Computational Linguistics

CSC 485/2501
Fall 2023

4C

4c. Language Modelling and Grammar

Gerald Penn
Department of Computer Science, University of Toronto

Language Modelling (Shannon, 1951; Jelinek, 1976)

$$\hat{w} = \underset{w_n}{\operatorname{argmax}}\, P(w_n \mid w_1 \ldots w_{n-1})$$

Examples:
- SkipGram (`word2vec`)
- BERT
- GPT

Language Modelling (Shannon, 1951; Jelinek, 1976)

$$\hat{w} = \underset{w_n}{\operatorname{argmax}}\, P(w_n \mid w_1 \ldots w_{n-1})$$

Example sentences:

Athens is the capital __

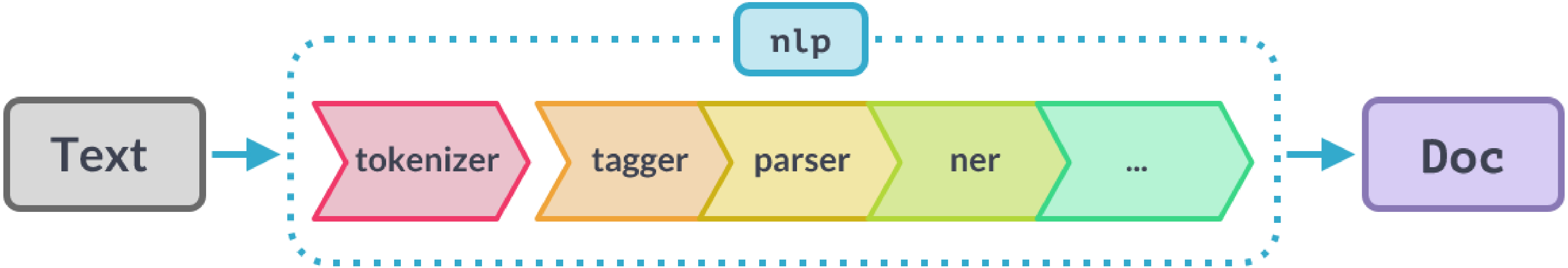
Athens is the capital of __

What do you need to know to predict the first?

What do you need to know to predict the second?

“BERT Rediscovers the Classical NLP Pipeline”

Tenney et al. (2019)



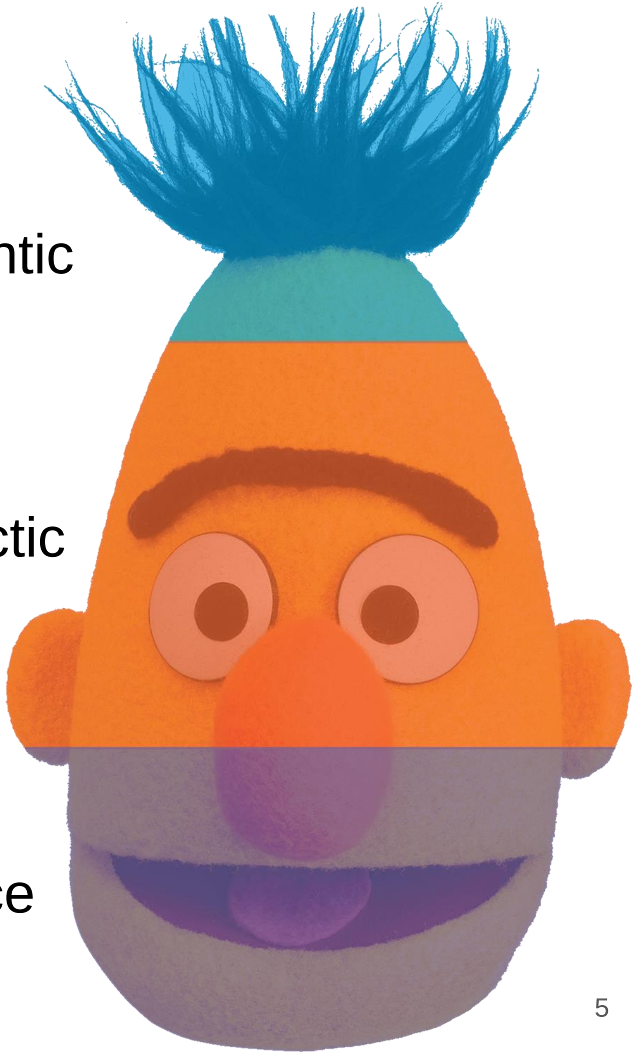
BERT recapitulates the “NLP pipeline?”

“Surface information at the bottom, syntactic information in the middle, semantic information at the top.”

Jawahar et al. (2019)

“It appears that basic syntactic information appears earlier in the network, while high-level semantic information appears at higher layers.”

Tenney et al. (2019)



Kendall's τ

$\tau($

Layer	SentLen (Surface)	WC (Surface)	TreeDepth (Syntactic)	TopConst (Syntactic)	BShift (Syntactic)	Tense (Semantic)	SubjNum (Semantic)	ObjNum (Semantic)	SOMO (Semantic)	CoordInv (Semantic)
1	93.9 (2.0)	24.9 (24.8)	35.9 (6.1)	63.6 (9.0)	50.3 (0.3)	82.2 (18.4)	77.6 (10.2)	76.7 (26.3)	49.9 (-0.1)	53.9 (3.9)
2	95.9 (3.4)	65.0 (64.8)	40.6 (11.3)	71.3 (16.1)	55.8 (5.8)	85.9 (23.5)	82.5 (15.3)	80.6 (17.1)	53.8 (4.4)	58.5 (8.5)
3	**96.2 (3.9)**	66.5 (66.0)	39.7 (10.4)	71.5 (18.5)	64.9 (14.9)	86.6 (23.8)	82.0 (14.6)	80.3 (16.6)	55.8 (5.9)	59.3 (9.3)
4	94.2 (2.3)	**69.8 (69.6)**	39.4 (10.8)	71.3 (18.3)	74.4 (24.5)	87.6 (25.2)	81.9 (15.0)	81.4 (19.1)	59.0 (8.5)	58.1 (8.1)
5	92.0 (0.5)	69.2 (69.0)	40.6 (11.8)	81.3 (30.8)	81.4 (31.4)	89.5 (26.7)	85.8 (19.4)	81.2 (18.6)	60.2 (10.3)	64.1 (14.1)
6	88.4 (-3.0)	63.5 (63.4)	**41.3 (13.0)**	83.3 (36.6)	82.9 (32.9)	89.8 (27.6)	**88.1 (21.9)**	82.0 (20.1)	60.7 (10.2)	71.1 (21.2)
7	83.7 (-7.7)	56.9 (56.7)	40.1 (12.0)	**84.1 (39.5)**	83.0 (32.9)	89.9 (27.5)	87.4 (22.2)	**82.2 (21.1)**	61.6 (11.7)	74.8 (24.9)
8	82.9 (-8.1)	51.1 (51.0)	39.2 (10.3)	84.0 (39.5)	83.9 (33.9)	89.9 (27.6)	87.5 (22.2)	81.2 (19.7)	62.1 (12.2)	76.4 (26.4)
9	80.1 (-11.1)	47.9 (47.8)	38.5 (10.8)	83.1 (39.8)	**87.0 (37.1)**	**90.0 (28.0)**	87.6 (22.9)	81.8 (20.5)	63.4 (13.4)	**78.7 (28.9)**
10	77.0 (-14.0)	43.4 (43.2)	38.1 (9.9)	81.7 (39.8)	86.7 (36.7)	89.7 (27.6)	87.1 (22.6)	80.5 (19.9)	63.3 (12.7)	78.4 (28.1)
11	73.9 (-17.0)	42.8 (42.7)	36.3 (7.9)	80.3 (39.1)	86.8 (36.8)	89.9 (27.8)	85.7 (21.9)	78.9 (18.6)	64.4 (14.5)	77.6 (27.9)
12	69.5 (-21.4)	49.1 (49.0)	34.7 (6.9)	76.5 (37.2)	86.4 (36.4)	89.5 (27.7)	84.0 (20.2)	78.7 (18.4)	**65.2 (15.3)**	74.9 (25.4)

Table 2: Probing task performance for each BERT layer. The value within the parentheses corresponds to the difference in performance of trained vs. untrained BERT.

