Many (most?) “naturally occurring” test cases in MNLI may not require understanding of the sentence (Poliak et al. 2018, Gururangan et al. 2018, Naik et al. 2018, etc.)

HANS (Heuristic Analysis for NLI Systems)

- **Lexical overlap:**

  The judge was paid by the lawyer → The lawyer paid the judge.

- **The subsequence heuristic:**

  The lawyer read the book → The lawyer read.

- **The constituent heuristic:**

  After the lawyer called, the judge arrived. → The judge arrived.

(McCoy, Pavlick & Linzen, 2019, *ACL*)

HANS: Contradicting examples

- **Lexical overlap:**

  The doctor was paid by the actor. $\nrightarrow$ The doctor paid the actor.

- **The subsequence heuristic:**

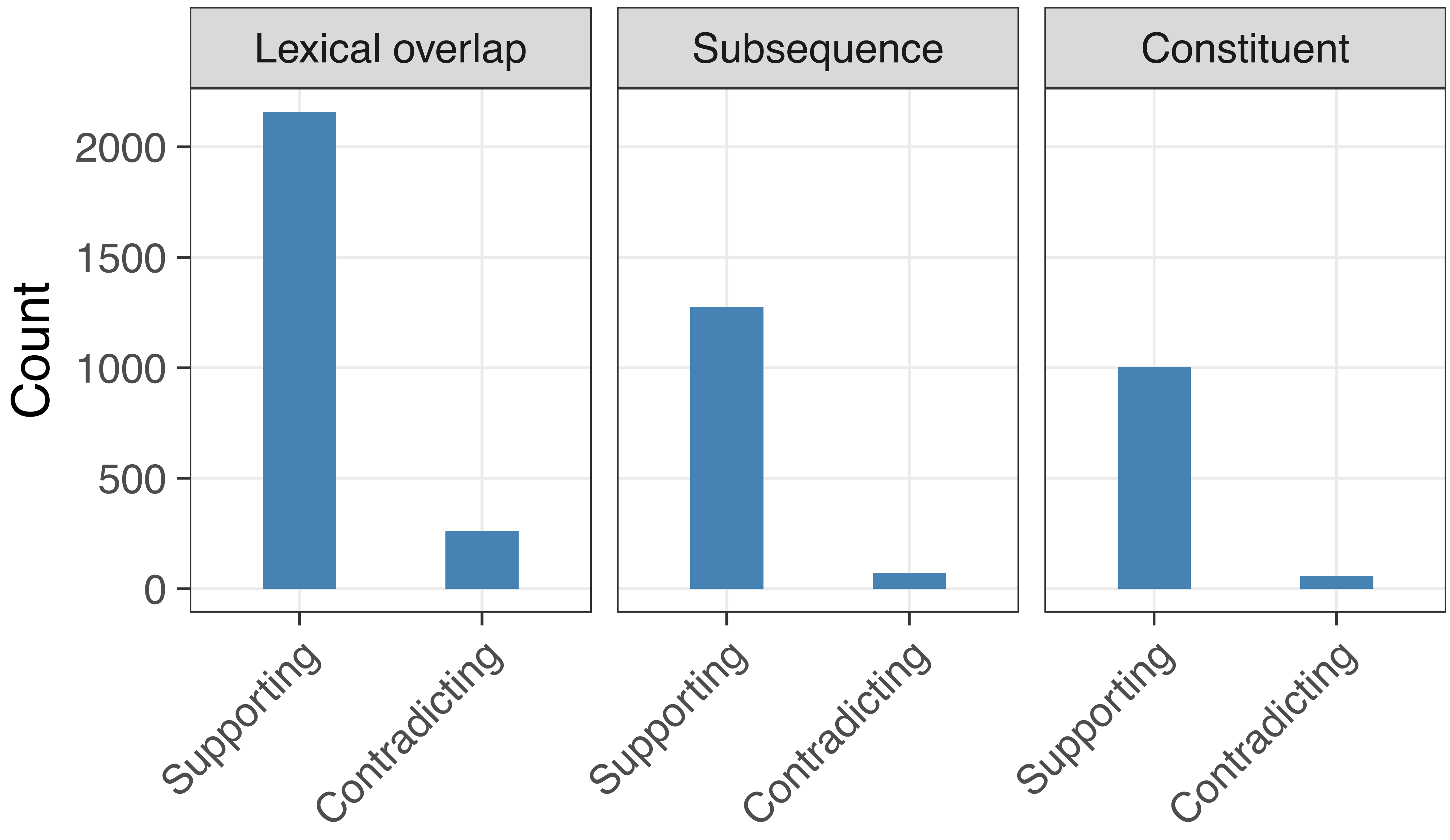
  The lawyer near the doctor danced. $\nrightarrow$ The doctor danced.

- **The constituent heuristic:**

  If the lawyer called, the judge arrived. $\nrightarrow$ The lawyer called.

(McCoy, Pavlick & Linzen, 2019, *ACL*)

