Interpretability and Analysis

LING 574 Deep Learning for NLP

Shane Steinert-Threlkeld

Today’s Plan

- NLP’s “Clever Hans” Moment: motivating interpretability and analysis
- Survey of several different methods:
  - Neuron-level
  - Psycholinguistic experiments
  - Diagnostic classifiers
  - Attention analysis
  - Adversarial datasets

NLP’s “Clever Hans Moment”

The Gradient

ME EDITOR'S NOTE OVERVIEWS PERSPECTIVES ABOUT SUBSCRIBE

Clever Hans

BERT

NLP's Clever Hans Moment has Arrived

26.AUG.2019

link

Clever Hans

- Early 1900s, a horse trained by his owner to do:
  - Addition
  - Division
  - Multiplication
  - Tell time
  - Read German
  - …

- Wow! Hans is really smart!

Clever Hans Effect

Clever Hans Effect

- Upon closer examination / experimentation…

Clever Hans Effect

- Upon closer examination / experimentation…
- Hans' success:

Clever Hans Effect

- Upon closer examination / experimentation…
- Hans' success:
  - 89% when questioner knows answer

Clever Hans Effect

- Upon closer examination / experimentation…
- Hans' success:
  - 89% when questioner knows answer
  - 6% when questioner doesn't know answer

Clever Hans Effect

- Upon closer examination / experimentation…
- Hans' success:
  - 89% when questioner knows answer
  - 6% when questioner doesn't know answer
- Further experiments: as Hans' taps got closer to correct answer, facial tension in questioner increased

Clever Hans Effect

- Upon closer examination / experimentation…
- Hans' success:
  - 89% when questioner knows answer
  - 6% when questioner doesn't know answer
- Further experiments: as Hans' taps got closer to correct answer, facial tension in questioner increased
- Hans didn't solve the task but exploited *a spuriously correlated cue*

Central question

- Do BERT et al's major successes at solving NLP tasks show that we have achieved robust natural language understanding in machines?

- Or: are we seeing a "Clever BERT" phenomenon?

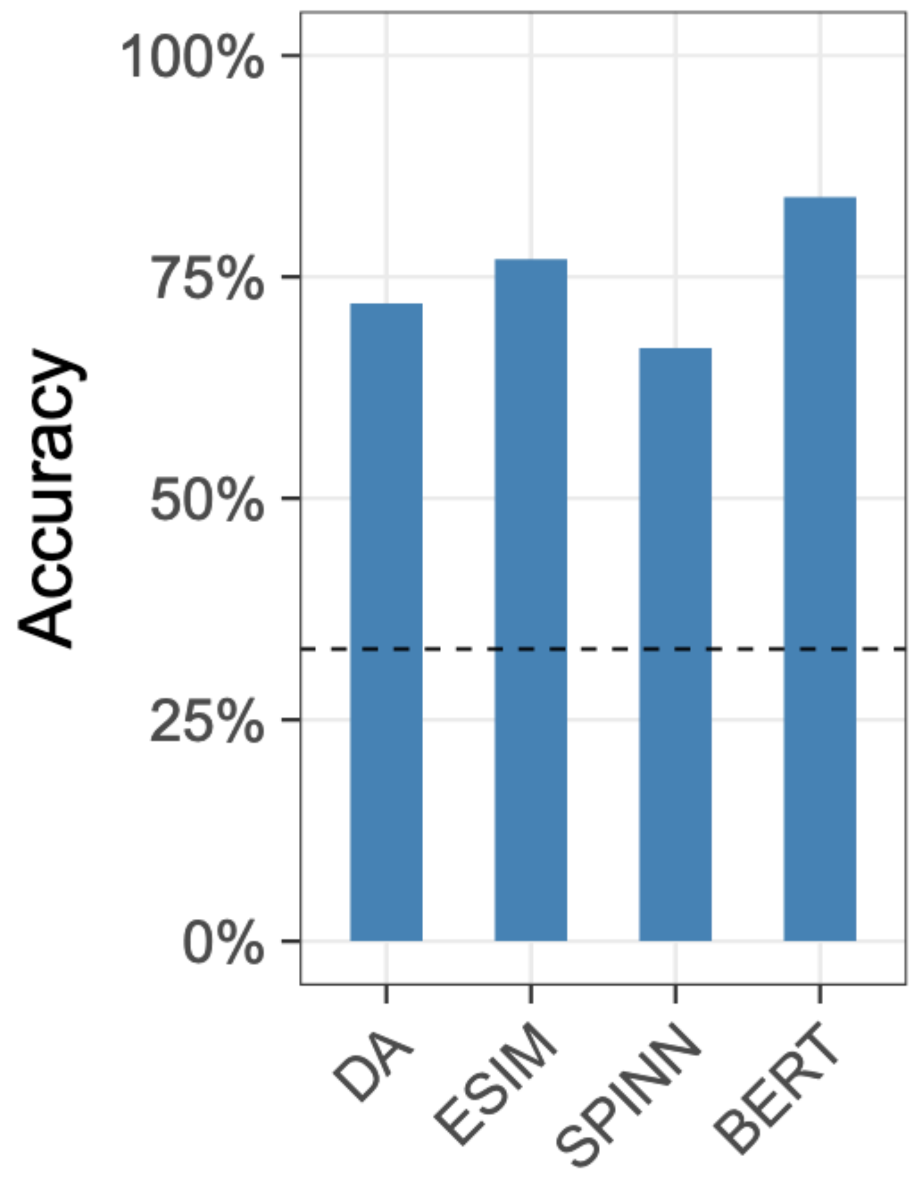
Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference

R. Thomas McCoy,[1] Ellie Pavlick,[2] & Tal Linzen[1]
[1]Department of Cognitive Science, Johns Hopkins University
[2]Department of Computer Science, Brown University
tom.mccoy@jhu.edu, ellie_pavlick@brown.edu, tal.linzen@jhu.edu

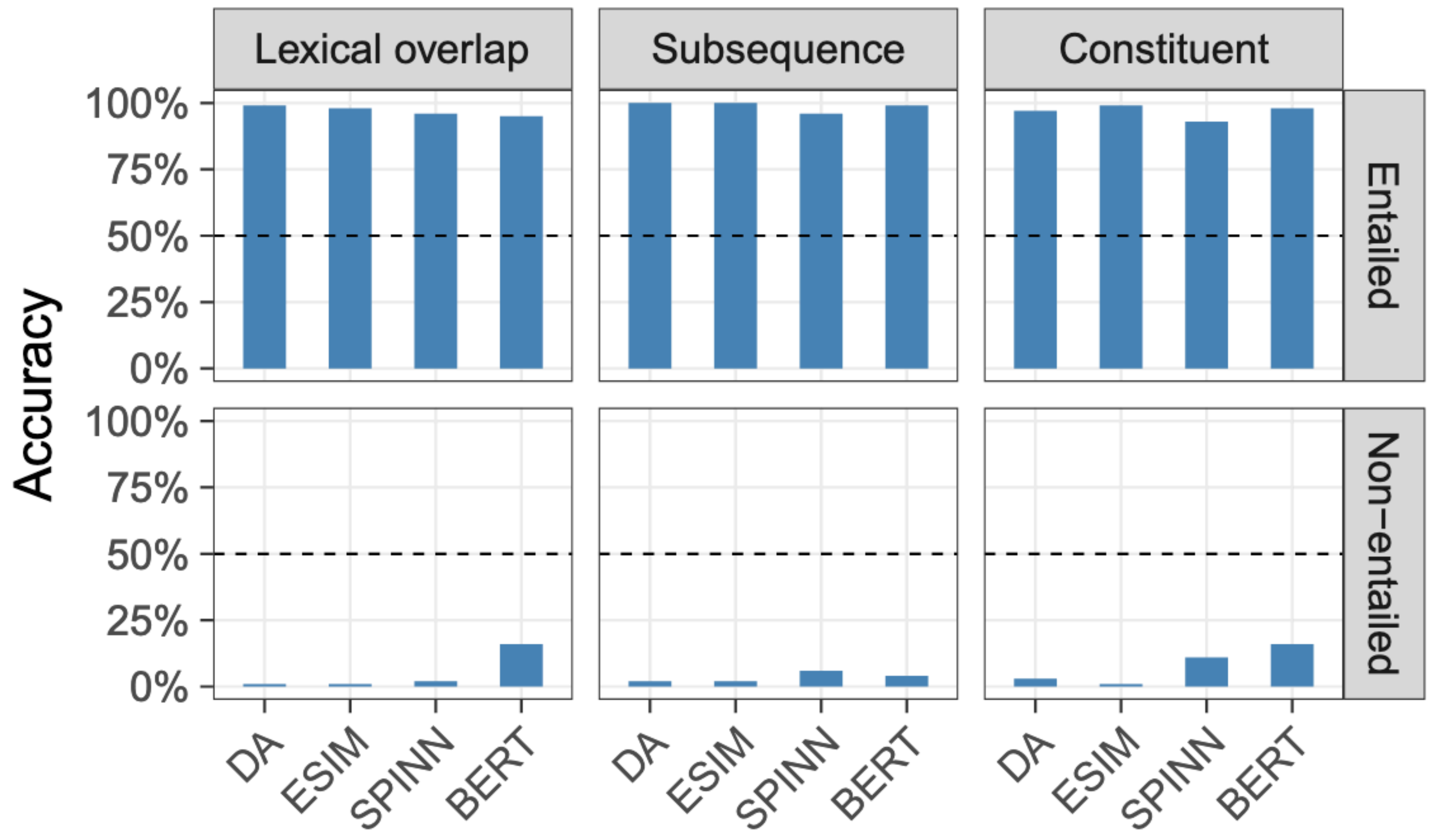
McCoy et al 2019

Heuristic	Premise	Hypothesis	Label
Lexical overlap heuristic	The banker near the judge saw the actor.	The banker saw the actor.	E
	The lawyer was advised by the actor.	The actor advised the lawyer.	E
	The doctors visited the lawyer.	The lawyer visited the doctors.	N
	The judge by the actor stopped the banker.	The banker stopped the actor.	N
Subsequence heuristic	The artist and the student called the judge.	The student called the judge.	E
	Angry tourists helped the lawyer.	Tourists helped the lawyer.	E
	The judges heard the actors resigned.	The judges heard the actors.	N
	The senator near the lawyer danced.	The lawyer danced.	N
Constituent heuristic	Before the actor slept, the senator ran.	The actor slept.	E
	The lawyer knew that the judges shouted.	The judges shouted.	E
	If the actor slept, the judge saw the artist.	The actor slept.	N
	The lawyers resigned, or the artist slept.	The artist slept.	N

Results



(a)



(b)

(performance improves if fine-tuned on this challenge set)

Why care?

- Effects of learning what neural language models understand:
  - Engineering: can help build better language technologies via improved models, data, training protocols, …
    - Trust, critical applications
  - Theoretical: can help us understand biases in different architectures (e.g. LSTMs vs Transformers), similarities to human learning biases
    - Which linguistic features / properties are *learnable* from raw text alone?
  - Ethical: e.g. do some models reflect problematic social biases more than others?

Visualization / neuron-level analysis

Main Idea

- Individual neurons in a network have activations that depend on the input
- Check to see whether any of them have activations which depend on / correlate with (linguistically) interesting features of the input
- [Think of the alleged “Jennifer Anniston cells”, aka grandmother cells]

Learning to Generate Reviews and Discovering Sentiment

Alec Radford [1] Rafal Jozefowicz [1] Ilya Sutskever [1]

Abstract

We explore the properties of byte-level recurrent language models. When given sufficient amounts of capacity, training data, and compute time, the representations learned by these models include disentangled features corresponding to high-level concepts. Specifically, we find a single unit which performs sentiment analysis. These representations, learned in an unsupervised manner, achieve state of the art on the binary subset of the Stanford Sentiment Treebank. They are also very data efficient. When using only a handful of labeled examples, our approach matches the performance of strong baselines trained on full datasets. We also demonstrate the sentiment unit has a direct influence on the generative process of the model. Simply fixing its value to be positive or negative generates samples with the corresponding positive or negative sentiment.

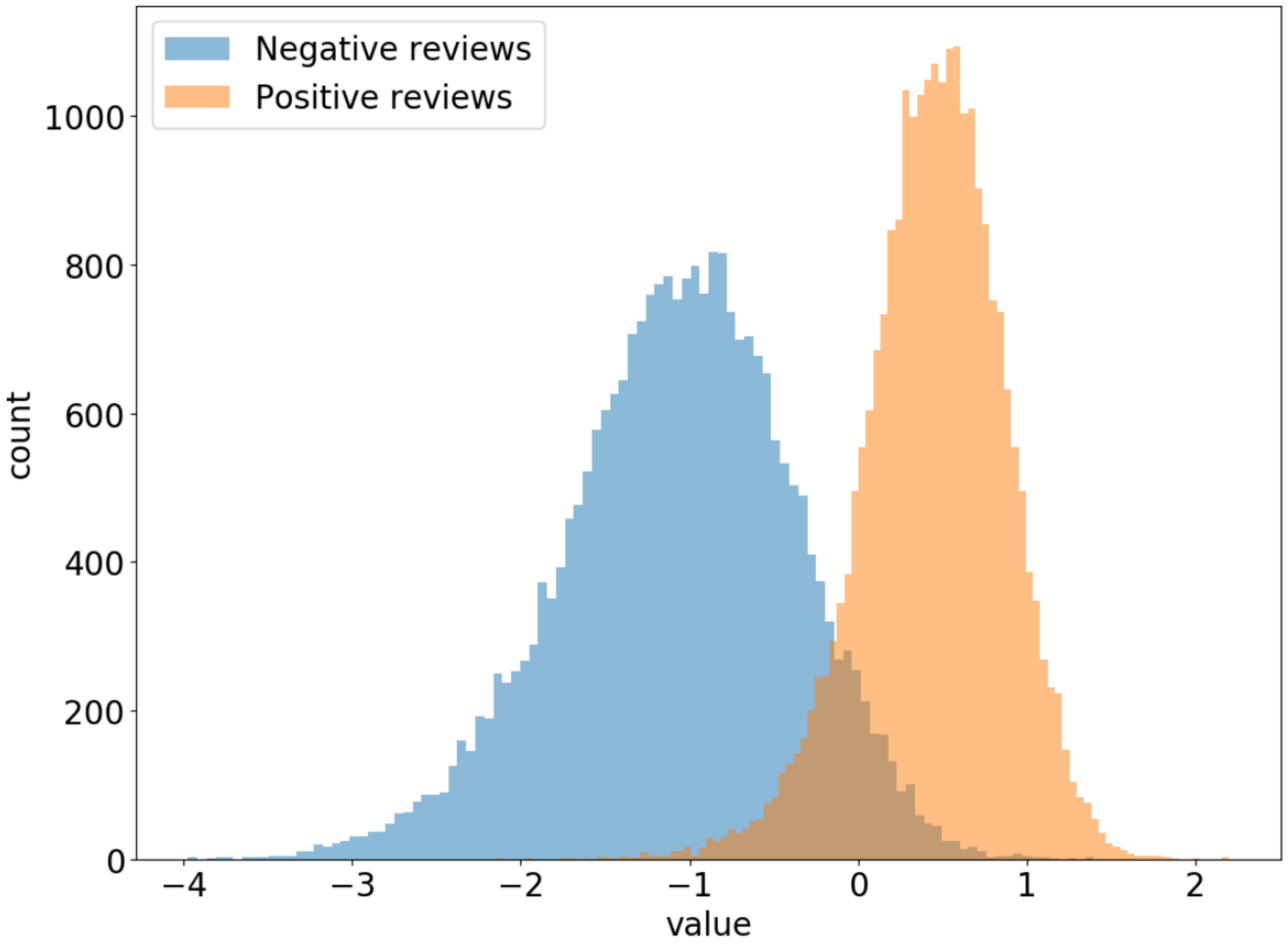
it is now commonplace to reuse these representations on a broad suite of related tasks - one of the most successful examples of transfer learning to date (Oquab et al., 2014).

There is also a long history of unsupervised representation learning (Olshausen & Field, 1997). Much of the early research into modern deep learning was developed and validated via this approach (Hinton & Salakhutdinov, 2006) (Huang et al., 2007) (Vincent et al., 2008) (Coates et al., 2010) (Le, 2013). Unsupervised learning is promising due to its ability to scale beyond only the subsets and domains of data that can be cleaned and labeled given resource, privacy, or other constraints. This advantage is also its difficulty. While supervised approaches have clear objectives that can be directly optimized, unsupervised approaches rely on proxy tasks such as reconstruction, density estimation, or generation, which do not directly encourage useful representations for specific tasks. As a result, much work has gone into designing objectives, priors, and architectures meant to encourage the learning of useful representations.

Approach

- Character-level language model (LSTM variant)
  - One layer; 4096 dim hidden state
  - Training: ~1 month on 4 GPUs
- Data: Amazon product reviews
- Fine-tune: sentiment analysis
  - NB: this data partially overlaps with training data [but a different task]

A sentiment neuron



Samples of the sentiment neuron

I found this to be a charming adaptation, very lively and full of fun. With the exception of a couple of major errors, the cast is wonderful. I have to echo some of the earlier comments -- Chynna Phillips is horribly miscast as a teenager. At 27, she's just too old (and, yes, it DOES show), and lacks the singing "chops" for Broadway-style music. Vanessa Williams is a decent-enough singer and, for a non-dancer, she's adequate. However, she is NOT Latina, and her character definitely is. She's also very STRIDENT throughout, which gets tiresome. The girls of Sweet Apple's Conrad Birdie fan club really sparkle -- with special kudos to Brigitta Dau and Chiara Zanni. I also enjoyed Tyne Daly's performance, though I'm not generally a fan of her work. Finally, the dancing Shriners are a riot, especially the dorky three in the bar. The movie is suitable for the whole family, and I highly recommend it.

