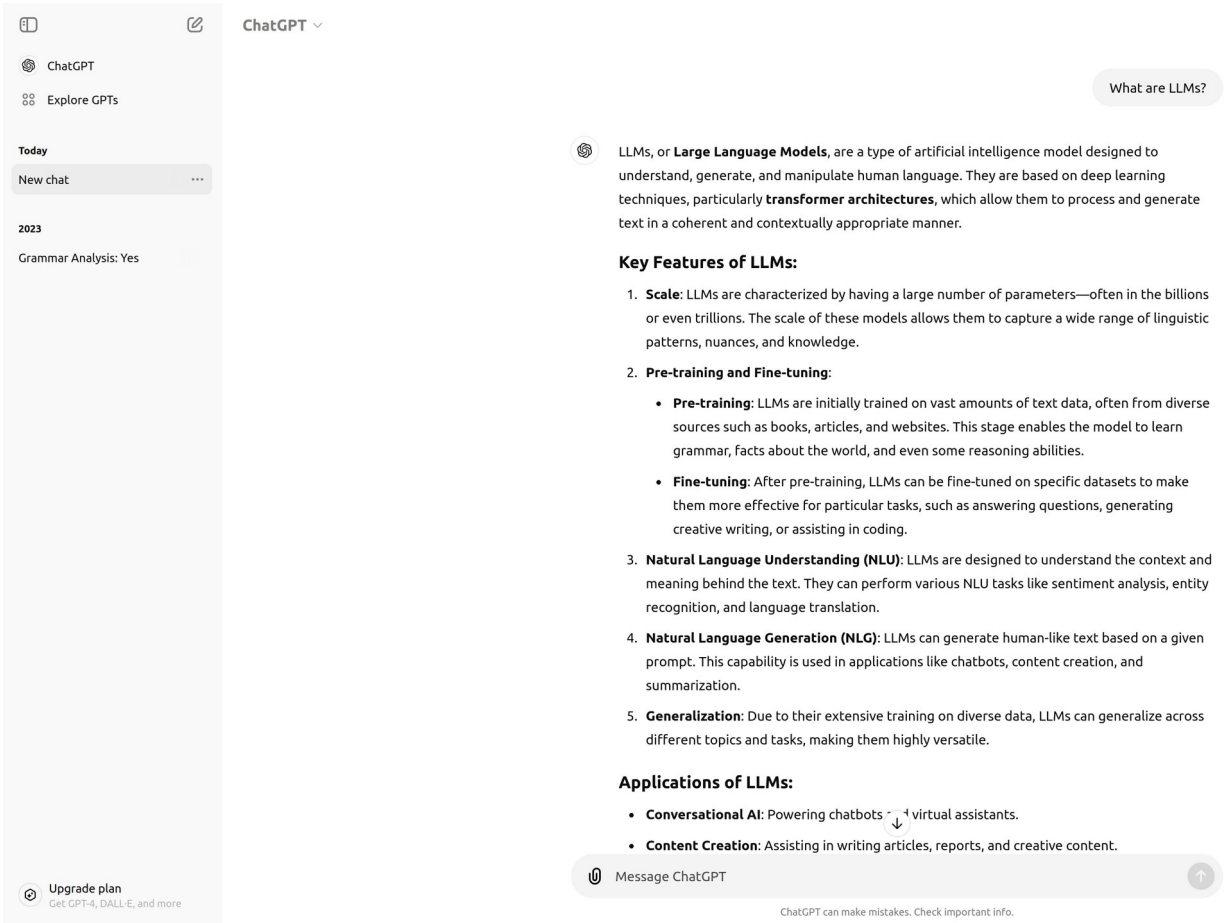
Linguistic Interpretation of Large Language Models: Some Foundational Challenges

Tommi Buder-Gröndahl

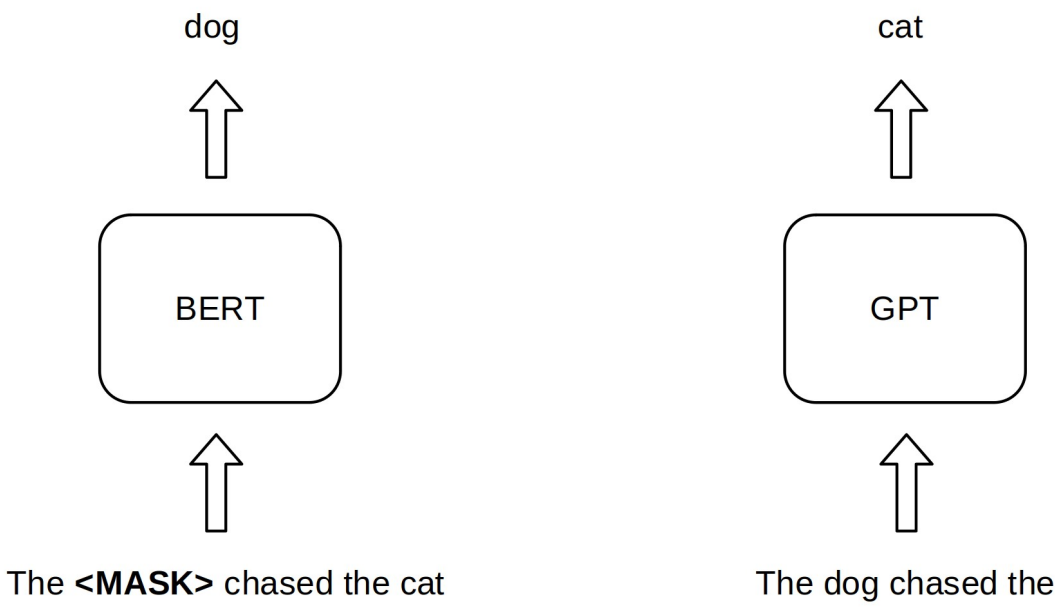
Large Language Models (LLMs)

- BERT, GPT, Claude, LLaMa, Mistral...



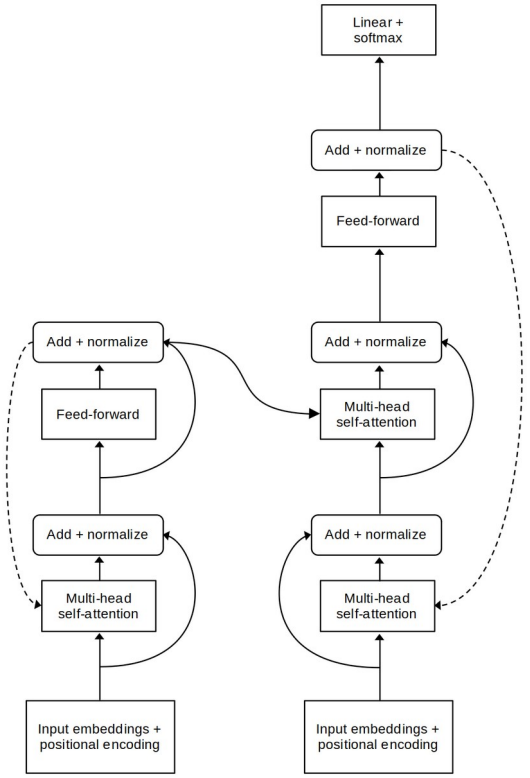
Large Language Models (LLMs)

- BERT, GPT, Claude, LLaMa, Mistral...
- *Pre-trained* on generic linguistic tasks: predicting masked work, predicting upcoming text, …



Large Language Models (LLMs)

- BERT, GPT, Claude, LLaMa, Mistral...
- *Pre-trained* on generic linguistic tasks: predicting masked work, predicting upcoming text, …
- Large *deep neural networks* (DNNs), currently mostly *Transformers* (Vaswani et al. 2017)

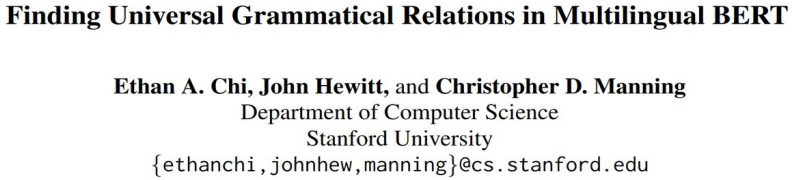


Large Language Models (LLMs)

- BERT, GPT, Claude, LLaMa, Mistral...
- *Pre-trained* on generic linguistic tasks: predicting masked work, predicting upcoming text, …
- Large *deep neural networks* (DNNs), currently mostly *Transformers* (Vaswani et al. 2017)
- Proposed to attain linguistic competence without innate language-specific capacities (vs. rule-based NLP, generative linguistics)

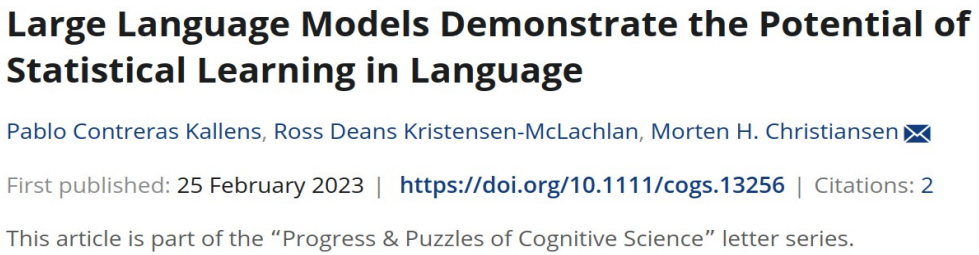
Emergent linguistic structure in artificial neural networks trained by self-supervision

Christopher D. Manning[a,1], Kevin Clark[a], John Hewitt[a], Urvashi Khandelwal[a], and Omer Levy[b]



Finding Universal Grammatical Relations in Multilingual BERT

Ethan A. Chi, John Hewitt, and **Christopher D. Manning**
Department of Computer Science
Stanford University
{ethanchi,johnhew,manning}@cs.stanford.edu



Large Language Models Demonstrate the Potential of Statistical Learning in Language

Pablo Contreras Kallens, Ross Deans Kristensen-McLachlan, Morten H. Christiansen

First published: 25 February 2023 | https://doi.org/10.1111/cogs.13256 | Citations: 2

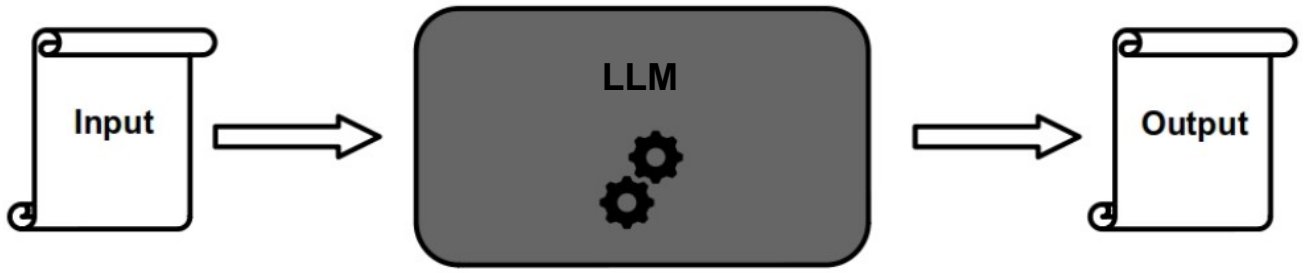
This article is part of the "Progress & Puzzles of Cognitive Science" letter series.

