Understanding language learning using computational methods

Lisa Pearl
University of California, Irvine
lpearl@uci.edu

Feb 18, 2013: Department of Cognitive Science
Johns Hopkins University
Language learning as ongoing mental computation
Language learning as ongoing mental computation

Language learning = given the available input,
Language learning as ongoing mental computation

Language learning = given the available input, information processing done by human minds

Input

Who did he find?
What happened?

Mental computation

lʊkætðəkɪɾi
Language learning as ongoing mental computation

Language learning = given the available input, information processing done by human minds to build a system of linguistic knowledge.
Language learning as ongoing mental computation

Language learning = given the available input, information processing done by human minds to build a system of linguistic knowledge whose output we observe.

Look at the kitty

Abstraction & generalization

Input

Who did he find?
What happened?

Output

Where’s the kitty?
Investigating language learning

Many different questions about this mental computation
Investigating language learning

Many different questions about this mental computation

What learning strategies comprise it?
(Phillips & Pearl in prep., Phillips & Pearl 2012, Pearl et al. 2011, Pearl et al. 2010)
Investigating language learning

Many different questions about this mental computation

What learning strategies comprise it?

What learning biases do children need to succeed at it?

(Pearl & Mis in rev., Pearl & Sprouse forthcoming, Pearl & Sprouse 2013, Pearl & Mis 2011, Pearl & Lidz 2009, Pearl 2008, Pearl & Weinberg 2007)
Investigating language learning

Many different questions about this mental computation

What learning strategies comprise it?

What learning biases do children need to succeed at it?

What knowledge representations can be learned using it?
(Pearl et al. in prep., Pearl 2011, Pearl 2009)
Investigating language learning

Many different questions about this mental computation

What learning strategies comprise it?

What learning biases do children need to succeed at it?

What knowledge representations can be learned using it?

When do children learn different aspects of the linguistic system using it, what data are available to them to do so, and what factors underlie their output?

(Pearl & Sarnecka in prep., Pearl & Braunwald in prep., Caponigro, Pearl et al. 2012, Caponigro, Pearl et al. 2011)
Methods of empirical investigation
Theoretical methods:

**What** knowledge of language is (and what children have to learn)

LOOK at the Kitty

lkætðəkiri

\[ [+\text{stop}] \mid [+\text{consonant}] \mid [+\text{alveolar}] \rightarrow [\text{r}] \mid [+\text{vowel}] \mid [+\text{stressed}] \mid [+\text{vowel}] \mid [-\text{stressed}] \]
Methods of empirical investigation

Experimental methods:
*When* knowledge is acquired, what the **input** looks like, & plausible capabilities underlying **how** acquisition works

\[
\frac{p(kitty)}{p(ki)} \\
p(H1 | \cdot) \propto p(\cdot | H1) \cdot p(H1)
\]
Methods of empirical investigation

Computational methods:
Strategies for how children acquire knowledge, sophisticated quantitative analysis of children’s input & output

XP-YP-ZP...

start-XP-YP + 1
...

“What did…”
Today’s Plan

Using computational methods to look at two questions about children’s mental computation
Today’s Plan

Using computational methods to look at two questions about children’s mental computation

What learning strategies comprise it?
Looking for strategies that are useful, useable, and work better with limited cognitive resources
Today’s Plan

Using computational methods to look at two questions about children’s mental computation

What learning strategies comprise it?
Looking for strategies that are useful, useable, and work better with limited cognitive resources

What learning biases do children need to succeed at it?
Understanding the nature of children’s language learning toolkit
Today’s Plan

Using computational methods to look at two questions about children’s mental computation

**What learning strategies comprise it?**
Looking for strategies that are useful, useable, and work better with limited cognitive resources

**What learning biases do children need to succeed at it?**
Understanding the nature of children’s language learning toolkit

Case study:
Word segmentation

Case study:
Word segmentation
Investigating learning strategies

For any potential strategy:

Is it \textit{useful}?

What is \textbf{possible} to learn from the available data?

- Ideal/rational models, computational-level approach

- What data representations are useful? What learning assumptions are useful?
Investigating learning strategies

For any potential strategy:
Is it useful?
Is it useable?

What is possible for children to learn from the available data?
  • Constrained/process models, algorithmic-level approach

  • Are these representations and assumptions still useful if cognitive resources are limited?
Invesga2ng	
  
  learning	
  
  strategies

For any potential strategy:

Is it useful?

Is it useable?

Does it work better when cognitive resources are constrained?

“Less is more” hypothesis of Newport (1990): Children do better precisely because they have more limited cognitive abilities.

• Also adults (sometimes) when their abilities are inhibited

(Cochran et al. 1999, Kersten et al. 2001 but see Perfors 2011)

• What learning strategies have this property?
Case study: Word segmentation

A big deal: Basis for more complex linguistic knowledge

look at the kitty

look at the the kitty

LOOK at the Kitty

phonology

syntax

semantics
Case study: Word segmentation

Also, we have pretty good empirical grounding. We know a lot about

(1) the data available (CHILDES)

(2) what cues children are sensitive to when

Case study: Word segmentation

Cognitive modeling: Given a corpus of fluent speech or text, we want to identify the words (units useful for mapping meaning).

whatsthat
thekitty
yeah
wheresthekitty

whats that
the kitty
yeah
wheres the kitty
Word segmentation strategies

- Language-dependent cues: phonotactics, allophonic variation, metrical (stress) patterns, effects of coarticulation

Problem: Since these vary cross-linguistically, need to know some words in the language to figure them out. But these cues are used to help identify words in the first place...
Word segmentation strategies

• Language-independent cue: probability of sequences of units like phonemes or syllables

• Potential: Early bootstrapping
  – Thiessen & Saffran 2003: statistical information used earlier than other cues
Bayesian inference:
A strategy that can use sequence probabilities

- The Bayesian learner seeks to identify an explanatory linguistic hypothesis that
  - accounts for the observed data
  - conforms to prior expectations

\[ P(h \mid d) \propto P(d \mid h) \, P(h) \]

posterior likelihood prior
Bayesian inference:
A strategy that can use sequence probabilities

• The Bayesian learner seeks to identify an explanatory linguistic hypothesis that
  – accounts for the observed data
  – conforms to prior expectations

\[ P(h \mid d) \propto P(d \mid h) \, P(h) \]

posterior  likelihood  prior

Ideal learner: Is this a useful strategy for word segmentation?
Bayesian inference:
A strategy that can use sequence probabilities

- The Bayesian learner seeks to identify an explanatory linguistic hypothesis that
  - accounts for the observed data
  - conforms to prior expectations

\[ P(h \mid d) \propto P(d \mid h) \, P(h) \]

posterior likelihood prior

Ideal learner: Is this a useful strategy for word segmentation?