Judy Holliday struck gold in 1950 withe George Cukor's film version of "Born Yesterday," and from that point forward, her career consisted of trying to find material good enough to allow her to strike gold again. It never happened. In "It Should Happen to You" (I can't think of a blander title, by the way), Holliday does yet one more variation on the dumb blonde who's maybe not so dumb after all, but everything about this movie feels warmed over and half hearted. Even Jack Lemmon, in what I believe was his first film role, can't muster up enough energy to enliven this recycled comedy. The audience knows how the movie will end virtually from the beginning, so mostly it just sits around waiting for the film to catch up. Maybe if you're enamored of Holliday you'll enjoy this; otherwise I wouldn't bother. Grade: C

Sentiment unit does all the work!

Table 2. IMDB sentiment classification

Method	Error
FullUnlabeledBoW (Maas et al., 2011)	11.11%
NB-SVM trigram (Mesnil et al., 2014)	8.13%
Sentiment unit (ours)	7.70%
SA-LSTM (Dai & Le, 2015)	7.24%
byte mLSTM (ours)	7.12%
TopicRNN (Dieng et al., 2016)	6.24%
Virtual Adv (Miyato et al., 2016)	5.91%

The Emergence of Number and Syntax Units in LSTM Language Models

Yair Lakretz
Cognitive Neuroimaging Unit
NeuroSpin center
91191, Gif-sur-Yvette, France
yair.lakretz@gmail.com

German Kruszewski
Facebook AI Research
Paris, France
germank@gmail.com

Theo Desbordes
Facebook AI Research
Paris, France
tdesbordes@fb.com

Dieuwke Hupkes
ILLC, University of Amsterdam
Amsterdam, Netherlands
d.hupkes@uva.nl

Stanislas Dehaene
Cognitive Neuroimaging Unit
NeuroSpin center
91191, Gif-sur-Yvette, France
stanislas.dehaene@gmail.com

Marco Baroni
Facebook AI Research
Paris, France
mbaroni@fb.com

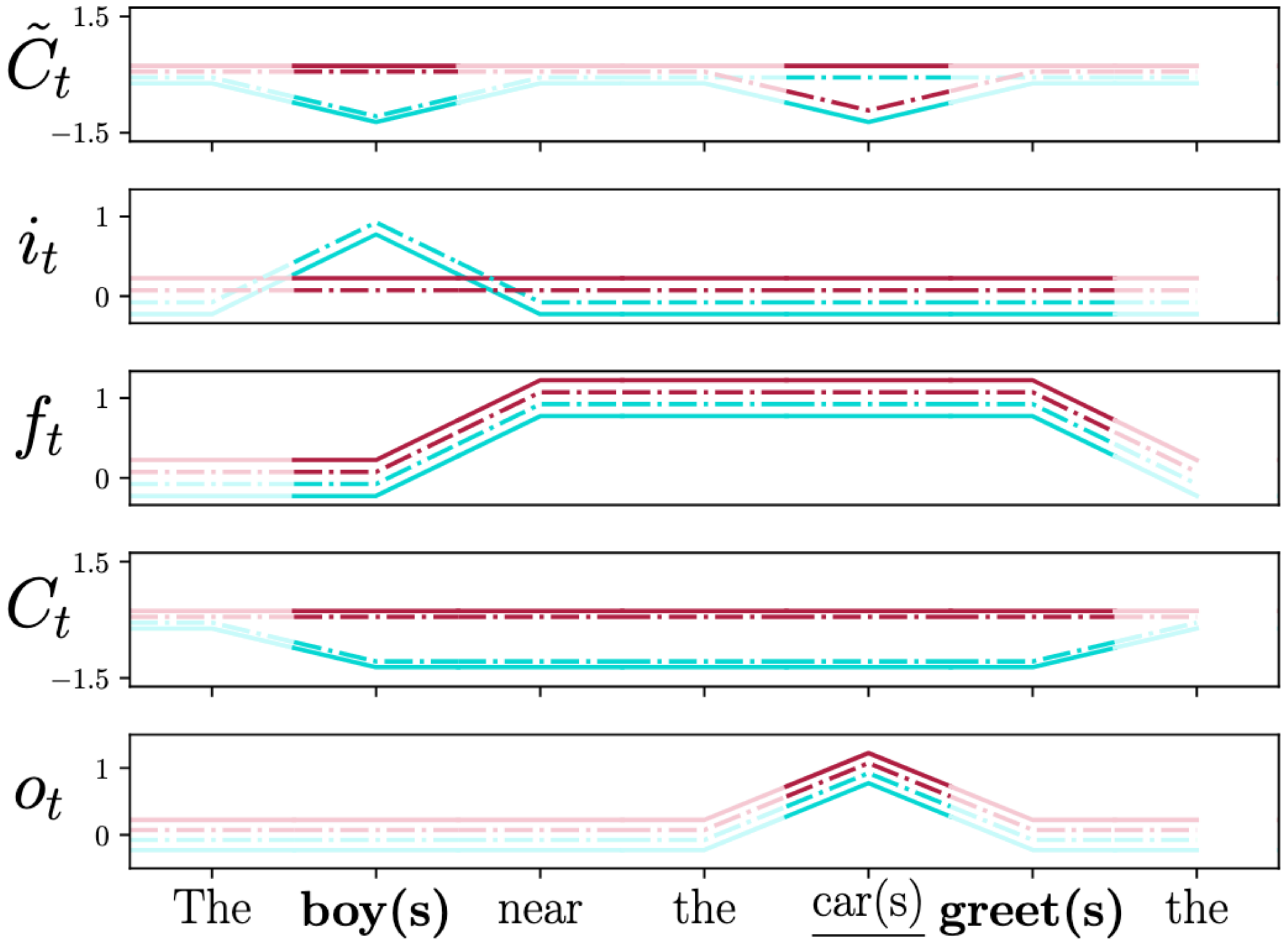
Approach

- Evaluating the Gulordava et al 2018 LSTM LM (trained on Wikipedia)
- Number agreement tasks: as in Linzen et al 2016 (to be discussed shortly!)
  - Plus synthetic:

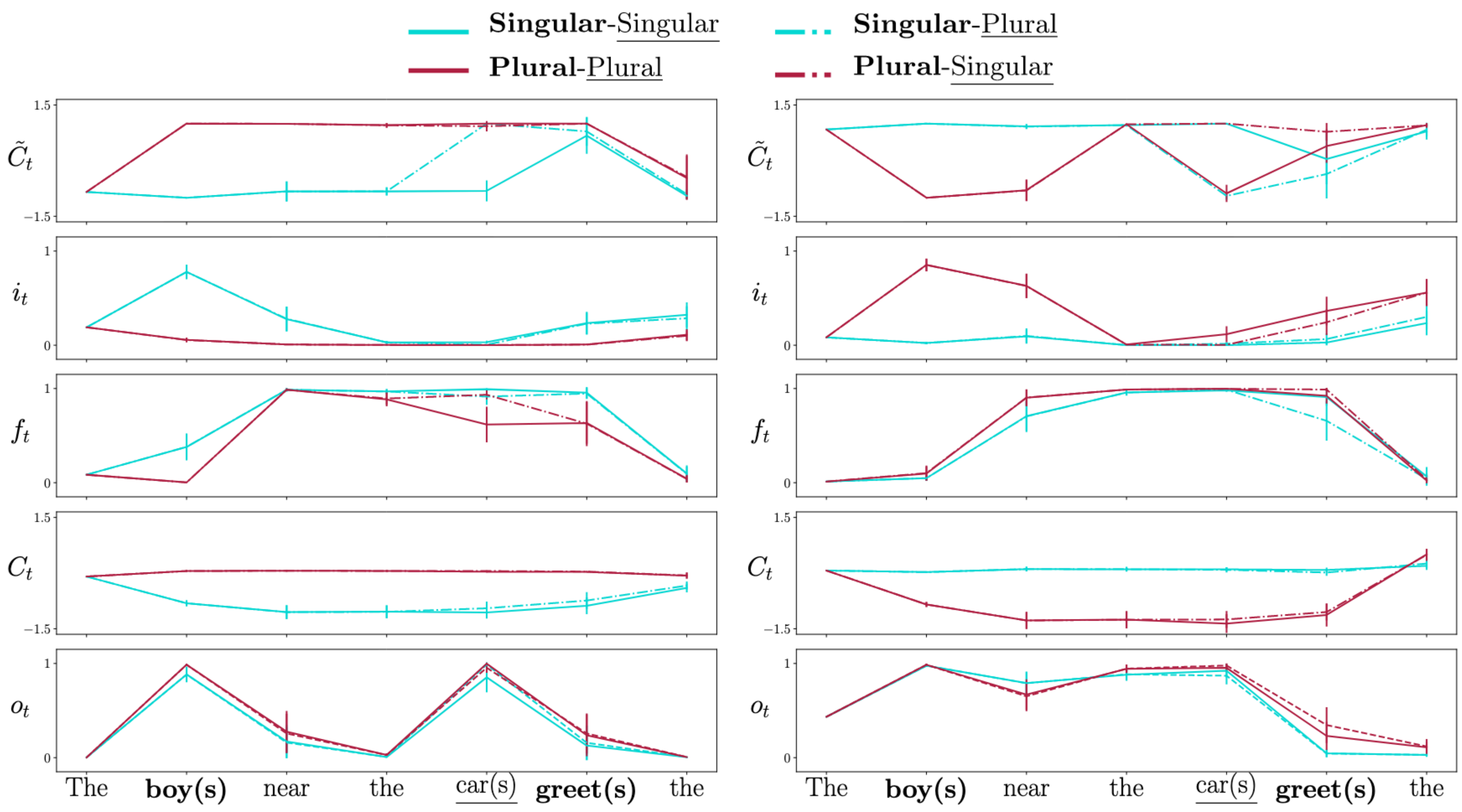
Simple	the **boy** **greets** the guy
Adv	the **boy** probably **greets** the guy
2Adv	the **boy** most probably **greets** the guy
CoAdv	the **boy** openly and deliberately **greets** the guy
NamePP	the **boy** near Pat **greets** the guy
NounPP	the **boy** near the car **greets** the guy
NounPPAdv	the **boy** near the car kindly **greets** the guy

- Find important cells by *ablation*: set activation to 0, see if performance suffers. (Also by regression; more in a minute)

Cell dynamics for storing number info



Learned cell dynamics for number info



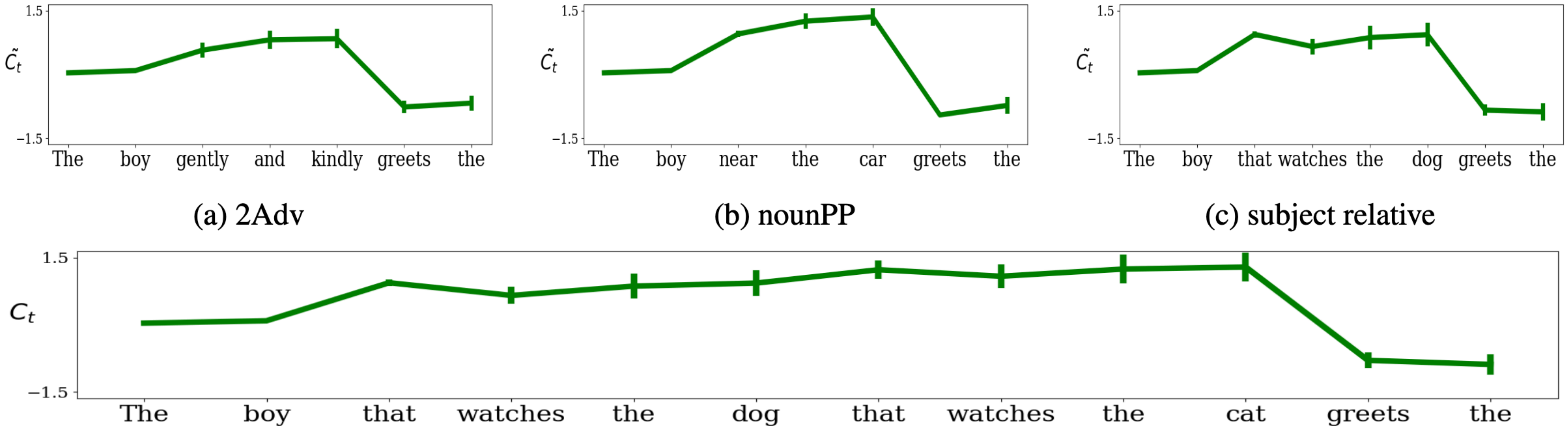
(a) 988 (singular)

(b) 776 (plural)

Finding a syntax unit

- Predict, via linear regression, from the cell:
  - Depth of the word in syntactic parse of the sentence
  - (Works pretty well: $R^2 = 0.85$. More on this idea later.)
- Identify cells that are assigned very high weight in the regression

Cell dynamics for a syntax unit



Neuron-level analysis: summary

Neuron-level analysis: summary

- Very promising and exciting when it does work: a good look "inside the black box", with very interpretable neural/cell dynamics. But:

Neuron-level analysis: summary

- Very promising and exciting when it does work: a good look “inside the black box”, with very interpretable neural/cell dynamics. But:
- “A needle in a haystack”: how to find the “good” neurons?
  - Some principled methods (ablation, regression); not all of them scale well
  - But also:
    - Is there a neuron that tracks property P?
    - Not: what are you tracking?

Neuron-level analysis: summary

- Very promising and exciting when it does work: a good look "inside the black box", with very interpretable neural/cell dynamics. But:
- "A needle in a haystack": how to find the "good" neurons?
  - Some principled methods (ablation, regression); not all of them scale well
  - But also:
    - Is there a neuron that tracks property P?
    - Not: what are you tracking?
- Deleting interpretable neurons may not effect performance in the original or downstream task (Morcos et al 2018)

Psycholinguistic methods

Animating Idea

- NLMs are a bit of a "black box". How can we figure out what they're doing?
- Well: humans are also (approximately) black boxes!
- So: let's treat NLMs the way we treat people when we try to figure out the nature of their linguistic knowledge.
  - In other words: treat NLMs as if they were participants in the kinds of experiments that (psycho-)linguists perform.
  - [NB: lots more to do here!]

Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies

Tal Linzen[1,2] **Emmanuel Dupoux**[1]
LSCP[1] & IJN[2], CNRS,
EHESS and ENS, PSL Research University
`{tal.linzen, emmanuel.dupoux}@ens.fr`

Yoav Goldberg
Computer Science Department
Bar Ilan University
`yoav.goldberg@gmail.com`

Abstract

The success of long short-term memory (LSTM) neural networks in language processing is typically attributed to their ability to capture long-distance statistical regularities. Linguistic regularities are often sensitive to syntactic structure; can such dependencies be captured by LSTMs, which do not have explicit structural representations? We begin addressing this question using number agreement in English subject-verb dependencies. We probe the architecture's grammatical competence both using training objectives with an explicit grammatical target (number prediction, grammaticality judgments) and using language models. In the strongly supervised settings,

(Hochreiter and Schmidhuber, 1997) or gated recurrent units (GRU) (Cho et al., 2014), has led to significant gains in language modeling (Mikolov et al., 2010; Sundermeyer et al., 2012), parsing (Vinyals et al., 2015; Kiperwasser and Goldberg, 2016; Dyer et al., 2016), machine translation (Bahdanau et al., 2015) and other tasks.

The effectiveness of RNNs[1] is attributed to their ability to capture statistical contingencies that may span an arbitrary number of words. The word *France*, for example, is more likely to occur somewhere in a sentence that begins with *Paris* than in a sentence that begins with *Penguins*. The fact that an arbitrary number of words can intervene between the mutually predictive words implies that they cannot be captured

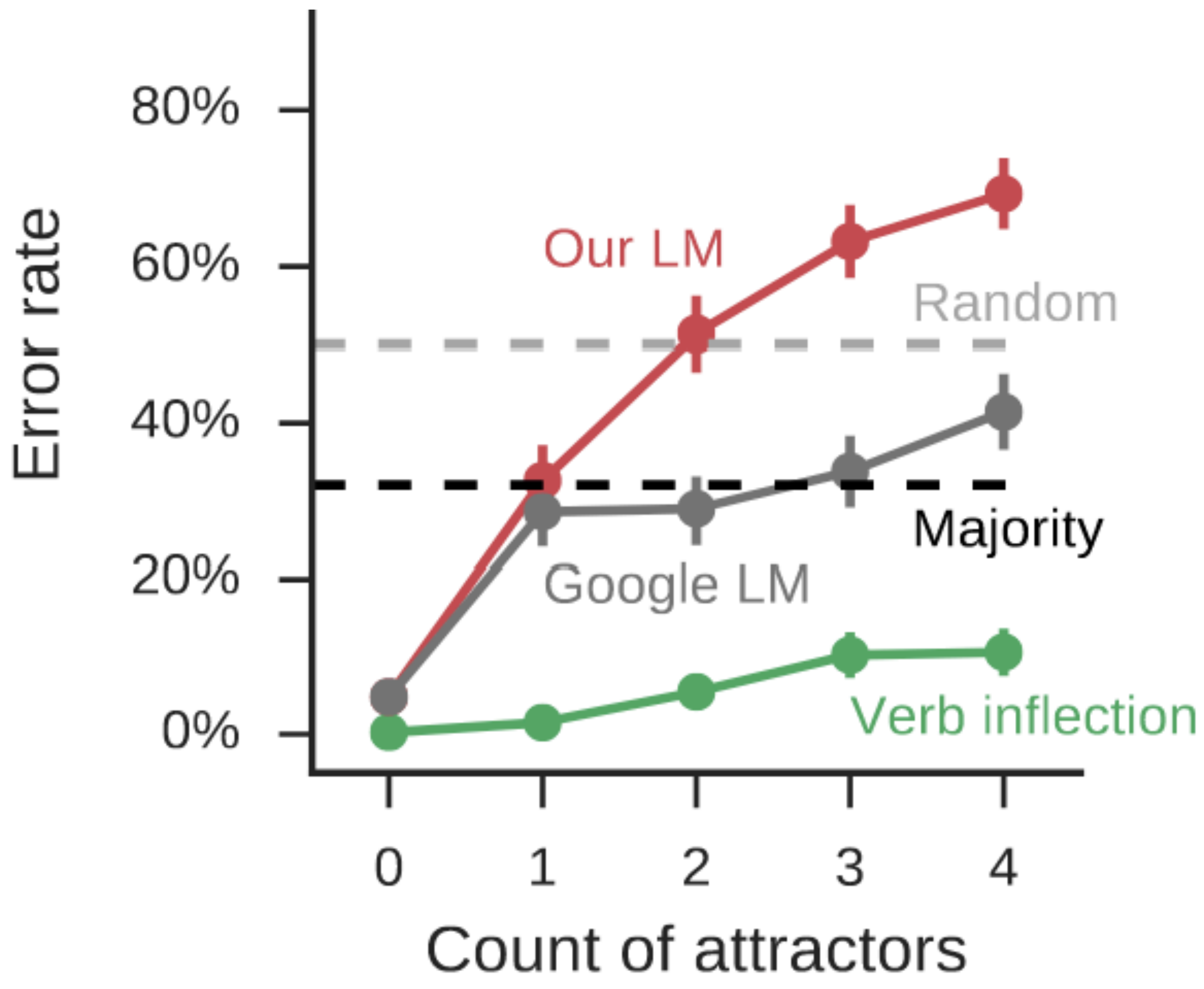
Subject-verb agreement

- Adjacent:
  - The key is on the table [SS]
  - * The key are on the table [SP]
  - * The keys is on the table [PS]
  - The keys are on the table [PP]
- Arbitrarily many *attractors* (nouns w/ different number) in between:
  - But even the **city** with several tall buildings and many thriving industries **is** struggling.