$) = 0.596$

$\tau($

Layer	SentLen (Surface)	WC (Surface)	TreeDepth (Syntactic)	TopConst (Syntactic)	BShift (Syntactic)	Tense (Semantic)	SubjNum (Semantic)	ObjNum (Semantic)	SOMO (Semantic)	CoordInv (Semantic)
1	[illegible]	[illegible]	35.9 (6.1)	63.6 (9.0)	50.3 (0.3)	82.2 (18.4)	77.6 (10.2)	76.7 (26.3)	49.9 (-0.1)	53.9 (3.9)
2	[illegible]	[illegible]	40.6 (11.3)	71.3 (16.1)	55.8 (5.8)	85.9 (23.5)	82.5 (15.3)	80.6 (17.1)	53.8 (4.4)	58.5 (8.5)
3	[illegible]	[illegible]	39.7 (10.4)	71.5 (18.5)	64.9 (14.9)	86.6 (23.8)	82.0 (14.6)	80.3 (16.6)	55.8 (5.9)	59.3 (9.3)
4	[illegible]	[illegible]	39.4 (10.8)	71.3 (18.3)	74.4 (24.5)	87.6 (25.2)	81.9 (15.0)	81.4 (19.1)	59.0 (8.5)	58.1 (8.1)
5	[illegible]	[illegible]	40.6 (11.8)	81.3 (30.8)	81.4 (31.4)	89.5 (26.7)	85.8 (19.4)	81.2 (18.6)	60.2 (10.3)	64.1 (14.1)
6	[illegible]	[illegible]	**41.3 (13.0)**	83.3 (36.6)	82.9 (32.9)	89.8 (27.6)	**88.1 (21.9)**	82.0 (20.1)	60.7 (10.2)	71.1 (21.2)
7	[illegible]	[illegible]	40.1 (12.0)	**84.1 (39.5)**	83.0 (32.9)	89.9 (27.5)	87.4 (22.2)	**82.2 (21.1)**	61.6 (11.7)	74.8 (24.9)
8	[illegible]	[illegible]	39.2 (10.3)	84.0 (39.5)	83.9 (33.9)	89.9 (27.6)	87.5 (22.2)	81.2 (19.7)	62.1 (12.2)	76.4 (26.4)
9	[illegible]	[illegible]	38.5 (10.8)	83.1 (39.8)	**87.0 (37.1)**	**90.0 (28.0)**	87.6 (22.9)	81.8 (20.5)	63.4 (13.4)	**78.7 (28.9)**
10	[illegible]	[illegible]	38.1 (9.9)	81.7 (39.8)	86.7 (36.7)	89.7 (27.6)	87.1 (22.6)	80.5 (19.9)	63.3 (12.7)	78.4 (28.1)
11	[illegible]	[illegible]	36.3 (7.9)	80.3 (39.1)	86.8 (36.8)	89.9 (27.8)	85.7 (21.9)	78.9 (18.6)	64.4 (14.5)	77.6 (27.9)
12	[illegible]	[illegible]	34.7 (6.9)	76.5 (37.2)	86.4 (36.4)	89.5 (27.7)	84.0 (20.2)	78.7 (18.4)	**65.2 (15.3)**	74.9 (25.4)

Table 2: Probing task performance for each BERT layer. The value within the parentheses corresponds to the difference in performance of trained vs. untrained BERT.

$) = 0.269$

Surface | Syntactic | Semantic

Kendall's τ (non-parametric)

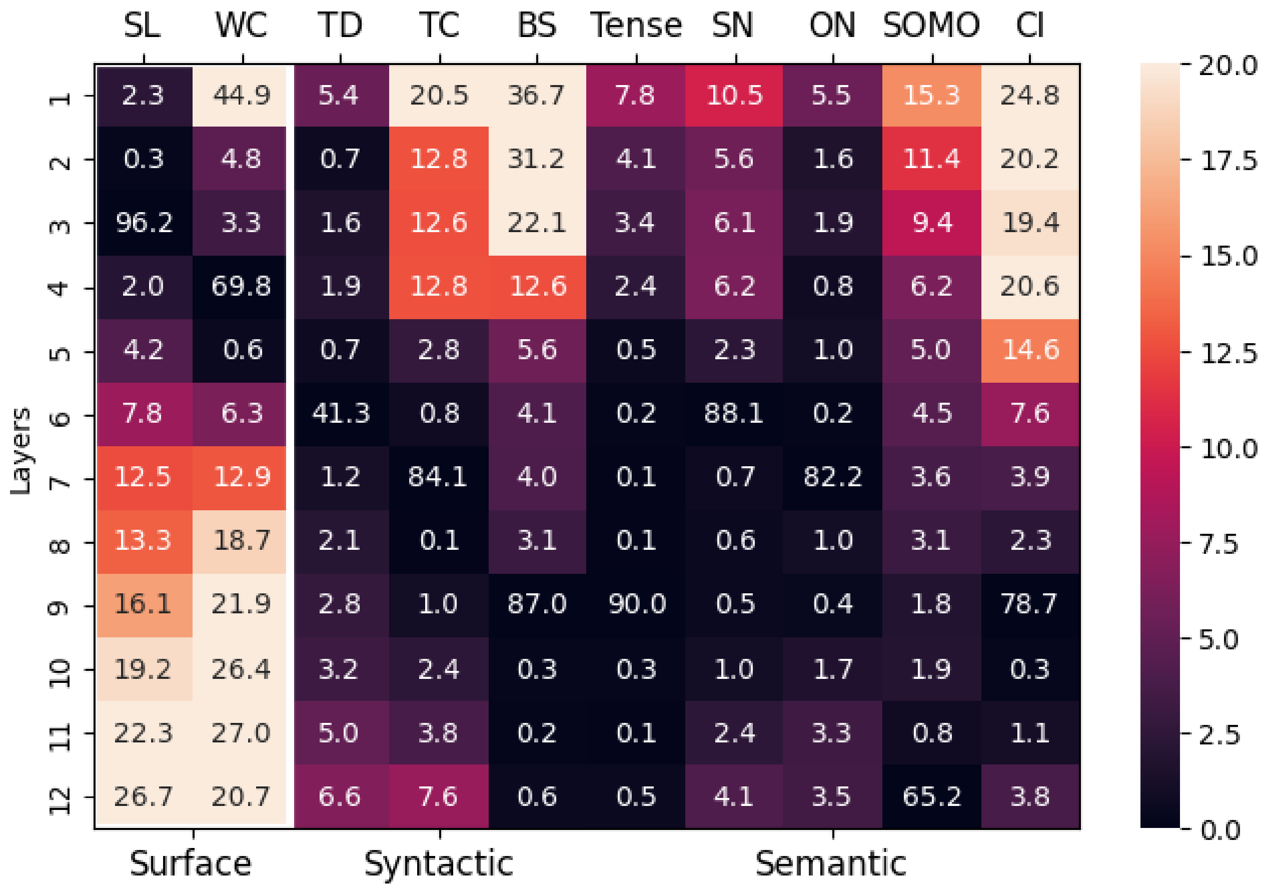
Determines the strength of association between two random variables based upon the number of pairs of paired samples that are "concordant":

