CS 489: Introduction to Natural Language Processing

Lecture 12: Probing Neural LMs I: Syntactic Knowledge

Instructor: Freda Shi

October 29, 2026



UNIVERSITY OF WATERLOO

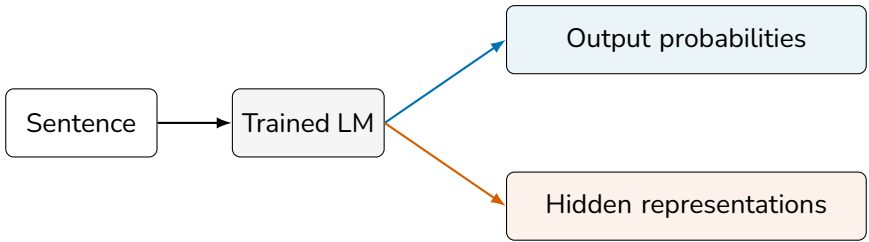
Outline

Lecture 11: We represented syntax explicitly with trees.

What syntactic information do neural language models learn?

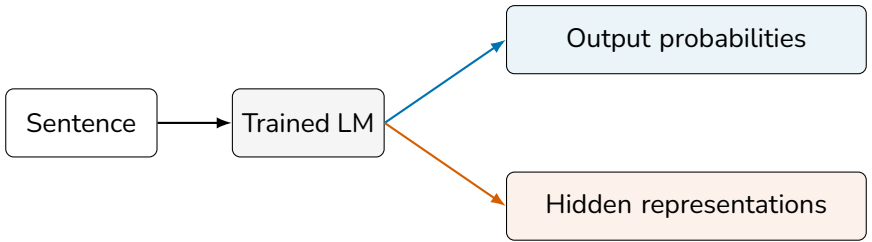
1. Do their predictions respect syntactic dependencies?
2. Can we recover trees from their hidden states?
3. What conclusions do these experiments support?

Two Kinds of Evidence



Behavioral test: Does the model prefer a grammatical form?

Two Kinds of Evidence



Behavioral test: Does the model prefer a grammatical form?

Representation probe: Can a trained readout recover syntactic information from frozen hidden states?

Subject-Verb Agreement

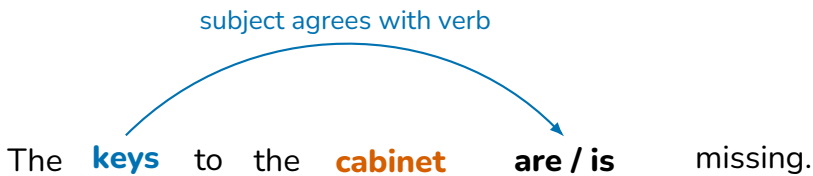
Which verb form completes this sentence?

The **keys** to the **cabinet** **are / is** missing.

[Linzen, Dupoux, & Goldberg (2016). Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies. *TACL*.]

Subject-Verb Agreement

Which verb form completes this sentence?



Are: the subject head is plural **keys**, not singular **cabinet**.

An **attractor** is an intervening noun whose number differs from the subject's. A nearest-noun heuristic predicts the wrong form here.

[Linzen, Dupoux, & Goldberg (2016). Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies. *TACL*.]

Counting Attractors

Intervening nouns are not necessarily attractors.

Sentence	Attractors
The keys **are** missing.	
The keys to the cabinets **are** missing.	
The keys to the cabinet **are** missing.	
The keys to the cabinet near the door **are** missing.	

Counting Attractors

Intervening nouns are not necessarily attractors.

Sentence	Attractors
The keys **are** missing.	0
The keys to the cabinets **are** missing.	0
The keys to the cabinet **are** missing.	1
The keys to the cabinet near the door **are** missing.	2

Counting Attractors

Intervening nouns are not necessarily attractors.

Sentence	Attractors
The keys **are** missing.	0
The keys to the cabinets **are** missing.	0
The keys to the cabinet **are** missing.	1
The keys to the cabinet near the door **are** missing.	2

Compare matched constructions and report accuracy by attractor count.
Otherwise sentence length, structure, and noun number can be confounded.