Modern language models refute Chomsky's approach to language

Steven T. Piantadosi[a,b]
[a]UC Berkeley, Psychology [b]Helen Wills Neuroscience Institute

Model interpretation

- LLMs are “black boxes”; how do they process language?

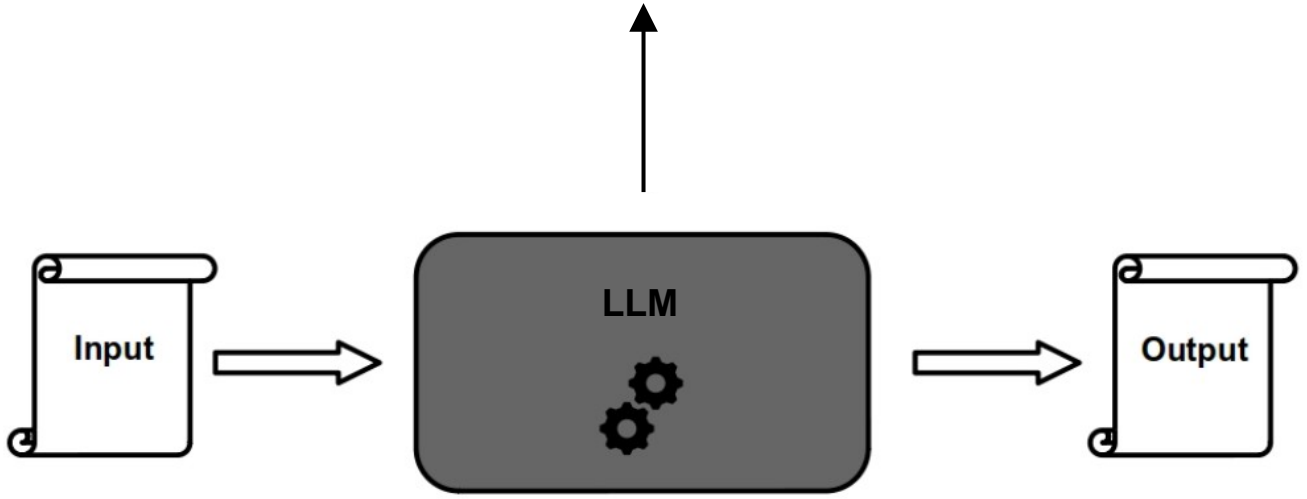


Model interpretation

- LLMs are “black boxes”; how do they process language?

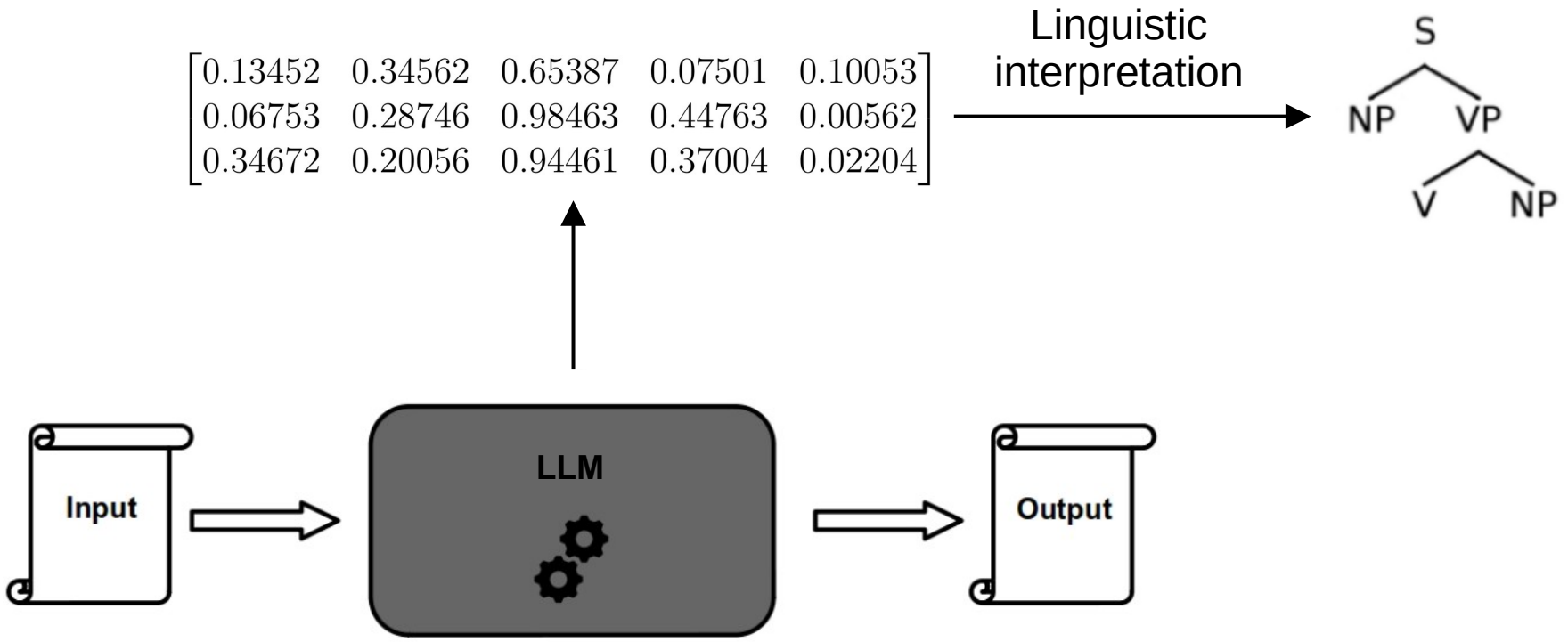
$$\begin{bmatrix} 0.13452 & 0.34562 & 0.65387 & 0.07501 & 0.10053 \\ 0.06753 & 0.28746 & 0.98463 & 0.44763 & 0.00562 \\ 0.34672 & 0.20056 & 0.94461 & 0.37004 & 0.02204 \end{bmatrix}$$

???



Model interpretation

- LLMs are “black boxes”; how do they process language?



Model interpretation

- Prevalent claim: LLMs have *internal linguistic representations*

"Our goal is to design a simple method for testing whether **a neural network embeds each sentence's dependency parse tree** in its contextual word representations – a structural hypothesis." (Hewitt & Manning, 2019, 4129–4130)
"Investigating how **BERT represents syntax**, we describe evidence that attention matrices **contain grammatical representations.**" (Coenen et al., 2019, 8592)
"In this work, we investigate the **linguistic structure implicitly learned by BERT's representations.**" (Jawahar et al., 2019, 3652)
"Another theme that emerges in several studies is the **hierarchical nature of the learned representations.**" (Belinkov & Glass, 2019, 52)
"We propose a methodology and offer the first detailed analysis of BERT's capacity to **capture different kinds of linguistic information by encoding it in its self-attention weights.**" (Kovaleva et al., 2019, 4365)
"We find that **the model represents the steps of the traditional NLP pipeline in an interpretable and localizable way,** and that the regions responsible for each step appear in the expected sequence: POS tagging, parsing, NER, semantic roles, then coreference." (Tenney, Das, & Pavlick, 2019, 4593)

Model interpretation

- Problems:
    1. What does this really mean?
    2. How to determine what kinds of representations (if any) LLMs have?

- Historically, this is *not* how connectionist language models have usually been interpreted.

Eliminative vs. implementational connectionism

- Eliminative connectionism:

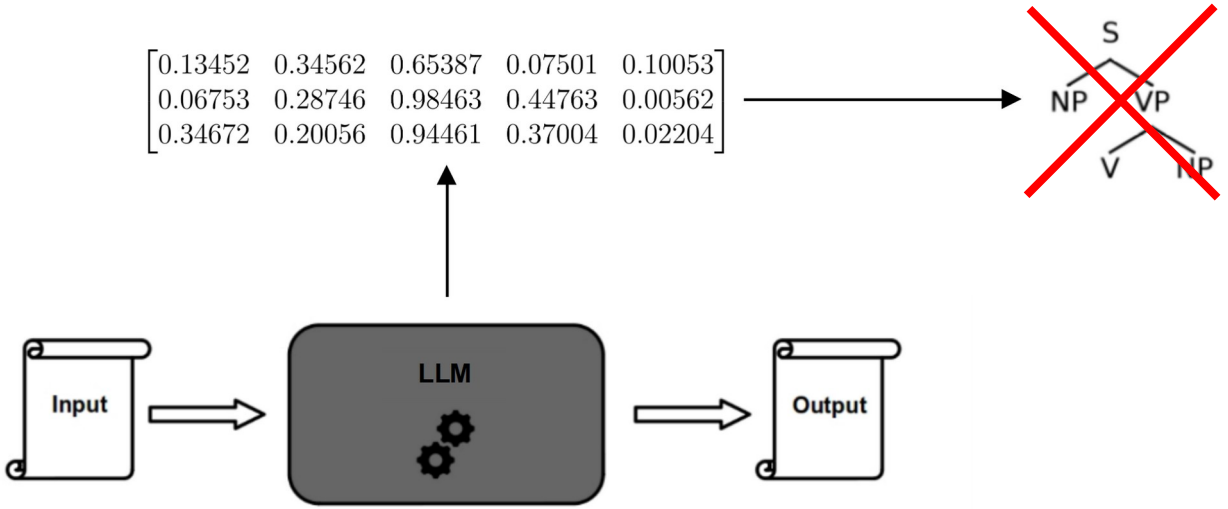
  "(...) a reasonable account of the acquisition of past tense can be provided without recourse (...) to the notion of a 'rule' as anything more than a description of the language. (...) The child need not figure out what the rules are, nor even that there are rules." (Rumelhart and McClelland 1986: 267)

- Implementational connectionism:

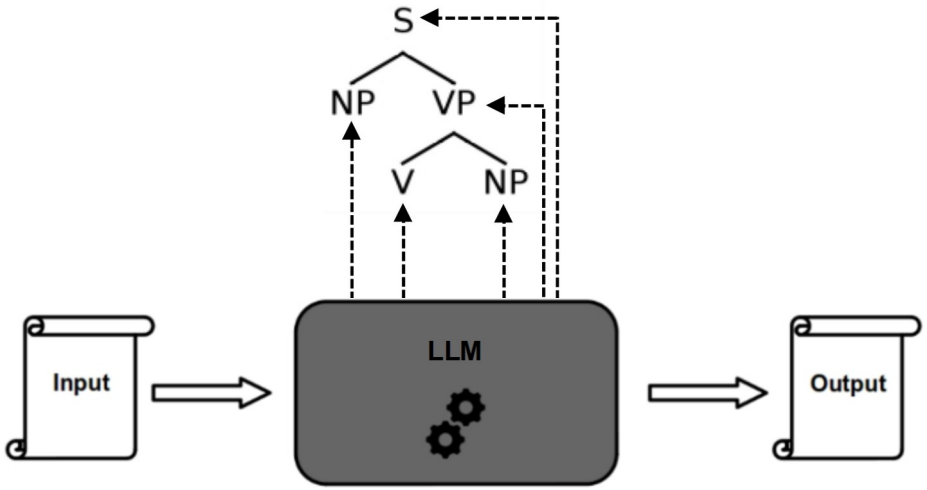
  "(…) the way the overall output of one network feeds into the input of another would be isomorphic to the structure of the symbol manipulations captured in the statements of rules." (Pinker and Price 1988: 76)

Eliminative vs. implementational connectionism

- Eliminative connectionism:



- Implementational connectionism:



Eliminative vs. implementational connectionism

"Whenever I fire a linguist our system performance improves" (attributed to Jelinek 1988)

Language Resources and Evaluation (2005) 39: 25–34
DOI 10.1007/s10579-005-2693-4

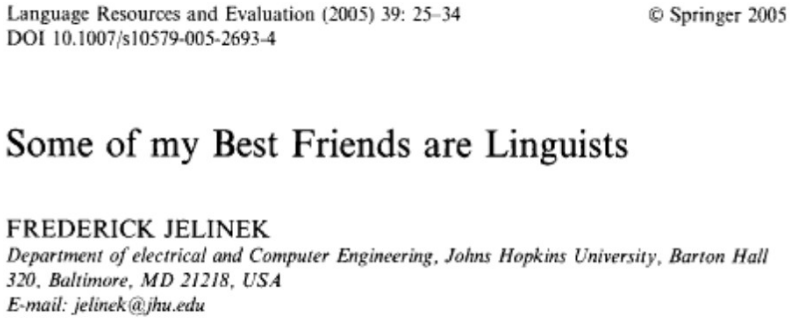
Some of my Best Friends are Linguists

FREDERICK JELINEK
Department of electrical and Computer Engineering, Johns Hopkins University, Barton Hall 320, Baltimore, MD 21218, USA
E-mail: jelinek@jhu.edu

Eliminative vs. implementational connectionism

"Whenever I fire a linguist our system performance improves" (attributed to Jelinek 1988)

"Due to the otherwise opaque, black-box nature of [LLMs], researchers have employed aspects of linguistic theory in order to characterize their behavior. Questions central to syntax — the study of the hierarchical structure of language — have factored heavily into such work." (Kulmizev and Nivre 2022: 02)



Language Resources and Evaluation (2005) 39: 25–34
DOI 10.1007/s10579-005-2693-4

Some of my Best Friends are Linguists

FREDERICK JELINEK
Department of electrical and Computer Engineering, Johns Hopkins University, Barton Hall 320, Baltimore, MD 21218, USA
E-mail: jelinek@jhu.edu

frontiers | Frontiers in Artificial Intelligence

TYPE Conceptual Analysis
PUBLISHED 17 October 2022
DOI 10.3389/frai.2022.796788

Check for updates

OPEN ACCESS

EDITED BY
Sebastian Padó,
University of Stuttgart, Germany

REVIEWED BY
William Schuler,
The Ohio State University,
United States
Felice Dell'Orletta.

Schrödinger's tree—On syntax and neural language models

Artur Kulmizev[1]* and Joakim Nivre[1,2]

[1]Computational Linguistics Group, Department of Linguistics and Philology, Uppsala University, Uppsala, Sweden, [2]RISE Research Institutes of Sweden, Kista, Sweden

Eliminative vs. implementational connectionism

- There has been a quick shift to implementational connectionism in the LLM-literature
- Is this justified, and how does this impact the linguistic ramifications of LLMs?

Papers

Tommi Buder-Gröndahl. The ambiguity of BERTology: What do large language models represent? *Synthese* 203: 15, 2023.

Tommi Buder-Gröndahl. What does parameter-free probing really uncover? *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers):* 327–336, 2024.

Ambiguity of BERTology

- Meta-theoretical problem: what does “representation of X” mean?
    1. *Content reading:* system-internal vehicle that carries information about content X
    2. *Vehicle reading:* system-internal vehicle that instantiates X

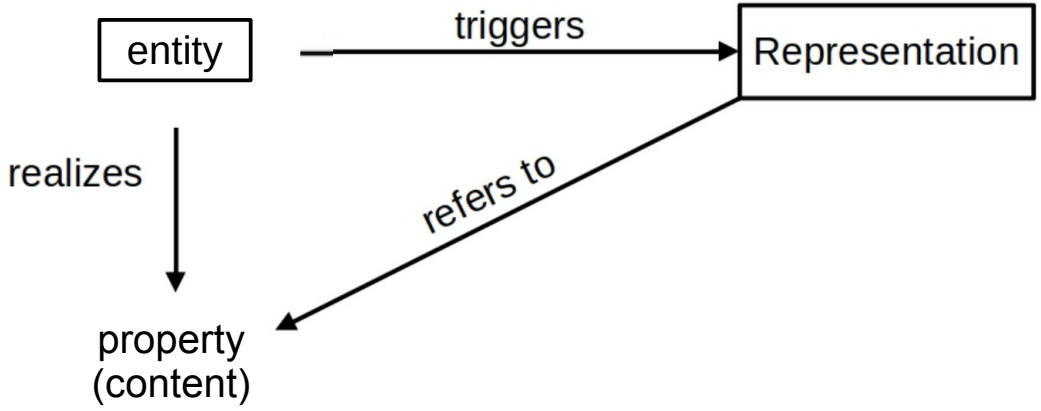
Ambiguity of BERTology

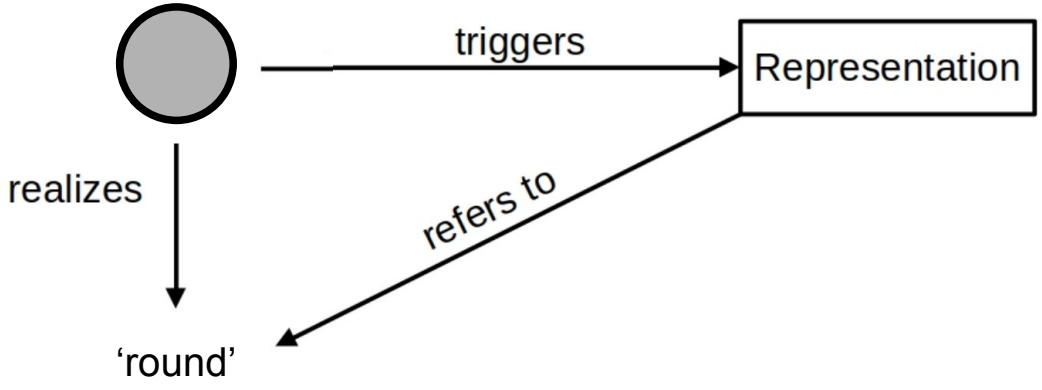
- Meta-theoretical problem: what does “representation of X” mean?
    1. *Content reading:* system-internal vehicle that carries information about content X
    2. *Vehicle reading:* system-internal vehicle that instantiates X

- *Both* have problems in interpreting claims that LLMs contain representations of abstract syntax
    1. Content reading is in danger of being *trivially false*
    2. Vehicle reading is in danger of being *trivially true*

Ambiguity of BERTology

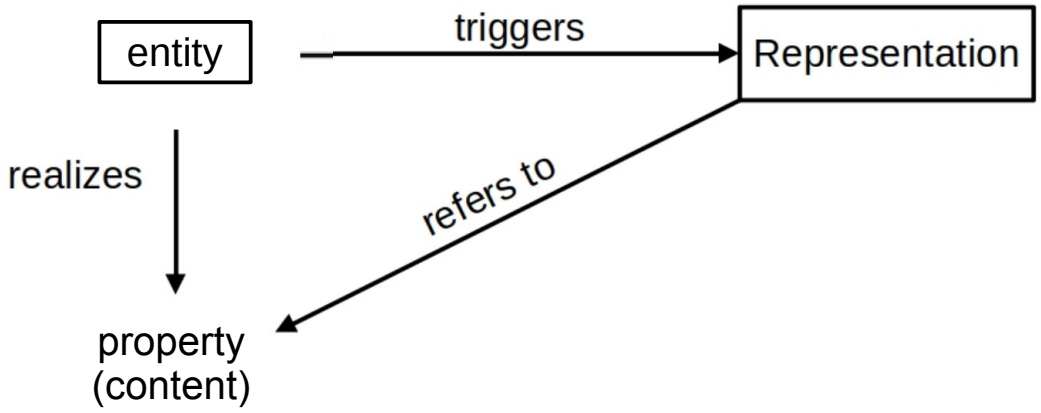
- Content reading is intuitive, but contrasts ubiquitous assumptions of formal linguistic theory
  - Typical idea: content is based on information picked up from the data (Dretske 1981, Fodor 1990, Millikan 2017, Neander 2017, Shea 2018, Cappelen & Dever 2021)