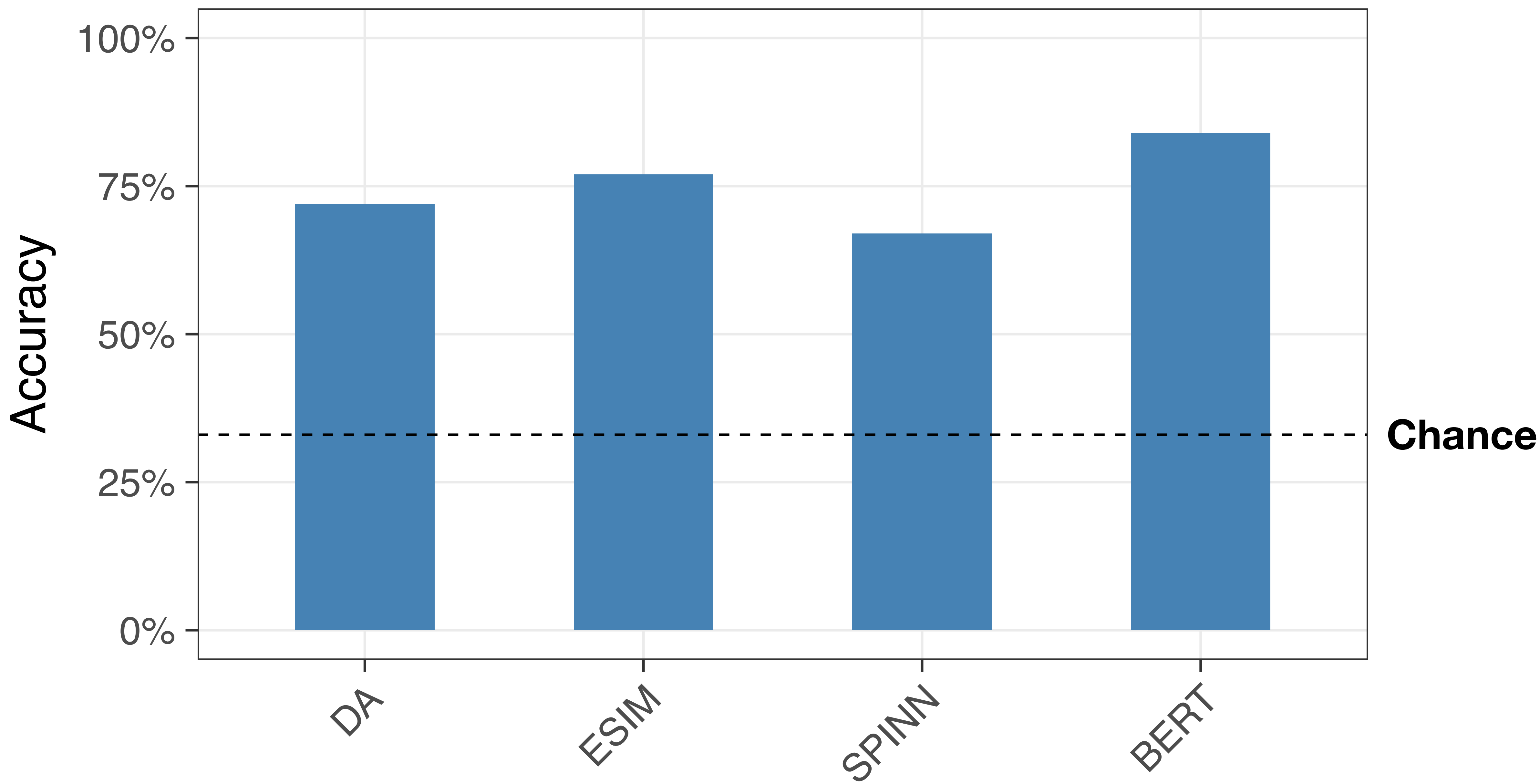
Why do we think neural NLI models might adopt these heuristics?