Scoring the Verb in an Autoregressive LM

Hold the prefix fixed:

$$c = \text{“The keys to the cabinet”}.$$

Compare the correct inflection with the incorrect one:

$$P_{\text{LM}}(\text{are} \mid c) \quad > \quad P_{\text{LM}}(\text{is} \mid c) \ ?$$

Each candidate is one token here; this test scores only the critical verb.

Scoring the Verb in an Autoregressive LM

Hold the prefix fixed:

$$c = \text{“The keys to the cabinet”}.$$

Compare the correct inflection with the incorrect one:

$$P_{\text{LM}}(\text{are} \mid c) \quad > \quad P_{\text{LM}}(\text{is} \mid c) \ ?$$

Illustrative probabilities: $P(\text{are} \mid c) = 0.12$, $P(\text{is} \mid c) = 0.03$.

$$\log P(\text{are} \mid c) - \log P(\text{is} \mid c) = \log 4 > 0.$$

The model passes this item even if another word is its most likely continuation.

Each candidate is one token here; this test scores only the critical verb.

Targeted Syntactic Evaluation

Marvin & Linzen (2018): Generate controlled contrasts instead of relying only on naturally occurring sentences.

A **minimal pair** differs in a small, targeted change that affects grammaticality.

✓ The farmer that the parents love **swims**.

* The farmer that the parents love **swim**.

[Marvin & Linzen (2018). Targeted Syntactic Evaluation of Language Models. *EMNLP*.]

Targeted Syntactic Evaluation

Marvin & Linzen (2018): Generate controlled contrasts instead of relying only on naturally occurring sentences.

A **minimal pair** differs in a small, targeted change that affects grammaticality.

✓ The farmer that the parents love **swims**.

* The farmer that the parents love **swim**.

- Vary the construction, subject number, and intervening nouns systematically.
- Keep the other words fixed within each pair.
- Templates reduce confounds; they do not eliminate lexical shortcuts.

[Marvin & Linzen (2018). Targeted Syntactic Evaluation of Language Models. *EMNLP*.]

Scoring a Full Minimal Pair

For an autoregressive LM, score the tokenized sentence x using

$$s(x) = \log P_{\mathrm{LM}}(x) = \sum_{t=1}^{|x|} \log P_{\mathrm{LM}}(x_t \mid x_{<t}).$$

For grammatical/ungrammatical pairs (g_m, u_m):

$$\text{Accuracy} = \frac{1}{M} \sum_{m=1}^{M} \mathbf{1}[s(g_m) > s(u_m)] \,.$$

Marvin & Linzen (2018) and BLiMP compare whole-sentence probabilities.

Scoring a Full Minimal Pair

For an autoregressive LM, score the tokenized sentence x using

$$s(x) = \log P_{\mathrm{LM}}(x) = \sum_{t=1}^{|x|} \log P_{\mathrm{LM}}(x_t \mid x_{<t}).$$

For grammatical/ungrammatical pairs (g_m, u_m):

$$\text{Accuracy} = \frac{1}{M} \sum_{m=1}^{M} \mathbf{1}[s(g_m) > s(u_m)] \,.$$

Different from critical-verb scoring:

- The shared prefix contributes the same score to both sentences.
- After the change, later tokens have different contexts and can affect the result.

Marvin & Linzen (2018) and BLiMP compare whole-sentence probabilities.

Minimal Pairs Beyond Agreement

Reflexive anaphora

- The authors amused **themselves** / ***himself**.

Here the reflexive refers back to the plural subject, **authors**.

Minimal Pairs Beyond Agreement

Reflexive anaphora

- The authors amused **themselves** / ***himself**.

Here the reflexive refers back to the plural subject, **authors**.

Negative polarity items (NPIs)

- **No** authors have **ever** laughed.
- ***The** authors have **ever** laughed.

In this contrast, **no** licenses **ever**. NPIs also occur in other environments, including questions.

Controlling Lexical Content

A model might exploit familiar word combinations rather than the intended dependency.