Layer

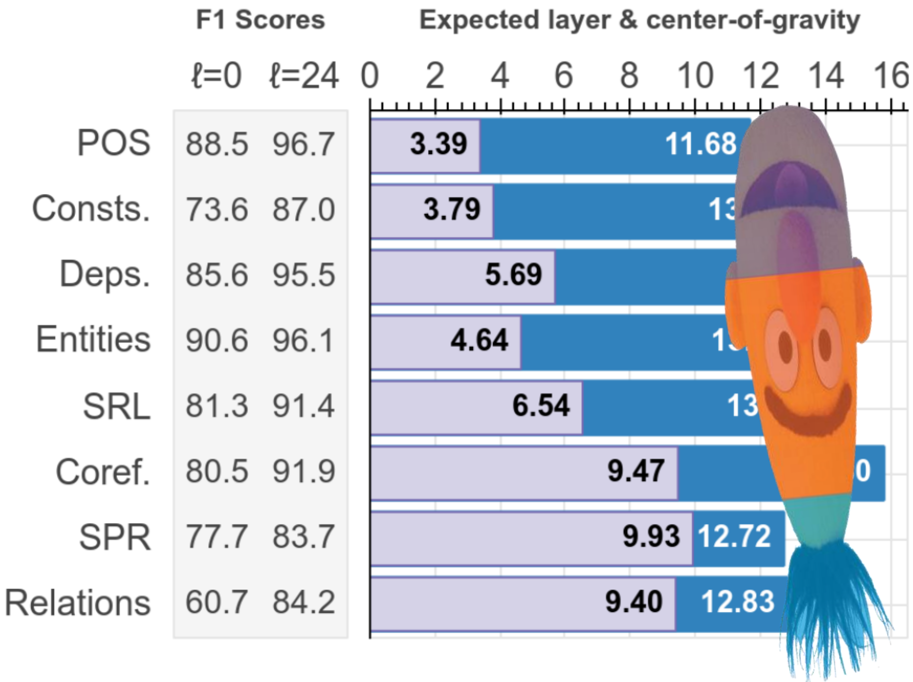
A	1	1
B	2	2
C	3	4
D	4	3
E	5	6
F	6	5
G	7	8
H	8	7
I	9	10
J	10	9
K	11	12
L	12	11

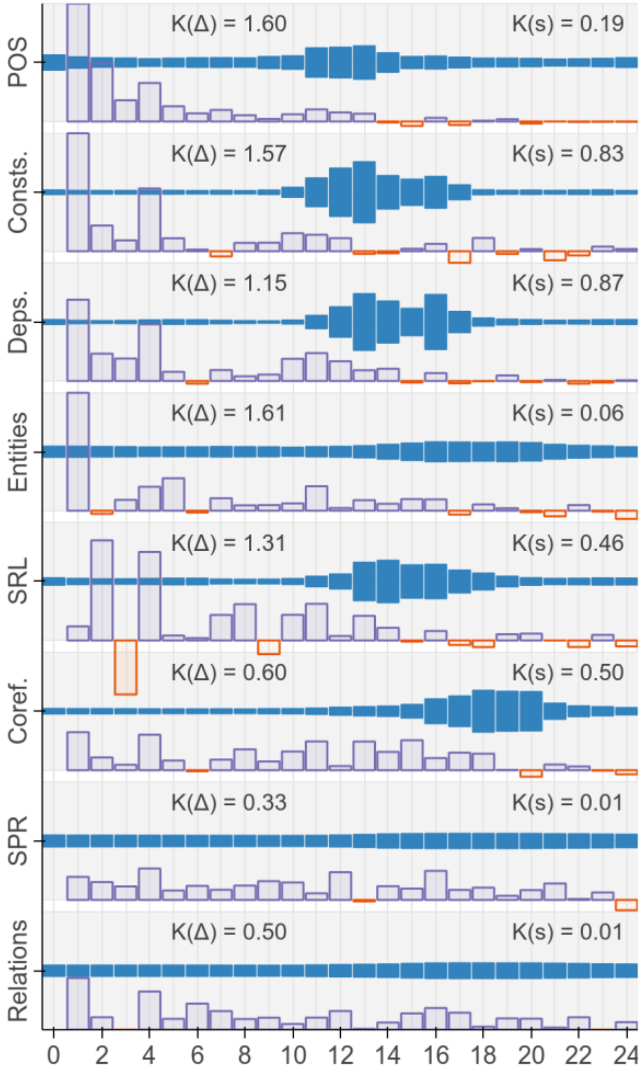
Ordinal ranks

Jawahar et al. (2019) Probing Result



Tenney et al. (2019) Center of Gravity





Pearson r = 0.319, p = 0.44

Weak correlation between layer and COG

Limitation of Tenney et al.'s (2019) Architecture

- Tenney et al. used the **same set of scalar attention weights** for every input sentence: cannot capture **variance of attention patterns across sentences**.
- The probe examines one (or two) span representations: cannot observe task knowledge across **token positions**.

SOLUTION

Self-attention Pooling

(Lee et al., 2017):

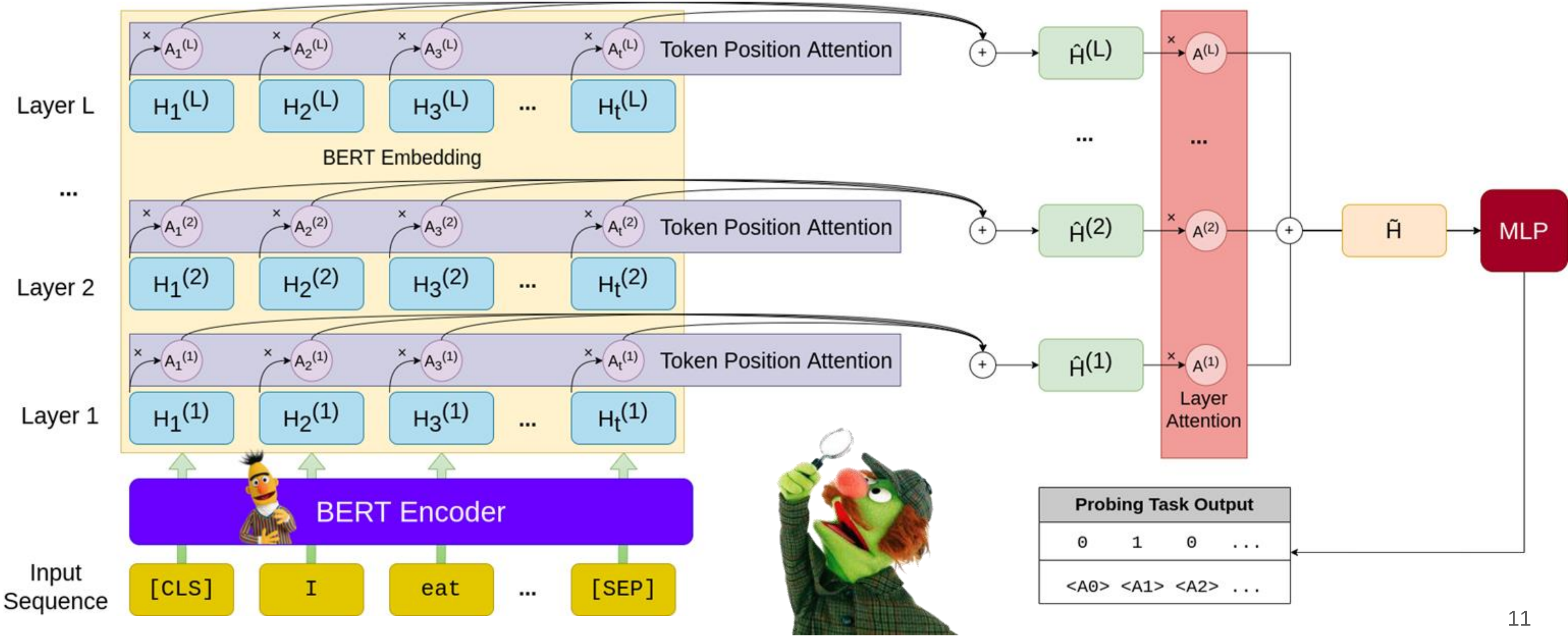
$$\alpha_t = \boldsymbol{w}_\alpha \cdot \boxed{\mathrm{FFNN}_\alpha(\boldsymbol{x}_t^*)}$$