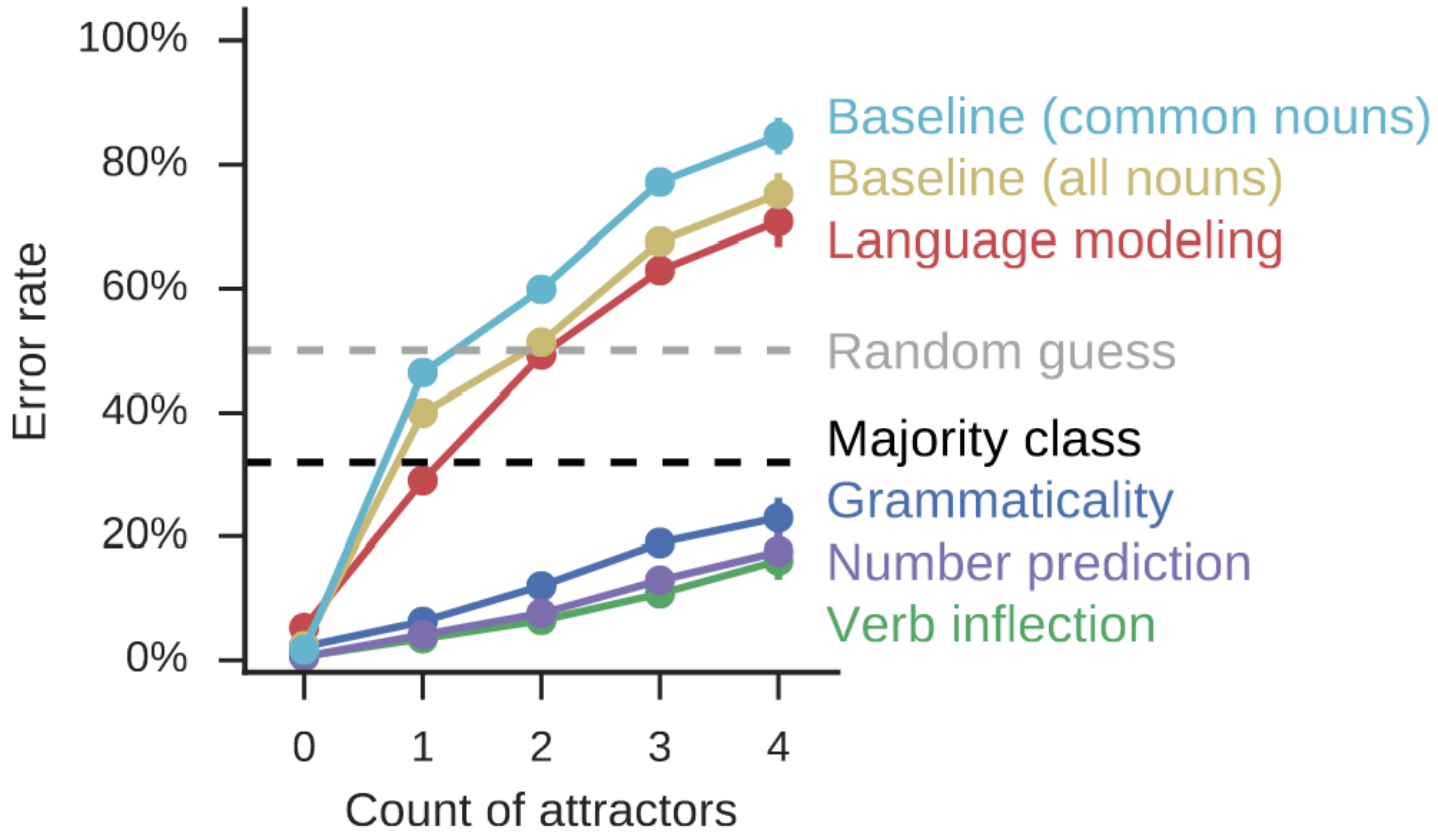
Method

- Does LM predict the right form of the verb?
  - "The keys on the cabinet …"
  - $P_{LM}(\text{are}) > P_{LM}(\text{is})?$
- Single layer LSTM w/ 50 hidden units
- NB: a lot more in the paper than we'll talk about here.
- Later: other methods for getting LM grammaticality judgments.

Accuracy vs. Attractors



Effect of Task



Take Home

- LSTMs can in general learn hierarchical dependencies
- But language modeling *may* not provide enough signal on its own
  - i.e. explicit supervision on the task is required

Colorless green recurrent networks dream hierarchically

Kristina Gulordava*
Department of Linguistics
University of Geneva
kristina.gulordava@unige.ch

Piotr Bojanowski
Facebook AI Research
Paris
bojanowski@fb.com

Edouard Grave
Facebook AI Research
New York
egrave@fb.com

Tal Linzen
Department of Cognitive Science
Johns Hopkins University
tal.linzen@jhu.edu

Marco Baroni
Facebook AI Research
Paris
mbaroni@fb.com

Abstract

Recurrent neural networks (RNNs) have achieved impressive results in a variety of linguistic processing tasks, suggesting that they can induce non-trivial properties of language. We investigate here to what extent RNNs learn to track abstract hierarchical syntactic structure. We test whether RNNs trained with a generic language modeling objective in four languages (Italian, English, Hebrew, Russian) can predict long-distance number agreement in various constructions. We include in our

achieved impressive results in large-scale tasks such as language modeling for speech recognition and machine translation, and are by now standard tools for sequential natural language tasks (e.g., Mikolov et al., 2010; Graves, 2012; Wu et al., 2016). This suggests that RNNs may learn to track grammatical structure even when trained on noisier natural data. The conjecture is supported by the success of RNNs as feature extractors for syntactic parsing (e.g., Cross and Huang, 2016; Kiperwasser and Goldberg, 2016; Zhang et al., 2017).

Innovations

- Same basic protocol, but:
  - More constructions / contexts to test agreement on
  - Multiple languages
  - Comparison to human judgments (in Italian)
  - Nonsense (nonce) constructions: think “colorless green ideas sleep furiously”
    - It **presents** the case for marriage equality and **states** …
    - It **stays** the shuttle for honesty insurance and **finds** …
- [Note: no “wug” / pseudo-words (“It blergs the shuttle …”); why not?]

Four languages; two constructions

		N V V	V NP conj V
Italian	Original	$93.3_{\pm 4.1}$	$83.3_{\pm 10.4}$
	Nonce	$92.5_{\pm 2.1}$	$78.5_{\pm 1.7}$
English	Original	$89.6_{\pm 3.6}$	$67.5_{\pm 5.2}$
	Nonce	$68.7_{\pm 0.9}$	$82.5_{\pm 4.8}$
Hebrew	Original	$86.7_{\pm 9.3}$	$83.3_{\pm 5.9}$
	Nonce	$65.7_{\pm 4.1}$	$83.1_{\pm 2.8}$
Russian	Original	-	$95.2_{\pm 1.9}$
	Nonce	-	$86.7_{\pm 1.6}$

Four languages; two constructions

		N V V	V NP conj V
Italian	Original	$93.3_{\pm 4.1}$	$83.3_{\pm 10.4}$
	Nonce	$92.5_{\pm 2.1}$	$78.5_{\pm 1.7}$
English	Original	$89.6_{\pm 3.6}$	$67.5_{\pm 5.2}$
	Nonce	$68.7_{\pm 0.9}$	$82.5_{\pm 4.8}$
Hebrew	Original	$86.7_{\pm 9.3}$	$83.3_{\pm 5.9}$
	Nonce	$65.7_{\pm 4.1}$	$83.1_{\pm 2.8}$
Russian	Original	-	$95.2_{\pm 1.9}$
	Nonce	-	$86.7_{\pm 1.6}$

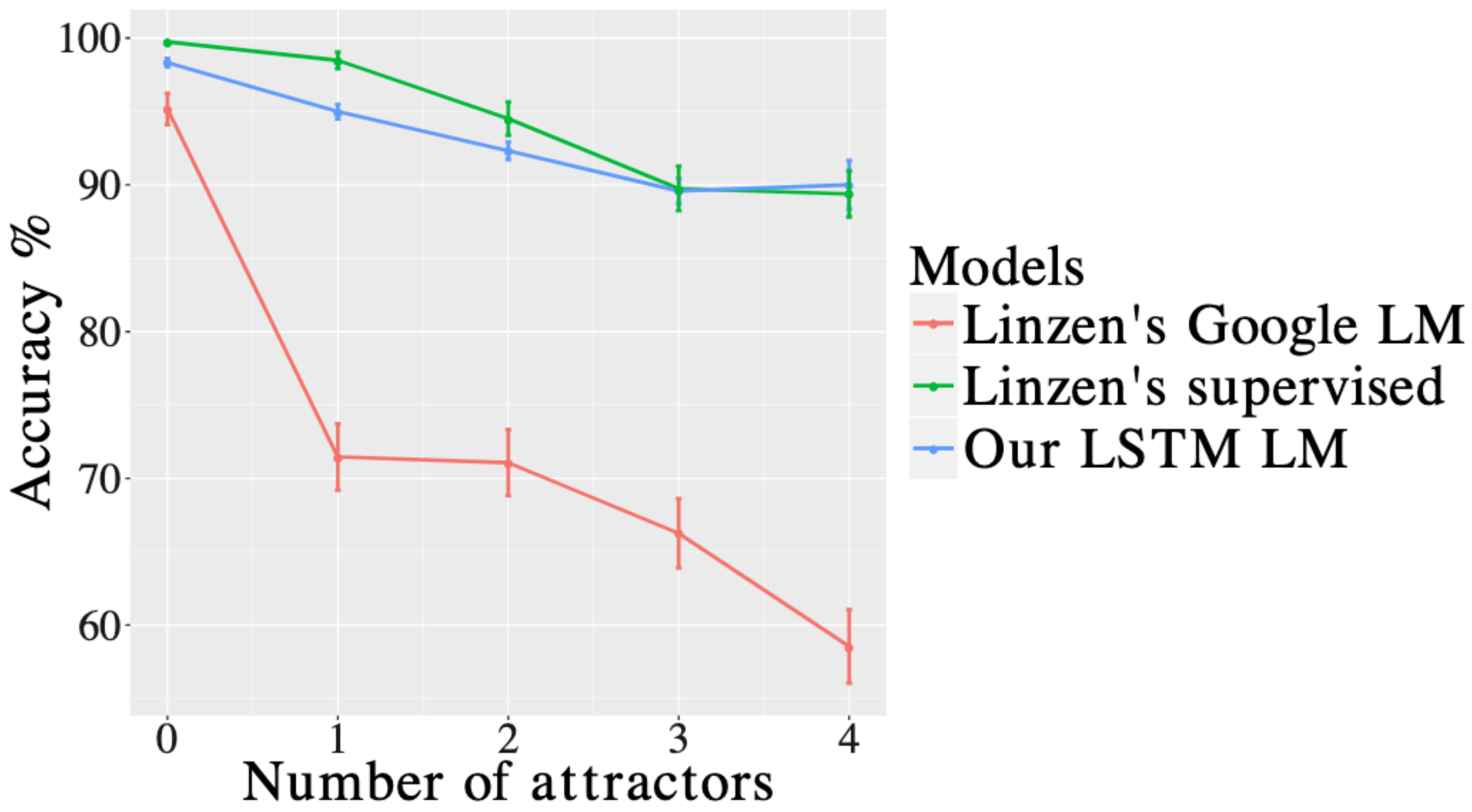
Maybe English's poor morphology and high POS ambiguity:
"If you **have** any questions or **need/needs**, …"

Comparison with Italians

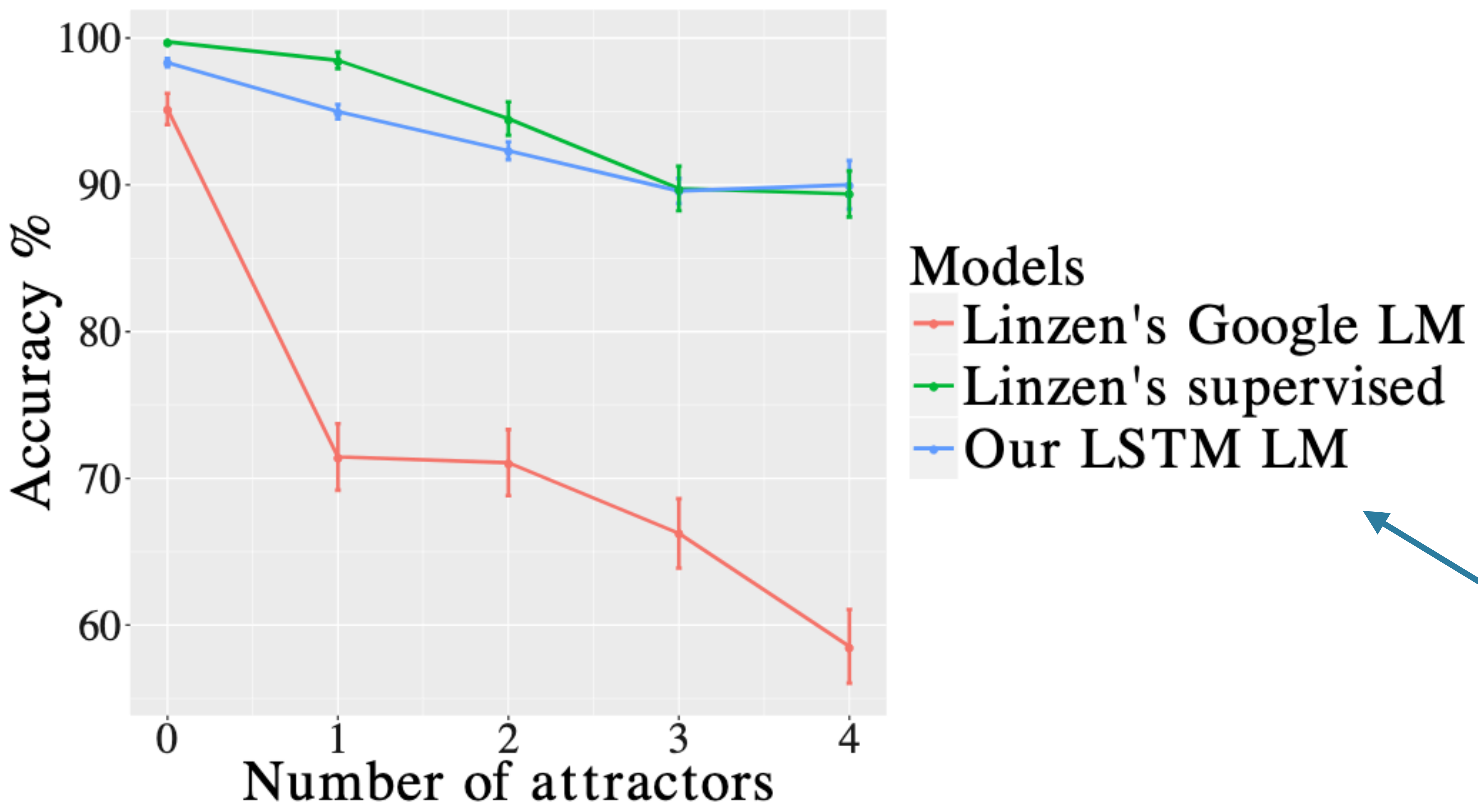
Construction	#original	Original Subjects	Original LSTM	Nonce Subjects	Nonce LSTM
DET [AdjP] NOUN	14	98.7	$98.6_{\pm 3.2}$	98.1	$91.7_{\pm 0.4}$
NOUN [RelC / PartP] clitic VERB	6	93.1	$100_{\pm 0.0}$	95.4	$97.8_{\pm 0.8}$
NOUN [RelC / PartP] VERB	27	97.0	$93.3_{\pm 4.1}$	92.3	$92.5_{\pm 2.1}$
ADJ [conjoined ADJs] ADJ	13	98.5	$100_{\pm 0.0}$	98.0	$98.1_{\pm 1.1}$
NOUN [AdjP] relpron VERB	10	95.9	$98.0_{\pm 4.5}$	89.5	$84.0_{\pm 3.3}$
NOUN [PP] ADVERB ADJ	13	91.5	$98.5_{\pm 3.4}$	79.4	$76.9_{\pm 1.4}$
NOUN [PP] VERB (participial)	18	87.1	$77.8_{\pm 3.9}$	73.4	$71.1_{\pm 3.3}$
VERB [NP] CONJ VERB	18	94.0	$83.3_{\pm 10.4}$	86.8	$78.5_{\pm 1.7}$
(Micro) average		94.5	$92.1_{\pm 1.6}$	88.4	$85.5_{\pm 0.7}$

Table 3: Subject and LSTM accuracy on the Italian test set, by construction and averaged.