Natural: The keys to the cabinet **are** missing.

Nonce: The poems to the onion **are** missing.

Course example of the nonce-sentence method used in Gulordava et al. (2018) and Goldberg (2019).

[Gulordava et al. (2018). Colorless Green Recurrent Networks Dream Hierarchically. *NAACL*.]

Controlling Lexical Content

A model might exploit familiar word combinations rather than the intended dependency.

Natural: The keys to the cabinet **are** missing.

Nonce: The poems to the onion **are** missing.

Replace content words while preserving their syntactic categories and morphology. The sentence becomes semantically unusual, but the agreement target stays plural.

Success weakens a purely lexical explanation; it does not rule out every surface heuristic.

Course example of the nonce-sentence method used in Gulordava et al. (2018) and Goldberg (2019).

[Gulordava et al. (2018). Colorless Green Recurrent Networks Dream Hierarchically. *NAACL*.]

Scoring Agreement with BERT

Goldberg (2019): Mask the target verb in the complete sentence.

The author that the **editors** `[MASK]` laughs.

$$P_{\text{BERT}}(\text{like} \mid x_{\backslash v}) \quad > \quad P_{\text{BERT}}(\text{likes} \mid x_{\backslash v})\ ?$$

Here $x_{\backslash v}$ is the sentence with its target verb replaced by `[MASK]`.

[Goldberg (2019). Assessing BERT's Syntactic Abilities. arXiv:1901.05287.]

Scoring Agreement with BERT

Goldberg (2019): Mask the target verb in the complete sentence.

The author that the **editors** `[MASK]` laughs.

$$P_{\text{BERT}}(\text{like} \mid x_{\backslash v}) \quad > \quad P_{\text{BERT}}(\text{likes} \mid x_{\backslash v})\ ?$$

Here $x_{\backslash v}$ is the sentence with its target verb replaced by `[MASK]`.

- BERT sees both left and right context, unlike a prefix-only LM.
- Only single-token candidates are retained; natural/nonce sets also exclude *is/are*.
- This masked-token score is not a full-sentence probability.

[Goldberg (2019). Assessing BERT's Syntactic Abilities. arXiv:1901.05287.]

Goldberg (2019): Agreement Results

Evaluation set	BERT-base	BERT-large
Natural sentences, 1–4 attractors	96–97%	96–97%
Nonce sentences	83%	80%

Goldberg (2019): Agreement Results

Evaluation set	BERT-base	BERT-large
Natural sentences, 1–4 attractors	96–97%	96–97%
Nonce sentences	83%	80%

Strong agreement behavior persists on the nonce set. Larger is not consistently better in these evaluations.

BLiMP: A Broader Set of Minimal Pairs

BLiMP: The Benchmark of Linguistic Minimal Pairs for English (Warstadt et al., 2020).

- **67 paradigms** across **12 linguistic phenomena**.
- **1,000 pairs per paradigm**: 67,000 pairs in total.
- Automatically generated from expert-designed grammars.
- Evaluate pretrained LMs without training on the benchmark.

[Warstadt et al. (2020). BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *TACL*, corrected version.]

BLiMP: A Broader Set of Minimal Pairs

BLiMP: The Benchmark of Linguistic Minimal Pairs for English (Warstadt et al., 2020).

- **67 paradigms** across **12 linguistic phenomena**.
- **1,000 pairs per paradigm**: 67,000 pairs in total.
- Automatically generated from expert-designed grammars.
- Evaluate pretrained LMs without training on the benchmark.

Agreement is only one part of the benchmark. Other phenomena include binding, argument structure, filler-gap dependencies, island effects, and NPI licensing.

[Warstadt et al. (2020). BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *TACL*, corrected version.]