$$a_{i,t} = \frac{\exp(\alpha_t)}{\sum\limits_{k=\mathrm{START}(i)}^{\mathrm{END}(i)} \exp(\alpha_k)}$$

$$\hat{\boldsymbol{x}}_i = \sum_{t=\mathrm{START}(i)}^{\mathrm{END}(i)} a_{i,t} \cdot \boldsymbol{x}_t$$

GridLoc Probe

- Token Position
- Layer
- Randomness & Training

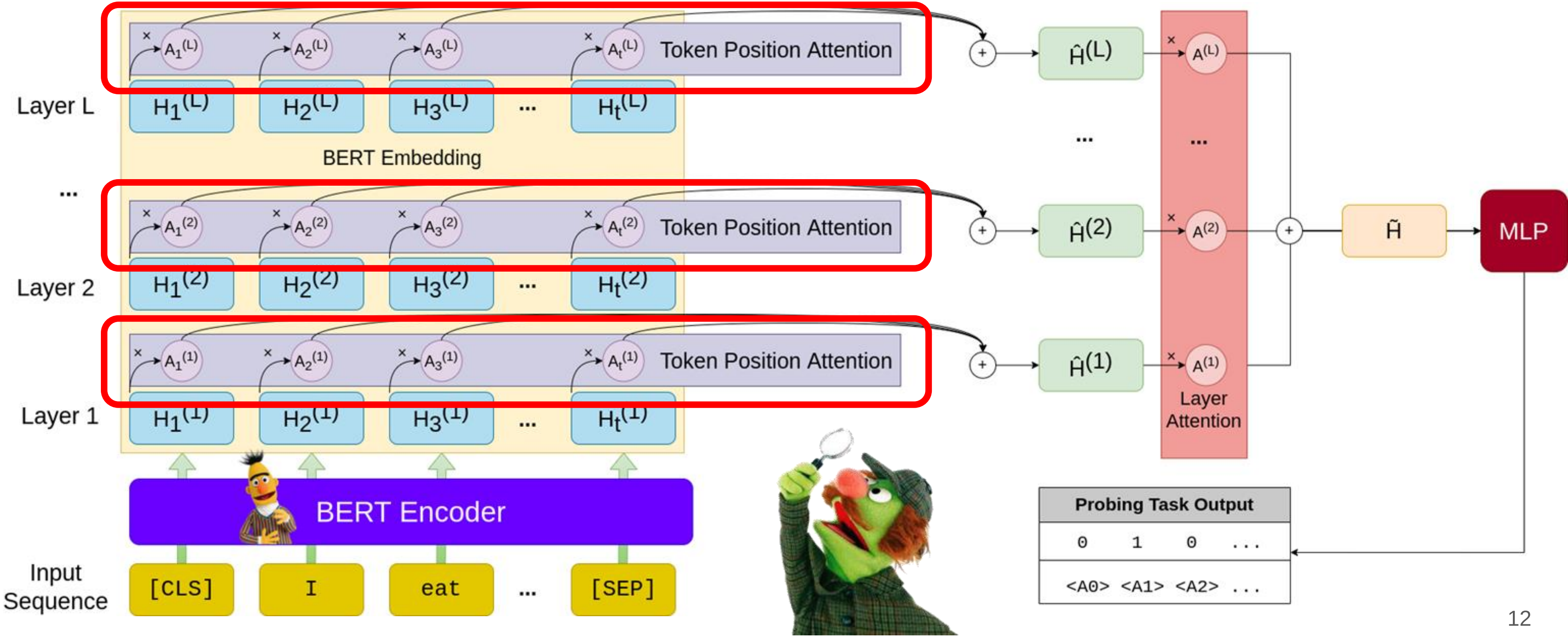


GridLoc Probe

Token position attention:

$$\mathbf{A}^{\text{token},(\ell)} = \text{softmax}(\mathbf{w}_{\text{token}} \cdot \text{RNN}(\mathbf{H}^{(\ell)}))$$

- Token Position
- Layer
- Randomness & Training

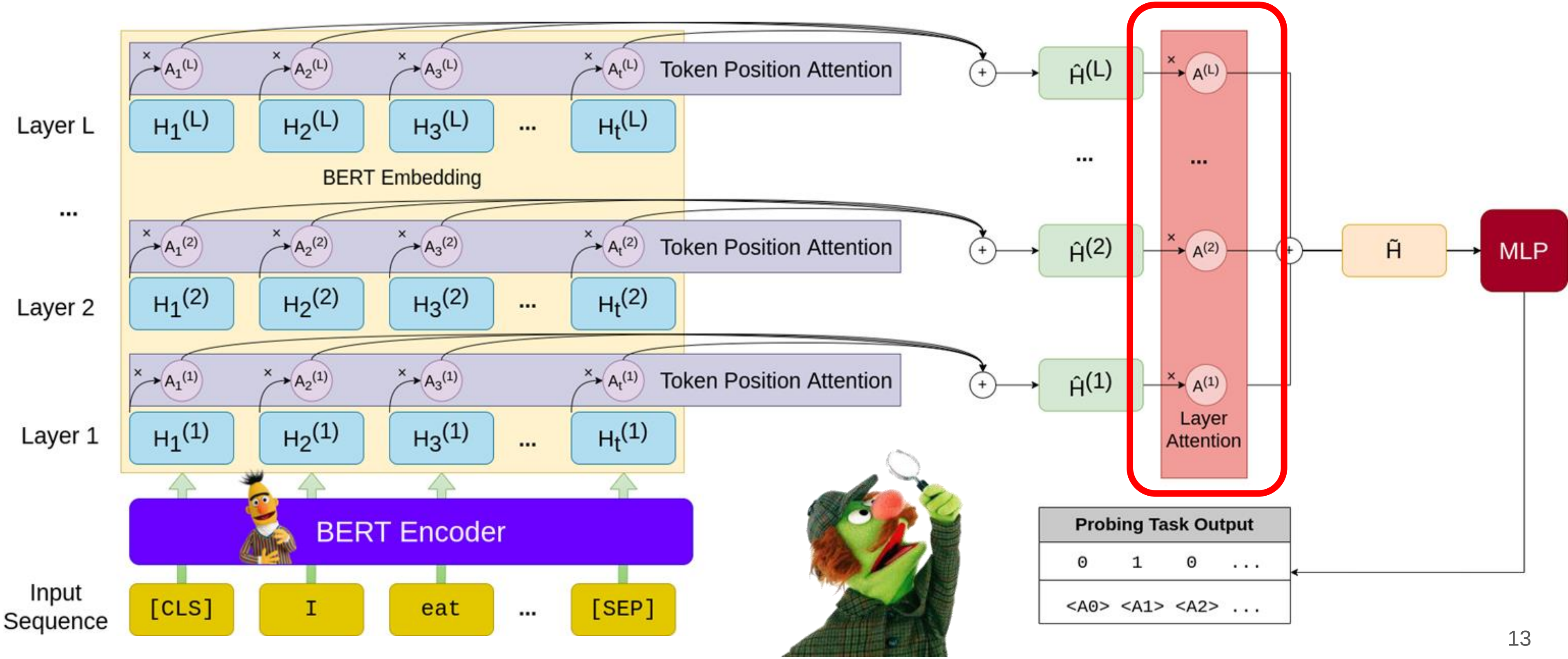


GridLoc Probe

Layer attention:

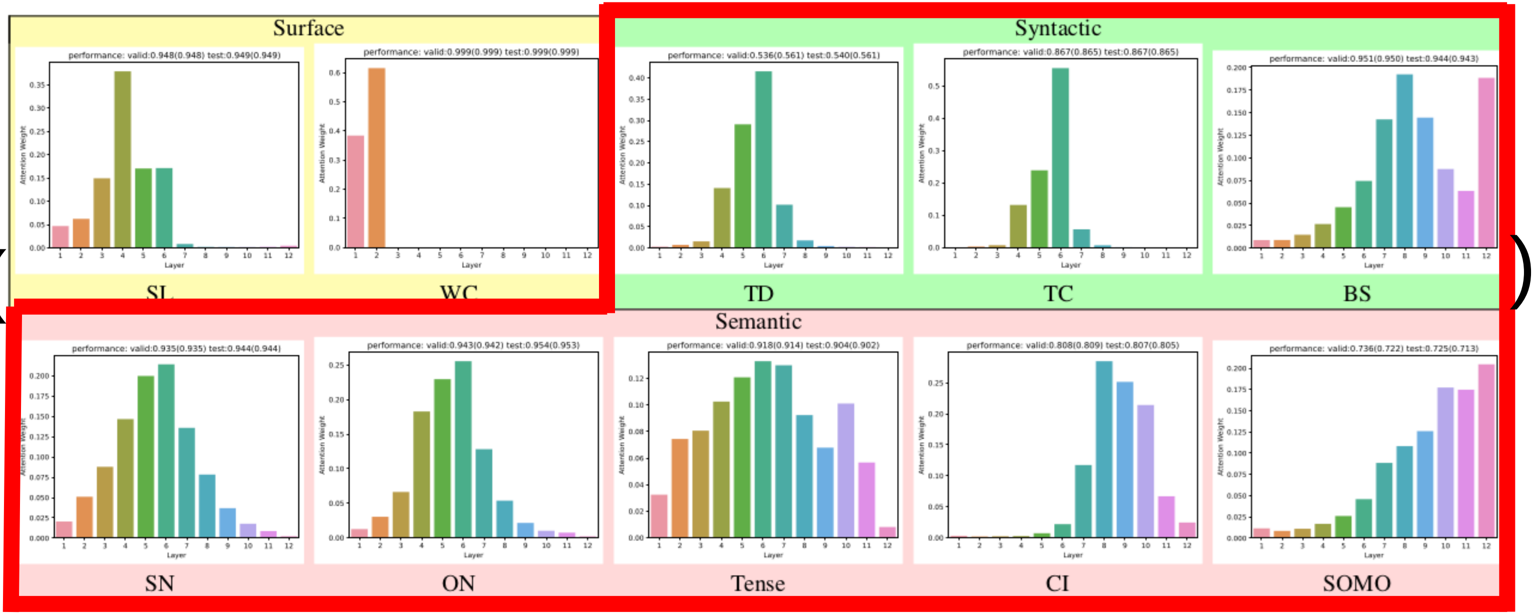
$$\mathbf{A}^{\text{layer}} = \text{softmax}(\mathbf{w}_{\text{layer}} \cdot \hat{\mathbf{H}}^{(\ell)})$$

- Token Position
- **Layer**
- Randomness & Training



Layers Alone do Not Rediscover the CNLP

$\tau\,($

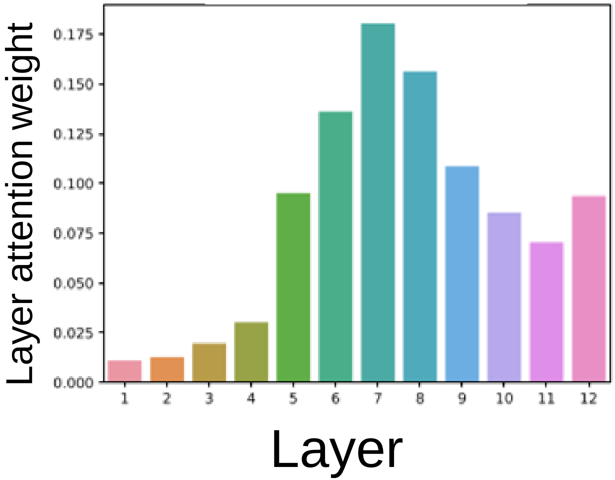


$) = 0.134$

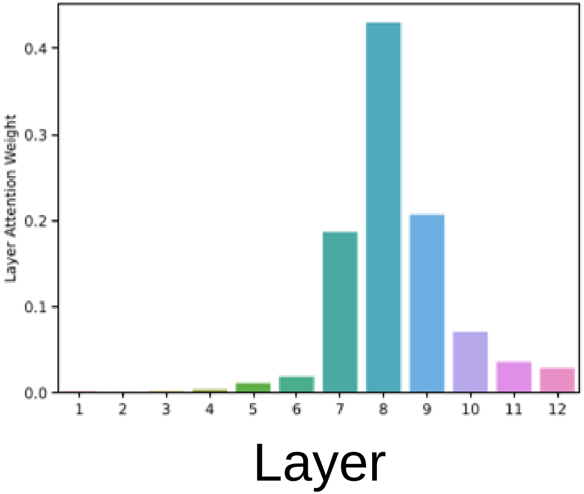
syntactic + semantic

Layer Variance across Sentences

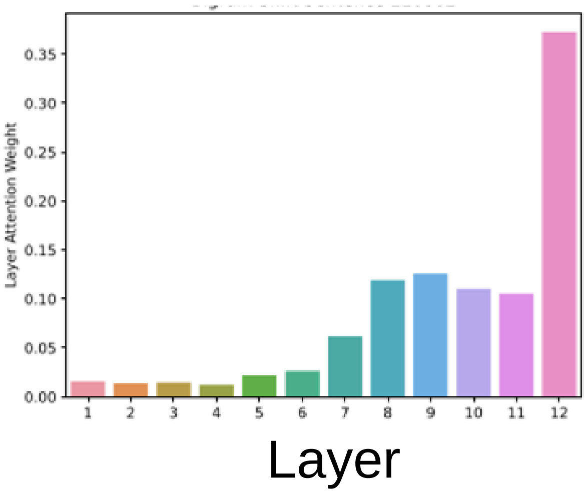
Bigram Shift sentence 110000



Bigram Shift sentence 110001



Bigram Shift sentence 110002



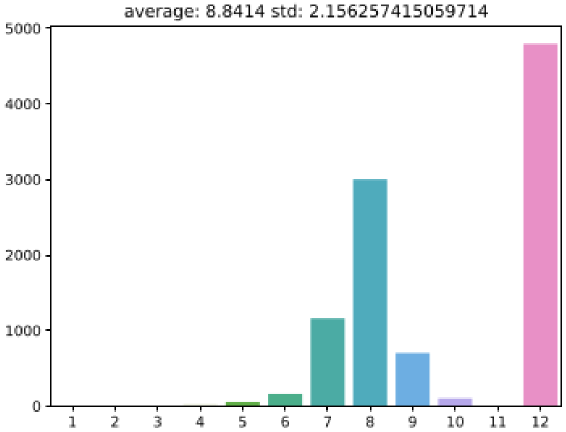
First 3 sentences of the Bigram Shift task test split.