Constrained learner: Is this a strategy useable by children? Is there any evidence it’s better when the learner is constrained?
Bayesian segmentation
(Goldwater et al. 2009)

Data: unsegmented corpus (transcriptions)
Hypotheses: sequences of word tokens

$$P(h \mid d) \propto P(d \mid h) P(h)$$

posterior likelihood prior

whatsthat thekitty yeah wheresthekitty

whats that the kitty yeah wheres the kitty
Bayesian segmentation  
(Goldwater et al. 2009)

Data: unsegmented corpus (transcriptions)  
Hypotheses: sequences of word tokens

\[ P(h \mid d) \propto P(d \mid h) \ P(h) \]

posterior  
likelihood  
prior

Implicit task: Identify the list of lexicon items that make up the sequences of word tokens, which make up the observed fluent speech data.

Lexicon: what, that, the, kitty, yeah, where

what that the kitty yeah where
what the kitty yeah where
Lexicon: what, that, the, kitty, yeah, where
Bayesian segmentation
(Goldwater et al. 2009)

Data: unsegmented corpus (transcriptions)
Hypotheses: sequences of word tokens

\[ P(h \mid d) \propto P(d \mid h) P(h) \]

posterior  likelihood  prior

= 1 if concatenating words forms corpus
= 0 otherwise.

Corpus: “lookatthekitty”

- \( P(d \mid h) = 1 \)
  - loo k at th eki tty
  - look at the kitty

- \( P(d \mid h) = 0 \)
  - i like penguins
  - look at the doggie
  - a b c
Bayesian segmentation
(Goldwater et al. 2009)

Data: unsegmented corpus (transcriptions)
Hypotheses: sequences of word tokens

\[ P(h \mid d) \propto P(d \mid h) \, P(h) \]

- posterior
- likelihood
- prior

= 1 if concatenating words forms corpus
= 0 otherwise.

Encodes learning assumptions or biases in the learner:
- prefer short words
- prefer fewer words
Bayesian segmentation  
(Goldwater et al. 2009)

Data: unsegmented corpus (transcriptions)  
Hypotheses: sequences of word tokens

Optimal solution is the segmentation with highest posterior probability.

\[ P(h \mid d) \propto P(d \mid h) \ P(h) \]

posterior    likelihood    prior

= 1 if concatenating words forms corpus
= 0 otherwise.

Encodes learning assumptions or biases in the learner:

• prefer short words
• prefer fewer words
Bayesian segmentation: Ideal vs. Constrained

Learner assumptions:

- Basic unit of representation = phoneme
- Very naïve language model:
  - Words are independent units (unigram assumption)
  - or
  - Words are units that predict other words (bigram assumption)

*Pearl, Goldwater, & Steyvers 2011, 2010*
Bayesian learners

Bayesian learners examined:

Ideal

Constrained

Pearl, Goldwater, & Steyvers 2011, 2010
Bayesian learners

Ideal learner (Batch Optimal: BatchOpt)

- Process data in a batch (perfect memory)
- Have enough processing resources to exhaustively search potential segmentations
- Select optimal segmentation

Pearl, Goldwater, & Steyvers 2011, 2010
Bayesian learners

Constrained learner (Online Optimal: OnlineOpt)

- Process data incrementally
- Have enough processing resources to exhaustively search potential segmentations
- Select optimal segmentation

*Pearl, Goldwater, & Steyvers 2011, 2010*
Bayesian learners

Constrained learner (Online Sub-optimal: OnlineSubOpt)
- Process data incrementally
- Have enough processing resources to exhaustively search potential segmentations
- Select segmentation probabilistically

Pearl, Goldwater, & Steyvers 2011, 2010
Bayesian learners

Constrained learner (Online Limited Working Memory: OnlineMem)

- Process data incrementally
- Limited working memory buffer, so cannot do exhaustive search: Focus instead on more recent data (recency bias)
- Select optimal segmentation

Pearl, Goldwater, & Steyvers 2011, 2010
Learner input

Pearl-Brent derived American English corpus, sub-section of speech directed at children 9 months or younger

- 28,391 utterances, 96,723 words
- 3.4 words per utterance, 4.2 syllables per utterance

hear the kitty Morgie
Sammy wants out
okay the kitty is out
what's Morgie gonna do
what's Morgie gonna
oh no no
no eating dog food
what was that
was a grunt
okay

Pearl, Goldwater, & Steyvers 2011, 2010
Bayesian segmentation: Ideal vs. Constrained

There’s a “less is more” effect for some constrained (OnlineMem) learners who have a unigram assumption.

Correct word token identification: 54% ideal vs. 64% constrained

Correct segmentation: “look at the doggie. look at the kitty.”
Best guess of learner: “lookat the doggie. lookat thekitty.”

Word Token Precision (P) = 2/5 (0.4), Word Token Recall (R) = 2/8 (0.25)
Word Token F-score = 2 * (P*R)/(P+R) = 0.31

Pearl, Goldwater, & Steyvers 2011, 2010
Bayesian segmentation: Ideal vs. Constrained

Why?
Their cognitive limitations caused them not to notice frequently occurring predictable sequences of short words. So, they didn’t try to make them one word, which is an undersegmentation error that the ideal learners often made.
Bayesian segmentation: Cognitive plausibility

What happens if we make the learning process we’re modeling look even more like the learning process children are using?

To do this, maybe we should revisit some of our modeling assumptions:

Basic unit of representation = phoneme?
Perceptual units for infants

Word segmentation timeline:
Statistical learning at the beginning of segmentation, before 7.5 months

What representations do infants have at this point?
• Phonemes around ~10 months (Werker & Tees 1984)
• Syllables around 3 months (Eimas 1999, Jusczyk & Derrah 1987)
Bayesian segmentation: Ideal vs. Constrained

Updated learner assumptions:

- Basic unit of representation = syllable
- Very naïve language model:
  Words are independent units (unigram assumption)
  or
  Words are units that predict other words (bigram assumption)

Phillips & Pearl 2012, in prep
Bayesian learning over syllables

<table>
<thead>
<tr>
<th>Method</th>
<th>Unigram</th>
<th>Bigram</th>
</tr>
</thead>
<tbody>
<tr>
<td>BatchOpt</td>
<td>53.1</td>
<td>77.1</td>
</tr>
<tr>
<td>OnlineOpt</td>
<td>58.8</td>
<td>75.1</td>
</tr>
<tr>
<td>OnlineSubOpt</td>
<td>63.7</td>
<td>77.8</td>
</tr>
<tr>
<td>OnlineMem</td>
<td>55.1</td>
<td>86.3</td>
</tr>
</tbody>
</table>

F = 2 * Prec * Rec

<table>
<thead>
<tr>
<th></th>
<th>Prec + Rec</th>
<th>Precision:</th>
<th>Recall:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>#correct / #found</td>
<td>#found / #true</td>
</tr>
</tbody>
</table>

Results averaged over 5 randomly generated test sets (~2800 utterances) that were separate from the training sets (~25200 utterances), all generated from the Pearl-Brent derived corpus.