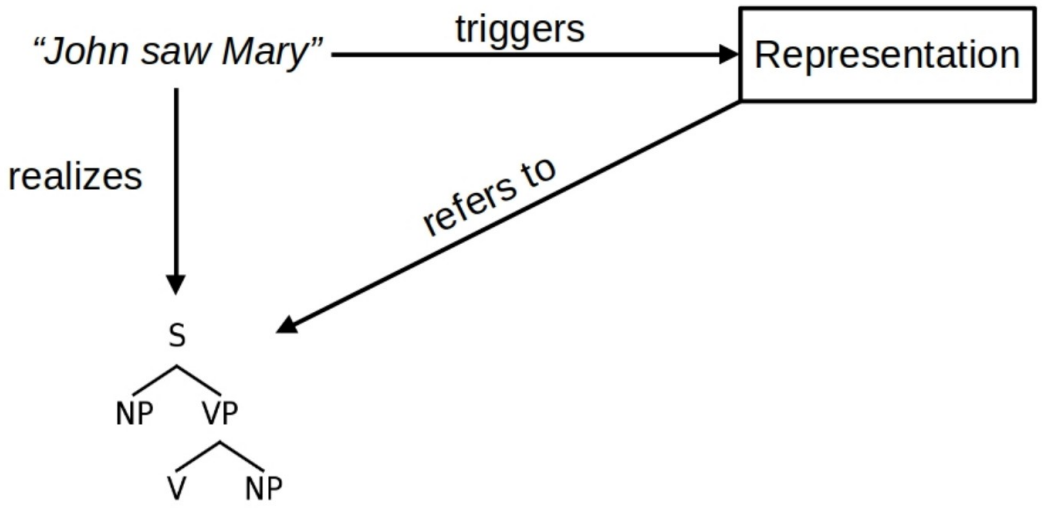




Ambiguity of BERTology

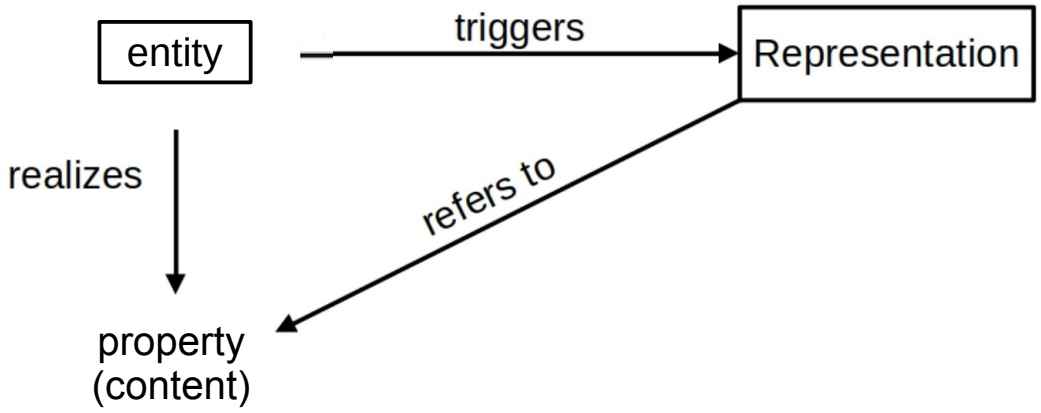
- Content reading is intuitive, but contrasts ubiquitous assumptions of formal linguistic theory
  - Typical idea: content is based on information picked up from the data (Dretske 1981, Fodor 1990, Millikan 2017, Neander 2017, Shea 2018, Cappelen & Dever 2021)

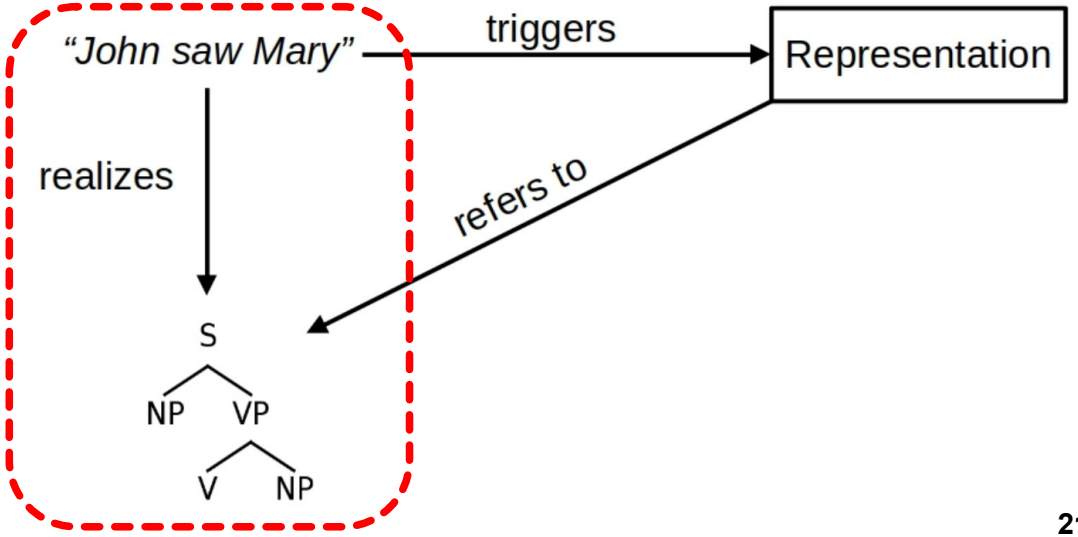




Ambiguity of BERTology

- Content reading is intuitive, but contrasts ubiquitous assumptions of formal linguistic theory
  - Typical idea: content is based on information picked up from the data (Dretske 1981, Fodor 1990, Millikan 2017, Neander 2017, Shea 2018, Cappelen & Dever 2021)
  - Such information must be *in the data* to begin with
  - But abstract syntax does *not* reduce to properties of linear strings





Ambiguity of BERTology

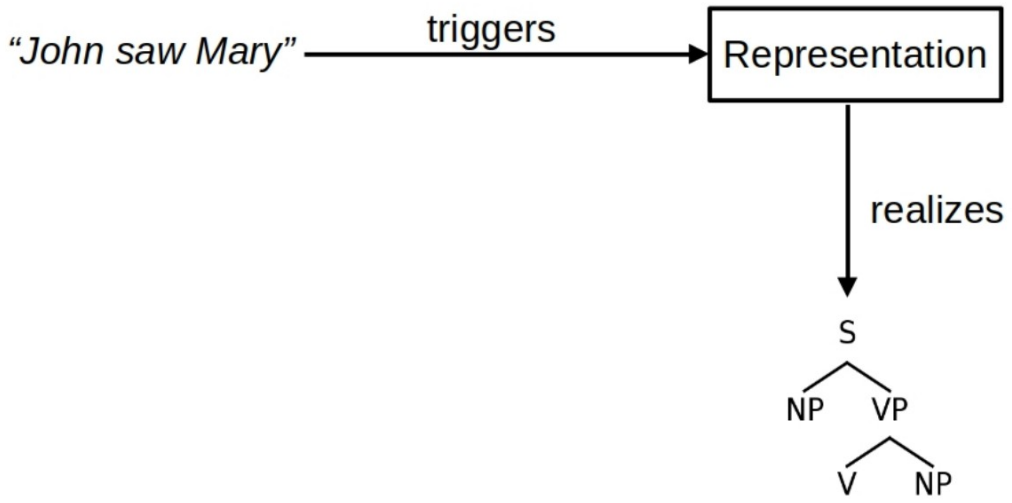
- Content reading is intuitive, but contrasts ubiquitous assumptions of formal linguistic theory
  - Typical idea: content is based on information picked up from the data (Dretske 1981, Fodor 1990, Millikan 2017, Neander 2017, Shea 2018, Cappelen & Dever 2021)
  - Such information must be *in the data* to begin with
  - But abstract syntax does *not* reduce to properties of linear strings

"syntax determines units of combined lexical items that are not identifiable or individuated in terms of linear order or any other perceptible property associated with morphophonemic form." (Collins 2023: 7)

"the perspective in [Chomsky (1975)] is top-down rather than bottom up. (…) the 'representations' are not derived from the utterance." (Adger 2022: 251)

Ambiguity of BERTology

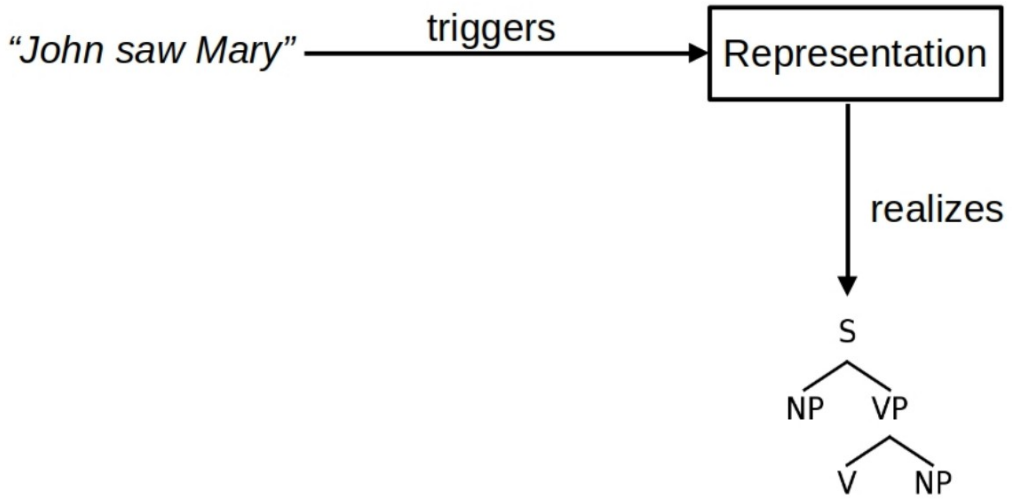
- Vehicle-reading succumbs to a different triviality problem
  - Basic idea: cognitive states themselves realize abstract linguistic structure



Ambiguity of BERTology

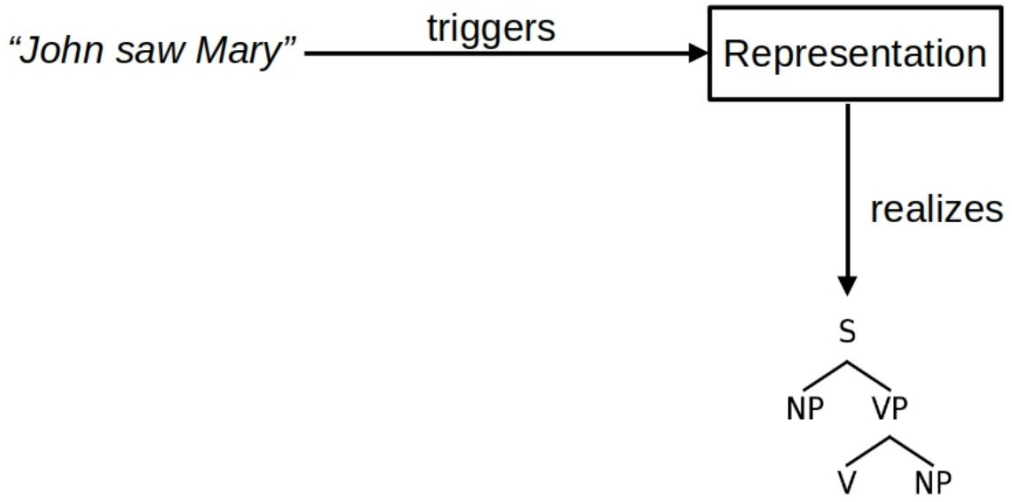
- Vehicle-reading succumbs to a different triviality problem
  - Basic idea: cognitive states themselves realize abstract linguistic structure