Same GridLoc probe model at the same epoch.

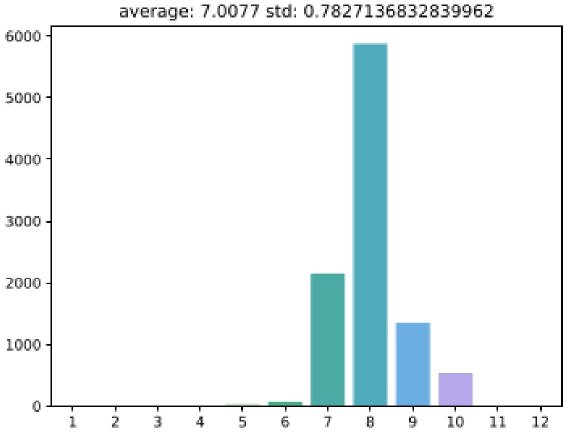
Very different layer attention weights.

Layer Variance across Random Seeds

Probe results are not immune to random initialization effects!



Seed: 0, Best Epoch: 7



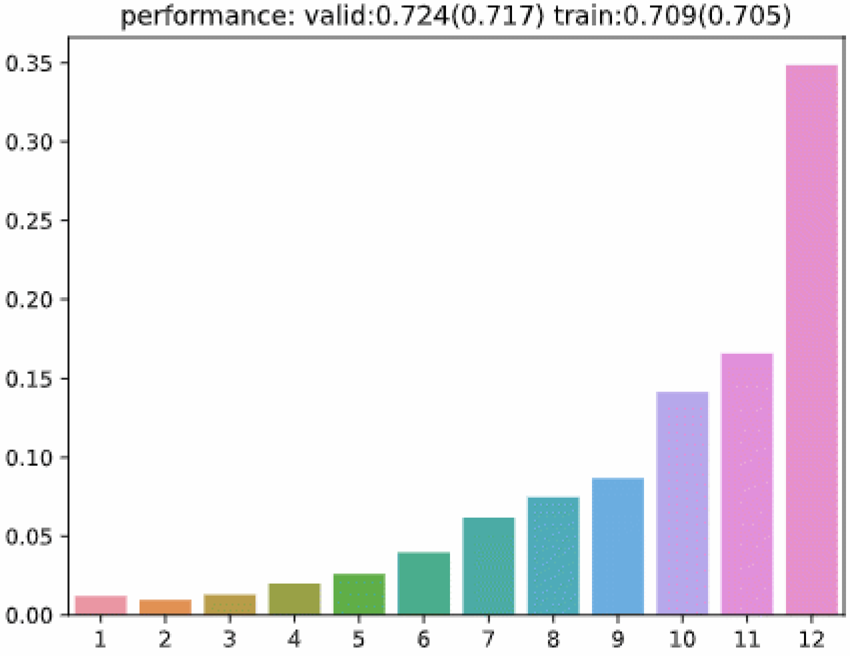
Seed: 1, Best Epoch: 8

Distribution of the best-performing layer over the Bigram Shift test set sentences for two probing runs with different random seeds.

Layer Variance through Training Time

Average layer attention weight distribution change through training iteration.

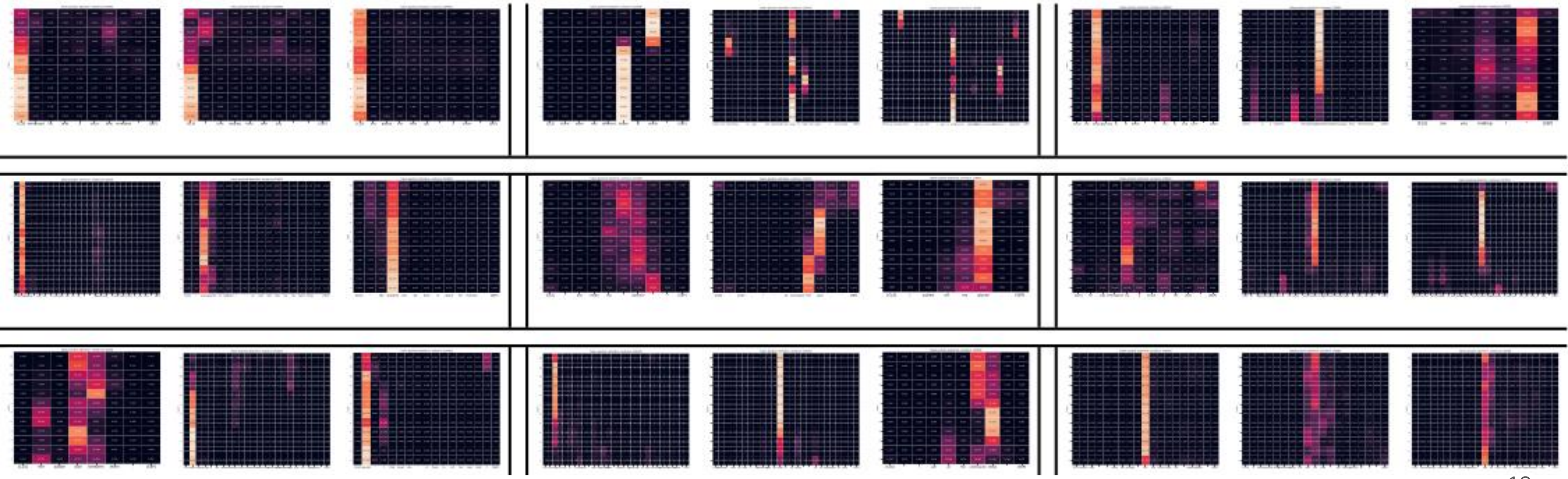
(SOMO, seed:0, best epoch: 3)

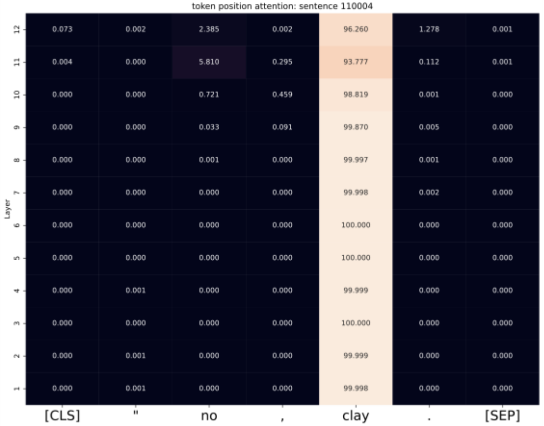


Consistently Idiosyncratic Token Positions

For most sentences, the token position attention at every layer attends to the same token, hence the bright vertical line.

The choice of that token position is not arbitrary — there are linguistic reasons for them.