Phillips & Pearl 2012, in prep
Bayesian learning over syllables

<table>
<thead>
<tr>
<th></th>
<th>Unigram</th>
<th>Bigram</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BatchOpt</strong></td>
<td>53.1</td>
<td>77.1</td>
</tr>
<tr>
<td><strong>OnlineOpt</strong></td>
<td>58.8</td>
<td>75.1</td>
</tr>
<tr>
<td><strong>OnlineSubOpt</strong></td>
<td>63.7</td>
<td>77.8</td>
</tr>
<tr>
<td><strong>OnlineMem</strong></td>
<td>55.1</td>
<td>86.3</td>
</tr>
</tbody>
</table>

F = $2 \times \text{Prec} \times \text{Rec} \over \text{Prec} + \text{Rec}$

**Flea**

A learner who assumes words are not predictive of other words performs significantly better when its abilities are constrained.

*More robust “less is more” effect than the phoneme-based unigram learner: All three constrained learners do better.*

Phillips & Pearl 2012, in prep
Bayesian learning over syllables

Word token F-scores

<table>
<thead>
<tr>
<th></th>
<th>Unigram</th>
<th>Bigram</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BatchOpt</strong></td>
<td>53.1</td>
<td>77.1</td>
</tr>
<tr>
<td><strong>OnlineOpt</strong></td>
<td>58.8</td>
<td>75.1</td>
</tr>
<tr>
<td><strong>OnlineSubOpt</strong></td>
<td>63.7</td>
<td>77.8</td>
</tr>
<tr>
<td><strong>OnlineMem</strong></td>
<td>55.1</td>
<td>86.3</td>
</tr>
</tbody>
</table>

F = 2 * Prec * Rec

Prec + Rec

Precision:
#correct / #found

Recall:
#found / #true

One of the more constrained learners who assumes words are predictive of other words performs significantly better than the ideal learner.

*New “less is more” effect: Phoneme-based bigram learners didn’t show this.*

*Phillips & Pearl 2012, in prep*
The utility of cognitively plausible modeling assumptions

In learners with either the unigram or the bigram assumption, we find what looks like a “less is more” effect.

By trying to make the model represent the input the way we think children do, we have reproduced behavior that we think children have.

View input as streams of syllables

Perform better with limited abilities

Phillips & Pearl 2012, in prep
What’s causing “less is more”?

Unigram learners benefit in a similar way to the phoneme-based learners in Pearl et al. 2011, 2010:

Constrained learners don’t create the undersegmentation errors that ideal learners do for frequently occurring sequences of short words. (They don’t notice them as much.)

“at the”  →  “atthe”
What’s causing “less is more”?

Bigram learners wouldn’t make this error though, because they have a way to represent predictable sequences. But the constrained OnlineMem bigram learner is significantly outperforming the ideal BatchOpt bigram learner (86.3 to 77.1)...

“at the” “atthe”
What’s causing “less is more”?

If we look at the recall scores for these bigram learners, we notice that token recall is higher for the constrained learner while lexicon recall (word types) is higher for the ideal learner.

(Lexicon scores factor out frequency of word tokens.)

<table>
<thead>
<tr>
<th></th>
<th>Token recall</th>
<th>Lexicon recall</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Ideal Bigram</strong></td>
<td>72.5</td>
<td><strong>79.7</strong></td>
</tr>
<tr>
<td><strong>OnlineMem Bigram</strong></td>
<td><strong>85.4</strong></td>
<td>76.8</td>
</tr>
</tbody>
</table>

Correct segmentation: “look at the doggie. look at the kitty.”
Best guess of learner: “lookat the doggie. lookat thekitty.”
Word Token Precision = 2/5 (0.4), Word Token Recall = 2/8 (0.25)
Lexicon Precision = 2/4 (0.5), Lexicon Recall = 2/5 (0.4)

*Phillips & Pearl 2012, in prep*
What’s causing “less is more”?

One idea: The constrained learner is correctly segmenting more frequent words (with more tokens per word) while the ideal learner is correctly segmenting more word types (words in the lexicon).

<table>
<thead>
<tr>
<th></th>
<th>Token recall</th>
<th>Lexicon recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ideal Bigram</td>
<td>72.5</td>
<td>79.7</td>
</tr>
<tr>
<td>OnlineMem Bigram</td>
<td>85.4</td>
<td>76.8</td>
</tr>
</tbody>
</table>

Phillips & Pearl 2012, in prep
What’s causing “less is more”?

It turns out that the constrained learner does identify words that are on average more frequent than the ideal learner’s words.

<table>
<thead>
<tr>
<th>Avg Log Frequency of Words Identified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ideal Bigram</td>
</tr>
<tr>
<td>-5.99</td>
</tr>
<tr>
<td>OnlineMem Bigram</td>
</tr>
<tr>
<td>-5.74</td>
</tr>
</tbody>
</table>

Note: Smaller negative number indicates more frequent
\((-5.99 = \text{probability } 10^{-5.99}, -5.74 = \text{probability } 10^{-5.74})\)

Possible interpretation: Constrained learner does well on more “important” words that occur more often.

*Phillips & Pearl 2012, in prep*
Understanding the learning process

Case study: Bayesian inference as an initial strategy for word segmentation

✓ Is it useful?

Ideal learners using this strategy perform fairly well, given realistic child-directed speech data.

Phillips & Pearl 2012, in prep
Understanding the learning process

Case study: Bayesian inference as an initial strategy for word segmentation

✓ Is it useful?

✓ Is it useable?

Constrained learners can still use this strategy and do quite well.

Phillips & Pearl 2012, in prep
Understanding the learning process

Case study: Bayesian inference as an initial strategy for word segmentation

✓ Is it useful?
✓ Is it useable?
✓ Does it work better when cognitive resources are constrained?

By representing the input in a way infants are likely to do, we find a stronger “less is more” effect, with constrained learners outperforming ideal learners.

Phillips & Pearl 2012, in prep
Now what?

Cross-linguistic investigation:
Does this learning strategy have these properties for languages besides English (especially languages with different morphology and syllable properties)?

Underway: Phillips & Pearl, in prep b
→ Spanish, Italian, German, Hungarian, Japanese, Farsi
Now what?

We know that infants are sensitive to additional information in the input. These cues can be incorporated into the learning process. Do we then find that Bayesian inference still performs well? Do other strategies?