On the Linzen et al 2016 Data



On the Linzen et al 2016 Data



Be careful with what you can conclude from one experiment!

Take Home

- Language modeling *may* after all provide enough of a signal to learn hierarchical syntactic dependencies
  - But may be very sensitive to hyper-parameters, including training data
  - [NB: the Gulordava et al model is a lot smaller than the Google LM]
  - “suggests that the input itself contains enough information to trigger some form of syntactic learning in a system, such as an RNN, that does not contain an explicit prior bias in favour of syntactic structures”
- Good model and data (we’ve used in several projects) to play with (https://github.com/facebookresearch/colorlessgreenRNNs)
- A follow-up, with more constructions than just subject/verb agreement, and artificially generated data: https://www.aclweb.org/anthology/D18-1151/

Diagnostic classifiers (“probing”)

Main Idea

- What's in a representation (a vector)? How can we tell?
- For example: does an LSTM's memory encode grammatical number?
  - If we're lucky: a single cell might, as we saw earlier. (Sparse representation)
  - In general: *if we can easily predict the number from the memory*, it's "already in there".
- Given a representation, train a simple model (usually a linear classifier) to predict a property of interest (usually linguistic) from that representation.

Note on Terminology

Journal of Artificial Intelligence Research 61 (2018) 907-926 Submitted 10/17; published 04/18

Visualisation and 'Diagnostic Classifiers' Reveal how Recurrent and Recursive Neural Networks Process Hierarchical Structure

Dieuwke Hupkes D.HUPKES@UVA.NL
Sara Veldhoen SARA.VELDHOEN@GMAIL.COM
Willem Zuidema ZUIDEMA@UVA.NL
ILLC, University of Amsterdam
P.O.Box 94242
1090 CE Amsterdam, Netherlands

- Roughly synonyms: diagnostic classifiers, probing classifiers, auxiliary prediction tasks, …
- [Basically: very simple transfer learning, with frozen "base model"]

Linguistic Knowledge and Transferability of Contextual Representations

Nelson F. Liu[♠♡*] **Matt Gardner**[♣] **Yonatan Belinkov**[◇]
Matthew E. Peters[♣] **Noah A. Smith**[♠♣]
[♠]Paul G. Allen School of Computer Science & Engineering,
University of Washington, Seattle, WA, USA
[♡]Department of Linguistics, University of Washington, Seattle, WA, USA
[♣]Allen Institute for Artificial Intelligence, Seattle, WA, USA
[◇]Harvard John A. Paulson School of Engineering and Applied Sciences and
MIT Computer Science and Artificial Intelligence Laboratory, Cambridge, MA, USA
{nfliu,nasmith}@cs.washington.edu
{mattg,matthewp}@allenai.org, belinkov@seas.harvard.edu

Abstract

Contextual word representations derived from large-scale neural language models are successful across a diverse set of NLP tasks, suggesting that they encode useful and transferable features of language. To shed light on the linguistic knowledge they capture, we study the representations produced by several recent pretrained contextualizers (variants of ELMo, the OpenAI transformer language model, and BERT) with a suite of sixteen diverse probing tasks. We find that linear models trained on top of frozen contextual repre-

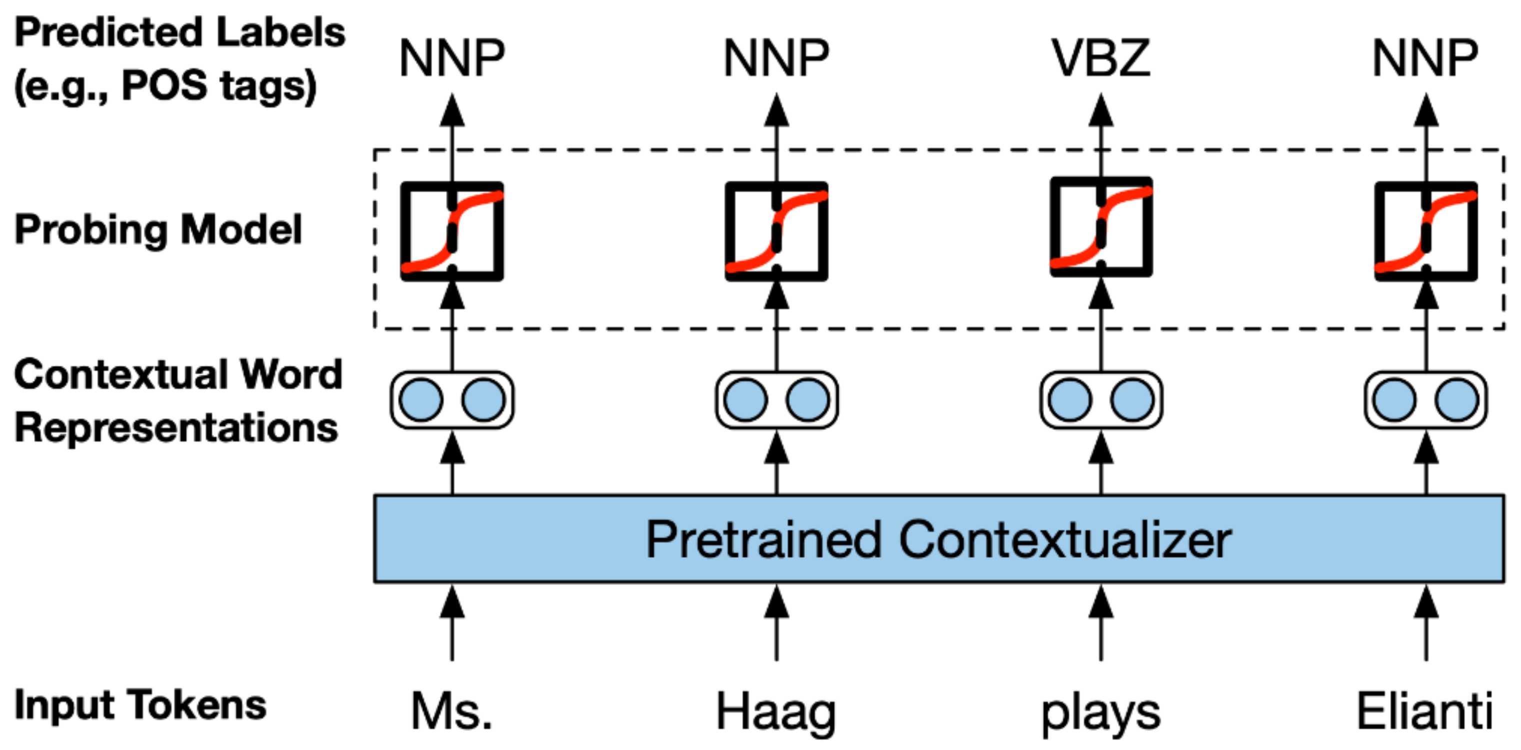


Figure 1: An illustration of the probing model setup used to study the linguistic knowledge within contextual word representations.

Tagging Results

Pretrained Representation		POS						Supersense ID		
	Avg.	CCG	PTB	EWT	Chunk	NER	ST	GED	PS-Role	PS-Fxn	EF
ELMo (original) best layer	81.58	93.31	97.26	95.61	90.04	82.85	93.82	29.37	75.44	84.87	73.20
ELMo (4-layer) best layer	81.58	93.81	**97.31**	95.60	89.78	82.06	**94.18**	29.24	74.78	85.96	73.03
ELMo (transformer) best layer	80.97	92.68	97.09	95.13	93.06	81.21	93.78	30.80	72.81	82.24	70.88
OpenAI transformer best layer	75.01	82.69	93.82	91.28	86.06	58.14	87.81	33.10	66.23	76.97	74.03
BERT (base, cased) best layer	84.09	93.67	96.95	95.21	92.64	82.71	93.72	43.30	**79.61**	87.94	75.11
BERT (large, cased) best layer	**85.07**	**94.28**	96.73	**95.80**	**93.64**	**84.44**	93.83	**46.46**	79.17	**90.13**	**76.25**
GloVe (840B.300d)	59.94	71.58	90.49	83.93	62.28	53.22	80.92	14.94	40.79	51.54	49.70
Previous state of the art (without pretraining)	83.44	94.7	97.96	95.82	95.77	91.38	95.15	39.83	66.89	78.29	77.10

Tagging Results

Pretrained Representation			POS						Supersense ID		
	Avg.	CCG	PTB	EWT	Chunk	NER	ST	GED	PS-Role	PS-Fxn	EF
ELMo (original) best layer	81.58	93.31	97.26	95.61	90.04	82.85	93.82	29.37	75.44	84.87	73.20
ELMo (4-layer) best layer	81.58	93.81	**97.31**	95.60	89.78	82.06	**94.18**	29.24	74.78	85.96	73.03
ELMo (transformer) best layer	80.97	92.68	97.09	95.13	93.06	81.21	93.78	30.80	72.81	82.24	70.88
OpenAI transformer best layer	75.01	82.69	93.82	91.28	86.06	58.14	87.81	33.10	66.23	76.97	74.03
BERT (base, cased) best layer	84.09	93.67	96.95	95.21	92.64	82.71	93.72	43.30	**79.61**	87.94	75.11
BERT (large, cased) best layer	**85.07**	**94.28**	96.73	**95.80**	**93.64**	**84.44**	93.83	**46.46**	79.17	**90.13**	**76.25**
GloVe (840B.300d)	59.94	71.58	90.49	83.93	62.28	53.22	80.92	14.94	40.79	51.54	49.70
Previous state of the art (without pretraining)	83.44	94.7	97.96	95.82	95.77	91.38	95.15	39.83	66.89	78.29	77.10

Context matters!

Coreference

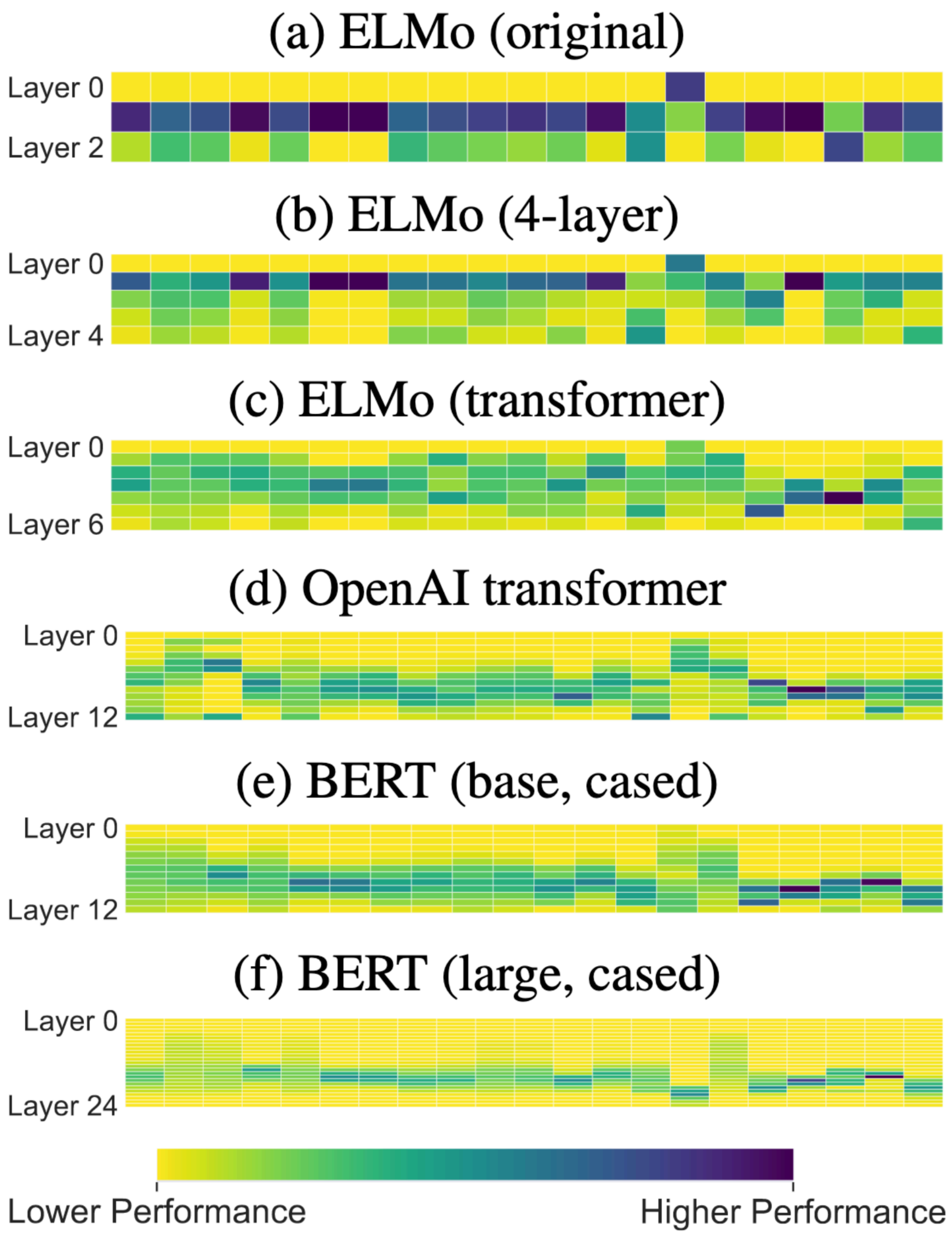
D.5 Pairwise Relations (ELMo and OpenAI Transformer)

Pretrained Representation	Syntactic Dep. Arc Prediction		Syntactic Dep. Arc Classification		Semantic Dep. Arc Prediction	Semantic Dep. Arc Classification	Coreference Arc Prediction
	PTB	EWT	PTB	EWT			
ELMo (original), Layer 0	78.27	77.73	82.05	78.52	70.65	77.48	72.89
ELMo (original), Layer 1	89.04	86.46	96.13	93.01	87.71	93.31	71.33
ELMo (original), Layer 2	88.33	85.34	94.72	91.32	86.44	90.22	68.46
ELMo (original), Scalar Mix	89.30	86.56	95.81	91.69	87.79	93.13	73.24
ELMo (4-layer), Layer 0	78.09	77.57	82.13	77.99	69.96	77.22	73.57
ELMo (4-layer), Layer 1	88.79	86.31	96.20	93.20	87.15	93.27	72.93
ELMo (4-layer), Layer 2	87.33	84.75	95.38	91.87	85.29	90.57	71.78
ELMo (4-layer), Layer 3	86.74	84.17	95.06	91.55	84.44	90.04	70.11
ELMo (4-layer), Layer 4	87.61	85.09	94.14	90.68	85.81	89.45	68.36
ELMo (4-layer), Scalar Mix	88.98	85.94	95.82	91.77	87.39	93.25	73.88
ELMo (transformer), Layer 0	78.10	78.04	81.09	77.67	70.11	77.11	72.50
ELMo (transformer), Layer 1	88.24	85.48	93.62	89.18	85.16	90.66	72.47
ELMo (transformer), Layer 2	88.87	84.72	94.14	89.40	85.97	91.29	73.03
ELMo (transformer), Layer 3	89.01	84.62	94.07	89.17	86.83	90.35	72.62
ELMo (transformer), Layer 4	88.55	85.62	94.14	89.00	86.00	89.04	71.80
ELMo (transformer), Layer 5	88.09	83.23	92.70	88.84	85.79	89.66	71.62
ELMo (transformer), Layer 6	87.22	83.28	92.55	87.13	84.71	87.21	66.35
ELMo (transformer), Scalar Mix	90.74	86.39	96.40	91.06	89.18	94.35	75.52
OpenAI transformer, Layer 0	80.80	79.10	83.35	80.32	76.39	80.50	72.58
OpenAI transformer, Layer 1	81.91	79.99	88.22	84.51	77.70	83.88	75.23
OpenAI transformer, Layer 2	82.56	80.22	89.34	85.99	78.47	85.85	75.77
OpenAI transformer, Layer 3	82.87	81.21	90.89	87.67	78.91	87.76	75.81
OpenAI transformer, Layer 4	83.69	82.07	92.21	89.24	80.51	89.59	75.99
OpenAI transformer, Layer 5	84.53	82.77	93.12	90.34	81.95	90.25	76.05
OpenAI transformer, Layer 6	85.47	83.89	93.71	90.63	83.88	90.99	74.43
OpenAI transformer, Layer 7	86.32	84.15	93.95	90.82	85.15	91.18	74.05
OpenAI transformer, Layer 8	86.84	84.06	94.16	91.02	85.23	90.86	74.20
OpenAI transformer, Layer 9	87.00	84.47	93.95	90.77	85.95	90.85	74.57
OpenAI transformer, Layer 10	86.76	84.28	93.40	90.26	85.17	89.94	73.86
OpenAI transformer, Layer 11	85.84	83.42	92.82	89.07	83.39	88.46	72.03
OpenAI transformer, Layer 12	85.06	83.02	92.37	89.08	81.88	87.47	70.44
OpenAI transformer, Scalar Mix	87.18	85.30	94.51	91.55	86.13	91.55	76.47
GloVe (840B.300d)	74.14	73.94	77.54	72.74	68.94	71.84	72.96

Table 9: Pairwise relation task performance of a linear probing model trained on top of the ELMo and OpenAI contextualizers, compared against a GloVe-based probing baseline.