BLiMP: Results Vary by Phenomenon

Phenomenon	5-gram	LSTM	GPT-2-large	Human
Anaphor agreement	47.9%	91.7%	99.3%	97.5%
Subject-verb agreement	60.3%	80.1%	86.4%	90.9%
Island effects	57.2%	46.6%	72.7%	84.9%
NPI licensing	45.5%	51.7%	76.8%	88.1%
Overall	61.2%	69.8%	83.0%	88.6%

Corrected arXiv v4, Table 3 (selected rows/models); overall uses all 67 paradigms. Human: raw individual forced-choice judgments on sampled pairs.

BLiMP: Results Vary by Phenomenon

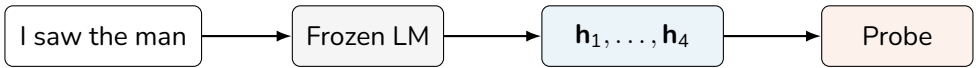
Phenomenon	5-gram	LSTM	GPT-2-large	Human
Anaphor agreement	47.9%	91.7%	99.3%	97.5%
Subject-verb agreement	60.3%	80.1%	86.4%	90.9%
Island effects	57.2%	46.6%	72.7%	84.9%
NPI licensing	45.5%	51.7%	76.8%	88.1%
Overall	61.2%	69.8%	83.0%	88.6%

GPT-2 is strong on anaphor agreement but weaker on islands and NPIs. A single overall score hides these differences.

Corrected arXiv v4, Table 3 (selected rows/models); overall uses all 67 paradigms. Human: raw individual forced-choice judgments on sampled pairs.

From Predictions to Hidden States

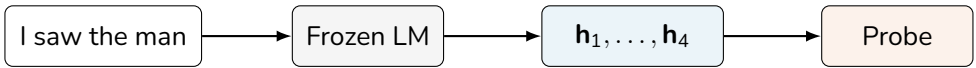
Can we recover syntax from representations computed by a **frozen LM**?



[Hewitt & Manning (2019). A Structural Probe for Finding Syntax in Word Representations. *NAACL*.]

From Predictions to Hidden States

Can we recover syntax from representations computed by a **frozen LM**?



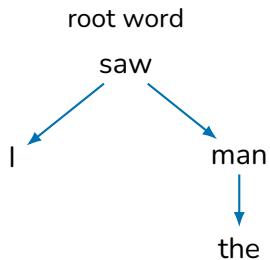
Hewitt & Manning (2019): Fit a linear map so that squared distances between transformed word vectors approximate dependency-tree distances.

Syntax labels train the **probe**, not the language model.

[Hewitt & Manning (2019). A Structural Probe for Finding Syntax in Word Representations. *NAACL*.]

Gold Targets: A Dependency Tree

Reuse **I saw the man** from Lecture 11.

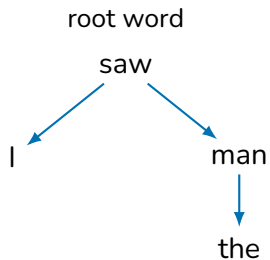


Distance: Count edges on the undirected path between two words.

Depth: Count edges from the root word; **saw** has depth 0. We exclude the artificial ROOT node.

Gold Targets: A Dependency Tree

Reuse **I saw the man** from Lecture 11.



d^{tree}	I	saw	the	man
I	0	1	3	2
saw	1	0	2	1
the	3	2	0	1
man	2	1	1	0
Depth	1	0	2	1

Distance: Count edges on the undirected path between two words.

Depth: Count edges from the root word; **saw** has depth 0. We exclude the artificial ROOT node.

The Distance Probe

At a fixed layer, each word has a frozen vector $\mathbf{h}_i \in \mathbb{R}^d$. Learn a shared map $\mathbf{B}_{\text{dist}} \in \mathbb{R}^{k \times d}$:

$$\mathbf{z}_i = \mathbf{B}_{\text{dist}} \mathbf{h}_i.$$

Hewitt & Manning (2019), Section 2.