- Ex: Input representation. Infants represent stressed and unstressed syllables separately (Pelucchi, Hay, & Saffran 2009)

  tea = /tí/ + 1
  pre tty = /ti/ + 1
Now what?

There are more ways to implement cognitive limitations. Do we find a stronger “less is more” effect when we implement other kinds?

– Ex: What if memory limitations also cause the lexicon items the learner is hypothesizing (and their respective counts) to decay?

*tea* = 15 times...or 18...or 12...

*pretty* = 100 times...or 120...or 80...
Now what?

Target state issue:
Even the ideal learners don’t achieve perfect (adult-like) word segmentation. How do we know if the lexicon any of the learners produce is “good enough”?

Sequential task check: Even if the results aren’t perfectly adult-like, is the lexicon obtained still useful for tasks that rely on that lexicon?

Ex: Identifying language-dependent cues to word segmentation

Ex: word-meaning mapping

Ex: grammatical categorization
Now what?

We know that infants are solving multiple language learning problems simultaneously. Do we find that Bayesian inference is useable and better with cognitive limitations when multiple learning tasks are involved?

Ex: word segmentation & phoneme identification

(We have some indication it could be useful: Feldman et al. 2009)
Now what?

Identifying learning strategies that are not only **useful**, but **useable** and **better with cognitive limitations** for the many different tasks of language acquisition.

How to do this: Translate computational-level (“rational”) learning strategies to algorithmic-level (“process”) learning strategies – can also show us which demonstrate a “less is more” effect.
Today’s Plan

Using **computational methods** to look at two questions about children’s mental computation

- What learning strategies comprise it?
  Looking for strategies that are useful, useable, and work better with limited cognitive resources

- What learning biases do children need to succeed at it?
  Understanding the nature of children’s language learning toolkit

Case study: Syntactic Islands
Children’s language learning toolkit: Some relevant dimensions

What kinds of learning biases could there be?

Pearl & Sprouse 2013, Pearl & Mis in rev.
Children’s language learning toolkit: Some relevant dimensions

What kinds of learning biases could there be?

– **innate** vs. **derived** from prior (language) experience

Pearl & Sprouse 2013, Pearl & Mis in rev.
Children’s language learning toolkit: Some relevant dimensions

What kinds of learning biases could there be?

- **innate** vs. **derived** from prior (language) experience
- **domain-specific** vs. **domain-general**

*Pearl & Sprouse 2013, Pearl & Mis in rev.*
Children’s language learning toolkit: Some relevant dimensions

What kinds of learning biases could there be?

− **innate** vs. **derived** from prior (language) experience
− **domain-specific** vs. **domain-general**
− **hypothesis space** vs. **learning mechanism**

*Pearl & Sprouse 2013, Pearl & Mis in rev.*
Children’s language learning toolkit: Universal Grammar connections

Universal Grammar is a particular kind of learning bias: **innate** & **domain-specific**.
(It doesn’t specify **hypothesis space** vs. **learning mechanism**.)

Pearl & Sprouse 2013, Pearl & Mis in rev.
Children’s language learning toolkit: Universal Grammar connections

Ideas for the biases in Universal Grammar often come from examining specific language learning problems, and figuring out what learning biases would be needed to solve those problems.

Pearl & Sprouse 2013, Pearl & Mis in rev.
Children’s language learning toolkit: Identifying the necessary biases

Note: This methodology can be used to simply identify the necessary biases, whatever kind they might be.

Pearl & Sprouse 2013, Pearl & Mis in rev.
Specifying learning problems

Initial state:
Specifying learning problems

Initial state:
- initial knowledge state
  ex: grammatical categories exist and can be identified
  ex: phrase structure exists and can be identified

\[ N^0, N', NP, DP, ... \]

Pearl & Mis in rev.
Specifying learning problems

Initial state:
  - initial knowledge state
    ex: grammatical categories exist and can be identified
    ex: phrase structure exists and can be identified
  - learning biases & capabilities
    ex: frequency information can be tracked
    ex: distributional information can be leveraged

\[ N^0, N', NP, DP, \ldots \]

\[ N^0 = N^0 + 1 \]

\[ p = 0.12 \]

\[ 0.12 \times 0.5 = 0.06 \]
Specifying learning problems

**Initial state:** initial knowledge state + learning biases & capabilities

**Data intake:**
Specifying learning problems

Initial state: initial knowledge state + learning biases & capabilities

Data intake:
- data perceived as relevant for learning (Fodor 1998)
  ex: all wh-utterances for learning about wh-dependencies
  ex: syntactic data for learning syntactic knowledge
[can be defined by knowledge & biases/capabilities in the initial state]
Specifying learning problems

Initial state: initial knowledge state + learning biases & capabilities

Data intake: data perceived as relevant for learning

Learning period:
Specifying learning problems

**Initial state:** initial knowledge state + learning biases & capabilities

**Data intake:** data perceived as relevant for learning

**Learning period:**
- how long children have to reach the target knowledge state
  
  Ex: 3 years, ~1,000,000 data points
Specifying learning problems

Initial state: initial knowledge state + learning biases & capabilities

Data intake: data perceived as relevant for learning

Learning period: how long children have to learn

Target state:
Specifying learning problems

**Initial state:** initial knowledge state + learning biases & capabilities

**Data intake:** data perceived as relevant for learning

**Learning period:** how long children have to learn

**Target state:**
- the knowledge children are trying to attain
  
  Ex: *Where did Jack think the necklace from ___ was too expensive?*

*Pearl & Mis in rev.*
Specifying learning problems

**Initial state:** initial knowledge state + learning biases & capabilities

**Data intake:** data perceived as relevant for learning

**Learning period:** how long children have to learn

**Target state:** the knowledge children must attain

*Pearl & Mis in rev.*
Specifying learning problems

**Initial state:** initial knowledge state + learning biases & capabilities

**Data intake:** data perceived as relevant for learning

**Learning period:** how long children have to learn

**Target state:** the knowledge children must attain

**Hard learning problem** (induction problem): Given a specific initial state, data intake, and learning period, the target state is *not* the only knowledge state that could be reached.
Case study: Syntactic islands

Why?

Syntactic islands are a type of linguistic knowledge that has been used to argue that innate, domain-specific (Universal Grammar) learning biases are necessary.

Pearl & Sprouse 2013
Syntactic islands

Dependencies can exist between two non-adjacent items. They do not appear to be constrained by length (Chomsky 1965, Ross 1967), but rather by whether the dependency crosses certain structures (called “syntactic islands”).
Syntactic islands

Dependencies can exist between two non-adjacent items. They do not appear to be constrained by length (Chomsky 1965, Ross 1967), but rather by whether the dependency crosses certain structures (called “syntactic islands”).

What does Jack think __?