No significant improvement over global embedding baseline [BERT does a bit better, so adirectionality seems to matter]

Layer-wise Prediction



(each column is a different task)

Effect of Pretraining Task

Pretraining Task	Layer Average Target Task Performance			
	0	1	2	Mix
CCG	56.70	64.45	63.71	66.06
Chunk	54.27	62.69	63.25	63.96
POS	56.21	63.86	64.15	65.13
Parent	54.57	62.46	61.67	64.31
GParent	55.50	62.94	62.91	64.96
GGParent	54.83	61.10	59.84	63.81
Syn. Arc Prediction	53.63	59.94	58.62	62.43
Syn. Arc Classification	56.15	64.41	63.60	66.07
Sem. Arc Prediction	53.19	54.69	53.04	59.84
Sem. Arc Classification	56.28	62.41	61.47	64.67
Conj	50.24	49.93	48.42	56.92
BiLM	66.53	65.91	65.82	66.49
GloVe (840B.300d)	60.55			
Untrained ELMo (original)	52.14	39.26	39.39	54.42
ELMo (original) (BiLM on 1B Benchmark)	64.40	79.05	77.72	78.90

- See also:
  - Zhang and Bowman 2018
  - Peters et al 2018b
  - Blevins et al 2018

WHAT DO YOU LEARN FROM CONTEXT? PROBING FOR SENTENCE STRUCTURE IN CONTEXTUALIZED WORD REPRESENTATIONS

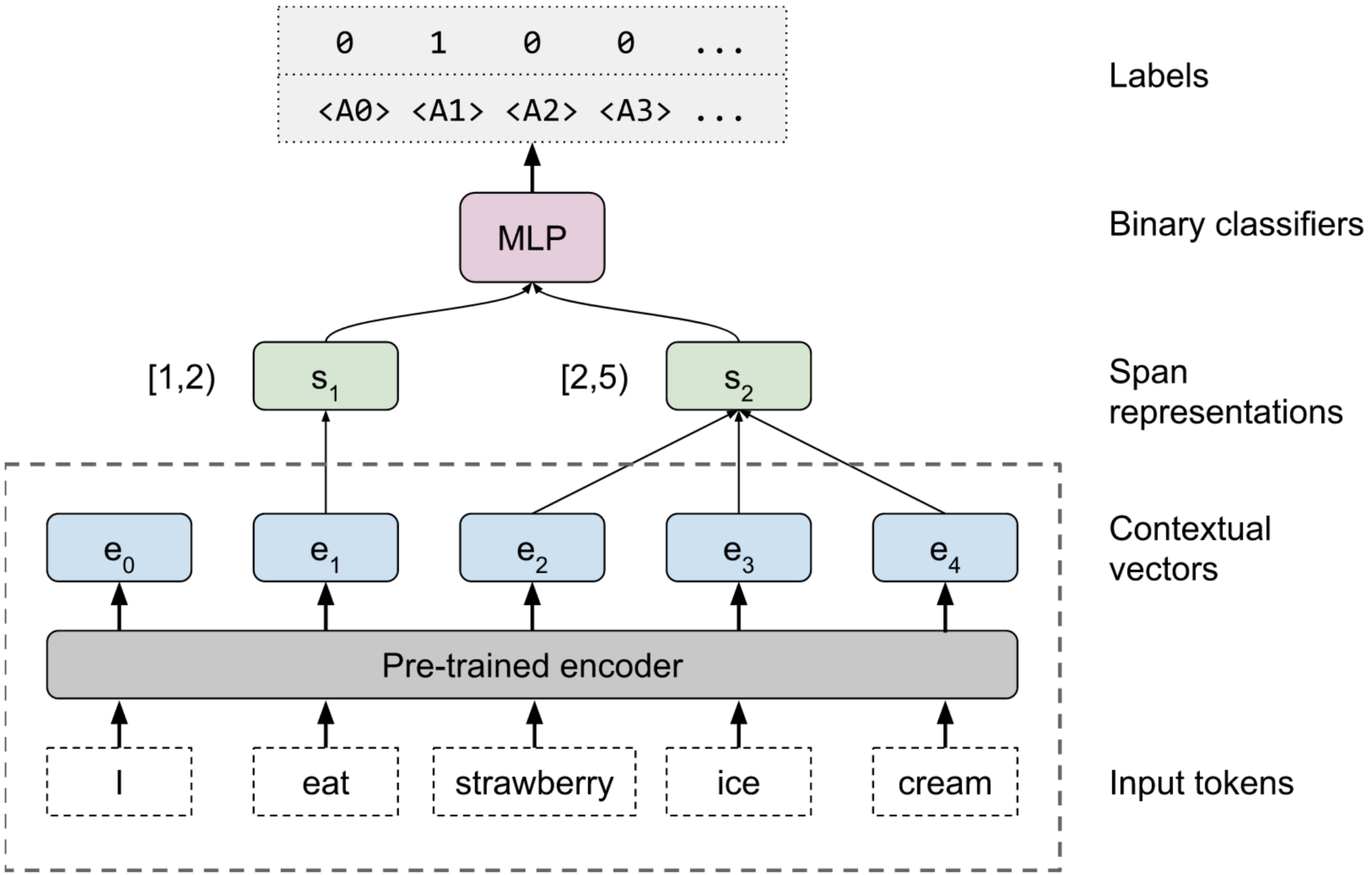
Ian Tenney,[*1] **Patrick Xia,**[2] **Berlin Chen,**[3] **Alex Wang,**[4] **Adam Poliak,**[2]
R. Thomas McCoy,[2] **Najoung Kim,**[2] **Benjamin Van Durme,**[2] **Samuel R. Bowman,**[4]
Dipanjan Das,[1] **and Ellie Pavlick**[1,5]

[1]Google AI Language, [2]Johns Hopkins University, [3]Swarthmore College,
[4]New York University, [5]Brown University

ABSTRACT

Contextualized representation models such as ELMo (Peters et al., 2018a) and BERT (Devlin et al., 2018) have recently achieved state-of-the-art results on a diverse array of downstream NLP tasks. Building on recent token-level probing work, we introduce a novel *edge probing* task design and construct a broad suite of sub-sentence tasks derived from the traditional structured NLP pipeline. We probe word-level contextual representations from four recent models and investigate how they encode sentence structure across a range of syntactic, semantic, local, and long-range phenomena. We find that existing models trained on language modeling and translation produce strong representations for syntactic phenomena, but only offer comparably small improvements on semantic tasks over a non-contextual baseline.

Edge Probing Set-up



Results

	CoVe Lex.	CoVe Full	CoVe Abs. Δ	ELMo Lex.	ELMo Full	ELMo Abs. Δ	GPT Lex.	GPT cat	GPT mix
Part-of-Speech	85.7	94.0	8.4	90.4	**96.7**	6.3	88.2	94.9	95.0
Constituents	56.1	81.6	25.4	69.1	**84.6**	15.4	65.1	81.3	**84.6**
Dependencies	75.0	83.6	8.6	80.4	**93.9**	13.6	77.7	92.1	**94.1**
Entities	88.4	90.3	1.9	92.0	**95.6**	3.5	88.6	92.9	92.5
SRL (all)	59.7	80.4	20.7	74.1	**90.1**	16.0	67.7	86.0	89.7
Core roles	*56.2*	*81.0*	*24.7*	*73.6*	***92.6***	*19.0*	*65.1*	*88.0*	*92.0*
Non-core roles	*67.7*	*78.8*	*11.1*	*75.4*	***84.1***	*8.8*	*73.9*	*81.3*	***84.1***
OntoNotes coref.	72.9	79.2	6.3	75.3	84.0	8.7	71.8	83.6	**86.3**
SPR1	73.7	77.1	3.4	80.1	**84.8**	4.7	79.2	83.5	83.1
SPR2	76.6	80.2	3.6	82.1	83.1	1.0	82.2	**83.8**	**83.5**
Winograd coref.	52.1	**54.3**	2.2	**54.3**	**53.5**	-0.8	51.7	52.6	**53.8**
Rel. (SemEval)	51.0	60.6	9.6	55.7	77.8	22.1	58.2	**81.3**	**81.0**
Macro Average	69.1	78.1	9.0	75.4	**84.4**	9.1	73.0	83.2	**84.4**

	BERT-base F1 Score Lex.	BERT-base F1 Score cat	BERT-base F1 Score mix	BERT-base Abs. Δ ELMo	BERT-large F1 Score Lex.	BERT-large F1 Score cat	BERT-large F1 Score mix	BERT-large Abs. Δ (base)	BERT-large Abs. Δ ELMo
Part-of-Speech	88.4	**97.0**	96.7	0.0	88.1	96.5	**96.9**	0.2	0.2
Constituents	68.4	83.7	86.7	2.1	69.0	80.1	**87.0**	0.4	2.5
Dependencies	80.1	93.0	95.1	1.1	80.2	91.5	**95.4**	0.3	1.4
Entities	90.9	96.1	96.2	0.6	91.8	96.2	**96.5**	0.3	0.9
SRL (all)	75.4	89.4	91.3	1.2	76.5	88.2	**92.3**	1.0	2.2
Core roles	*74.9*	*91.4*	*93.6*	*1.0*	*76.3*	*89.9*	***94.6***	*1.0*	*2.0*
Non-core roles	*76.4*	*84.7*	*85.9*	*1.8*	*76.9*	*84.1*	***86.9***	*1.0*	*2.8*
OntoNotes coref.	74.9	88.7	90.2	6.3	75.7	89.6	**91.4**	1.2	7.4
SPR1	79.2	84.7	**86.1**	1.3	79.6	85.1	**85.8**	-0.3	1.0
SPR2	81.7	83.0	**83.8**	0.7	81.6	83.2	**84.1**	0.3	1.0
Winograd coref.	54.3	53.6	54.9	1.4	53.0	53.8	**61.4**	6.5	7.8
Rel. (SemEval)	57.4	78.3	82.0	4.2	56.2	77.6	**82.4**	0.5	4.6
Macro Average	75.1	84.8	86.3	1.9	75.2	84.2	**87.3**	1.0	2.9

Conclusion

- “in general, contextualized embeddings improve over their non-contextualized counterparts largely on syntactic tasks (e.g. constituent labeling) in comparison to semantic tasks (e.g. coreference), suggesting that these embeddings encode syntax more so than higher-level semantics”

Is it in the probe or the representation?

Designing and Interpreting Probes with Control Tasks

John Hewitt
Stanford University
johnhew@stanford.edu

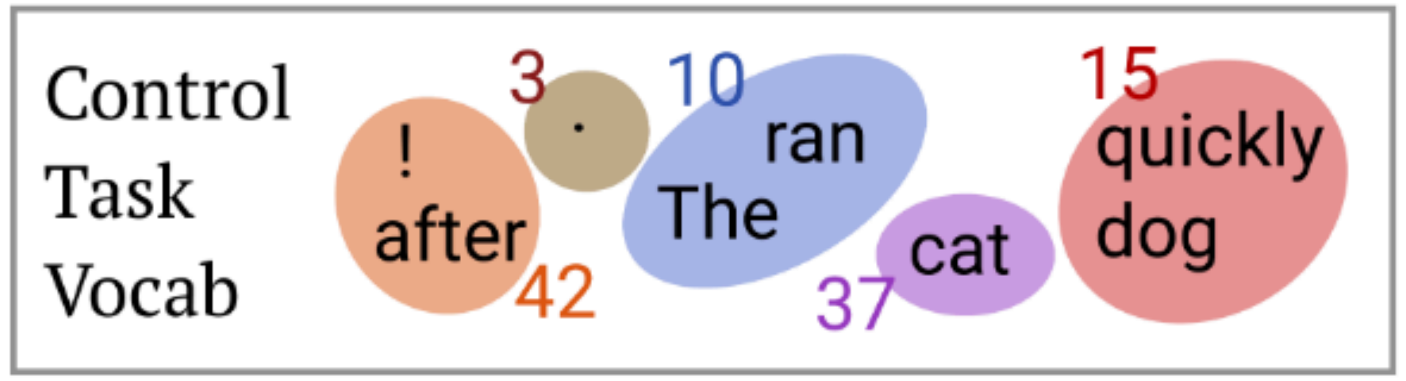
Percy Liang
Stanford University
pliang@cs.stanford.edu

Is it in the probe or the representation?

Designing and Interpreting Probes with Control Tasks

John Hewitt
Stanford University
johnhew@stanford.edu

Percy Liang
Stanford University
pliang@cs.stanford.edu



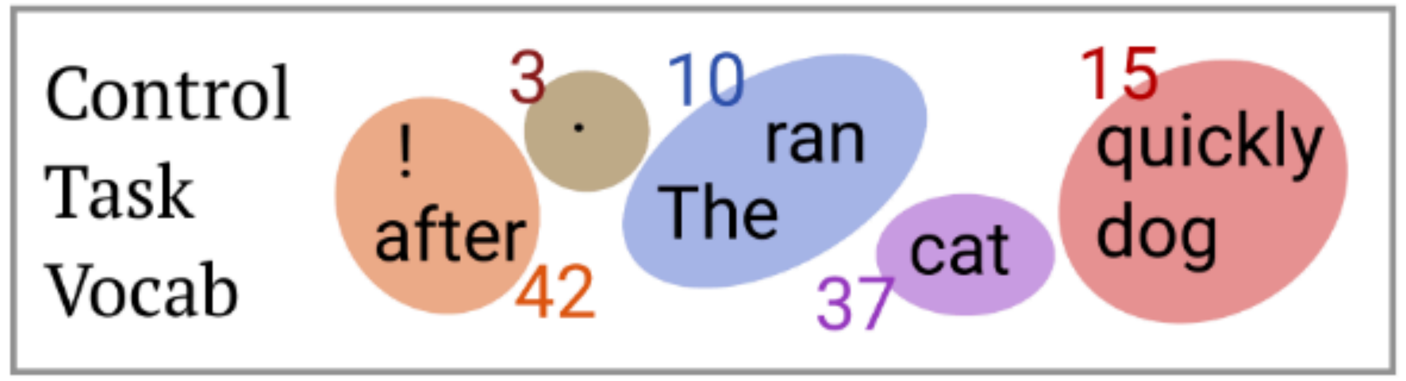
Sentence 1	The	cat	ran	quickly	.
Part-of-speech	DT	NN	VBD	RB	.
Control task	**10**	**37**	**10**	**15**	**3**
Sentence 2	The	dog	ran	after	!
Part-of-speech	DT	NN	VBD	IN	.
Control task	**10**	**15**	**10**	**42**	**42**

Is it in the probe or the representation?

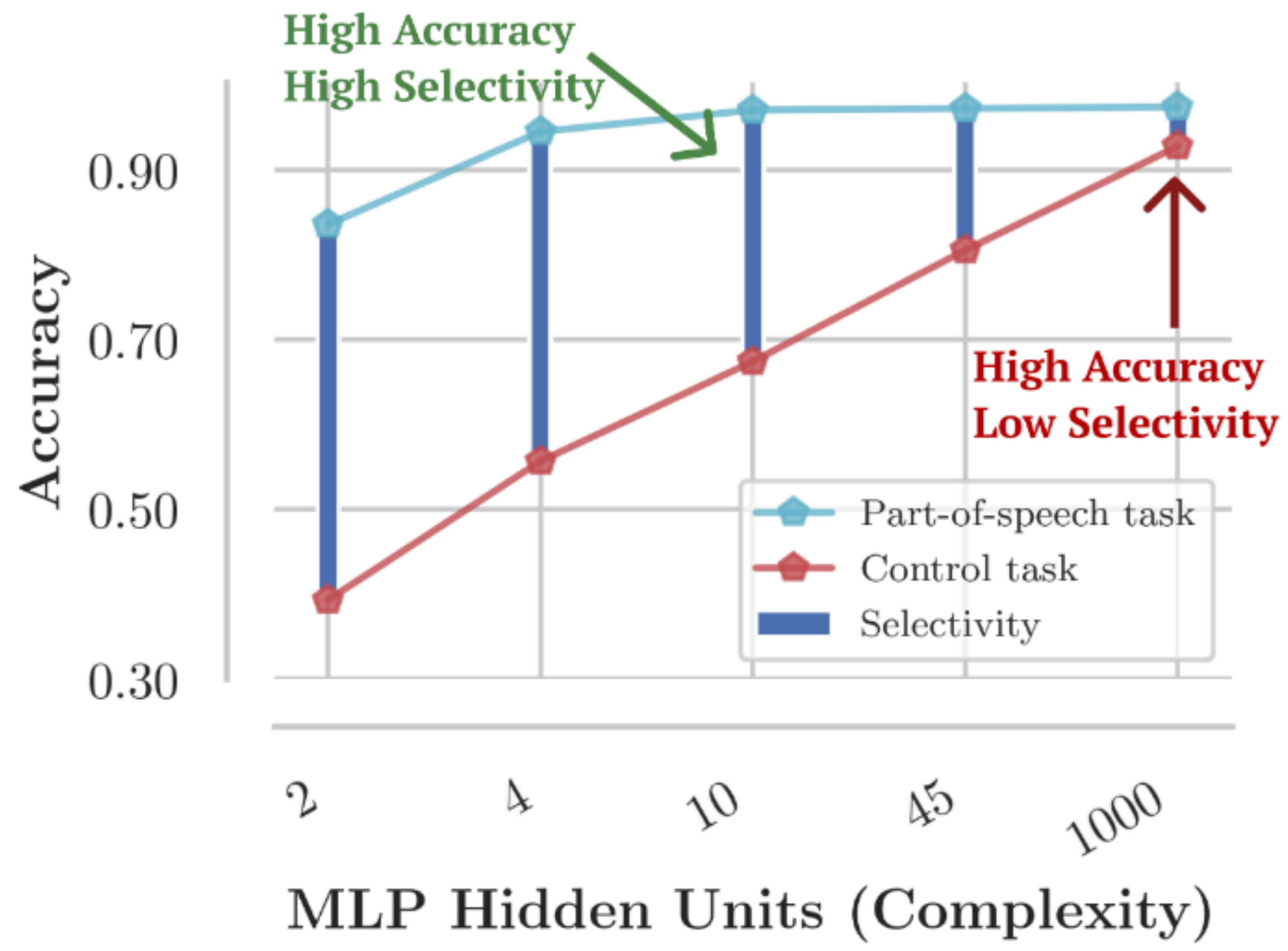
Designing and Interpreting Probes with Control Tasks

John Hewitt
Stanford University
johnhew@stanford.edu

Percy Liang
Stanford University
pliang@cs.stanford.edu



Sentence 1	The	cat	ran	quickly	.
Part-of-speech	DT	NN	VBD	RB	.
Control task	**10**	**37**	**10**	**15**	**3**
Sentence 2	The	dog	ran	after	!
Part-of-speech	DT	NN	VBD	IN	.
Control task	**10**	**15**	**10**	**42**	**42**



Summary

- Use simple classifiers to see what can be extracted from a model's representations.
- Some clear trends:
  - Contextualized representations have more info than global ones (GloVe e.g.)
    - Especially for syntax
  - Layer-wise: early recurrent layers are more transferrable, less clear on Transformers
  - Language modeling a very good task for building transferrable representations

Summary, cont.