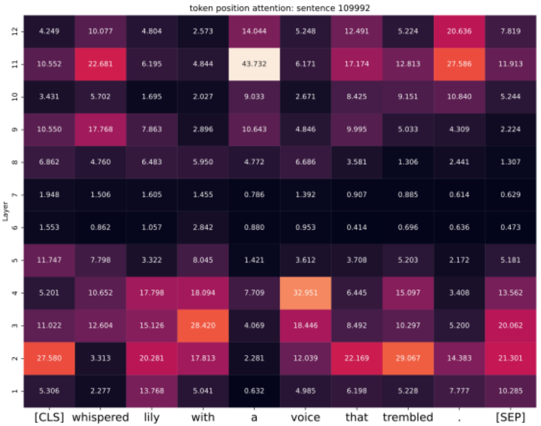


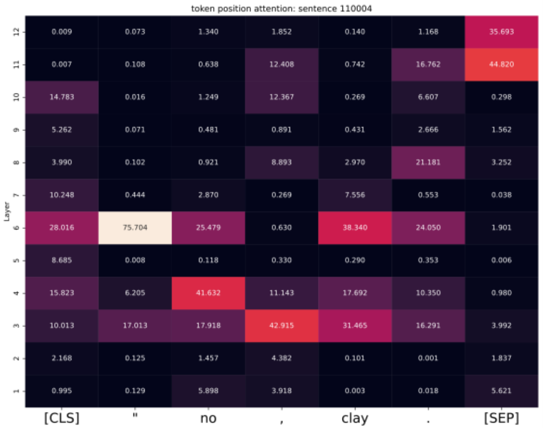
Token Position?

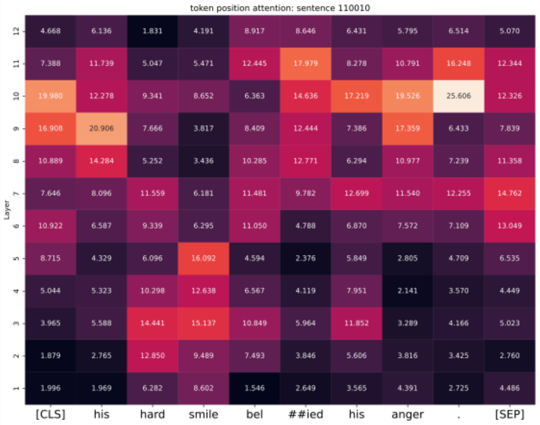
Sentence Length (sent id: 109992)

Word Content (sent id: 110004)

Tense (sent id: 110010)



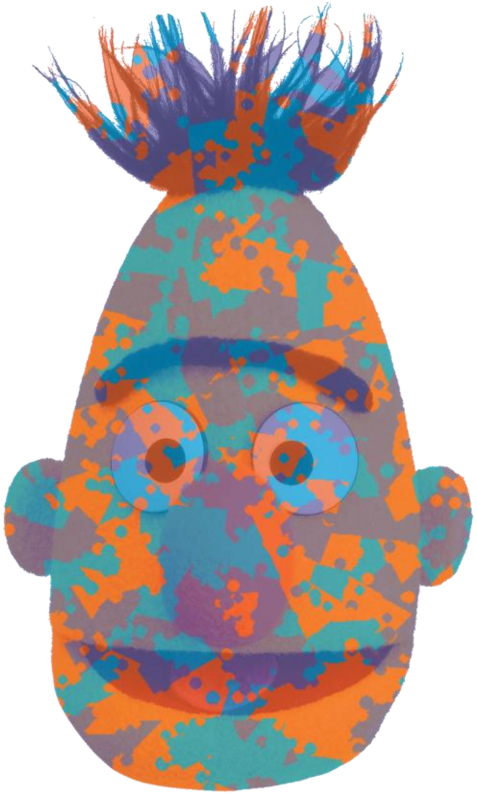




Layer?

Conclusion

- Did BERT rediscover a CNLP? Not in a naïve, architectural sense.
- Probing results regarding BERT layers are unstable; the distribution along token positions is relatively more stable.
- No evidence that pseudo-cognitive appeals to layer depth are to be preferred as the mode of explanation for BERT’s inner workings.



Grammaticality

“well formed; in accordance with the productive rules of the grammar of a language”
- lexico.com (Oxford)

From *grammatical*, “of or pertaining to grammar”
16th century: ≈ *literal*
18th century: a state of linguistic purity
19th century: relating to mere arrangement of words, as opposed to logical form or structure

Grammaticality vs. Probability

“I think we are forced to conclude that ... probabilistic models give **no** *particular insight into some of the basic problems of syntactic structure.”*
- Chomsky (1957)

Grammaticality vs. Probability (Chomsky, 1955)

colorless green ideas sleep furiously

furiously sleep ideas green colorless

Grammaticality vs. Probability (Saul & Pereira, 1997)

colorless green ideas sleep furiously

(-40.44514457)

furiously sleep ideas green colorless

(-51.41419769)

This is not only a probabilistic model, but a probabilistic language model (*Agglomerative Markov Process).*

(-39.5588693)

colorless sleep green ideas furiously

colorless ideas furiously green sleep

colorless sleep furiously green ideas

colorless green ideas sleep furiously

(-40.44514457)

furiously sleep ideas green colorless

(-51.41419769)

green furiously colorless ideas sleep

green ideas sleep colorless furiously

(-51.69151925)

Scandal!

Our ACL 2019 submission: *What Chomsky (1957) originally claimed still essentially holds: current language models do not have the ability to produce grammaticality judgements.*

ACL 2019 reviewer: *The treatment of the research literature … comes across as inflammatory.*

CGISF too small?
CoLA (Warstadt et al., 2019)

10,657 (English) examples taken from linguistics papers.

LSTM LM + threshold:

- 65.2% in-domain accuracy
- 71.1% Out-of-domain Accuracy

Not bad?

CGISF too small!

CoLA (Warstadt et al., 2019)

10,657 (English) examples taken from linguistics papers.

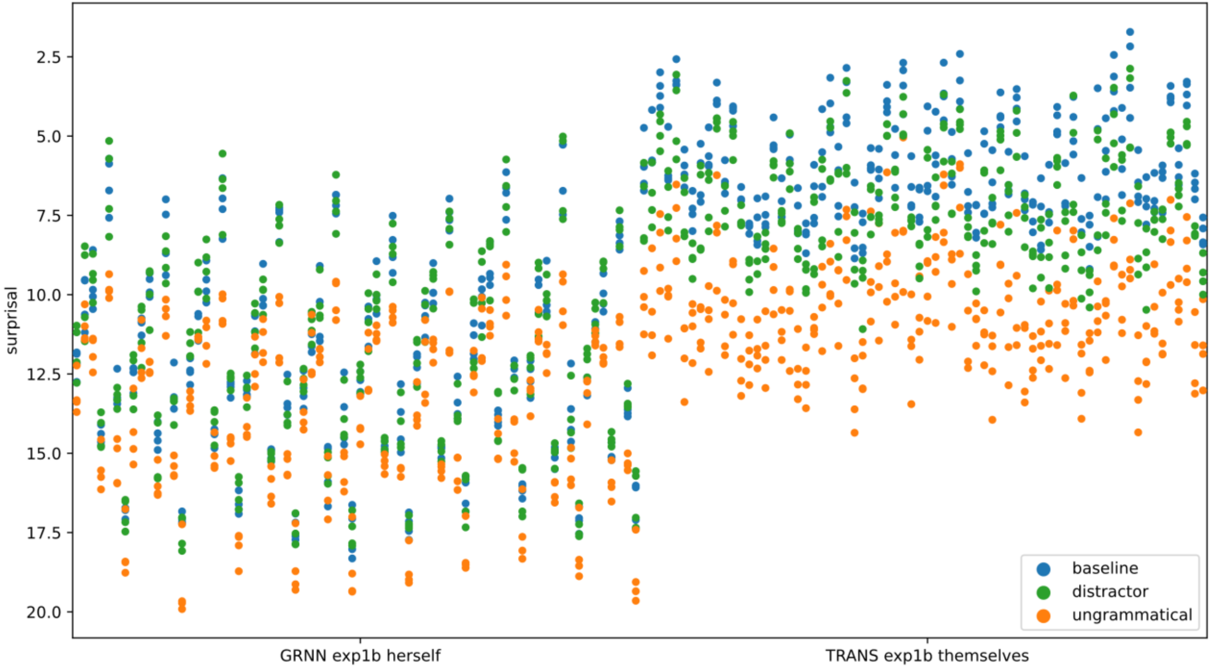
LSTM LM + threshold:

- 65.2% in-domain accuracy
- 71.1% Out-of-domain Accuracy

Not bad?