What does Jack think that Lily said that Sarah heard that Jareth believed __?
Syntactic islands

Dependencies can exist between two non-adjacent items. They do not appear to be constrained by length (Chomsky 1965, Ross 1967), but rather by whether the dependency crosses certain structures (called “syntactic islands”).

Some example islands

Complex NP island:
*What did you make [the claim that Jack bought __]?

Subject island:
*What do you think [the joke about __] offended Jack?

Whether island:
*What do you wonder [whether Jack bought __]?

Adjunct island:
*What do you worry [if Jack buys __]?

Pearl & Sprouse 2013
Syntactic islands

Predominant theory in generative syntax:
Syntactic islands require innate, domain-specific learning biases about the hypothesis space

(1) A dependency cannot cross two or more bounding nodes.

[Diagram showing subjacency]

Pearl & Sprouse 2013
Syntactic islands

Predominant theory in generative syntax:
Syntactic islands require innate, domain-specific learning biases about the hypothesis space

(1) A dependency cannot cross two or more bounding nodes.

(2) Bounding nodes: language-specific
(CP, IP, and/or NP – must learn which ones are relevant for language)

\[ \text{Wh} \quad \ldots \quad [\text{BN2}] \quad \ldots \quad [\text{BN1} \ldots \ldots ] \]\n
\{CP, IP, NP\}?  

Pearl & Sprouse 2013
Syntactic islands

Predominant theory in generative syntax:
Syntactic islands require **innate**, **domain-specific** learning biases about the hypothesis space... in addition to whatever else they might require

\[ BN = \{CP, IP, NP\} \]

Not 2+ bounding nodes (BNs)

*Pearl & Sprouse* 2013
Syntactic islands

How do we investigate this?

(1) Explicitly define the target knowledge state, using adult acceptability judgments.

(2) Identify the data available in the input, using realistic samples. (Is there an induction problem, given what we think children’s data intake is?)

(3) Implement a probabilistic learner that can learn about syntactic islands and see what kind of learning biases it requires. This requires making the initial state and learning period explicit.
The target state:
Adult knowledge of syntactic islands

Sprouse et al. (2012) collected magnitude estimation judgments for four different islands, using a factorial definition that controlled for two salient properties of island-crossing dependencies:
- length of dependency (matrix vs. embedded)
- presence of an island structure (non-island vs. island)
The target state: 
Adult knowledge of syntactic islands

Sprouse et al. (2012) collected magnitude estimation judgments for four different islands, using a factorial definition that controlled for two salient properties of island-crossing dependencies:
- length of dependency (matrix vs. embedded)
- presence of an island structure (non-island vs. island)

Complex NP islands

Who __ claimed that Lily forgot the necklace?  
What did the teacher claim that Lily forgot __?  
Who __ made the claim that Lily forgot the necklace?  
*What did the teacher make the claim that Lily forgot __?  

Pearl & Sprouse 2013
The target state:  
Adult knowledge of syntactic islands

Sprouse et al. (2012) collected magnitude estimation judgments for four different islands, using a factorial definition that controlled for two salient properties of island-crossing dependencies:

- length of dependency (matrix vs. embedded)
- presence of an island structure (non-island vs. island)

Subject islands

Who ___ thinks the necklace is expensive?
What does Jack think ___ is expensive?
Who ___ thinks the necklace for Lily is expensive?
*Who does Jack think the necklace for ___ is expensive? 

matrix | non-island  
embedded | non-island  
matrix | island  
embedded | island

Pearl & Sprouse 2013
Sprouse et al. (2012) collected magnitude estimation judgments for four different islands, using a factorial definition that controlled for two salient properties of island-crossing dependencies:
- **length** of dependency (matrix vs. embedded)
- presence of an island structure (non-island vs. island)

Whether islands

Who ___ thinks that Jack stole the necklace?

What does the teacher think that Jack stole ___ ?

Who ___ wonders whether Jack stole the necklace?

*What does the teacher wonder whether Jack stole ___ ?

**Pearl & Sprouse 2013**
The target state:
Adult knowledge of syntactic islands

Sprouse et al. (2012) collected magnitude estimation judgments for four different islands, using a factorial definition that controlled for two salient properties of island-crossing dependencies:
- **length** of dependency (matrix vs. embedded)
- presence of an **island** structure (non-island vs. island)

Adjunct islands

Who ___ thinks that Lily forgot the necklace?  
What does the teacher think that Lily forgot ___?  
Who ___ worries if Lily forgot the necklace?  
*What does the teacher worry if Lily forgot ___?  

*Pearl & Sprouse 2013
The target state:
Adult knowledge of syntactic islands

Syntactic island = superadditive interaction of the two factors (additional unacceptability that arises when the two factors are combined, above and beyond the independent contribution of each factor).

Pearl & Sprouse 2013
The target state:
Adult knowledge of syntactic islands

Sprouse et al. (2012)’s data on the four island types (173 subjects)

Superadditivity present for all islands tested

Knowledge that dependencies cannot cross these island structures is part of the adult knowledge state
Specifying the learning problem: Syntactic islands

target state: knowledge of grammatical and ungrammatical dependencies, as indicated by Sprouse et al. (2012) judgment data
The data in the input

Data from five corpora of child-directed speech (Brown-Adam, Brown-Eve, Brown-Sarah, Suppes, Valian) from CHILDES (MacWhinney 2000): speech to 25 children between the ages of one and five years old.

Total words: 813,036
Utterances containing a wh-dependency: 31,247

Sprouse et al. (2012) stimuli types:

<table>
<thead>
<tr>
<th></th>
<th>MATRIX + NON-ISLAND</th>
<th>EMBEDDED + NON-ISLAND</th>
<th>MATRIX + ISLAND</th>
<th>EMBEDDED + ISLAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complex NP</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Subject</td>
<td>7</td>
<td>29</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Whether</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Adjunct</td>
<td>7</td>
<td>295</td>
<td>15</td>
<td>0</td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
The data in the input

wh-dependency rarity
These kinds of wh-dependencies are fairly rare in general - the most frequent appears about 0.9% of the time (295 of 31,247).

Sprouse et al. (2012) stimuli types (out of 31,247):

<table>
<thead>
<tr>
<th></th>
<th>MATRIX + NON-ISLAND</th>
<th>EMBEDDED + NON-ISLAND</th>
<th>MATRIX + ISLAND</th>
<th>EMBEDDED + ISLAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complex NP</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Subject</td>
<td>7</td>
<td>29</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Whether</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Adjunct</td>
<td>7</td>
<td>295</td>
<td>15</td>
<td>0</td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
The data in the input

Being grammatical doesn’t necessarily mean a *wh*-dependency will appear in the input at all.