- Promises:
  - Lets us learn what's encoded in a model's opaque representation
- Shortcomings:
  - Comparison/control (cf H+L)
  - Correlation vs causation: encoding != used by the model
    - New methods try to overcome this ("amnesic probing", causal model inference)

Probing Classifiers: Promises, Shortcomings, and Advances

Yonatan Belinkov*
Technion – Israel Institute of Technology
belinkov@technion.ac.il

Probing classifiers have emerged as one of the prominent methodologies for interpreting and analyzing deep neural network models of natural language processing. The basic idea is simple—a classifier is trained to predict some linguistic property from a model's representations—and has been used to examine a wide variety of models and properties. However, recent studies have demonstrated various methodological limitations of this approach. This squib critically reviews the probing classifiers framework, highlighting their promises, shortcomings, and advances.

Attention-based

What does BERT look at? An Analysis of BERT's Attention

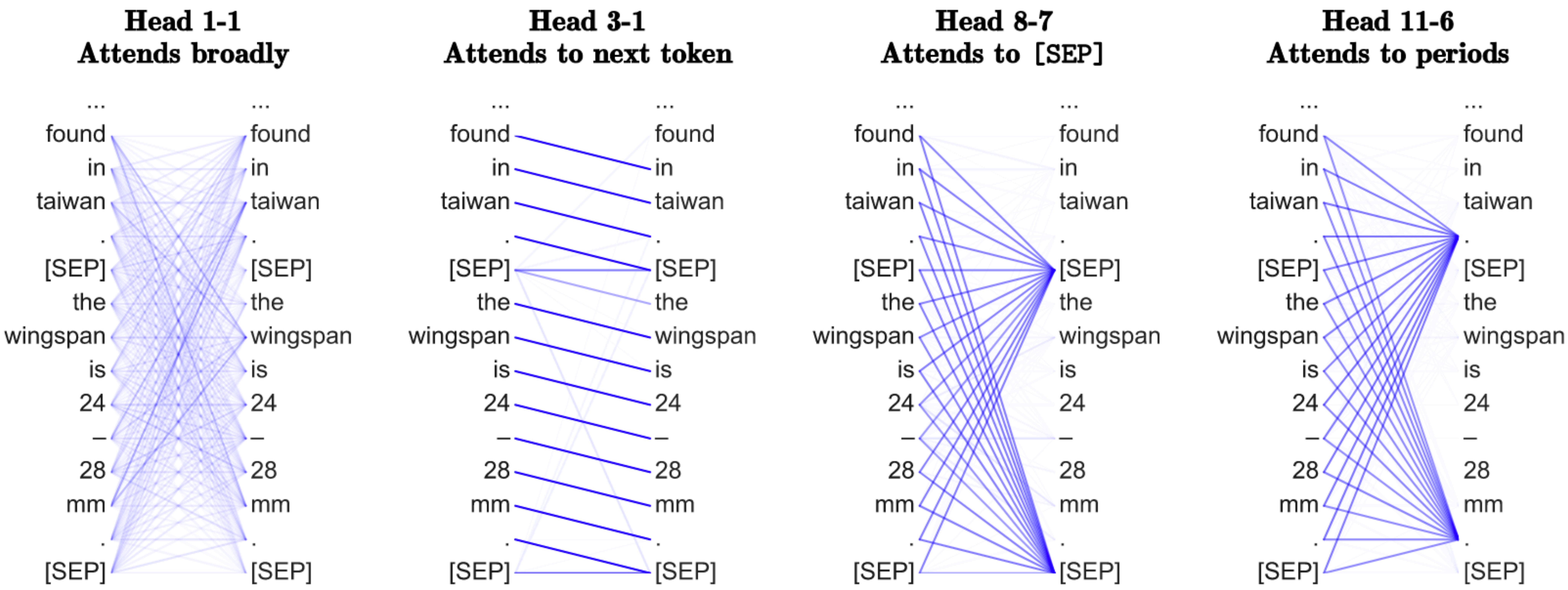
Kevin Clark[†] **Urvashi Khandelwal**[†] **Omer Levy**[‡] **Christopher D. Manning**[†]
[†]Computer Science Department, Stanford University
[‡]Facebook AI Research
{kevclark,urvashik,manning}@cs.stanford.edu
omerlevy@fb.com

Abstract

Large pre-trained neural networks such as BERT have had great recent success in NLP, motivating a growing body of research investigating what aspects of language they are able to learn from unlabeled data. Most recent analysis has focused on model outputs (e.g., lan-

study[1] the *attention maps* of a pre-trained model. Attention (Bahdanau et al., 2015) has been a highly successful neural network component. It is naturally interpretable because an attention weight has a clear meaning: how much a particular word will be weighted when computing the next representation for the current word. Our analysis fo-

Qualitative Patterns



Attention Head as Classifier

- No new training required
- Do any of these work for pairwise classification tasks “off-the-shelf”?

Attention Head as Classifier

$$\alpha_j = q \cdot k_j$$

$$e_j = e^{\alpha_j} / \Sigma_j e^{\alpha_j}$$

$$c = \Sigma_j e_j v_j$$

$$\text{class}(q) = \arg\max_j \alpha_j$$

- No new training required
- Do any of these work for pairwise classification tasks “off-the-shelf”?

Dependency Parsing

Relation	Head	Accuracy	Baseline
All	7-6	34.5	26.3 (1)
`prep`	7-4	66.7	61.8 (-1)
`pobj`	9-6	**76.3**	34.6 (-2)
`det`	8-11	**94.3**	51.7 (1)
`nn`	4-10	70.4	70.2 (1)
`nsubj`	8-2	58.5	45.5 (1)
`amod`	4-10	75.6	68.3 (1)
`dobj`	8-10	**86.8**	40.0 (-2)
`advmod`	7-6	48.8	40.2 (1)
`aux`	4-10	81.1	71.5 (1)
`poss`	7-6	**80.5**	47.7 (1)
`auxpass`	4-10	**82.5**	40.5 (1)
`ccomp`	8-1	**48.8**	12.4 (-2)
`mark`	8-2	**50.7**	14.5 (2)
`prt`	6-7	**99.1**	91.4 (-1)

Coreference

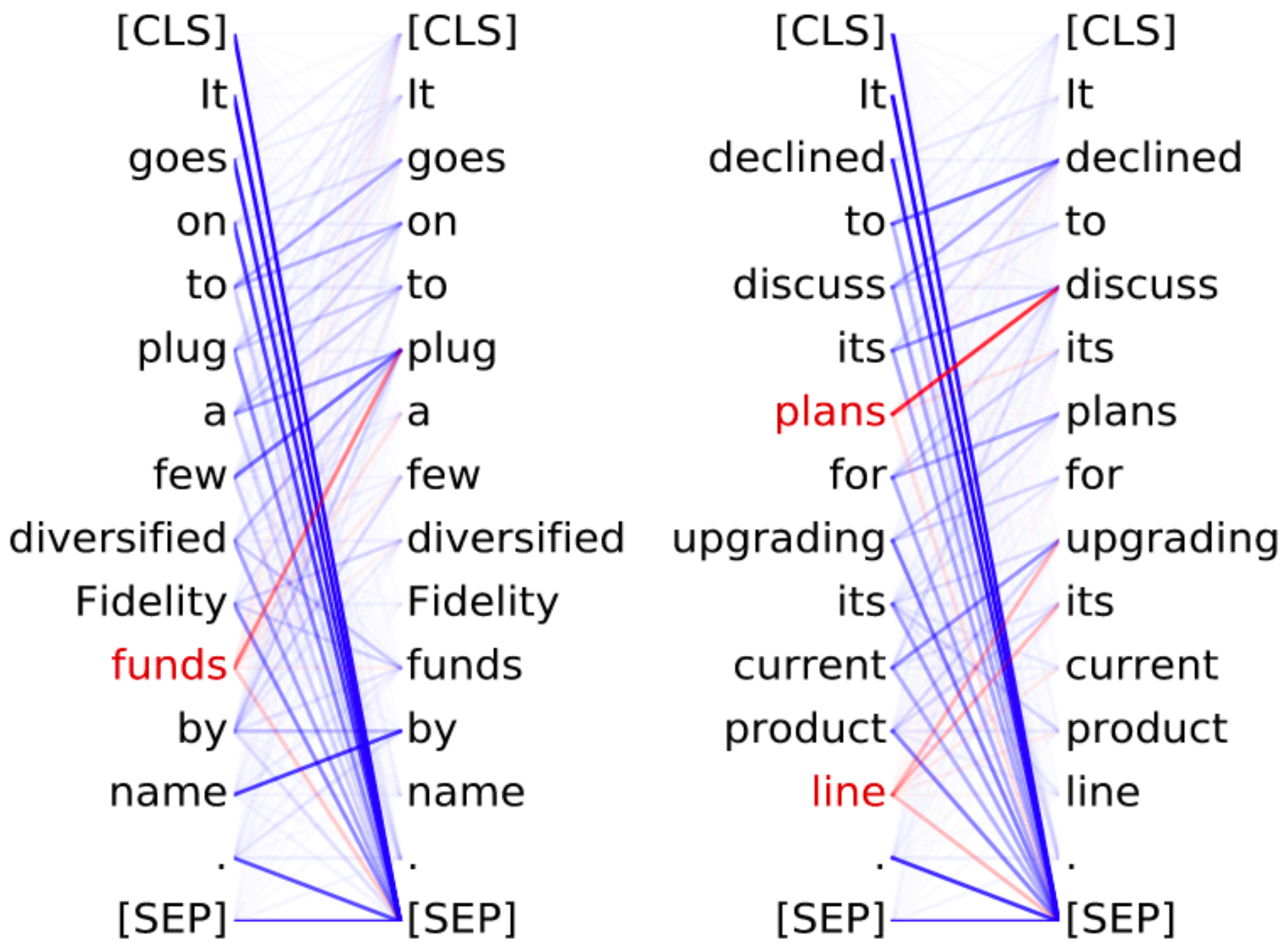
Model	All	Pronoun	Proper	Nominal
Nearest	27	29	29	19
Head-word match	52	47	67	40
Rule-based	69	70	77	60
Neural coref	83*	–	–	–
Head 5-4	65	64	73	58

*Only roughly comparable because on non-truncated documents and with different mention detection.

Examples

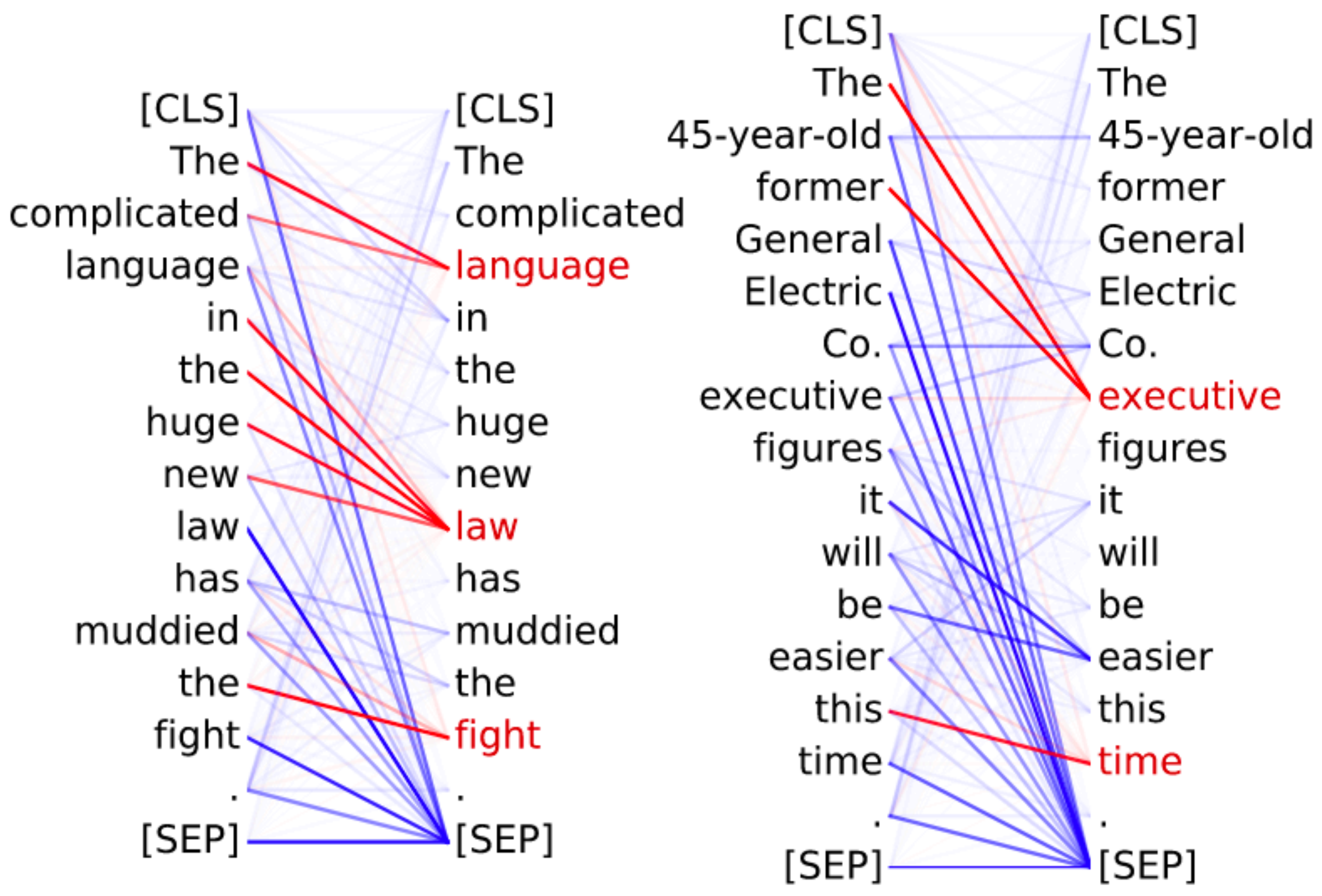
Head 8-10

- **Direct objects** attend to their verbs
- 86.8% accuracy at the `dobj` relation



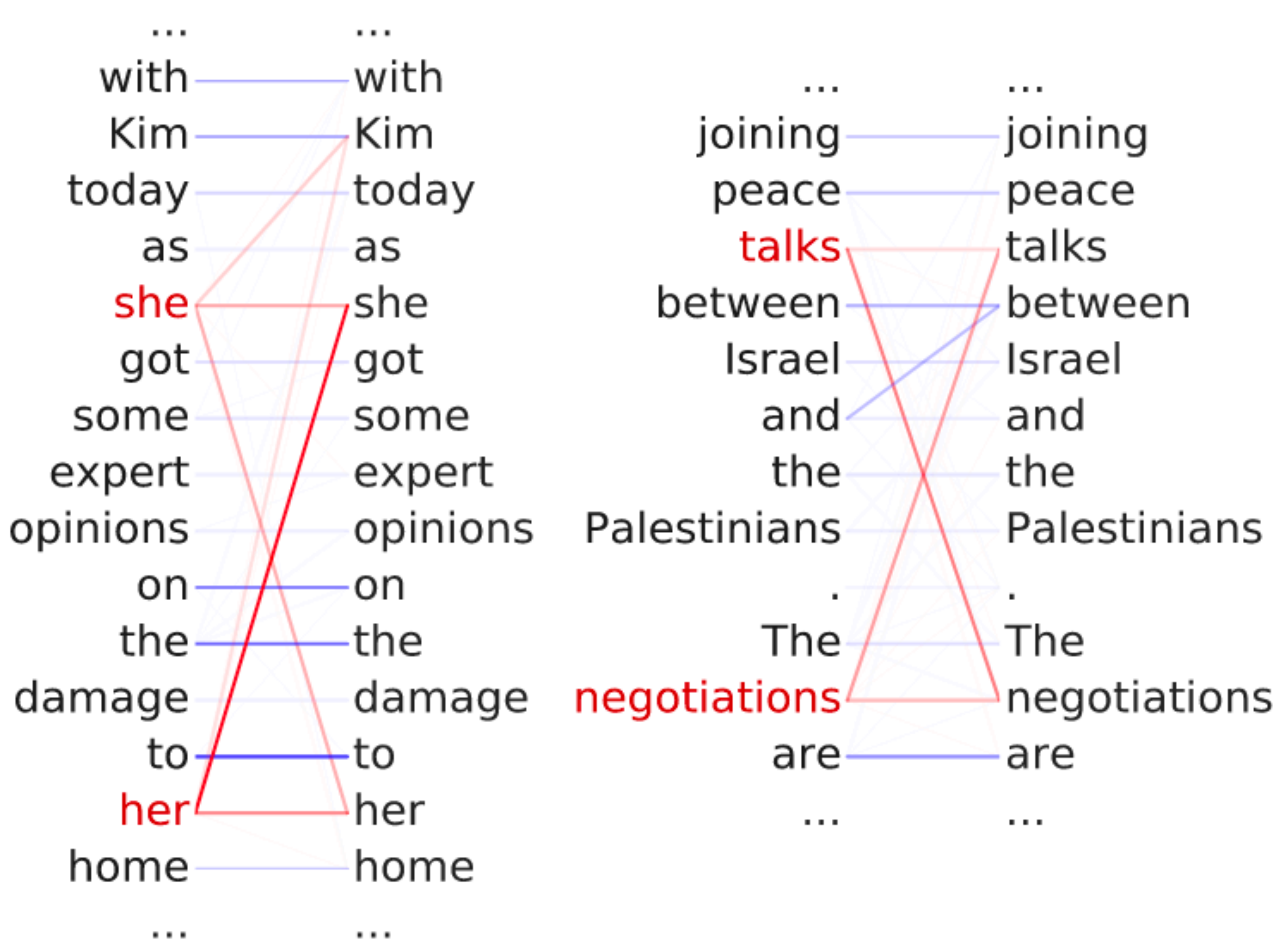
Head 8-11

- **Noun modifiers** (e.g., determiners) attend to their noun
- 94.3% accuracy at the `det` relation