The Distance Probe

At a fixed layer, each word has a frozen vector $\mathbf{h}_i \in \mathbb{R}^d$. Learn a shared map $\mathbf{B}_{\text{dist}} \in \mathbb{R}^{k \times d}$:

$$\mathbf{z}_i = \mathbf{B}_{\text{dist}} \mathbf{h}_i.$$

Predict the tree distance using a **squared Euclidean distance**:

$$\widehat{d}(i, j) = \|\mathbf{z}_i - \mathbf{z}_j\|_2^2 = \|\mathbf{B}_{\text{dist}}(\mathbf{h}_i - \mathbf{h}_j)\|_2^2.$$

The same map is applied to every word in every sentence. The output dimension k bounds its rank; the paper also permits full-rank maps.

Hewitt & Manning (2019), Section 2.

Why Squared Distance? An Exact Toy Geometry

Use one coordinate per edge: saw $\to$ I, saw $\to$ man, man $\to$ the. Sum orthogonal unit vectors along the root-to-word path.

Word	Illustrative vector $\mathbf{z}_i$	$\|\mathbf{z}_i\|_2^2$
I	$(1, 0, 0)$	
saw	$(0, 0, 0)$	
the	$(0, 1, 1)$	
man	$(0, 1, 0)$	

An exact construction for this tree, not measured LM vectors.

Why Squared Distance? An Exact Toy Geometry

Use one coordinate per edge: saw $\to$ I, saw $\to$ man, man $\to$ the. Sum orthogonal unit vectors along the root-to-word path.

Word	Illustrative vector $\mathbf{z}_i$	$\|\mathbf{z}_i\|_2^2$
I	$(1,0,0)$	1
saw	$(0,0,0)$	0
the	$(0,1,1)$	2
man	$(0,1,0)$	1

$$\|\mathbf{z}_{\mathrm{I}} - \mathbf{z}_{\mathrm{the}}\|_2^2 = \|(1,-1,-1)\|_2^2 = 3 = d^{\mathrm{tree}}(\mathrm{I},\mathrm{the}).$$

Shared path components cancel; each remaining edge contributes 1.

An exact construction for this tree, not measured LM vectors.

Fitting the Distance Probe

Training data: sentences with gold dependency trees.

1. Run the frozen LM to obtain word representations.
2. Convert each gold tree into a matrix of path lengths.
3. Update only $\mathbf{B}_{\text{dist}}$ to reduce prediction error.

Fitting the Distance Probe

Training data: sentences with gold dependency trees.

1. Run the frozen LM to obtain word representations.
2. Convert each gold tree into a matrix of path lengths.
3. Update only $\mathbf{B}_{\text{dist}}$ to reduce prediction error.

For training sentences $s \in \mathcal{D}$, with n_s words:

$$\mathcal{L}_{\text{dist}} = \sum_{s \in \mathcal{D}} \frac{1}{n_s^2} \sum_{i,j=1}^{n_s} \left| \widehat{d}_s(i,j) - d_s^{\text{tree}}(i,j) \right| .$$

Use absolute error, averaged over word pairs within each sentence. Evaluate on held-out sentences, not the trees used to fit the probe.

The Depth Probe

Fit a **separate** map $\mathbf{B}_{\text{depth}}$ to predict each word's depth:

$$\widehat{\delta}(i) = \|\mathbf{B}_{\text{depth}}\mathbf{h}_i\|_2^2 \approx \delta^{\text{tree}}(i).$$

$$\mathcal{L}_{\text{depth}} = \sum_{s \in \mathcal{D}} \frac{1}{n_s} \sum_{i=1}^{n_s} \left| \widehat{\delta}_s(i) - \delta_s^{\text{tree}}(i) \right|.$$

The Depth Probe

Fit a **separate** map $\mathbf{B}_{\text{depth}}$ to predict each word's depth:

$$\widehat{\delta}(i) = \|\mathbf{B}_{\text{depth}}\mathbf{h}_i\|_2^2 \approx \delta^{\text{tree}}(i).$$

$$\mathcal{L}_{\text{depth}} = \sum_{s \in \mathcal{D}} \frac{1}{n_s} \sum_{i=1}^{n_s} \left| \widehat{\delta}_s(i) - \delta_s^{\text{tree}}(i) \right|.$$

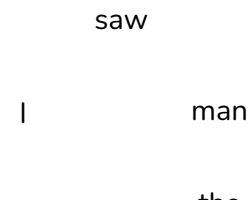
For **I saw the man**, the gold depths are $(1, 0, 2, 1)$. The exact toy vectors have these squared norms.