Sprouse et al. (2012) stimuli types (out of 31,247): 

<table>
<thead>
<tr>
<th></th>
<th>MATRIX + NON-ISLAND</th>
<th>EMBEDDED + NON-ISLAND</th>
<th>MATRIX + ISLAND</th>
<th>EMBEDDED + ISLAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complex NP</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Subject</td>
<td>7</td>
<td>29</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Whether</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Adjunct</td>
<td>7</td>
<td>295</td>
<td>15</td>
<td>0</td>
</tr>
</tbody>
</table>
The data in the input

Unless the child is sensitive to very small frequencies, it’s difficult to tell the difference between grammatical and ungrammatical dependencies sometimes...

Sprouse et al. (2012) stimuli types (out of 31,247):

<table>
<thead>
<tr>
<th></th>
<th>MATRIX + NON-ISLAND</th>
<th>EMBEDDED + NON-ISLAND</th>
<th>MATRIX + ISLAND</th>
<th>EMBEDDED + ISLAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complex NP</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Subject</td>
<td>7</td>
<td>29</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Whether</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Adjunct</td>
<td>7</td>
<td>295</td>
<td>15</td>
<td>0</td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
The data in the input

...and impossible to tell no matter what the rest of the time.

Sprouse et al. (2012) stimuli types (out of 31,247):  

<table>
<thead>
<tr>
<th></th>
<th>MATRIX + NON-ISLAND</th>
<th>EMBEDDED + NON-ISLAND</th>
<th>MATRIX + ISLAND</th>
<th>EMBEDDED + ISLAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complex NP</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Subject</td>
<td>7</td>
<td>29</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Whether</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Adjunct</td>
<td>7</td>
<td>295</td>
<td>15</td>
<td>0</td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
The data in the input

If children are relying only on direct evidence and keying grammaticality directly to frequency, this looks like a hard learning problem.

Sprouse et al. (2012) stimuli types (out of 31,247):

<table>
<thead>
<tr>
<th></th>
<th>MATRIX + NON-ISLAND</th>
<th>EMBEDDED + NON-ISLAND</th>
<th>MATRIX + ISLAND</th>
<th>EMBEDDED + ISLAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complex NP</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Subject</td>
<td>7</td>
<td>29</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Whether</td>
<td>7</td>
<td>295</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Adjunct</td>
<td>7</td>
<td>295</td>
<td>15</td>
<td>0</td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
Specifying the learning problem:
Syntactic islands

initial state:
  Bias: Learn only from direct evidence.

data intake: examples of specific $wh$-dependencies in the input

target state: knowledge of grammatical and ungrammatical dependencies, as indicated by Sprouse et al. (2012) judgment data
Building a computational learner

Idea: Use indirect positive evidence, too.

Similar in spirit to linguistic parameters: Data are deemed informative, even if they are not data about the specific phenomenon of interest.

Here: Dependencies other than the ones of interest (the Sprouse et al. 2012 stimuli) are useful to learn from.

Pearl & Sprouse 2013
Specifying the learning problem:
Syntactic islands

initial state:
- Bias: Learn only from direct evidence.
+ Bias: Learn from both direct and indirect evidence coming from wh-dependencies.

data intake: all wh-dependencies in the input

target state: knowledge of grammatical and ungrammatical dependencies, as indicated by Sprouse et al. (2012) judgment data
Building a computational learner

Learning Bias: Children track the occurrence of structures that can be derived from phrase structure trees during parsing - container nodes.

\[
[\text{CP} \text{ Who} \text{ did } [\text{IP} \text{ she } [\text{VP} \text{ like } __]]]?
\]

Container node sequence: IP-VP

\[
[\text{CP} \text{ Who} \text{ did } [\text{IP} \text{ she } [\text{VP} \text{ think } [\text{CP} [\text{IP} [\text{NP} \text{ the gift}] \text{ VP} \text{ was } [\text{PP} \text{ from } __]]]]]]]]?
\]

Container node sequence: IP-VP-CP-IP-VP-PP

Pearl & Sprouse 2013
Building a computational learner

Children’s hypotheses are about what container node sequences are grammatical for dependencies in the language.

Grammatical:
- IP-VP-NP
- IP-VP-CP-IP-VP
- IP-VP-PP
- IP-VP-CP-IP-NP-PP

Ungrammatical:
- IP-VP-NP-CP-IP-VP

Pearl & Sprouse 2013
Specifying the learning problem: Syntactic islands

initial state:
- Bias: Learn from both direct and indirect evidence coming from *wh*-dependencies.
- **Capability**: Be able to parse data in the input into phrase structure trees.
- **Bias**: Characterize dependencies as sequences of container nodes.

data intake: all *wh*-dependencies in the input

target state: knowledge of grammatical and ungrammatical dependencies, as indicated by Sprouse et al. (2012) judgment data
What does the target knowledge look like?

Sprouse et al. (2012) stimuli:

<table>
<thead>
<tr>
<th>Complex NP islands</th>
<th>Subject islands</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>IP-VP-CP-IP-VP</td>
<td>IP-VP-CP-IP</td>
</tr>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>*IP-VP-NP-CP-IP-VP</td>
<td>*IP-VP-CP-IP-NP-PP</td>
</tr>
</tbody>
</table>

All the ungrammatical dependencies are distinct from all the grammatical dependencies for these syntactic islands.

*Pearl & Sprouse 2013*
What does the target knowledge look like?

Sprouse et al. (2012) stimuli:

<table>
<thead>
<tr>
<th>Whether islands</th>
<th>Adjunct islands</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>IP-VP-CP-IP-VP</td>
<td>IP-VP-CP-IP-VP</td>
</tr>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>*IP-VP-CP-IP-VP</td>
<td>*IP-VP-CP-IP-VP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>matrix</th>
<th>embedded</th>
<th>matrix</th>
<th>island</th>
<th>matrix</th>
<th>embedded</th>
<th>island</th>
<th>matrix</th>
<th>embedded</th>
<th>island</th>
<th>matrix</th>
<th>embedded</th>
<th>island</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-island</td>
<td></td>
<td></td>
<td>non-island</td>
<td></td>
<td>island</td>
<td></td>
<td></td>
<td>island</td>
<td></td>
<td></td>
<td>island</td>
<td></td>
<td></td>
</tr>
<tr>
<td>non-island</td>
<td></td>
<td></td>
<td>non-island</td>
<td></td>
<td>island</td>
<td></td>
<td></td>
<td>island</td>
<td></td>
<td></td>
<td>island</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
What does the target knowledge look like?

Sprouse et al. (2012) stimuli:

<table>
<thead>
<tr>
<th>Whether islands</th>
<th>Adjunct islands</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>IP-VP-CP-IP-VP</td>
<td>IP-VP-CP-IP-VP</td>
</tr>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>*IP-VP-CP-IP-VP</td>
<td>*IP-VP-CP-IP-VP</td>
</tr>
</tbody>
</table>

Uh oh - the ungrammatical dependencies look identical to some of the grammatical dependencies for these syntactic islands.