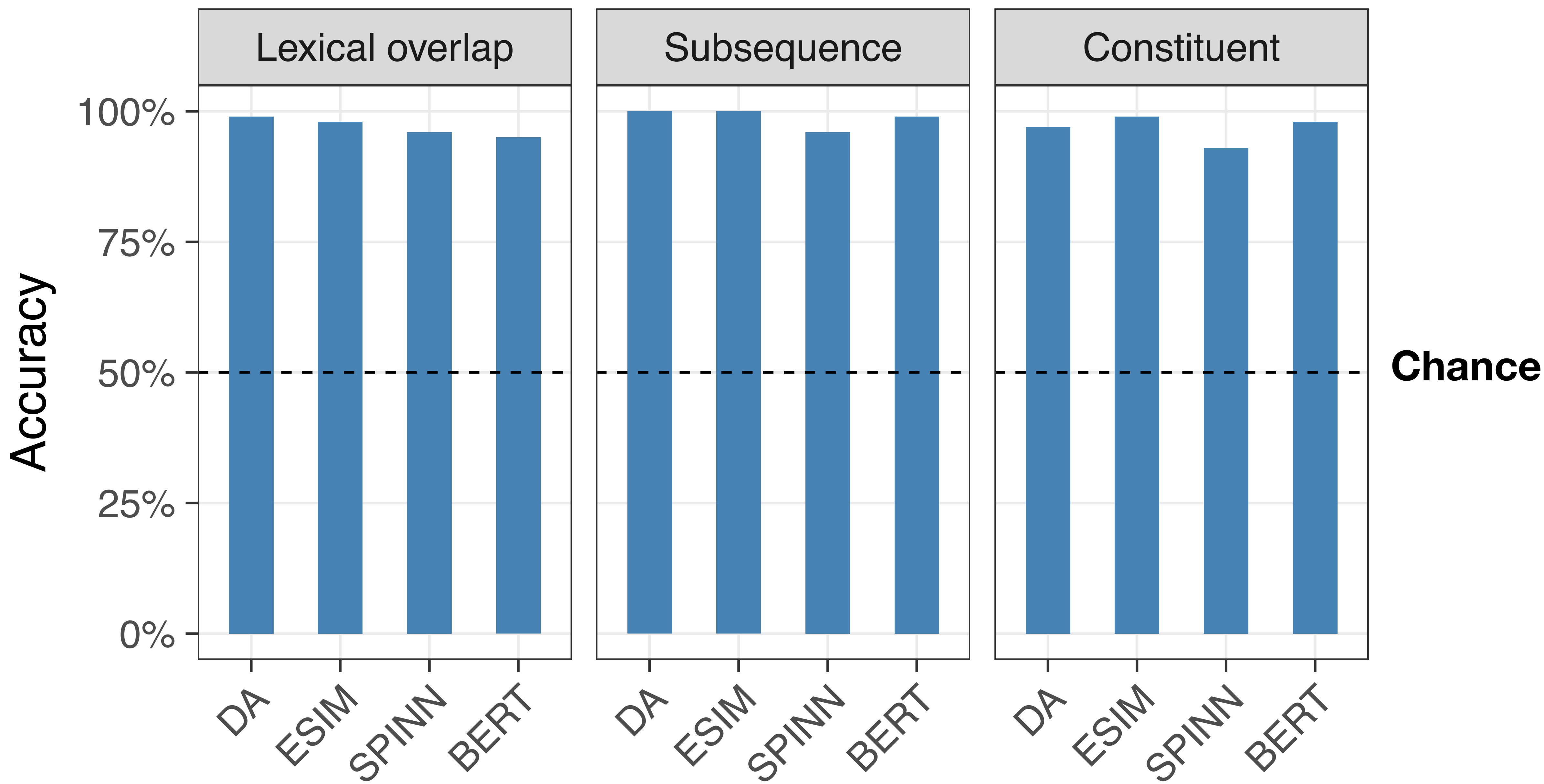
Experimental setup

- For each heuristic, we constructed five templates that support it and five that contradict it
- Four existing models from the literature, representing four sequence representation strategies
  - Decomposable Attention: Bag-of-words (Parikh et al. 2016)
  - ESIM: sequential RNN (Chen et al. 2017)
  - SPINN: Tree-shaped RNN (Bowman et al. 2016)
  - BERT (Devlin et al. 2019)
- All trained on MultiNLI (except BERT which was fine-tuned on MultiNLI)