Predict the root word with $\arg\min_i \widehat{\delta}(i)$. Distance and depth probes are trained separately; their fitted maps need not agree.

Recovering an Undirected Tree

A **minimum spanning tree** connects all words with minimum total edge weight. Add the cheapest edge that creates no cycle; stop after $n - 1$ edges.

Word pair	Toy $\widehat{d}$	Decision
I–man	0.9	
saw–man	1.0	
I–saw	1.1	
the–man	1.2	
saw–the	1.8	
I–the	2.4	

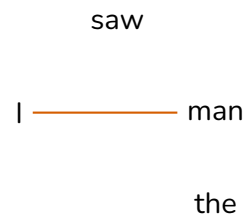


Illustrative imperfect probe scores, distinct from the exact geometry above.

Recovering an Undirected Tree

A **minimum spanning tree** connects all words with minimum total edge weight. Add the cheapest edge that creates no cycle; stop after $n - 1$ edges.

Word pair	Toy $\widehat{d}$	Decision
I–man	0.9	Add
saw–man	1.0	
I–saw	1.1	
the–man	1.2	
saw–the	1.8	
I–the	2.4	

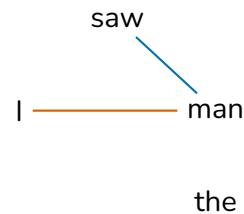


Illustrative imperfect probe scores, distinct from the exact geometry above.

Recovering an Undirected Tree

A **minimum spanning tree** connects all words with minimum total edge weight. Add the cheapest edge that creates no cycle; stop after $n - 1$ edges.

Word pair	Toy $\widehat{d}$	Decision
I–man	0.9	Add
saw–man	1.0	Add
I–saw	1.1	
the–man	1.2	
saw–the	1.8	
I–the	2.4	

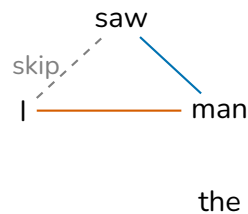


Illustrative imperfect probe scores, distinct from the exact geometry above.

Recovering an Undirected Tree

A **minimum spanning tree** connects all words with minimum total edge weight. Add the cheapest edge that creates no cycle; stop after $n - 1$ edges.

Word pair	Toy $\widehat{d}$	Decision
I–man	0.9	Add
saw–man	1.0	Add
I–saw	1.1	Skip: cycle
the–man	1.2	
saw–the	1.8	
I–the	2.4	

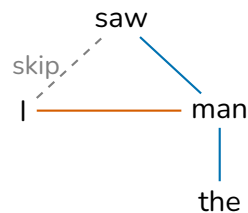


Illustrative imperfect probe scores, distinct from the exact geometry above.

Recovering an Undirected Tree

A **minimum spanning tree** connects all words with minimum total edge weight. Add the cheapest edge that creates no cycle; stop after $n - 1$ edges.

Word pair	Toy $\widehat{d}$	Decision
I–man	0.9	Add
saw–man	1.0	Add
I–saw	1.1	Skip: cycle
the–man	1.2	Add; stop
saw–the	1.8	
I–the	2.4	

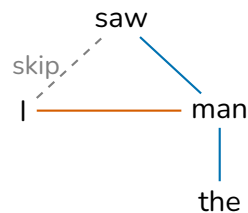


Illustrative imperfect probe scores, distinct from the exact geometry above.

Recovering an Undirected Tree

A **minimum spanning tree** connects all words with minimum total edge weight. Add the cheapest edge that creates no cycle; stop after $n - 1$ edges.

Word pair	Toy $\widehat{d}$	Decision
I–man	0.9	Add
saw–man	1.0	Add
I–saw	1.1	Skip: cycle
the–man	1.2	Add; stop
saw–the	1.8	
I–the	2.4	



UUAS (undirected unlabeled attachment score): 2/3 gold edges recovered. The edge I–man is wrong; distances alone do not supply directions or labels.