But, roughly 71% of their test set are labelled positively.

Grammaticality vs. Probability: Accuracy isn't the most suitable PBC is a better way to go



Hu et al.'s (2020): 100% accuracy 100% accuracy

0.25 PBC 0.73 PBC

Point-Biserial Correlations

- Grammaticality taken to be a binary variable (yes/no).
- The probability produced by a language model for a string of words is continuous.
- Point-biserial correlations: $$r_{pb} = \frac{M_1 - M_0}{s_n}\sqrt{pq}$$

- M_1 = mean of the continuous values assigned to samples that received the positive binary value.
- M_0 = mean of the continuous values assigned to the samples that received the negative binary value.
- S_n = standard dev. of all samples' continuous values.
- p = Proportion of samples with negative binary value.
- q = Proportion of samples with positive binary value.

What about GPT-2?

OpenAI's GPT-2 has been promoted as "an AI" that exemplifies an emergent understanding of language after mere unsupervised training on about 40GB of webpage text. It sounds really convincing in interviews:

- *Q: Which technologies are worth watching in 2020?*
  A: I would say it is hard to narrow down the list. The world is full of disruptive technologies with real and potentially huge global impacts. The most important is artificial intelligence, which is becoming exponentially more powerful. There is also the development of self-driving cars. There is a lot that we can do with artificial intelligence to improve the world….
- *Q: Are you worried that ai* [sic] *technology can be misused?*
  A: Yes, of course. But this is a global problem and we want to tackle it with global solutions….

--- "AI can do that", *The World in 2020 – The Economist*

Surely something this sophisticated can predict grammaticality, right?

Wrong

Model	Norm.	GPT-2		GPT-2 XL	
		LOG	EXP	LOG	EXP
GPT-2 Models	Raw	0.1839	0.0117	0.1476	0.0123
	Norm	0.2498	0.1643	0.2241	0.1592
	SLOR	0.2489	0.092	0.2729	0.0872

- Should conclusions about grammaticality be based upon scientific experimentation or self-congratulatory PR stunts?
- People are very good at attributing interpretations to natural phenomena that defy interpretation.

Legitimate Points of Concern

- Is grammaticality really a discrete variable?
  - Several have argued that a presumed correlation between neural language models and grammaticality suggests that grammaticality should be viewed as gradient (Lau et al., 2017; Sprouse et al., 2018).
- Eliciting grammaticality ≠ blindly probing the elephant.
  - Numerous papers on individual features of grammaticality (Linzen et al., 2016; Bernardy & Lappin, 2017; Gulordava et al., 2018).
- How do you sample grammaticality judgements?
  - Acceptability judgements (Sprouse & Almeida 2012; Sprouse et al., 2013) are not quite the same thing – experimental subjects can easily be misled by interpretability.
  - Round-trip machine translation of grammatical sentences for generating ungrammatical strings (Lau et al., 2014;2015).

The Deep Learning Advantage?

- There is now a robust thread of research that uses language models for tasks other than predicting the next word, not because they are the best approach, but because the people using them are scientifically illiterate:
    - What language consists of and how it works,
    - How to evaluate performance and progress in the task.
- When these models work well at all, they often get credit just for placing.
- Grammaticality prediction is one of these tasks.

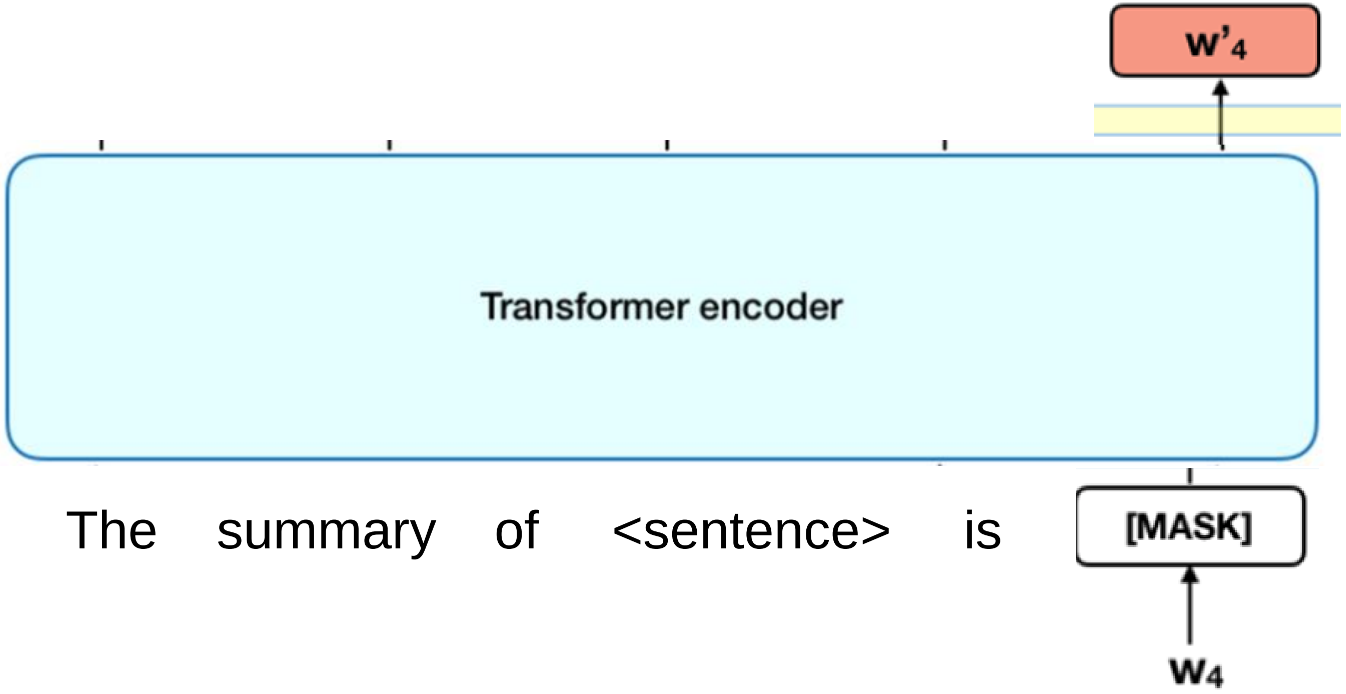
The Deep Learning Retort

- In the case of grammaticality, the reply by this community has been:
  - To blame linguists for coining a task (they didn’t) that is ill posed (it isn’t),
  - To shift to a different, easier task, relative grammaticality, which is also known to be more stable across samples of human annotations.
- Pedestrian attempts at promoting deep learning will often represent fields such as CL as blindly hunting for “hand-crafted” features in order to improve the performance of their classifiers.
- In fact, several discriminative pattern-recognition methods were already in widespread use before the start of the “deep learning revolution” that had made this approach very unattractive.

The Deep Learning Advantage

- Nevertheless, deep learning is adding value, but more in terms of:
  - Modularity of the different network layers that allows for separation and recombination,
  - Novelty of the approaches, even if performance isn't state of the art, and
  - the "liberated practitioner," who can now produce a baseline system with very little expertise that has a higher accuracy than earlier naïve baselines.

Encoder “LMs”– thinking outside the box



PromptBert (*Jiang et al., 2022*)