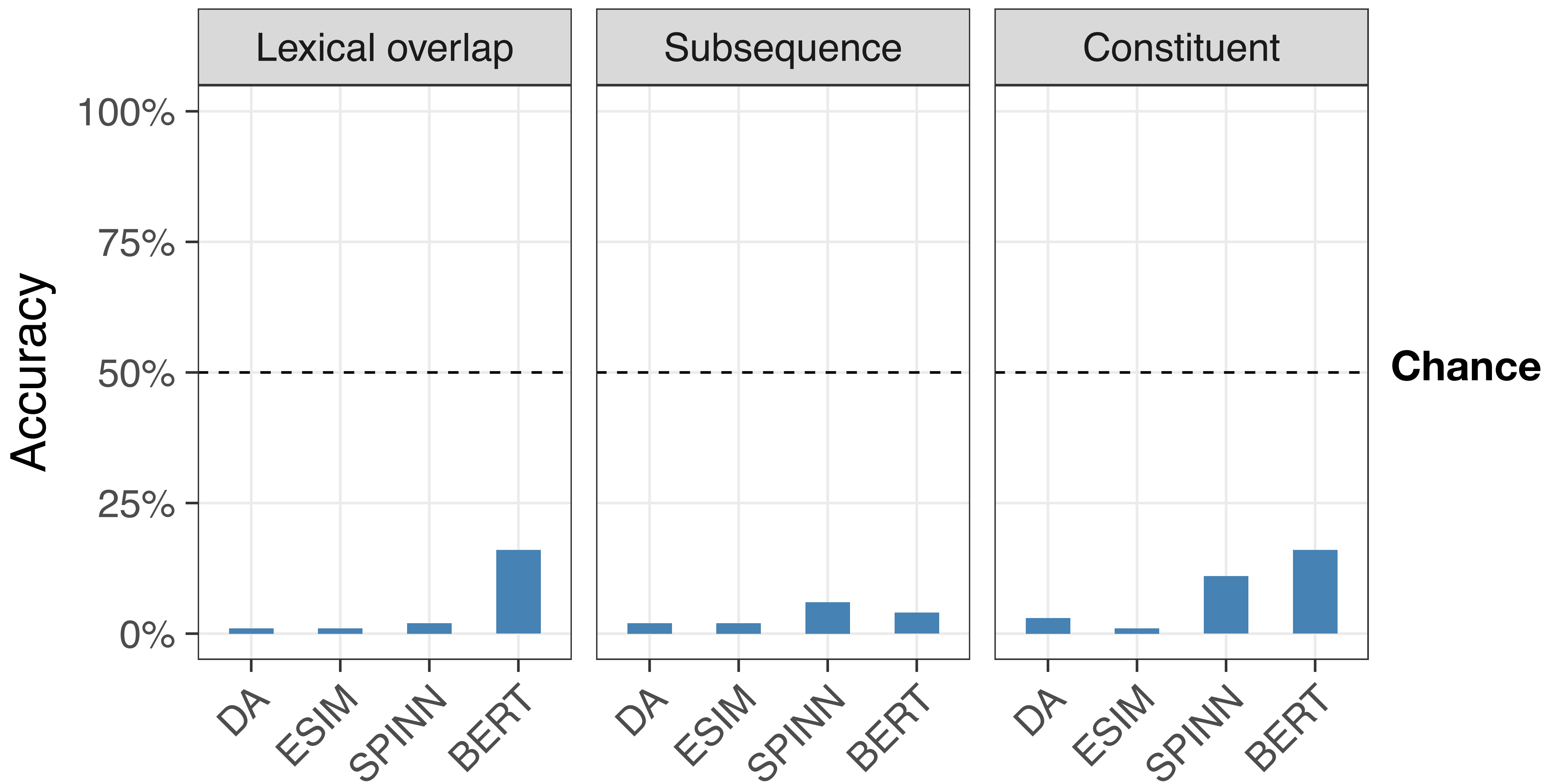
Results on MNLI



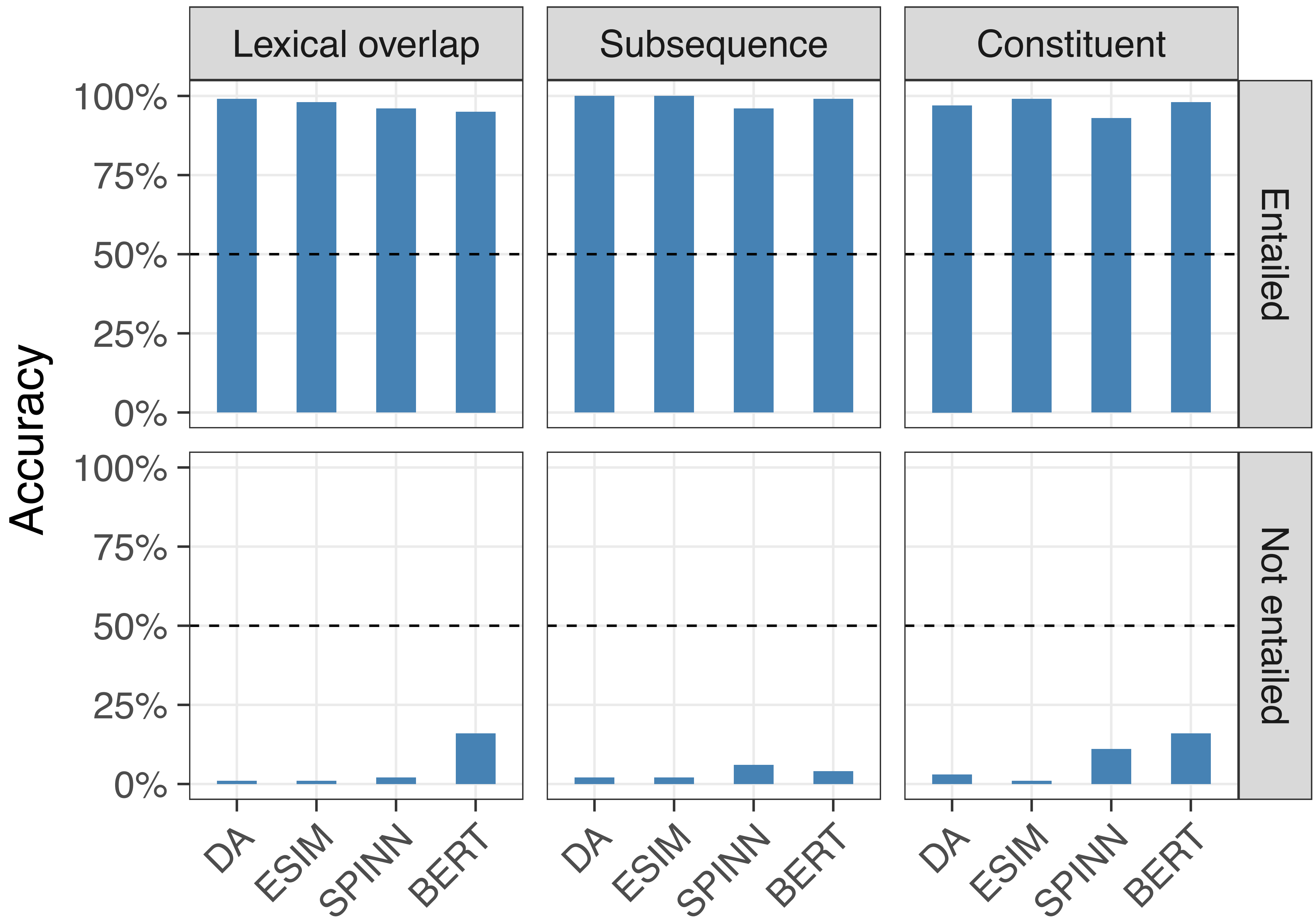
HANS (correct answer: entailed)



HANS (correct answer: not entailed)



HANS



Lexical overlap
Subsequence
Constituent
Entailed
Not entailed
Accuracy
100%
75%
50%
25%
0%
DA
ESIM
SPINN
BERT

HANS: Case-by-case results

Heuristic	Subcase	DA	ESIM	SPINN	BERT
Lexical overlap	Subject-object swap	0.00	0.00	0.03	0.00
	Sentences with PPs	0.00	0.00	0.01	0.25
	Sentences with relative clauses	0.04	0.04	0.06	0.18
	Conjunctions	0.00	0.00	0.01	0.39
	Passives	0.00	0.00	0.00	0.00
Subsequence	NP/S	0.04	0.02	0.09	0.02
	PP on subject	0.00	0.00	0.00	0.06
	Relative clause on subject	0.03	0.04	0.05	0.01
	MV/RR	0.04	0.03	0.03	0.00
	NP/Z	0.02	0.01	0.11	0.10
Constituent	Embedded under preposition	0.14	0.02	0.29	0.50
	Outside embedded clause	0.01	0.00	0.02	0.00
	Embedded under verb	0.00	0.00	0.01	0.22
	Disjunction	0.01	0.03	0.20	0.01
	Adverbs	0.00	0.00	0.00	0.08

HANS: Results

Heuristic	Subcase	DA	ESIM	SPINN	BERT
Lexical overlap	$\boxed{\text{Subject-object swap}}$	0.00	0.00	0.03	0.00
	Sentences with PPs	0.00	0.00	0.01	0.25

BERT trained on MNLI always predicts that

The lawyer advised the judge

entails

The judge advised the lawyer

HANS: discussion

- MNLI does not contain sufficient signal to indicate to a syntactically sophisticated model (BERT) that NLI requires syntax
- Our evaluation data sets should give us a realistic view of the abilities of our systems on the **task** as theoretically defined
- Augmenting the training data with HANS-like examples helps (and generalizes to other syntax-sensitive evaluations)