Head 5-4

- **Coreferent** mentions attend to their antecedents
- 65.1% accuracy at linking the head of a coreferent mention to the head of an antecedent



Revealing the Dark Secrets of BERT

Olga Kovaleva, Alexey Romanov, Anna Rogers, Anna Rumshisky
Department of Computer Science
University of Massachusetts Lowell
Lowell, MA 01854
{okovalev,arum,aromanov}@cs.uml.edu

Abstract

BERT-based architectures currently give state-of-the-art performance on many NLP tasks, but little is known about the exact mechanisms that contribute to its success. In the current work, we focus on the interpretation of self-attention, which is one of the fundamental underlying components of BERT. Using a subset of GLUE tasks and a set of handcrafted features-of-interest, we propose the methodology and carry out a qualitative and quantita-

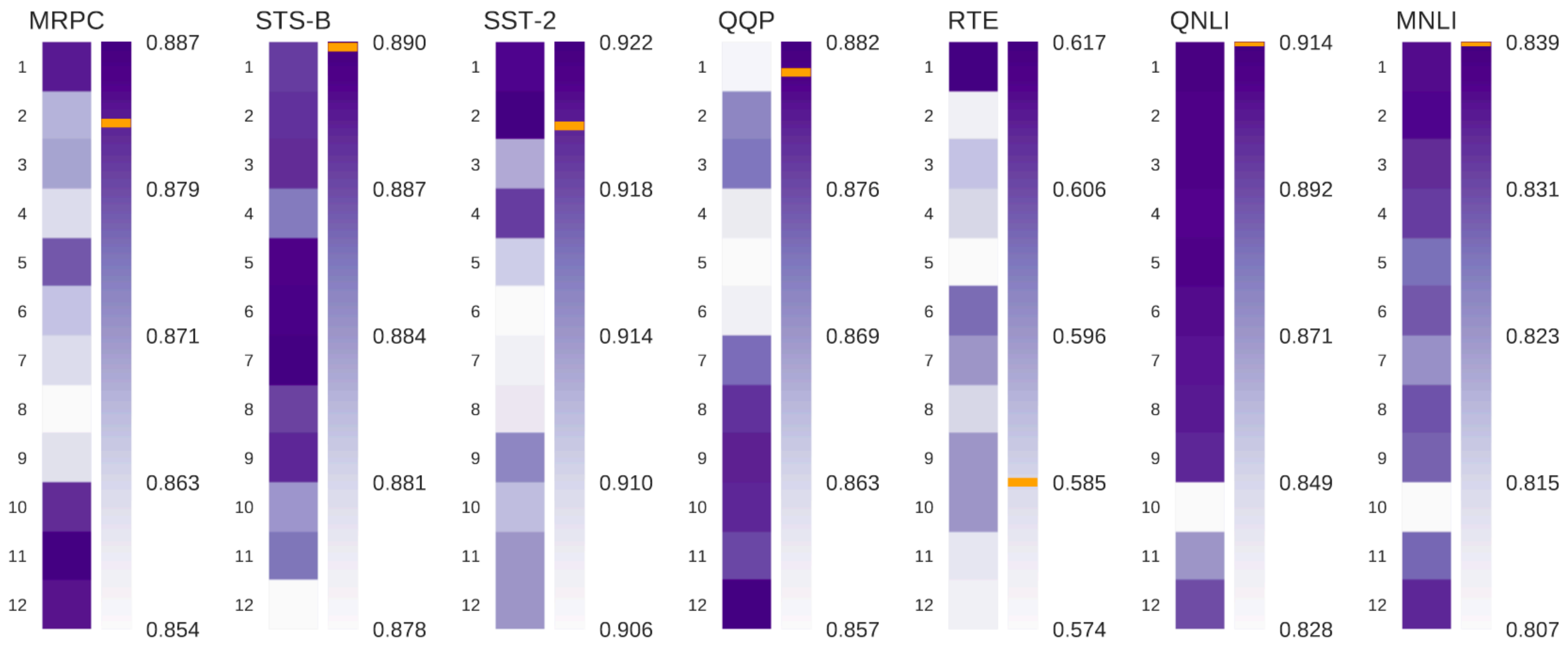
inference. State-of-the-art performance is usually obtained by fine-tuning the pre-trained model on the specific task. In particular, BERT-based models are currently dominating the leaderboards for SQuAD[1] (Rajpurkar et al., 2016) and GLUE benchmarks[2] (Wang et al., 2018).

However, the exact mechanisms that contribute to the BERT's outstanding performance still remain unclear. We address this problem through selecting a set of linguistic features of interest and

Overall

- Same observation as previous: many heads only pay attention to [SEP] and [CLS] tokens
- Changes in attention before and after fine-tuning
- Pruning some heads can actually improve performance (see also Voita et al on the original Transformer)

Pruning all attention in a layer



NB: pay attention to the scales

Summary

- Sometimes, attention heads seem to encode some linguistically interesting properties
  - But there appears to be lots of redundancy
  - And there’s much more terrain to explore here
- As before: we can ask if property P can be found in attention, but not what role (independently of a hypothesis) a head is playing
- For the curious: ongoing debate about the connection between attention and model predictions (not as applied to LMs yet): Attention is not explanation; Attention is not not explanation

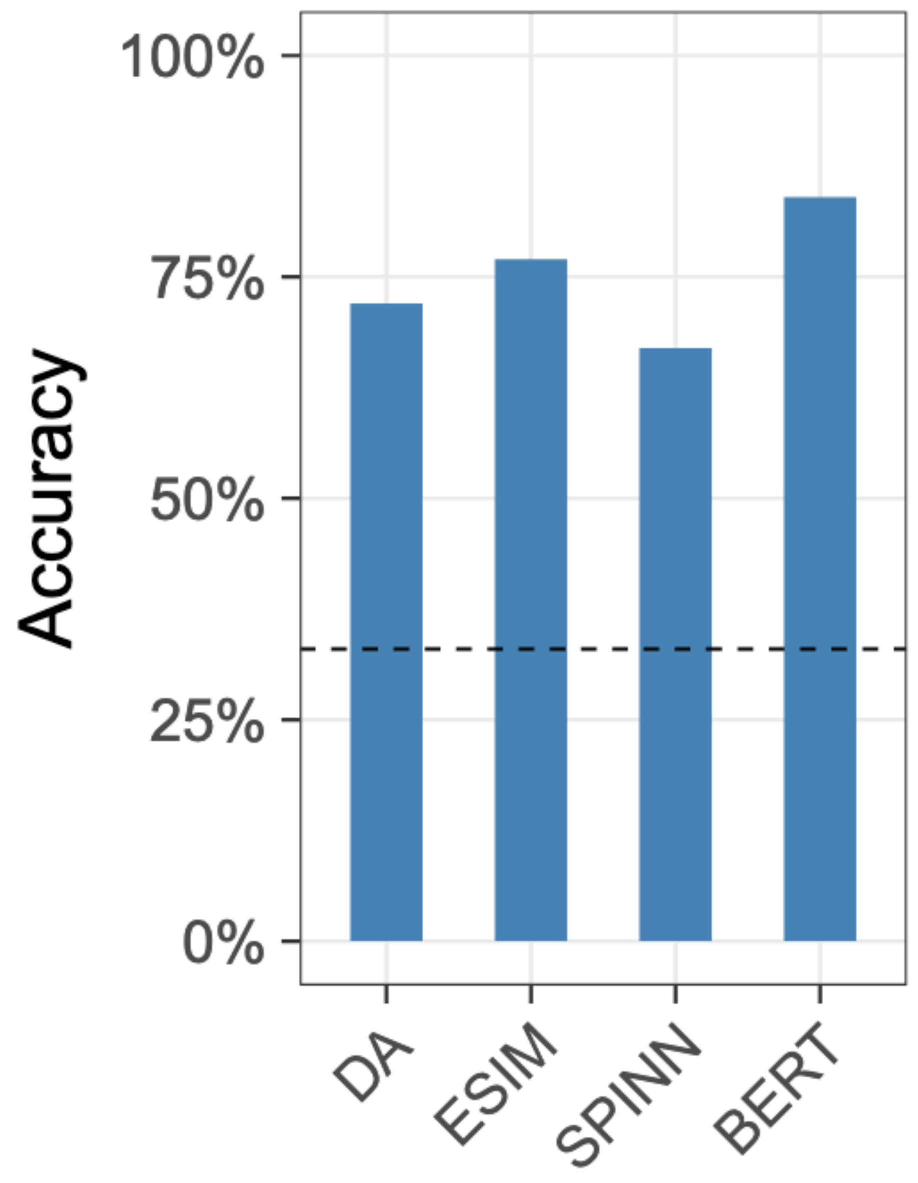
Adversarial Datasets

Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference

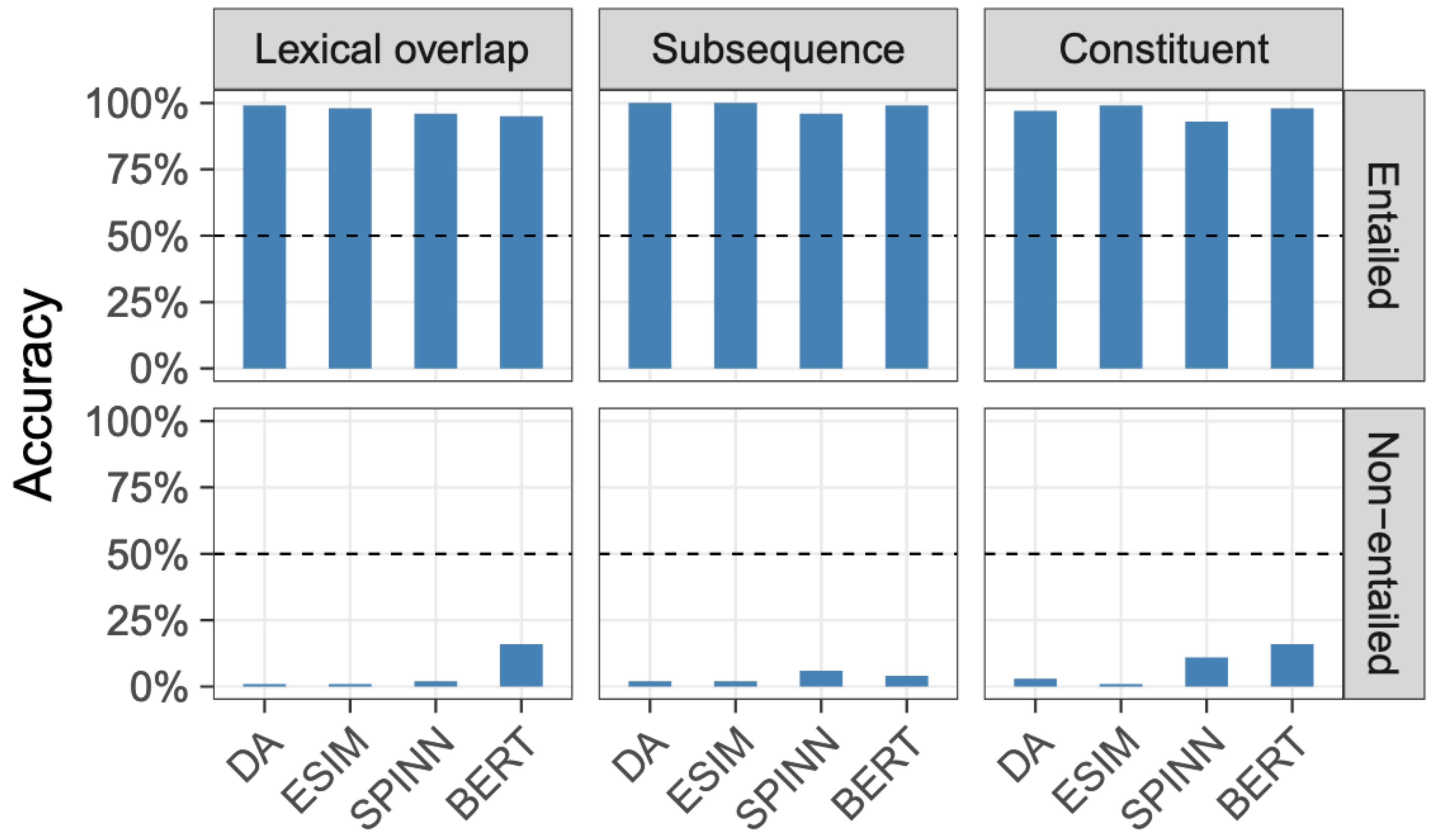
R. Thomas McCoy,[1] Ellie Pavlick,[2] & Tal Linzen[1]
[1]Department of Cognitive Science, Johns Hopkins University
[2]Department of Computer Science, Brown University
tom.mccoy@jhu.edu, ellie_pavlick@brown.edu, tal.linzen@jhu.edu

Heuristic	Premise	Hypothesis	Label
Lexical overlap heuristic	The banker near the judge saw the actor.	The banker saw the actor.	E
	The lawyer was advised by the actor.	The actor advised the lawyer.	E
	The doctors visited the lawyer.	The lawyer visited the doctors.	N
	The judge by the actor stopped the banker.	The banker stopped the actor.	N
Subsequence heuristic	The artist and the student called the judge.	The student called the judge.	E
	Angry tourists helped the lawyer.	Tourists helped the lawyer.	E
	The judges heard the actors resigned.	The judges heard the actors.	N
	The senator near the lawyer danced.	The lawyer danced.	N
Constituent heuristic	Before the actor slept, the senator ran.	The actor slept.	E
	The lawyer knew that the judges shouted.	The judges shouted.	E
	If the actor slept, the judge saw the artist.	The actor slept.	N
	The lawyers resigned, or the artist slept.	The artist slept.	N

Results



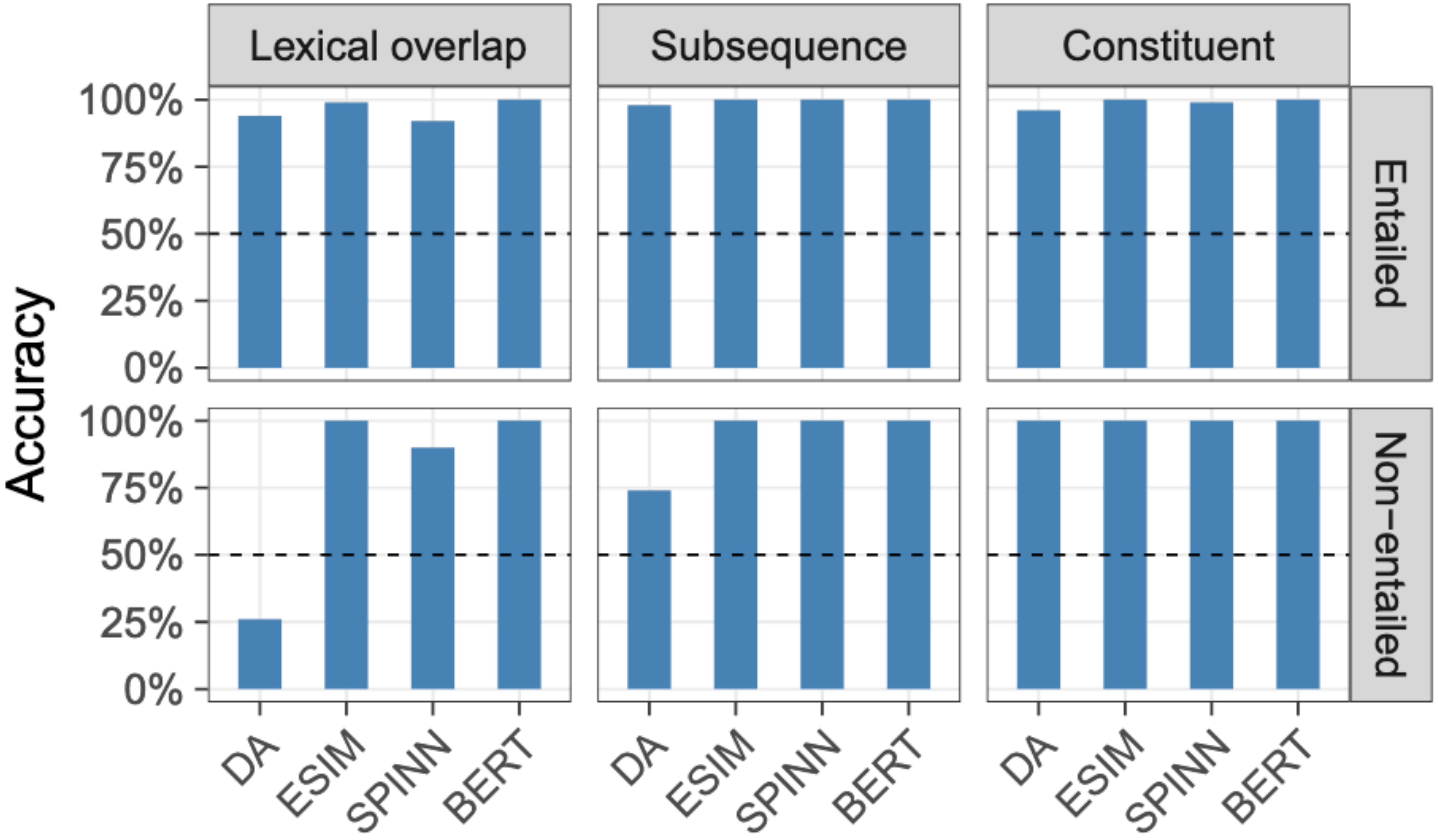
(a)



(b)

(performance improves if fine-tuned on this challenge set)

Fine-tuning augmented with examples



Conclusion

- Solving a dataset != solving a task
  - Models are very powerful, can be very “clever”
  - Adopt heuristics that exploit spurious cues in the data
- Careful design of “adversarial” data can both expose the heuristics being relied on and hopefully improve the representations learned

Probing Neural Network Comprehension of Natural Language Arguments

Timothy Niven and **Hung-Yu Kao**

Intelligent Knowledge Management Lab
Department of Computer Science and Information Engineering
National Cheng Kung University
Tainan, Taiwan
tim.niven.public@gmail.com, hykao@mail.ncku.edu.tw

Abstract

We are surprised to find that BERT's peak performance of 77% on the Argument Reasoning Comprehension Task reaches just three points below the average untrained human baseline. However, we show that this result is entirely accounted for by exploitation of spurious statistical cues in the dataset. We analyze the nature of these cues and demonstrate that a range of models all exploit them. This analysis informs the construction of an adversarial dataset on which all models achieve random accuracy. Our adversarial dataset provides a

Claim	Google is not a harmful monopoly
Reason	People can choose not to use Google
Warrant	Other search engines don't redirect to Google
Alternative	All other search engines redirect to Google

$$\textbf{Reason} \text{ (and since) } \textbf{Warrant} \rightarrow \textbf{Claim}$$
$$\textbf{Reason} \text{ (but since) } \textbf{Alternative} \rightarrow \neg\ \textbf{Claim}$$

Figure 1: An example of a data point from the ARCT test set and how it should be read. The inference from R and A to $\neg C$ is by design.