"'A mental representation of the grammar of the language' is just the mental structure (brain state) which is, at the relevant level of abstraction from physiological mechanisms, the grammar of the language."
(Adger 2022, 251)



Ambiguity of BERTology

- Vehicle-reading succumbs to a different triviality problem
  - Basic idea: cognitive states themselves realize abstract linguistic structure
  - But abstract structures are not literally “in” concrete systems → more indirect relation needed



Ambiguity of BERTology

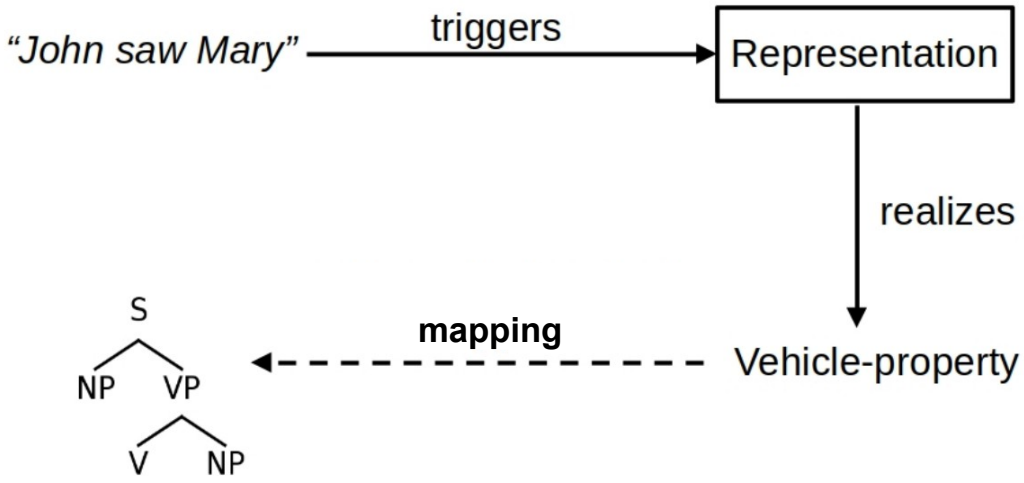
- Vehicle-reading succumbs to a different triviality problem
  - Basic idea: cognitive states themselves realize abstract linguistic structure
  - But abstract structures are not literally "in" concrete systems → more indirect relation needed

$$\text{Merge}(A, B) = \{A, B\}$$

"We don't have sets in our heads. So you have to know that when we develop a theory about our thinking, about our computation, internal processing and so on in terms of sets, that it's going have to be translated into some terms that are neurologically realizable."
(Chomsky 2012, 91)

Ambiguity of BERTology

- Vehicle-reading succumbs to a different triviality problem
  - Basic idea: cognitive states themselves realize abstract linguistic structure
  - But abstract structures are not literally “in” concrete systems → more indirect relation needed
  - *Mapping* concrete vehicles to abstract structures

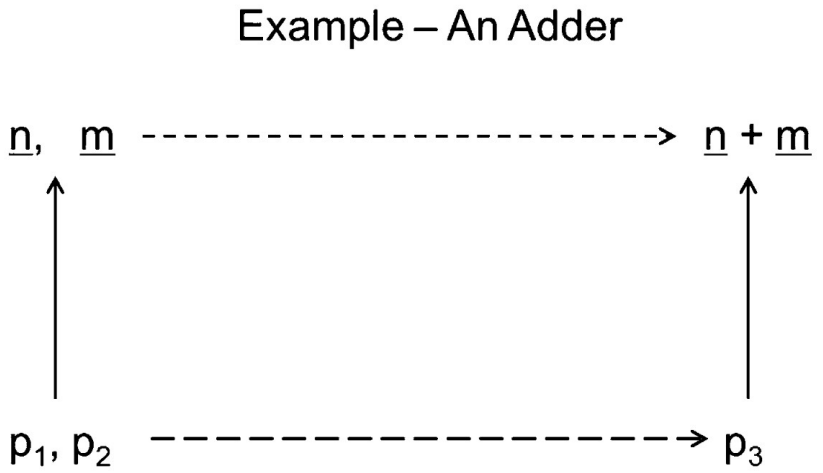


Ambiguity of BERTology

- Vehicle-reading succumbs to a different triviality problem
  - Basic idea: cognitive states themselves realize abstract linguistic structure
  - But abstract structures are not literally “in” concrete systems → more indirect relation needed
  - *Mapping* concrete vehicles to abstract structures
  - Mapping theories of computational implementation have well-known triviality problems
    - Any sufficiently complex system can be mapped to abstract structures (Searle 1992)
    - Anything can be mapped to finite-storage computation (Putnam 1988, Sprevak 2018)

Ambiguity of BERTology

- The mapping account could be salvaged by considering *explanatory virtues* of different mappings
- Abstract formalisms are used for *surrogative reasoning* about concrete systems (Swoyer 1991)



(Egan 2010: 115)

Ambiguity of BERTology

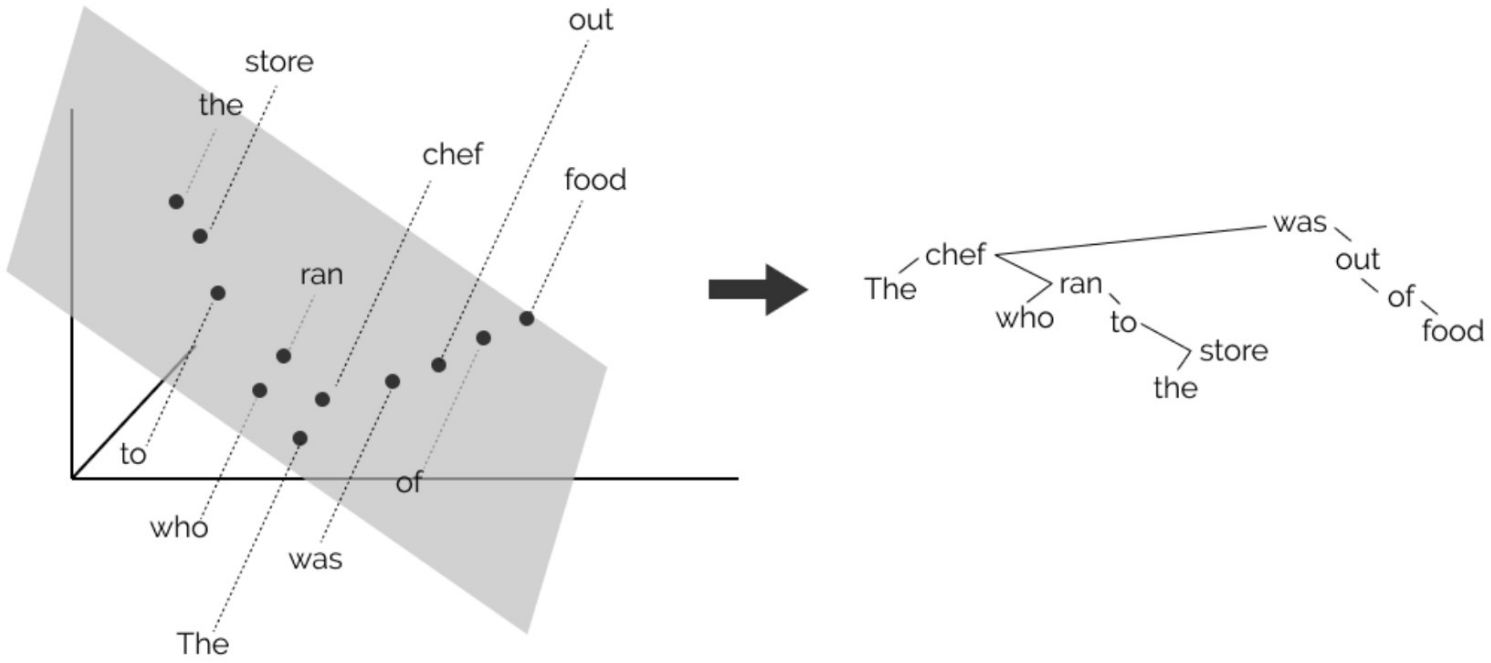
- The mapping account could be salvaged by considering *explanatory virtues* of different mappings
- Abstract formalisms are used for *surrogative reasoning* about concrete systems (Swoyer 1991)
- Some formalisms yield better surrogative reasoning than others
- Task: find linguistic formalism that yields the *best* surrogative reasoning about LLMs

Example – An Adder

(Egan 2010: 115)

Ambiguity of BERTology

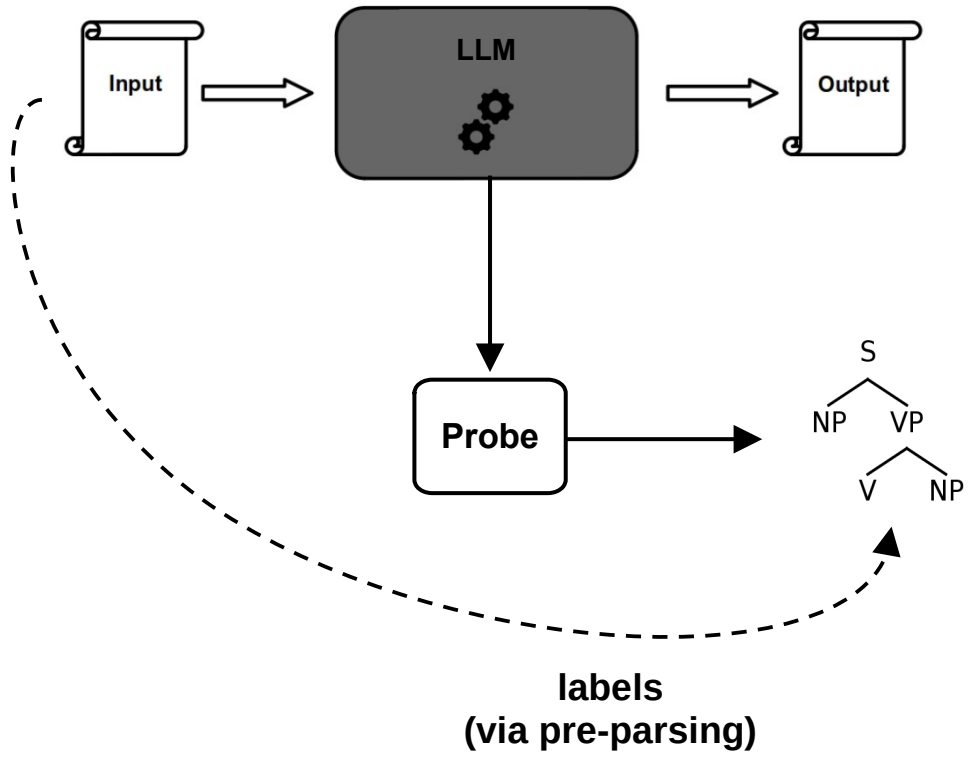
- Probing: mapping embeddings of pre-trained LLMs to linguistic labels



https://nlp.stanford.edu/~johnhew/structural-probe.html

Ambiguity of BERTology

- Probing: mapping embeddings of pre-trained LLMs to linguistic labels
- Typically *supervised*: labels obtained from human-made (or rule-based) annotations