*Pearl & Sprouse 2013*
Building a computational learner

Learning bias solution:
Have CP container nodes be more specified for the learner:
Use the lexical head to subcategorize the CP container node.

$CP_{\text{null}}, CP_{\text{that}}, CP_{\text{whether}}, CP_{\text{if}}, \text{etc.}$

The learner can then distinguish between these structures:

$\text{IP-VP-}CP_{\text{null/that}}-\text{IP-VP}$
$\text{IP-VP-}CP_{\text{whether/if}}-\text{IP-VP}$

Pearl & Sprouse 2013
What does the target knowledge look like?

Sprouse et al. (2012) stimuli:

<table>
<thead>
<tr>
<th>Complex NP islands</th>
<th>Subject islands</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>IP-VP-CP_{that} -IP-VP</td>
<td>IP-VP-CP_{null} -IP</td>
</tr>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>*IP-VP-NP-CP_{that} -IP-VP</td>
<td>*IP-VP-CP_{null} -IP-NP-PP</td>
</tr>
</tbody>
</table>

All the ungrammatical dependencies are still distinct from all the grammatical dependencies for these syntactic islands.

Pearl & Sprouse 2013
What does the target knowledge look like?

Sprouse et al. (2012) stimuli:

<table>
<thead>
<tr>
<th>Whether islands</th>
<th>Adjunct islands</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>IP-VP-CP_{that}-IP-VP</td>
<td>IP-VP-CP_{that}-IP-VP</td>
</tr>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
<tr>
<td>*IP-VP-CP_{whether}-IP-VP</td>
<td>*IP-VP-CP_{if}-IP-VP</td>
</tr>
</tbody>
</table>

Now the ungrammatical dependencies are distinct from all the grammatical dependencies for these syntactic islands, too.

Pearl & Sprouse 2013
Specifying the learning problem: Syntactic islands

initial state:
  Bias: Learn from both direct and indirect evidence coming from \textit{wh}-dependencies.
  Capability: Be able to parse data in the input into phrase structure trees.
  Bias: Characterize dependencies as sequences of container nodes.
  $+$Bias: \textit{Subcategorize container nodes by CP lexical content}.

data intake: all \textit{wh}-dependencies in the input

target state: knowledge of grammatical and ungrammatical dependencies, as indicated by Sprouse et al. (2012) judgment data

\textit{Pearl \& Sprouse 2013}
Building a computational learner

Learning Bias: Implicitly assign a probability to a container node sequence by tracking **trigrams of container nodes**. A sequence’s probability is the smoothed product of its trigrams.

Pearl & Sprouse 2013
Building a computational learner

Learning Bias: Implicitly assign a probability to a container node sequence by tracking trigrams of container nodes. A sequence’s probability is the smoothed product of its trigrams.

\[
[C_P \text{ Who did } [I_P \text{ she } [V_P \text{ think } [C_P [I_P [N_P \text{ the gift}] [V_P \text{ was } [P_P \text{ from ___}]]]]]]?]
\]

\[
\text{start-IP-VP-}C_{null}\text{-IP-VP-PP-end } = \\
\text{start-IP-VP} \\
\text{IP-VP-}C_{null} \\
V_P-}C_{null}\text{-IP} \\
C_{null}\text{-IP-VP} \\
\text{IP-VP-PP} \\
V_P-PP\text{-end}
\]
Building a computational learner

Learning Bias: Implicitly assign a probability to a container node sequence by tracking trigrams of container nodes. A sequence’s probability is the smoothed product of its trigrams.

\[
\text{[CP Who did [IP she [VP think [CP [IP [NP the] gift] [vp was [PP from __]]]]]]?}
\]

\[
\begin{align*}
\text{IP} & \quad \text{VP} & \quad \text{CP\textsubscript{null}} & \quad \text{IP} & \quad \text{VP} & \quad \text{PP} \\
\text{start-IP-VP-CP\textsubscript{null}-IP-VP-PP-end} &= \\
\text{start-IP-VP} & \quad \text{IP-VP-CP\textsubscript{null}} & \quad \text{VP-CP\textsubscript{null}-IP} & \quad \text{CP\textsubscript{null}-IP-VP} & \quad \text{IP-VP-PP} & \quad \text{VP-PP-end} \\
\text{Probability(IP-VP-CP\textsubscript{null}-IP-VP-PP)} &= p(\text{start-IP-VP-CP\textsubscript{null}-IP-VP-PP-end}) \\
&= p(\text{start-IP-VP}) \times p(\text{IP-VP-CP\textsubscript{null}}) \times p(\text{VP-CP\textsubscript{null}-IP}) \times p(\text{CP\textsubscript{null}-IP-VP}) \\
&\quad \times p(\text{IP-VP-PP}) \times p(\text{VP-PP-end})
\end{align*}
\]

Pearl & Sprouse 2013
Building a computational learner

Learning Bias: Implicitly assign a probability to a container node sequence by tracking trigrams of container nodes. A sequence’s probability is the smoothed product of its trigrams.

What this does:

• longer dependencies are less probable than shorter dependencies, all other things being equal

• individual trigram frequency matters: short dependencies made of infrequent trigrams will be less probable than longer dependencies made of frequent trigrams

Effect: the frequencies observed in the input can temper the detrimental effect of dependency length.
Specifying the learning problem: Syntactic islands

initial state:
Bias: Learn from both direct and indirect evidence coming from $wh$-dependencies.
Capability: Be able to parse data in the input into phrase structure trees.
Bias: Characterize dependencies as sequences of container nodes.
Bias: Subcategorize container nodes by CP lexical content.
+Bias: Track trigrams of container nodes in the input.
+Capability: Generate probability of $wh$-dependency from trigrams of container nodes characterizing it.

data intake: all $wh$-dependencies in the input

target state: knowledge of grammatical and ungrammatical dependencies, as indicated by Sprouse et al. (2012) judgment data

Pearl & Sprouse 2013
Learning process

Hear utterance

Parse utterance, characterizing dependencies as container node sequences

Identify trigrams and update trigram frequencies

XP-YP-ZP...

start-XP-YP + 1

Repeat until learning period ends

What did...

Pearl & Sprouse 2013
Generating grammaticality preferences

Parse structure, characterizing dependency as container node sequence

Identify trigrams

Calculate probability of container node sequence from trigrams

Probability = \( p(start-XP-YP) \times p(XP-YP-ZP) \times \ldots \)

Pearl & Sprouse 2013
Building a computational learner: Empirical grounding

Child-directed speech (Brown-Adam, Brown-Eve, Suppes, Valian) from CHILDES:

What kind of dependencies are present?