Illustrative imperfect probe scores, distinct from the exact geometry above.

Structural Probes: Results and Baselines

UUAS: fraction of gold undirected edges recovered.

Root accuracy: fraction of sentences with the correct least-deep word.

Representation	UUAS	Root accuracy
ELMo0 (no context)	26.8%	54.3%
PROJ0 (random BiLSTM baseline)	59.8%	64.4%
BERT-base (layer 7)	79.8%	88.0%

Hewitt & Manning (2019), Table 1 (selected results; punctuation excluded). PROJ0 uses pretrained ELMo0 inputs and a random BiLSTM.

Structural Probes: Results and Baselines

UUAS: fraction of gold undirected edges recovered.

Root accuracy: fraction of sentences with the correct least-deep word.

Representation	UUAS	Root accuracy
ELMo0 (no context)	26.8%	54.3%
PROJ0 (random BiLSTM baseline)	59.8%	64.4%
BERT-base (layer 7)	79.8%	88.0%

BERT's representations support substantial but imperfect tree recovery. The strong random-network baseline makes a raw probe score hard to interpret alone.

Hewitt & Manning (2019), Table 1 (selected results; punctuation excluded). PROJ0 uses pretrained ELMo0 inputs and a random BiLSTM.

Designing the Probe Experiment

1. **Fix the representation extraction.** Freeze the LM and specify how subword states become word vectors.
2. **Split by sentence into training, validation, and test sets.** Fit on training trees; choose layer, rank, and other settings on validation data.
3. **Report held-out results and matched baselines.** Use the same probe protocol with contextual, non-contextual, and random-network representations.

Designing the Probe Experiment

1. **Fix the representation extraction.** Freeze the LM and specify how subword states become word vectors.
2. **Split by sentence into training, validation, and test sets.** Fit on training trees; choose layer, rank, and other settings on validation data.
3. **Report held-out results and matched baselines.** Use the same probe protocol with contextual, non-contextual, and random-network representations.

Selecting the best layer on the test set leaks information into evaluation. Report layer-wise variation rather than only an unexplained peak score.

A Simple Probe Still Learns from Labels

A linear transformation limits the readout, but it is not a guarantee that the probe merely “finds” syntax.

- $\mathbf{B} \in \mathbb{R}^{k \times d}$ still has *kd* trainable parameters.
- The structural score $\|\mathbf{B}(\mathbf{h}_i - \mathbf{h}_j)\|^2$ is quadratic in the input difference.
- A strong score can reflect useful representations, probe capacity, or both.

A Simple Probe Still Learns from Labels

A linear transformation limits the readout, but it is not a guarantee that the probe merely “finds” syntax.

- $\mathbf{B} \in \mathbb{R}^{k \times d}$ still has *kd* trainable parameters.
- The structural score $\|\mathbf{B}(\mathbf{h}_i - \mathbf{h}_j)\|^2$ is quadratic in the input difference.
- A strong score can reflect useful representations, probe capacity, or both.

Compare capacity settings using the same representations and labeled data.

Accessible Information Is Not Necessarily Used

A probe asks:

Can a readout recover syntax from the model's hidden states?

Accessible Information Is Not Necessarily Used

A probe asks:

Can a readout recover syntax from the model's hidden states?

A causal question asks:

Does changing that information change the model's own predictions?

Accessible Information Is Not Necessarily Used

A probe asks:

Can a readout recover syntax from the model's hidden states?

A causal question asks:

Does changing that information change the model's own predictions?

Behavioral tests constrain what the model gets right. Probes constrain what information is accessible through a chosen readout.

Neither alone identifies the mechanism producing the behavior.

[Recommended reading: Belinkov (2022). Probing Classifiers: Promises, Shortcomings, and Advances. *Computational Linguistics*.]

Next Lecture

Mechanistic Interpretability and Knowledge Probing

From reading information out of hidden states to testing the computations that use it.

Circuits, activation patching, and factual knowledge.