Ambiguity of BERTology

- Probing: mapping embeddings of pre-trained LLMs to linguistic labels
- Typically *supervised*: labels obtained from human-made (or rule-based) annotations

$$d_B(\mathbf{h}_i, \mathbf{h}_j) = (B(\mathbf{h}_i - \mathbf{h}_j))^T (B(\mathbf{h}_i - \mathbf{h}_j))$$

d_B: parse tree distance between tokens
$\mathbf{h}_i$: encoding of *i*:th token
$\mathbf{h}_j$: encoding of *j*:th token
B = probe parameter matrix

(Hewitt & Manning 2019)

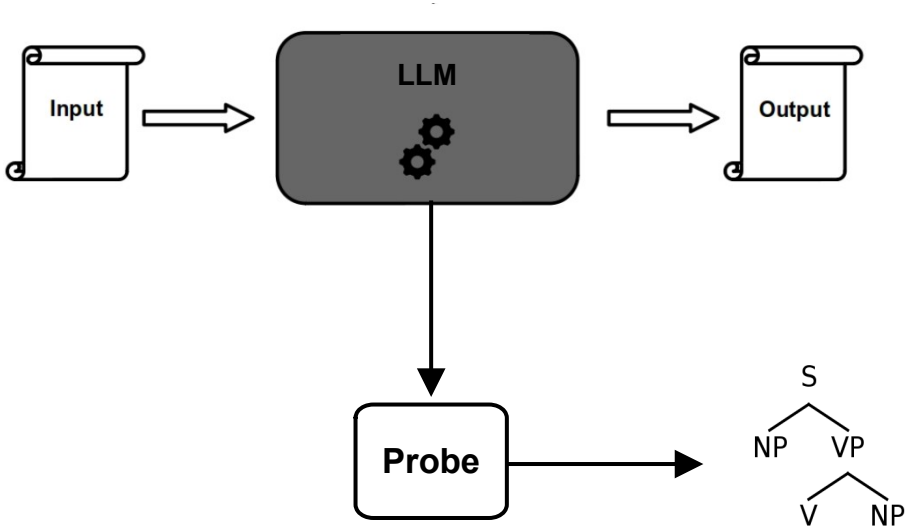
Ambiguity of BERTology

- Supervised probing is insufficient for evaluating grammatical formalisms for LLM-interpretation:

  What are the LLM-internal states that best predict formalism F?

  vs.

  What formalism F* best captures the LLM-internal pipeline?



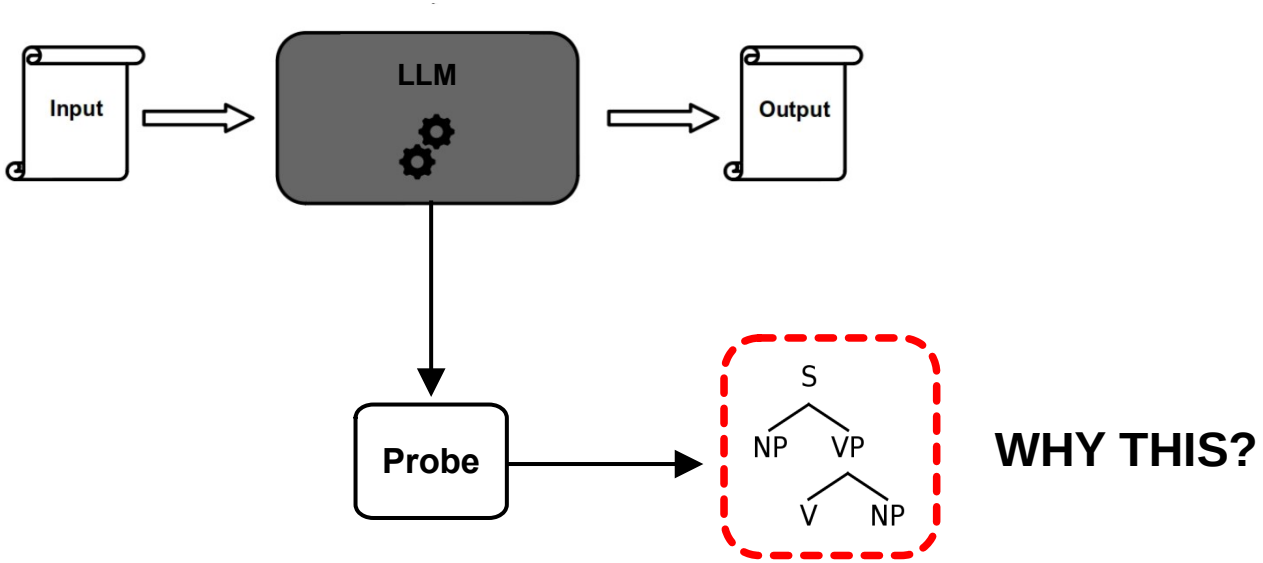
Ambiguity of BERTology

- Supervised probing is insufficient for evaluating grammatical formalisms for LLM-interpretation:

  What are the LLM-internal states that best predict formalism F?

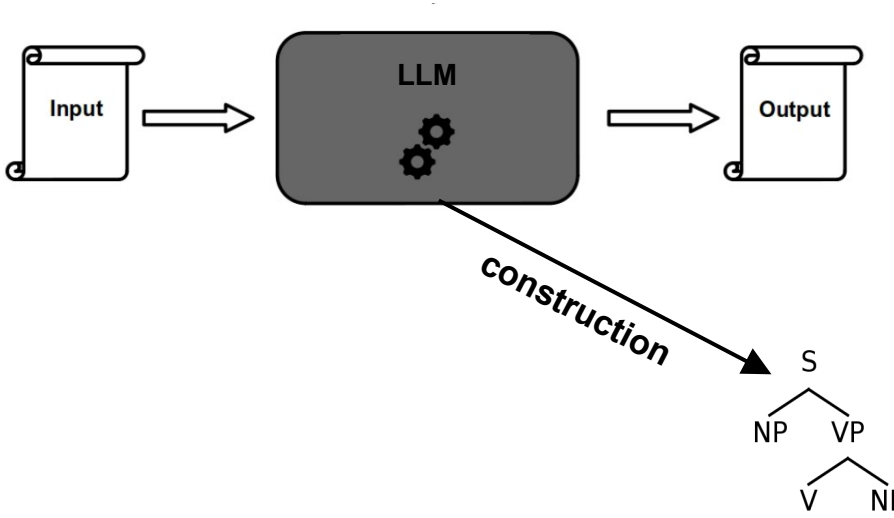
  vs.

  What formalism F* best captures the LLM-internal pipeline?



Parameter-free probing

- More “bottom-up” than supervised probing
- Constructing a syntactic parse directly from the LLM without training a separate probe
- No reliance on pre-determined labels



Parameter-free probing

- Perturbed masking
  - Parameter-free probing technique developed by Wu et al.
  - Generates syntactic dependencies from BERT-embeddings

$$f(x_i, x_j) = d(H_\theta(\mathbf{x} \backslash \{x_i\})_i, H_\theta(\mathbf{x} \backslash \{x_i, x_j\})_i)$$

x = $(x_1, \ldots, x_n)$: input sequence
θ: model parameters
$H_\theta(\mathbf{x} \setminus \{x_i\})_i$: embedding of i:th token of **x** with x_i masked
$H_\theta(\mathbf{x} \setminus \{x_i, x_j\})_i$: embedding of i:th token of **x** with x_i and x_j masked
d: Euclidean distance
$f(x_i, x_j)$: impact of x_j to the embedding of x_i

Parameter-free probing

- Perturbed masking
  - Parameter-free probing technique developed by Wu et al.
  - Generates syntactic dependencies from BERT-embeddings

> “In fact, there is actually no guarantee that our probe will find a strong correlation with human-designed syntax, since we do not introduce the human-designed syntax as supervision. What we found is the ‘natural’ syntax inherent in BERT, which is acquired from self-supervised learning on plain text.”
> (Wu et al 2020: 4173)

Parameter-free probing

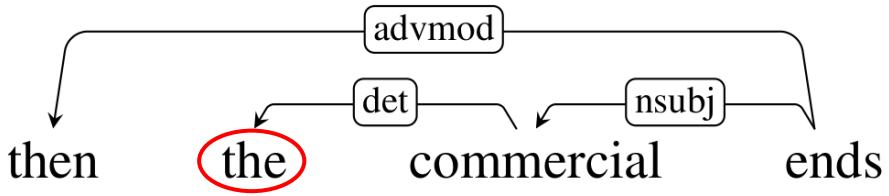
- Perturbed masking
  - Parameter-free probing technique developed by Wu et al.
  - Generates syntactic dependencies from BERT-embeddings

> "In fact, there is actually no guarantee that our probe will find a strong correlation with human-designed syntax, since we do not introduce the human-designed syntax as supervision. What we found is the 'natural' syntax inherent in BERT, which is acquired from self-supervised learning on plain text."
> (Wu et al 2020: 4173)

- Buder-Gröndahl (2024): replicating Wu et al's results, and comparing BERT-derived parses to Universal Dependencies (UD) in the English Parallel Universal Dependencies (PUD) treebank

Parameter-free probing

UD:



BERT:

$Dep(x)$: deprel assigned to x by UD
$Head_{UD}(x)$: head assigned to x by UD
$Head_{BERT}(x)$: head assigned to x by BERT
$H_U(x) = Dep(Head_{UD}(x))$
$H_B(x) = Dep(Head_{BERT}(x))$