Outline

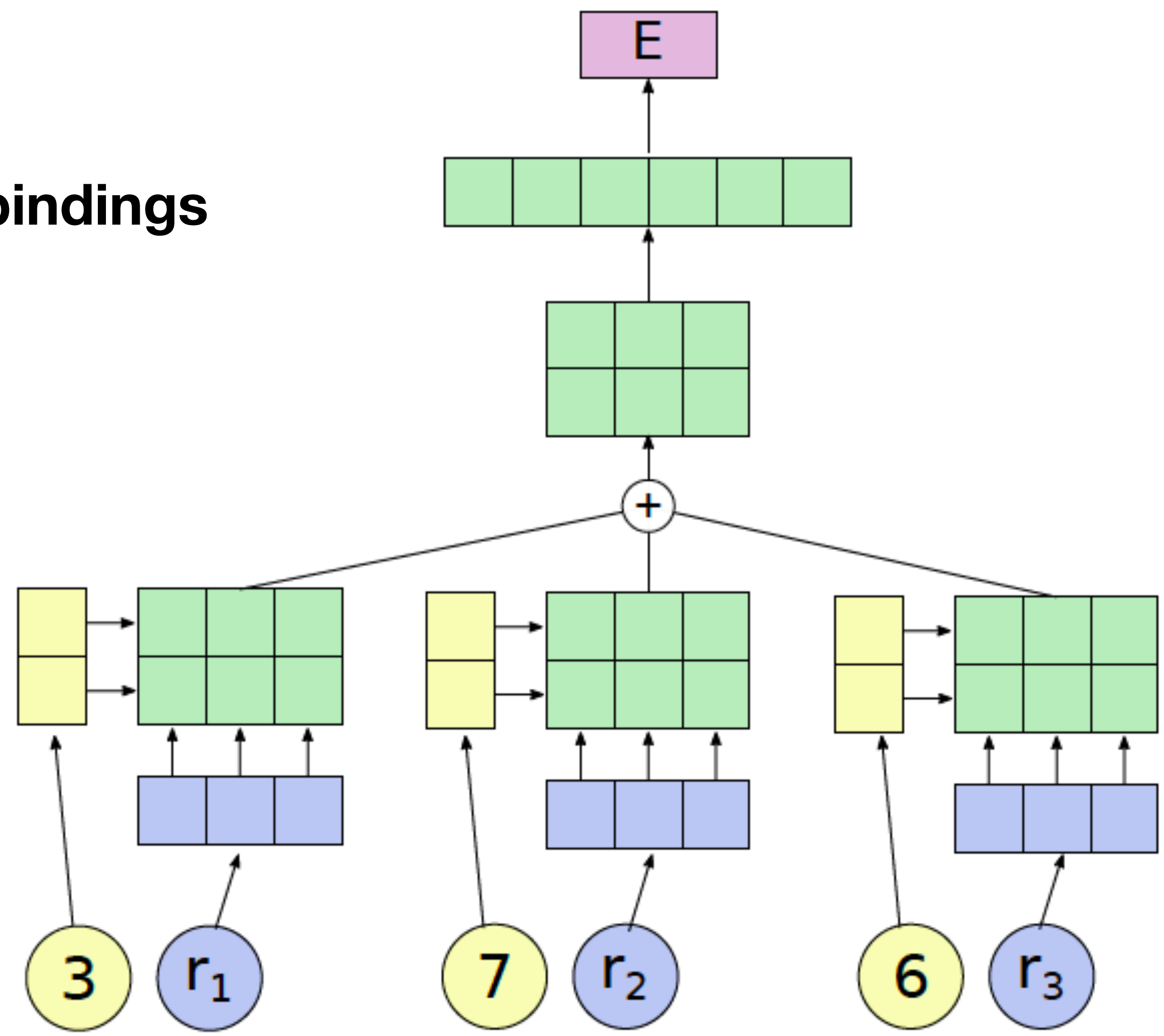
1. Syntactic evaluation of language models
2. Do recurrent neural network language models show human-like syntactic generalization?
3. Syntactic generalization in natural language inference
4. Bonus: measuring compositionality in neural network vector representations

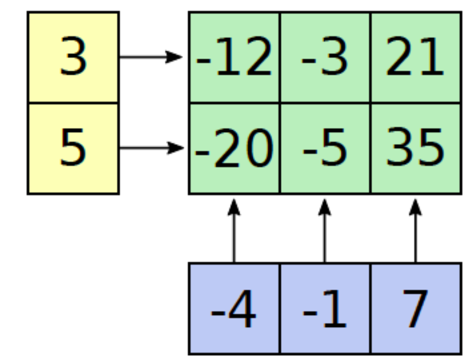
Measuring compositionality in neural network representations

- Compositional representations are necessary for generalization in sequence processing tasks
- Neural networks perform well on certain tasks using continuous vector representations
- How do these representations implicitly encode emergent compositional structure?

Method: Tensor Product Decomposition Networks

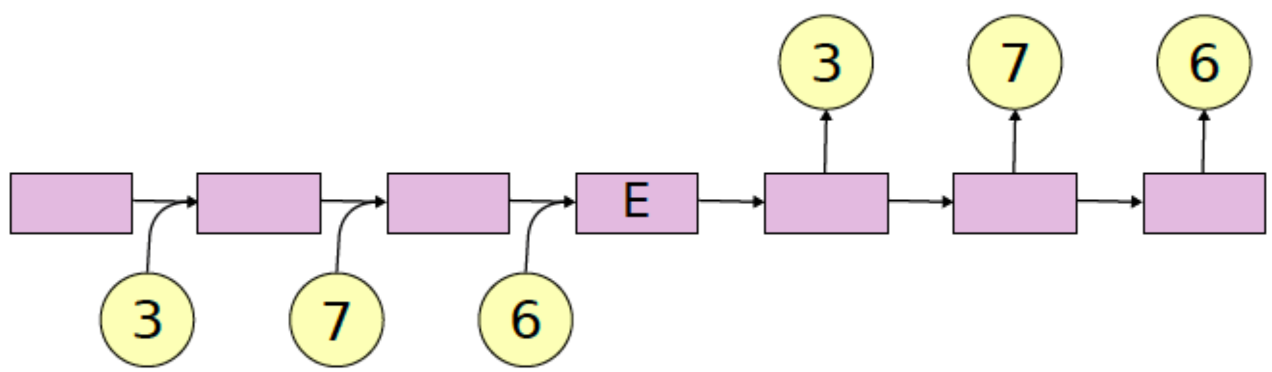
Sum of filler-role bindings





(Smolensky, 1990;
McCoy, Linzen, Dunbar & Smolensky, 2019, *ICLR*)

Test case: sequence autoencoding



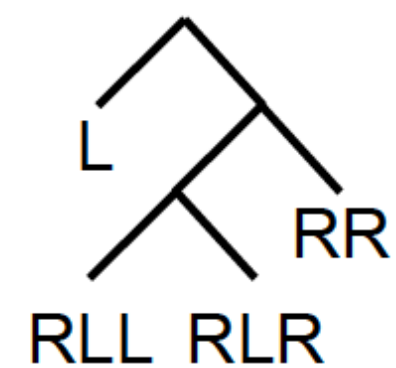
Hypothesis:

 = 4:first + 2:second + 7:third + 9:fourth

Experimental setup: role schemes

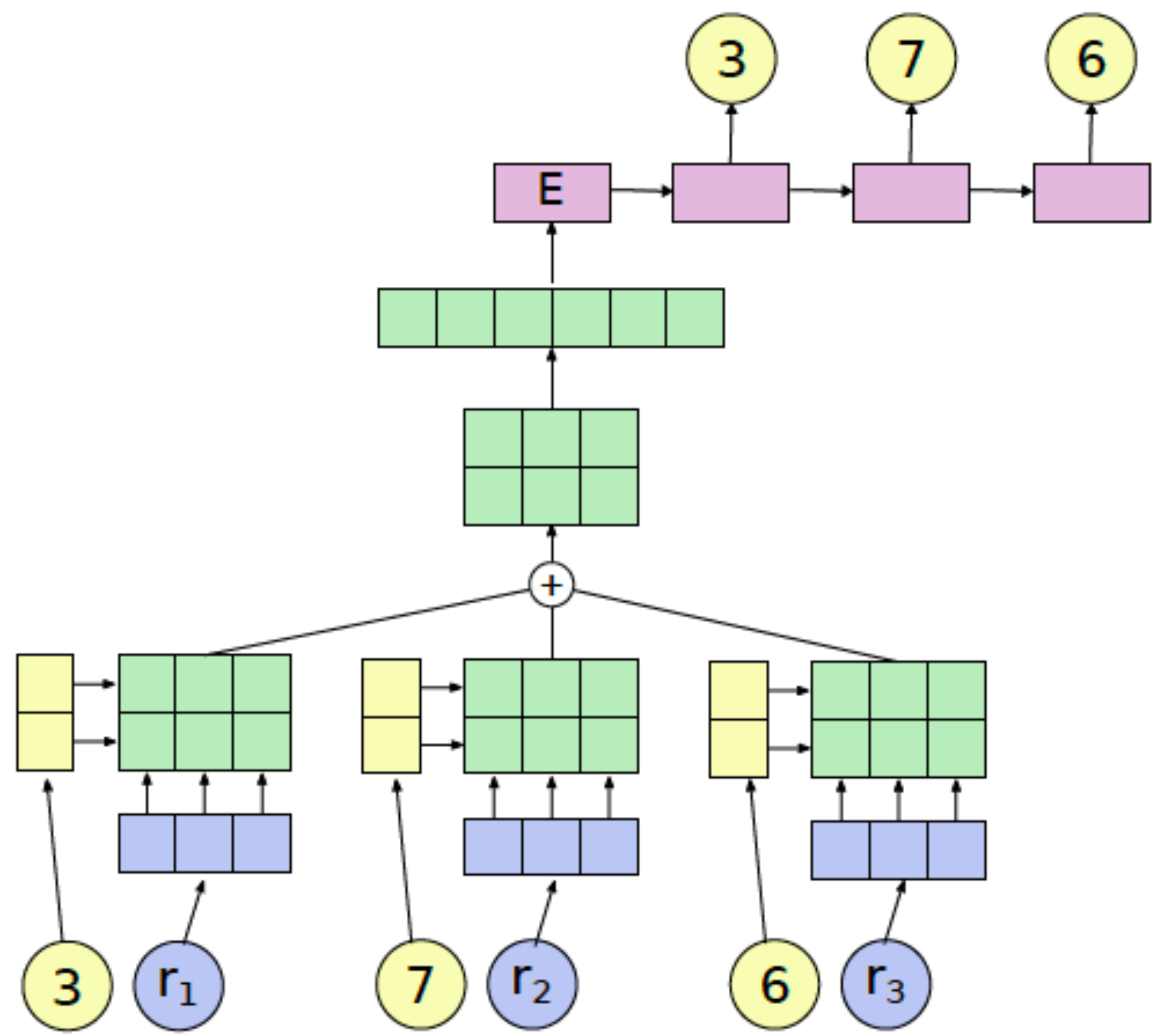
= 4:first + 2:second + 7:third + 9:fourth

	3	1	1	6
Left-to-right	0	1	2	3
Right-to-left	3	2	1	0
Bidirectional	(0, 3)	(1, 2)	(2, 1)	(3, 0)
Wickelroles	#_1	3_1	1_6	1_#
Tree	L	RLL	RLR	RR
Bag of words	r_0	r_0	r_0	r_0