The Argument Reasoning Comprehension Task (ARCT) (Habernal et al., 2018a) defers the prob-

Results, with and w/o adversarial set

	Test		
	Mean	**Median**	**Max**
BERT	**0.671** $\pm$ 0.09	**0.712**	**0.770**
BERT (W)	0.656 $\pm$ 0.05	0.675	0.712
BERT (R, W)	0.600 $\pm$ 0.10	0.574	0.750
BERT (C, W)	0.532 $\pm$ 0.09	0.503	0.732
BoV	0.564 $\pm$ 0.02	0.569	0.595
BoV (W)	0.567 $\pm$ 0.02	0.572	0.606
BoV (R, W)	0.554 $\pm$ 0.02	0.557	0.579
BoV (C, W)	0.545 $\pm$ 0.02	0.544	0.589
BiLSTM	0.552 $\pm$ 0.02	0.552	0.592
BiLSTM (W)	0.550 $\pm$ 0.02	0.547	0.577
BiLSTM (R, W)	0.547 $\pm$ 0.02	0.551	0.577
BiLSTM (C, W)	0.552 $\pm$ 0.02	0.550	0.601

Results, with and w/o adversarial set

	Test		
	Mean	**Median**	**Max**
BERT	**0.671** ± 0.09	**0.712**	**0.770**
BERT (W)	0.656 ± 0.05	0.675	0.712
BERT (R, W)	0.600 ± 0.10	0.574	0.750
BERT (C, W)	0.532 ± 0.09	0.503	0.732
BoV	0.564 ± 0.02	0.569	0.595
BoV (W)	0.567 ± 0.02	0.572	0.606
BoV (R, W)	0.554 ± 0.02	0.557	0.579
BoV (C, W)	0.545 ± 0.02	0.544	0.589
BiLSTM	0.552 ± 0.02	0.552	0.592
BiLSTM (W)	0.550 ± 0.02	0.547	0.577
BiLSTM (R, W)	0.547 ± 0.02	0.551	0.577
BiLSTM (C, W)	0.552 ± 0.02	0.550	0.601

	Original	**Adversarial**
Claim	Google is not a harmful monopoly	Google is a harmful monopoly
Reason	People can choose not to use Google	People can choose not to use Google
Warrant	Other search engines do not redirect to Google	All other search engines redirect to Google
Alternative	All other search engines redirect to Google	Other search engines do not redirect to Google

Results, with and w/o adversarial set

	Test		
	Mean	**Median**	**Max**
BERT	**0.671** $\pm$ 0.09	**0.712**	**0.770**
BERT (W)	0.656 $\pm$ 0.05	0.675	0.712
BERT (R, W)	0.600 $\pm$ 0.10	0.574	0.750
BERT (C, W)	0.532 $\pm$ 0.09	0.503	0.732
BoV	0.564 $\pm$ 0.02	0.569	0.595
BoV (W)	0.567 $\pm$ 0.02	0.572	0.606
BoV (R, W)	0.554 $\pm$ 0.02	0.557	0.579
BoV (C, W)	0.545 $\pm$ 0.02	0.544	0.589
BiLSTM	0.552 $\pm$ 0.02	0.552	0.592
BiLSTM (W)	0.550 $\pm$ 0.02	0.547	0.577
BiLSTM (R, W)	0.547 $\pm$ 0.02	0.551	0.577
BiLSTM (C, W)	0.552 $\pm$ 0.02	0.550	0.601

	Original	**Adversarial**
Claim	Google is not a harmful monopoly	Google is a harmful monopoly
Reason	People can choose not to use Google	People can choose not to use Google
Warrant	Other search engines do not redirect to Google	All other search engines redirect to Google
Alternative	All other search engines redirect to Google	Other search engines do not redirect to Google

eliminates reliance on "not" as a cue; found to be helpful

Results, with and w/o adversarial set

	Test Mean	Median	Max
BERT	**0.671** $\pm$ 0.09	**0.712**	**0.770**
BERT (W)	0.656 $\pm$ 0.05	0.675	0.712
BERT (R, W)	0.600 $\pm$ 0.10	0.574	0.750
BERT (C, W)	0.532 $\pm$ 0.09	0.503	0.732
BoV	0.564 $\pm$ 0.02	0.569	0.595
BoV (W)	0.567 $\pm$ 0.02	0.572	0.606
BoV (R, W)	0.554 $\pm$ 0.02	0.557	0.579
BoV (C, W)	0.545 $\pm$ 0.02	0.544	0.589
BiLSTM	0.552 $\pm$ 0.02	0.552	0.592
BiLSTM (W)	0.550 $\pm$ 0.02	0.547	0.577
BiLSTM (R, W)	0.547 $\pm$ 0.02	0.551	0.577
BiLSTM (C, W)	0.552 $\pm$ 0.02	0.550	0.601

	Test Mean	Median	Max
BERT	**0.504** $\pm$ 0.01	**0.505**	**0.533**
BERT (W)	0.501 $\pm$ 0.00	0.501	0.502
BERT (R, W)	0.500 $\pm$ 0.00	0.500	0.502
BERT (C, W)	0.501 $\pm$ 0.01	0.500	0.518

	Original	Adversarial
Claim	Google is not a harmful monopoly	Google is a harmful monopoly
Reason	People can choose not to use Google	People can choose not to use Google
Warrant	Other search engines do not redirect to Google	All other search engines redirect to Google
Alternative	All other search engines redirect to Google	Other search engines do not redirect to Google

eliminates reliance on “not” as a cue; found to be helpful

Results, with and w/o adversarial set

	Test		
	Mean	Median	Max
BERT	**0.671** $\pm$ 0.09	**0.712**	**0.770**
BERT (W)	0.656 $\pm$ 0.05	0.675	0.712
BERT (R, W)	0.600 $\pm$ 0.10	0.574	0.750
BERT (C, W)	0.532 $\pm$ 0.09	0.503	0.732
BoV	0.564 $\pm$ 0.02	0.569	0.595
BoV (W)	0.567 $\pm$ 0.02	0.572	0.606
BoV (R, W)	0.554 $\pm$ 0.02	0.557	0.579
BoV (C, W)	0.545 $\pm$ 0.02	0.544	0.589
BiLSTM	0.552 $\pm$ 0.02	0.552	0.592
BiLSTM (W)	0.550 $\pm$ 0.02	0.547	0.577
BiLSTM (R, W)	0.547 $\pm$ 0.02	0.551	0.577
BiLSTM (C, W)	0.552 $\pm$ 0.02	0.550	0.601

	Test		
	Mean	Median	Max
BERT	**0.504** $\pm$ 0.01	**0.505**	**0.533**
BERT (W)	0.501 $\pm$ 0.00	0.501	0.502
BERT (R, W)	0.500 $\pm$ 0.00	0.500	0.502
BERT (C, W)	0.501 $\pm$ 0.01	0.500	0.518

even though trained on adversarial examples

	Original	Adversarial
Claim	Google is not a harmful monopoly	Google is a harmful monopoly
Reason	People can choose not to use Google	People can choose not to use Google
Warrant	Other search engines do not redirect to Google	All other search engines redirect to Google
Alternative	All other search engines redirect to Google	Other search engines do not redirect to Google

eliminates reliance on “not” as a cue; found to be helpful

Adversarial Datasets

- Can help identify heuristics and/or statistical cues that models are relying on to make decisions
- Sometimes, but not always, the models just need to see some examples from the adversarial set to learn it
- NB: constructing such a set often relies on deep linguistic knowledge!

Interventions / Causal Analysis

Problem with Probing

- Recall the issue with diagnostic classifiers / probing:
  - We can learn that property X is encoded in representation R
  - But not: does the model use property X in making its decisions
- Main idea here: *causally intervene* on the model and/or data to figure out which properties the model is relying on
  - Somewhat analogous to individual neuron ablation
  - E.g. if we "remove all number information" from R, does the model's performance on a given task suffer

Amnesic Probing

Amnesic Probing: Behavioral Explanation with Amnesic Counterfactuals

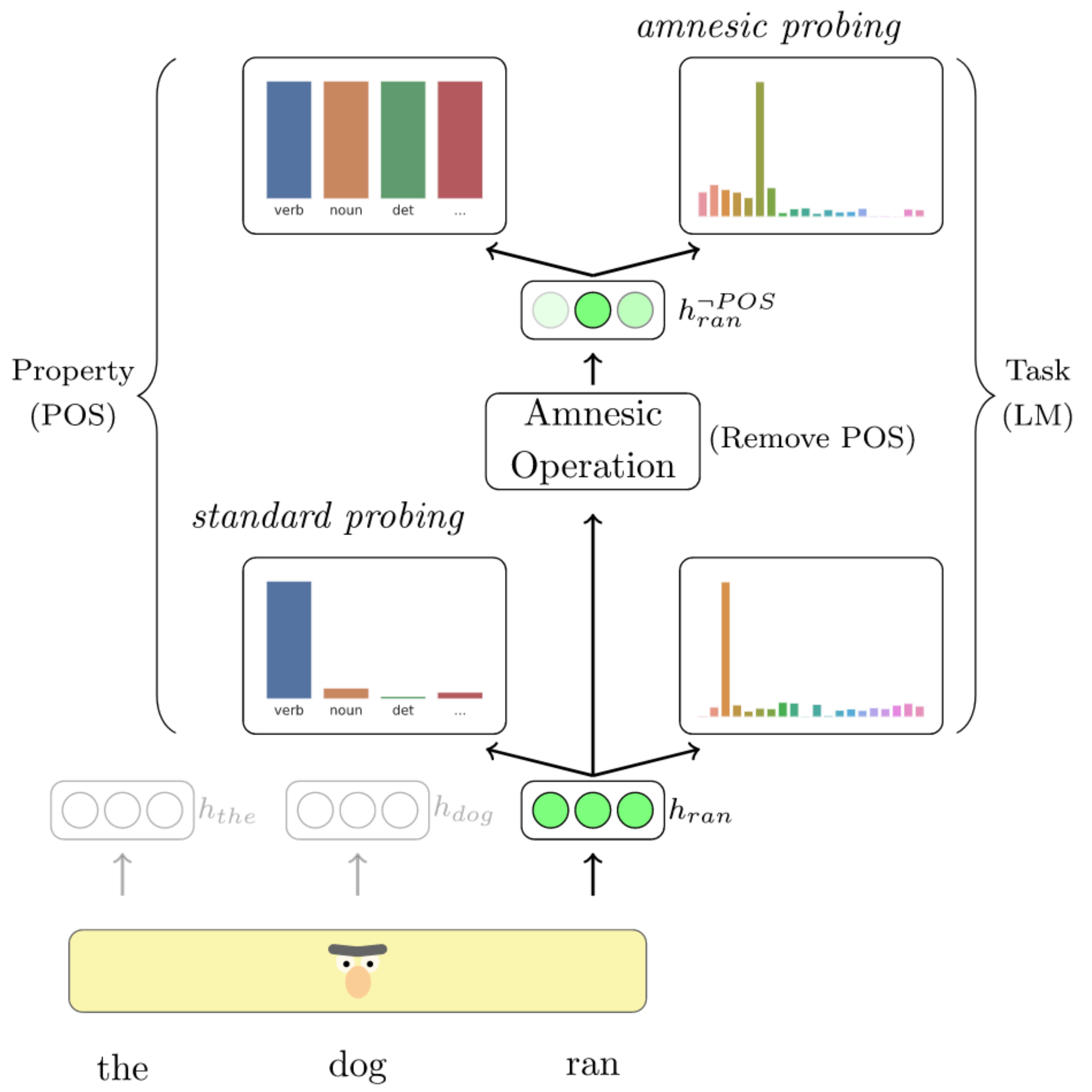
Yanai Elazar[1,2] **Shauli Ravfogel**[1,2] **Alon Jacovi**[1] **Yoav Goldberg**[1,2]
[1]Computer Science Department, Bar Ilan University
[2]Allen Institute for Artificial Intelligence
{yanaiela,shauli.ravfogel,alonjacovi,yoav.goldberg}@gmail.com

Abstract

A growing body of work makes use of *probing* in order to investigate the working of neural models, often considered black boxes. Recently, an ongoing debate emerged surrounding the limitations of the probing paradigm. In this work, we point out the inability to infer behavioral conclusions from probing results, and offer an alternative method that focuses on how the information is being used, rather than

in understanding how these models work and what is being encoded in them. One prominent methodology that attempts to shed light on those questions is *probing* (Conneau et al., 2018) (also known as *auxilliary prediction* [Adi et al., 2016] and *diagnostic classification* [Hupkes et al., 2018]). Under this methodology, one trains a simple model —a *probe*—to predict some desired information from the latent representations of the pre-trained model. High prediction performance is interpreted as evidence for the information being encoded

Amnesic Probing Method



Amnesic Probing Results

		dep	*f-pos*	*c-pos*	*ner*	*phrase start*	*phrase end*
Properties	N. dir	738	585	264	133	36	22
	N. classes	41	45	12	19	2	2
	Majority	11.44	13.22	31.76	86.09	59.25	58.51
Probing	Vanilla	76.00	89.50	92.34	93.53	85.12	83.09
LM-Acc	Vanilla	94.12	94.12	94.12	94.00	94.00	94.00
	Rand	12.31	56.47	89.65	92.56	93.75	93.86
	Selectivity	73.78	92.68	97.26	96.06	96.96	96.93
	Amnesic	7.05	12.31	61.92	83.14	94.21	94.32
LM-D_{KL}	Rand	8.11	4.61	0.36	0.08	0.01	0.01
	Amnesic	8.53	7.63	3.21	1.24	0.01	0.01

- Model relies differentially on different linguistic properties
- Probing performance does not entail model reliance

One last meta-point

Investigating BERT's Knowledge of Language: Five Analysis Methods with NPIs

Alex Warstadt,[†,1,2] **Yu Cao,**[†,3] **Ioana Grosu,**[†,2] **Wei Peng,**[†,3] **Hagen Blix,**[†,1]
Yining Nie,[†,1,2] **Anna Alsop,**[†,2] **Shikha Bordia,**[†,3] **Haokun Liu,**[†,3] **Alicia Parrish,**[†,2,3]
Sheng-Fu Wang,[†,3] **Jason Phang,**[†,1,3] **Anhad Mohananey,**[†,1,3] **Phu Mon Htut,**[†,3]
Paloma Jeretič,[†,1,2] **and Samuel R. Bowman**
New York University

[†]Equal contribution with roles given below; order assigned randomly. Correspondence: bowman@nyu.edu
[1]Framing and organizing the paper [2]Creating diagnostic data [3]Constructing and running experiments

Abstract

Though state-of-the-art sentence representation models can perform tasks requiring significant knowledge of grammar, it is an open question how best to evaluate their grammatical knowledge. We explore five experimental methods inspired by prior work evaluating pretrained sentence representation models. We use a single linguistic phenomenon, negative polarity item (NPI) licensing in English, as a case study for our experiments. NPIs like *any* are grammatical only if they appear in a *licensing environment* like negation (*Sue doesn't have any cats vs. *Sue has any cats*).

acceptability. Linzen et al. (2016), Warstadt et al. (2018), and Kann et al. (2019) use Boolean acceptability judgments inspired by methodologies in generative linguistics. However, we have not yet seen any substantial direct comparison between these methods, and it is not yet clear whether they tend to yield similar conclusions about what a given model knows.

We aim to better understand the trade-offs in task choice by comparing different methods inspired by previous work to evaluate sentence understanding models in a single empirical domain. We choose as our case study negative polarity

Negative polarity items

- NPIs are expressions like *any, ever* that are only grammatical in “negative” environments:
  - * Shaan has done *any* of the reading.
  - Shaan hasn’t done *any* of the reading.
- Question: does BERT “understand” NPIs?
- [NB: see also Marvin and Linzen 2018; Jumelet and Hupkes 2018; Jumelet et al 2021 (w/ yours truly :))]

Does BERT “understand” NPIs?

- It depends!
- “We find that BERT has significant knowledge of these features, but its success varies widely across different experimental methods. We conclude that a variety of methods is necessary to reveal all relevant aspects of a model’s grammatical knowledge in a given domain.”

Wrapping Up

Interpretability and Analysis

- Current NLP models are often a "black box", trained on huge amounts of data, which makes it very unclear what they are learning from their data
  - Engineering: build better models for the future [though caveat emptor]
  - Theoretical: what kinds of linguistic information are learnable (and not) from what kinds of data
  - Ethical: what harmful effects are learned from the data, and how can these be mitigated
- Methods briefly surveyed: neuron-level, psycholinguistic, diagnostic classifiers (+ causal variants), attention analysis, adversarial data
- A huge and growing area!