$Dep(the) = det,$
$H_U(the) = nsubj,$
$H_B(the) = root$

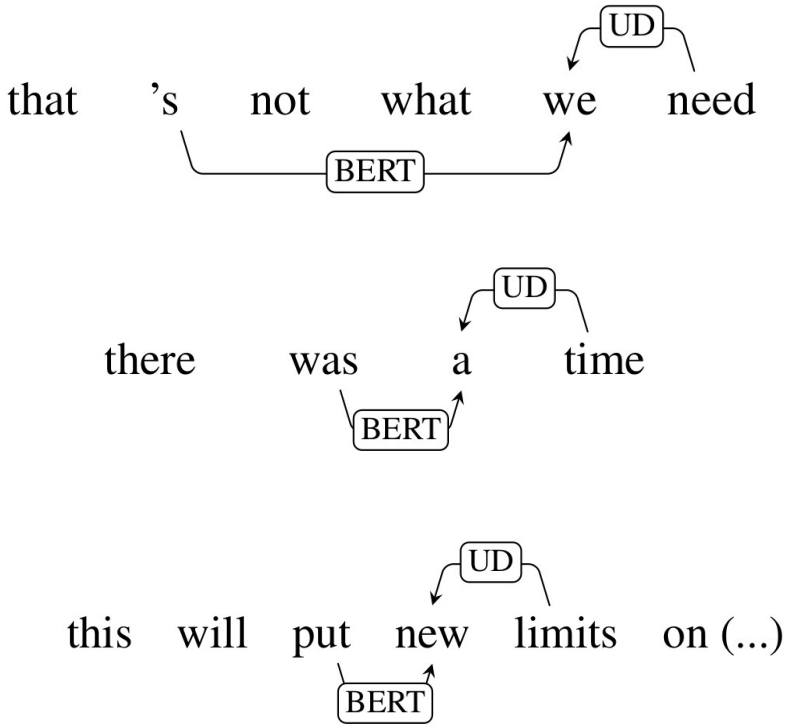
Parameter-free probing

- *Dependent-head shifts* between UD and BERT:
  - verbal argument structure
  - noun phrase structure
  - adjective/adverb modifiers
  - prepositional phrases

- General results
  - Shift ratio: 58%
  - 80% of *Dep*-types had a shift rate over 50%
  - Most common H_B: *root* (35% of all shifts)

Parameter-free probing

- BERT systematically over-assigned the root verb as a head
  - of embedded clause arguments
  - of determiners
  - of adjective/adverb modifiers



Parameter-free probing

- BERT systematically over-assigned the root verb as a head
  - of embedded clause arguments
  - of determiners
  - of adjective/adverb modifiers

Dep	H_U	Ratio	Count
nsubj	root	0.24	198
	acl:relcl	0.81	140
	ccomp	0.92	101
	advcl	0.79	80
	conj	0.83	68
	parataxis	0.64	46
nsubj:pass	root	0.29	38
	acl:relcl	0.94	32
	advcl	0.91	21
obj	advcl	0.66	86
	xcomp	0.75	82
	acl:relcl	0.78	58
	conj	0.66	58
	acl	0.73	52
	root	0.15	47
	ccomp	0.73	29

Table 1: Verbal argument structure: subjects and objects.

Dep	H_U	Ratio	Count
det	obl	0.52	261
	obj	0.67	253
	nsubj	0.54	208
	nmod	0.49	191
	conj	0.57	44
	nsubj:pass	0.54	43
	nmod:poss	0.64	23
	appos	0.68	21
nmod:poss	obj	0.70	56
	nmod	0.72	55
	obl	0.58	54
	nsubj	0.70	53
nummod	obl	0.69	55
	nmod	0.71	25

Table 2: Determiners, possessors, and numerals.

Dep	H_U	Ratio	Count
amod	obj	0.62	151
	obl	0.52	151
	nmod	0.53	132
	nsubj	0.53	118
	conj	0.63	56
	nsubj:pass	0.52	29
	compound	0.57	21
advmod	root	0.18	57
	conj	0.62	53
	advcl	0.72	51
	acl:relcl	0.73	40
	amod	0.73	36
	advmod	0.71	32
	nummod	0.75	27
	ccomp	0.68	27
	obl	0.72	21
	xcomp	0.72	21
nmod	obl	0.88	243
	obj	0.89	202
	nsubj	0.87	163
	nmod	0.84	127
	conj	0.88	59
	nsubj:pass	0.83	34
	appos	0.85	23
	root	0.38	20

Table 3: Adjectival, adverbial, and nominal modifiers.

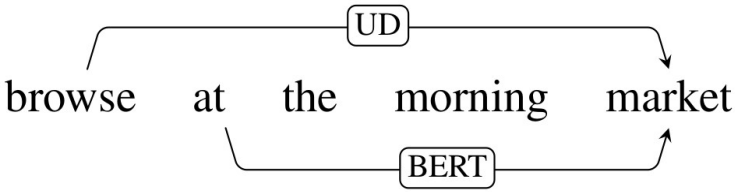
Parameter-free probing

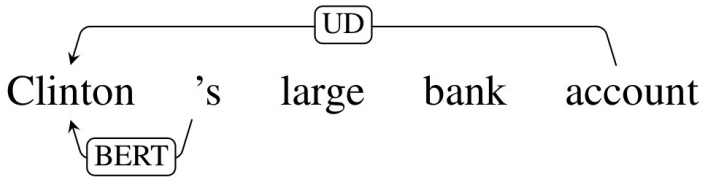
- BERT systematically over-assigned the root verb as a head
  - of embedded clause arguments
  - of determiners
  - of adjective/adverb modifiers

- This behavior is:
  - non-recursive
  - not analyzable by familiar syntactic frameworks

Parameter-free probing

- Some of BERT's behavior had a salient linguistic interpretation
  - headedness of prepositional phrases: noun in UD, preposition in BERT
  - headedness of possessive constructions: noun in UD, possessor in BERT





Parameter-free probing

- Some of BERT's behavior had a salient linguistic interpretation
  - headedness of prepositional phrases: noun in UD, preposition in BERT
  - headedness of possessive constructions: noun in UD, possessor in BERT

- This behavior is:
  - a deviation from BERT's general preference for the root as a head
  - linguistically analyzable: direction of dependency between functional and lexical heads
  - similar to alternative formalisms such as *surface-syntactic UD* (SUD) (Gerdes et al. 2018)

Parameter-free probing

- If perturbed masking really uncovers BERT's "natural syntax", it drastically differs from UD
- BERT vastly over-assigns the root as a head, resulting in a lack of recursion
- Some of BERT's deviations from UD have a linguistic analogy in the headedness direction between functional and lexical heads (prepositional phrases and possessives)

Summary

- BERTology has committed to representational realism (implementational connectionism)
  - This is ambiguous between vehicle- and content-readings of “linguistic representation”
  - Content-reading makes representation-claims of abstract syntax trivially false
  - Vehicle-reading succumbs to the triviality-problem in mapping accounts of implementation

- Vehicle-reading could be salvaged: which formalism optimally captures the LLM-pipeline?
  - Supervised probing is too weak to establish this: linguistic analysis is presupposed
  - Parameter-free probing is more “bottom-up”, but many results are linguistically incoherent

Extra slides: probing algorithms

- Mapping embeddings of pre-trained LLMs to linguistic labels
- Typically *supervised*: labels obtained from human-made (or rule-based) annotations

$$d_B(\mathbf{h}_i, \mathbf{h}_j) = (B(\mathbf{h}_i - \mathbf{h}_j))^T (B(\mathbf{h}_i - \mathbf{h}_j))$$

d_B: parse tree distance between tokens
$\mathbf{h}_i$: encoding of *i*:th token
$\mathbf{h}_j$: encoding of *j*:th token
B = probe parameter matrix

(Hewitt & Manning 2019)

Extra slides: probing algorithms

- Mapping embeddings of pre-trained LLMs to linguistic labels
- Typically *supervised*: labels obtained from human-made (or rule-based) annotations
- *Parameter-free probing*: unsupervised “bottom-up” alternative

$$f(x_i, x_j) = d(H_\theta(\mathbf{x}\backslash\{x_i\})_i, H_\theta(\mathbf{x}\backslash\{x_i, x_j\})_i)$$

f: impact between two tokens → syntactic relation (dependency/phrase)
d = Euclidean distance
$H_\theta(\mathbf{x})_i$: encoding of *i*:th token of input **x** (model parameters θ)
$\mathbf{x}\backslash\{x_i\}$: input x with *i*:th token masked
$\mathbf{x}\backslash\{x_i, x_j\}$: input x with *i*:th token masked

(Wu et al. 2020)

Extra slides: result tables

Dep	H_U	Ratio	Count
nsubj	root	0.24	198
	acl:relcl	0.81	140
	ccomp	0.92	101
	advcl	0.79	80
	conj	0.83	68
	parataxis	0.64	46
nsubj:pass	root	0.29	38
	acl:relcl	0.94	32
	advcl	0.91	21
obj	advcl	0.66	86
	xcomp	0.75	82
	acl:relcl	0.78	58
	conj	0.66	58
	acl	0.73	52
	root	0.15	47
	ccomp	0.73	29

Table 1: Verbal argument structure: subjects and objects.

Dep	H_U	Ratio	Count
det	obl	0.52	261
	obj	0.67	253
	nsubj	0.54	208
	nmod	0.49	191
	conj	0.57	44
	nsubj:pass	0.54	43
	nmod:poss	0.64	23
	appos	0.68	21
nmod:poss	obj	0.70	56
	nmod	0.72	55
	obl	0.58	54
	nsubj	0.70	53
nummod	obl	0.69	55
	nmod	0.71	25

Table 2: Determiners, possessors, and numerals.

Extra slides: result tables

Dep	H_U	Ratio	Count
amod	obj	0.62	151
	obl	0.52	151
	nmod	0.53	132
	nsubj	0.53	118
	conj	0.63	56
	nsubj:pass	0.52	29
	compound	0.57	21
advmod	root	0.18	57
	conj	0.62	53
	advcl	0.72	51
	acl:relcl	0.73	40
	amod	0.73	36
	advmod	0.71	32
	nummod	0.75	27
	ccomp	0.68	27
	obl	0.72	21
	xcomp	0.72	21
nmod	obl	0.88	243
	obj	0.89	202
	nsubj	0.87	163
	nmod	0.84	127
	conj	0.88	59
	nsubj:pass	0.83	34
	appos	0.85	23
	root	0.38	20

Table 3: Adjectival, adverbial, and nominal modifiers.