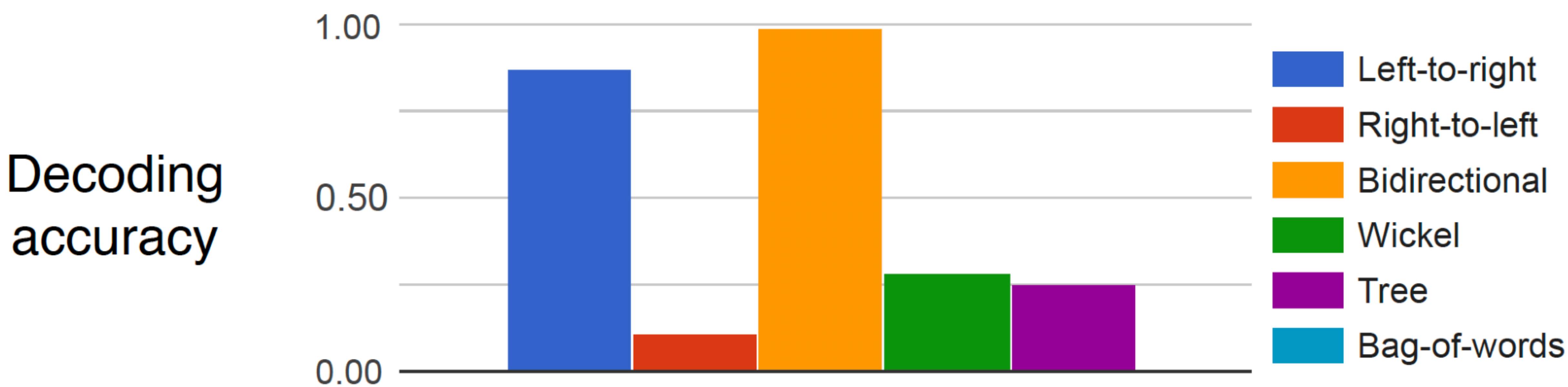


Tree roles

Evaluation: substitution accuracy

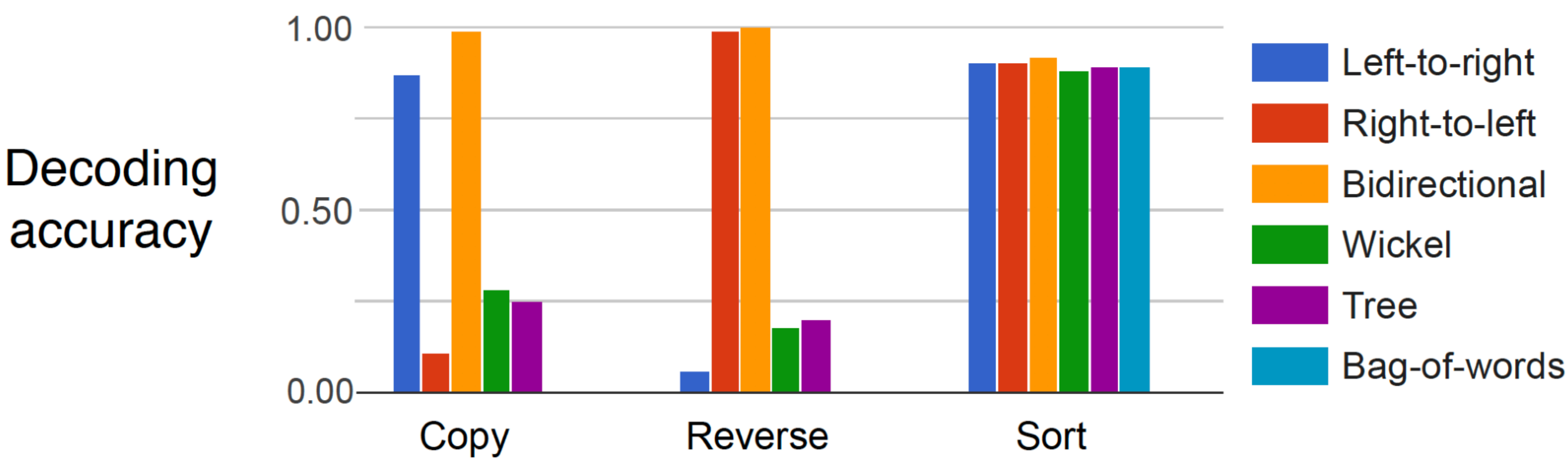


RNN autoencoders can be approximated almost perfectly



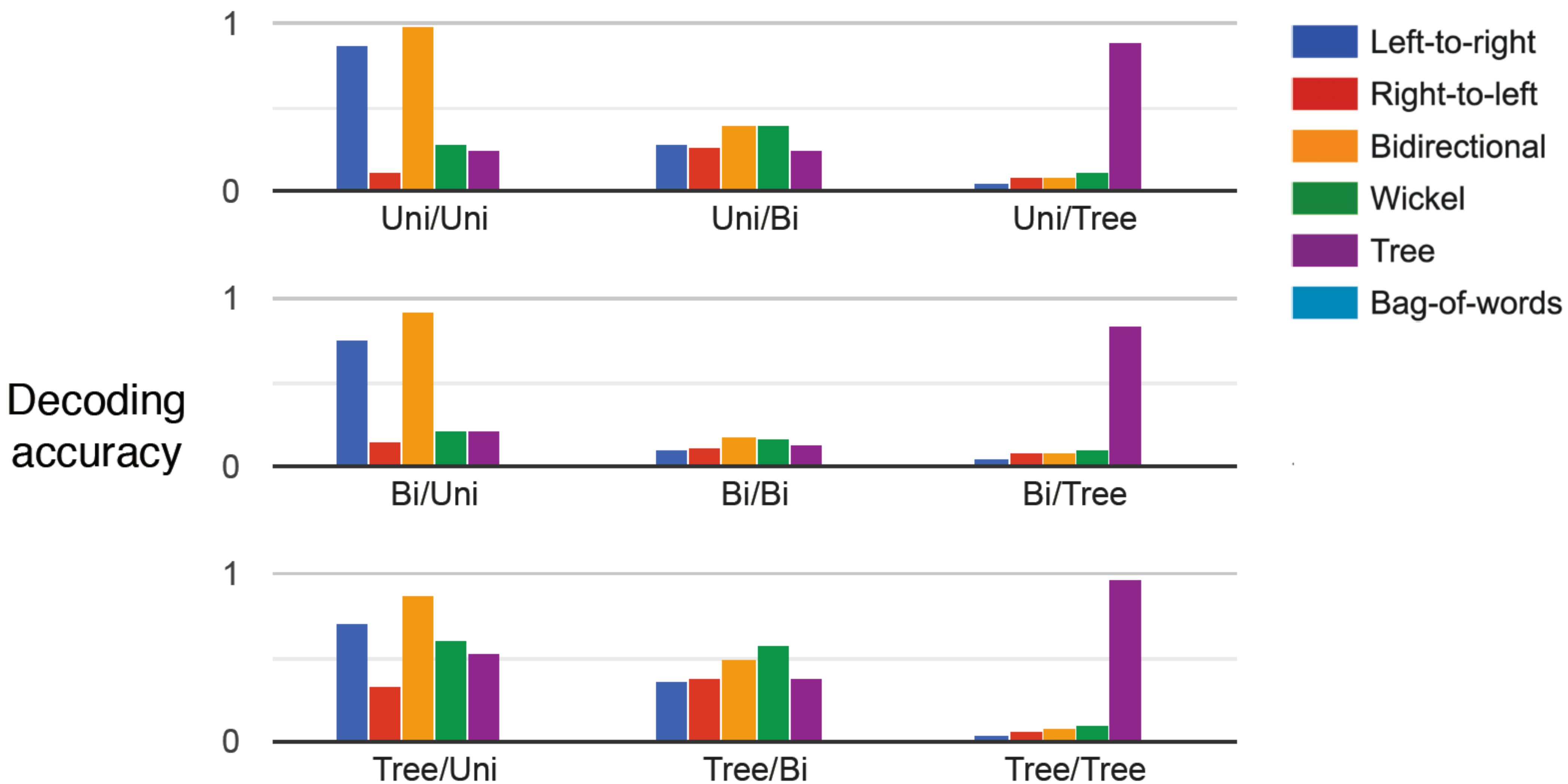
(McCoy, Linzen, Dunbar & Smolensky, 2019, *ICLR*)

Different tasks favor different role schemes



(McCoy, Linzen, Dunbar & Smolensky, 2019, *ICLR*)

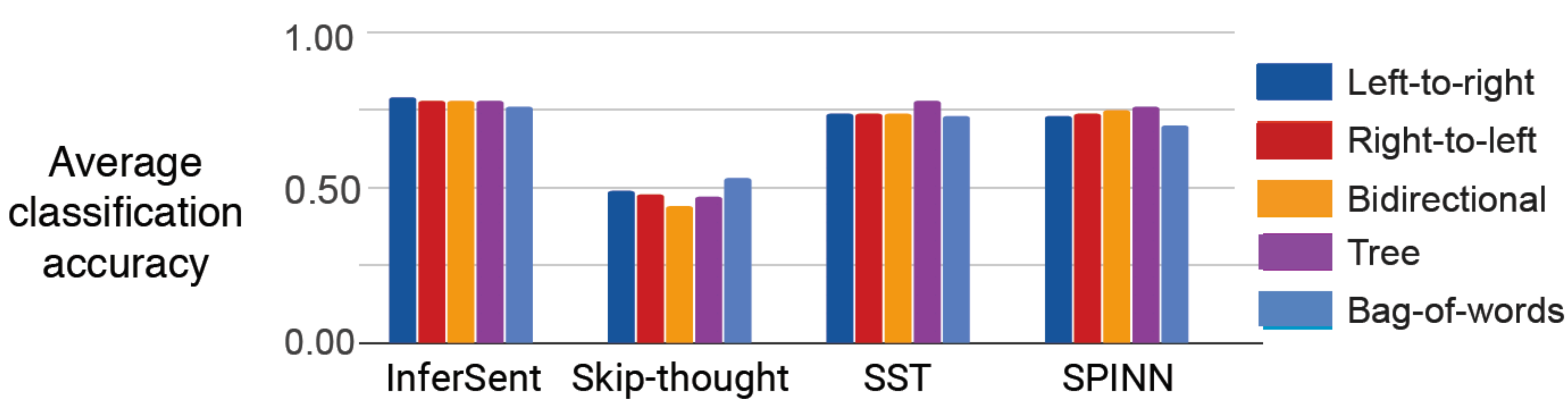
The decoder determines the learned role scheme



(McCoy, Linzen, Dunbar & Smolensky, 2019, *ICLR*)

What about vector sentence embeddings from NLP tasks?

	Model Type	Training task
InferSent	BiLSTM	Natural Language Inference
Skip-thought	LSTM	Previous/next sentence prediction
SST	Tree	Sentiment prediction
SPINN	Tree	Natural Language Inference



Interim discussion

- Sequence representation in RNN seq2seq networks can be decomposed as sums of filler-role binding vectors
- Depend on the task and decoder architecture in an interpretable way
- Sentence representations from NLP don't show similarly compositional properties

Post-doc plug!

- I am hiring two post-docs!
  - With Chris Honey (Psychological & Brain Sciences, JHU): neural network modeling of ECoG data from language paradigms
  - In my group: evaluation and syntactic generalization in neural networks

Thank you!

- **NSF:** GRFP 1746891, INSPIRE BCS-1344269
- **ERC:** ERC-2011-AdG-295810 (BOOTPHON)
- **ANR:** ANR-10-LABX-0087 (IEC), ANR-10-IDEX-0001-02 (PSL*), ANR-17-CE28-0009 (GEOMPHON), ANR-11-IDEX-0005 (USPC), and ANR-10-LABX-0083 (EFL)
- **Google:** Google Faculty Award

Thank you!

- Neural networks may succeed on frequent (and simpler) sentence types without mastering many linguistic phenomena

- LSTM LMs can approximate syntactic behavior in many sentences, but still struggle on complex sentences (e.g., relative clauses, reflexive anaphora binding)

- Our evaluation data sets should give us a realistic view of the abilities of our systems on the **task** as theoretically defined, rather than a specific data set (e.g., for MNLI)

- RNN can learn to represent sequences as sums of filler-role bindings (without specific supervision)

BERT’s pre-trained syntactic representations are actually quite good!

	BERT Base	BERT Large	LSTM (M&L)	Humans (M&L)	# Pairs (# M&L Pairs)
SUBJECT-VERB AGREEMENT:					
Simple	1.00	1.00	0.94	0.96	120 (140)
In a sentential complement	0.83	0.86	0.99	0.93	1440 (1680)
Short VP coordination	0.89	0.86	0.90	0.82	720 (840)
Long VP coordination	0.98	0.97	0.61	0.82	400 (400)
Across a prepositional phrase	0.85	0.85	0.57	0.85	19440 (22400)
Across a subject relative clause	0.84	0.85	0.56	0.88	9600 (11200)
Across an object relative clause	0.89	0.85	0.50	0.85	19680 (22400)
Across an object relative (no *that*)	0.86	0.81	0.52	0.82	19680 (22400)
In an object relative clause	0.95	0.99	0.84	0.78	15960 (22400)
In an object relative (no *that*)	0.79	0.82	0.71	0.79	15960 (22400)
REFLEXIVE ANAPHORA:					
Simple	0.94	0.92	0.83	0.96	280 (280)
In a sentential complement	0.89	0.86	0.86	0.91	3360 (3360)
Across a relative clause	0.80	0.76	0.55	0.87	22400 (22400)

(Goldberg, 2019)

What if we added HANS to the training set?