Dep	H_U	Ratio	Count
case	obl	0.72	877
	nmod	0.73	783
	nmod:poss	0.83	85
obl	root	0.47	283
	acl:relcl	0.97	117
	advcl	0.95	92
	acl	0.93	88
	conj	0.91	90
	xcomp	0.95	89
	ccomp	0.96	50
	parataxis	0.96	25

Table 4: Prepositional phrases.

Extra slides: result tables

Dep	Ratio	Count
case	0.7251	1799
punct	0.5135	1252
det	0.5433	1105
nmod	0.8500	912
obl	0.7082	869
amod	0.5402	719
nsubj	0.4683	650
compound	0.6675	538
conj	0.8176	511
mark	0.7964	442
obj	0.5011	438
cc	0.7615	431
advmod	0.5035	426
nmod:poss	0.6703	244
advcl	0.7158	209
aux	0.4474	183
acl:relcl	0.8483	179
xcomp	0.5815	157
nummod	0.6071	153
nsubj:pass	0.5720	135
acl	0.6895	131
appos	0.8310	118
flat	0.4978	114
cop	0.3270	103
ccomp	0.7259	98
aux:pass	0.2915	79
parataxis	0.5979	58
fixed	0.5243	54
root	0.0363	36
compound:prt	0.4714	33
nmod:tmod	0.6667	26
csubj	0.5926	16
expl	0.2459	15
obl:npmod	0.7000	14
obl:tmod	0.6111	11
nmod:npmod	0.5263	10
det:predet	0.8889	8
cc:preconj	0.5455	6
csubj:pass	1.0000	3
dislocated	1.0000	2
reparandum	1.0000	1
discourse	1.0000	1
iobj	0.1000	1

Table 6: All dependency-head shifts ordered by Dep ("Ratio": ratio of shifts from all tokens with the Dep).

H_U	Ratio	Count
obl	0.6802	2048
root	0.2664	1694
nmod	0.6788	1655
conj	0.7654	1292
obj	0.7283	946
nsubj	0.6651	872
advcl	0.7791	663
acl:relcl	0.8109	579
xcomp	0.8168	495
ccomp	0.8327	458
acl	0.7762	281
appos	0.7301	238
parataxis	0.7409	223
nsubj:pass	0.6494	176
amod	0.7368	140
nmod:poss	0.7707	121
compound	0.6289	100
advmod	0.7810	82
csubj	0.7703	57
nummod	0.8036	45
flat	0.8276	24
cc	0.8750	14
obl:npmod	0.6667	14
obl:tmod	0.5833	14
csubj:pass	0.8667	13
mark	0.6000	9
nmod:tmod	0.2857	8
case	0.1591	7
dislocated	1.0000	6
nmod:npmod	0.8571	6
iobj	0.8333	5
dep	1.0000	2
det	0.6667	2
cc:preconj	1.0000	1

Table 7: All dependency-head shifts ordered by H_U ("Ratio": ratio of shifts from all tokens with the H_U).

H_D	Ratio	Count
root	0.4763	4244
case	0.9684	1135
amod	0.9386	764
compound	0.9107	602
nsubj	0.5525	542
obl	0.3431	503
nmod	0.3771	474
det	0.9978	453
punct	1.0000	404
obj	0.5306	399
advmod	0.9425	377
cc	0.9936	310
conj	0.4107	276
mark	0.9636	159
nummod	0.9341	156
advcl	0.4519	155
cop	1.0000	122
nsubj:pass	0.5622	122
nmod:poss	0.7707	121
aux	1.0000	119
xcomp	0.5174	119
acl	0.5622	104
flat	0.9533	102
aux:pass	1.0000	92
acl:relcl	0.3571	75
parataxis	0.4621	67
ccomp	0.3907	59
appos	0.3931	57
fixed	1.0000	55
compound:prt	1.0000	33
nmod:tmod	0.5455	24
expl	1.0000	14
obl:npmod	0.6316	12
det:predet	1.0000	9
nmod:npmod	0.9000	9
csubj	0.3462	9
cc:preconj	1.0000	4
obl:tmod	0.2308	3
reparandum	0.6667	2
dislocated	1.0000	1
discourse	1.0000	1
vocative	1.0000	1
csubj:pass	0.3333	1

Table 8: All dependency-head shifts ordered by H_B ("Ratio": ratio of shifts from all tokens with the H_B).

Extra slides: result tables

Dep–H_U–H_B shift (count)		
case-obl-root (521)	case-nmod-root (231)	cc-conj-root (191)
det-obj-root (141)	det-nsubj-root (134)	case-nmod-obl (122)
punct-root-obl (117)	nmod-obl-root (107)	det-obl-case (101)
det-nmod-case (100)	case-nmod-obj (99)	obl-root-case (97)
mark-xcomp-root (87)	nmod-nsubj-root (85)	mark-advcl-root (84)
nmod-obj-root (83)	punct-root-nsubj (79)	case-nmod-nsubj (79)
case-nmod-nmod (73)	det-obl-amod (66)	nsubj-ccomp-root (66)
amod-obj-root (64)	det-obl-root (62)	amod-obl-root (61)
case-nmod:poss-root (56)	nmod-nmod-root (54)	punct-root-advmod (53)
case-obl-acl (52)	nsubj-acl:relcl-root (52)	amod-nsubj-root (49)
punct-root-punct (45)	compound-nsubj-root (45)	mark-ccomp-root (44)
compound-obl-root (44)	compound-nmod-root (43)	obl-xcomp-root (43)
obl-acl-root (43)	obl-acl:relcl-root (43)	punct-conj-cc (41)
obl-conj-root (41)	amod-obj-det (40)	obl-root-amod (40)
punct-root-nmod (38)	amod-nmod-root (38)	obl-advcl-root (38)
obl-root-compound (38)	nsubj-advcl-root (37)	obj-advcl-root (36)
nummod-obl-root (36)	punct-root-parataxis (35)	nsubj-root-amod (35)
obj-xcomp-root (35)	punct-conj-conj (35)	nmod-obl-case (34)
case-obl-advcl (33)	case-obl-conj (33)	punct-conj-root (32)
nmod-obj-case (32)	det-nmod-amod (31)	amod-nmod-case (31)
nmod-nmod-case (31)	nsubj-root-compound (31)	nmod:poss-obl-case (31)
punct-appos-root (30)	case-obl-acl:relcl (30)	conj-nmod-root (30)
case-nmod-det (29)	det-nsubj-amod (28)	nmod-obj-amod (28)
cc-conj-obl (27)	punct-conj-nmod (26)	case-nmod-conj (26)
det-nmod-root (26)	det-obj-advcl (26)	nmod-obl-compound (26)
det-nmod-compound (25)	nmod-conj-root (25)	compound-obj-root (25)
nsubj-conj-root (25)	obj-acl-root (25)	det-nsubj:pass-root (24)
obl-root-nmod (24)	conj-nsubj-root (24)	amod-obl-det (23)
nmod:poss-nmod-case (23)	nmod:poss-nsubj-root (23)	punct-conj-obl (22)
det-obj-amod (22)	obl-acl:relcl-case (22)	nsubj-root-case (22)
cc-conj-nmod (22)	advmod-advcl-root (22)	conj-nmod-cc (22)
nmod-nsubj-case (21)	obl-root-nummod (21)	flat-nsubj-root (21)
obj-acl:relcl-root (21)	acl-obj-root (21)	punct-root-det (20)
case-obl-xcomp (20)	nmod-obl-amod (20)	compound-obl-det (20)
compound-nmod-case (20)	obl-ccomp-root (20)	

Table 9: Dep–H_U–H_B shifts and their counts (minimum count: 20).

References

Adger, D. (2022). What are linguistic representations? Mind & Language, 37 (2), 248–260.

Belinkov, Y., & Glass, J. (2019). Analysis methods in neural language processing: A survey. Transactions of the Association for Computational Linguistics, 7 , 49–72.

Chomsky, N. (2012). The science of language. Cambridge: Cambridge University Press.

Coenen, A., Reif, E., Yuan, A., Kim, B., Pearce, A., Viégas, F., Wattenberg, M. (2019). Visualizing and measuring the geometry of BERT. Proceedings of the 33rd Conference on Neural Information Processing Systems (p. 8592-8600).

Collins, J. (2023). Internalist priorities in a philosophy of words. Synthese, 201 (3), 110.

Danilevsky, M., Qian, K., Aharonov, R., Katsis, Y., Kawas, B., Sen, P. (2020). A survey of the state of explainable AI for natural language processing. Proceedings of the 1st conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing (pp. 447–459).

Egan, F. (2010). Computation models: a modest role for content. Studies in History and Philosophy of Science, 41 (3), 253–259.

Hewitt, J., & Manning, C.D. (2019). A structural probe for finding syntax in word representations. Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers) (pp. 4129–4138).

Jawahar, G., Sagot, B., Seddah, D. (2019). What does BERT learn about the structure of language? Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 3651–3657).

Kovaleva, O., Romanov, A., Rogers, A., Rumshisky, A. (2019). Revealing the dark secrets of BERT. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (pp. 4365–4374).

References

Kulmizev, A., & Nivre, J. (2022). Schrödinger's tree—on syntax and neural language models. Frontiers in Artificial Intelligence, 5.

Kulmizev, A., Ravishankar, V., Abdou, M., Nivre, J. (2020). Do neural language models show preferences for syntactic formalisms? Proceedings of the 58th annual meeting of the association for computational linguistics (pp. 4077–4091).

Pinker, S., & Price, A. (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. Cognition, 28 (1–2), 73–193.

Putnam, H. (1988). Representation and reality. Cambridge: MIT Press.

Rumelhart, D.E., & McClelland, J.L. (1986). On learning the past tenses of English verbs. J.L. McClelland, D.E. Rumelhart, & T.P.R. Group (Eds.), Parallel distributed processing: Explorations in the microstructure of cognition: Vol. 2. psychological and biological models (pp. 216–271). Cambridge: MIT Press.

Searle, J. (1992). The Rediscovery of the Mind. Cambridge: MIT Press.

Sprevak, M. (2018). Triviality arguments about computational implementation. M. Sprevak & M. Colombo (Eds.), Routledge handbook of the computational mind (pp. 175–191). London: Routledge.

Swoyer, C. (1991). Structural representation and surrogative reasoning. Synthese, 87 (3), 449–508,

Tenney, I., Das, D., Pavlick, E. (2019). BERT rediscovers the classical NLP pipeline. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 4593–4601).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., . . . Polosukhins, I. (2017). Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing (pp. 6000–6010).

Wu, Z., Chen, Y, Kao, B. and Liu, Q. (2020). Perturbed masking: Parameter-free probing for analyzing and interpreting BERT. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4166–4176.