<table>
<thead>
<tr>
<th>Dependency</th>
<th>Percentage</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP-VP</td>
<td>76.7%</td>
<td>What did you see __?</td>
</tr>
<tr>
<td>IP</td>
<td>12.8%</td>
<td>What __ happened?</td>
</tr>
<tr>
<td>IP-VP-IP-VP</td>
<td>5.6%</td>
<td>What did she want to do __?</td>
</tr>
<tr>
<td>IP-VP-PP</td>
<td>2.5%</td>
<td>What did she read from __?</td>
</tr>
<tr>
<td>IP-VP-CP_{null}-IP-VP</td>
<td>1.1%</td>
<td>What did she think he said __?</td>
</tr>
</tbody>
</table>

...
Specifying the learning problem: Syntactic islands

initial state:
  Bias: Learn from both direct and indirect evidence coming from \textit{wh}-dependencies.
  Capability: Be able to parse data in the input into phrase structure trees.
  Bias: Characterize dependencies as sequences of container nodes.
  Bias: Subcategorize container nodes by CP lexical content.
  Bias: Track trigrams of container nodes in the input.
  Capability: Generate probability of \textit{wh}-dependency from trigrams of container nodes characterizing it.

data intake: all \textit{wh}-dependencies in the input

target state: knowledge of grammatical and ungrammatical dependencies, as indicated by Sprouse et al. (2012) judgment data

Pearl & Sprouse 2013
Building a computational learner: Empirical grounding

Hart & Risley 1995: Children hear approximately one million utterances in their first three years.

Assumption: learning period for modeled learners is 3 years (ex: between 2 and 5 years old for modeling children’s acquisition), so they would hear one million utterances.

Total learning period: 200,000 \textit{wh}-dependency data points (\textit{wh}-dependencies make up approximately 20\% of the input)

Pearl & Sprouse 2013
Specifying the learning problem:
Syntactic islands

initial state:
  Bias: Learn from both direct and indirect evidence coming from \(wh\)-dependencies.
  Capability: Be able to parse data in the input into phrase structure trees.
  Bias: Characterize dependencies as sequences of container nodes.
  Bias: Subcategorize container nodes by CP lexical content.
  Bias: Track trigrams of container nodes in the input.
  Capability: Generate probability of \(wh\)-dependency from trigrams of container nodes characterizing it.

data intake: all \(wh\)-dependencies in the input

learning period: \(~3\) years = \(~200,000\) \(wh\)-dependency data points

target state: knowledge of grammatical and ungrammatical dependencies, as indicated by Sprouse et al. (2012) judgment data

Pearl & Sprouse 2013
Success metrics

Compare learned grammaticality preferences to Sprouse et al. (2012) judgment data.

Then, for each island, we plot the predicted grammaticality preferences from the modeled learner on an interaction plot, using log probability of the dependency on the y-axis. Non-parallel lines indicate knowledge of islands.

Pearl & Sprouse 2013
Learning results

Superadditivity observed for all four islands:

This learner has knowledge of these syntactic islands!

That means this learner can solve this learning problem.

Now...what did it need to do so?

Pearl & Sprouse 2013
The nature of children’s toolkit

Now that the biases have been identified, we can think about what kind of biases they are.

Learn from all *wh*-dependencies
Parse data into phrase structure trees
Attend to container nodes & subcategorize by CP
Extract & track container node trigrams
Calculate dependency probability from trigrams

*Pearl & Sprouse 2013*
The nature of children’s toolkit

Are they **innate** or **derived**? (It may not be so clear for some biases.)

<table>
<thead>
<tr>
<th></th>
<th>Innate</th>
<th>Derived</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learn from all <em>wh</em>-dependencies</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Parse data into phrase structure trees</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Attend to container nodes &amp; subcategorize by CP</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Extract &amp; track container node trigrams</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>Calculate dependency probability from trigrams</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

---

*Pearl & Sprouse 2013*
# The nature of children’s toolkit

Are they **domain-specific** or **domain-general**?

<table>
<thead>
<tr>
<th></th>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learn from all <em>wh</em>-dependencies</td>
<td>?</td>
<td>?</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Parse data into phrase structure trees</td>
<td>?</td>
<td>?</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Attend to container nodes &amp; subcategorize by CP</td>
<td>?</td>
<td>?</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Extract &amp; track container node trigrams</td>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Calculate dependency probability from trigrams</td>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

*Pearl & Sprouse 2013*
The nature of children’s toolkit

Are they about the **hypothesis space** or the **learning mechanism**?

<table>
<thead>
<tr>
<th></th>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
<th>Hypothesis space</th>
<th>Learning mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learn from all <em>wh</em>-dependencies</td>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Parse data into phrase structure trees</td>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Attend to container nodes &amp; subcategorize by CP</td>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Extract &amp; track container node trigrams</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>Calculate dependency probability from trigrams</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

*Pearl & Sprouse 2013*
The nature of children’s toolkit

The Universal Grammar question:
Are any necessarily both innate and domain-specific? Maybe.

<table>
<thead>
<tr>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
<th>Hypothesis space</th>
<th>Learning mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Learn from all <em>wh</em>-dependencies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parse data into phrase structure trees</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Attend to container nodes &amp; subcategorize by CP</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>Extract &amp; track container node trigrams</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>Calculate dependency probability from trigrams</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
Main implications of this learner for Universal Grammar

(1) Even though there is a hard learning problem for these syntactic islands, it may not require Universal Grammar learning biases to solve it.

<table>
<thead>
<tr>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Learn from all *wh*-dependencies
Parse data into phrase structure trees
Attend to container nodes & subcategorize by CP
Extract & track container node trigrams
Calculate dependency probability from trigrams

*Pearl & Sprouse 2013*
Main implications of this learner for Universal Grammar

(2) Even if Universal Grammar (UG) learning biases are required, they are different from (and less specific than) the biases previously proposed.

<table>
<thead>
<tr>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

Learn from all *wh*-dependencies
Parse data into phrase structure trees
Attend to container nodes & subcategorize by CP
Extract & track container node trigrams
Calculate dependency probability from trigrams

*Pearl & Sprouse 2013*
Main implications of this learner for Universal Grammar

Ex: Even though an abstract linguistic representation is required (container nodes), no “constraint” on the number of these nodes in a dependency is required. This falls out automatically from other non-UG learning biases.

Learn from all wh-dependencies
Parse data into phrase structure trees
Attend to container nodes & subcategorize by CP
Extract & track container node trigrams
Calculate dependency probability from trigrams

<table>
<thead>
<tr>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Now what?

Investigate the biases that may be either innate or derived.

Can we create a learner that can derive them from the available linguistic information?

If we can, what are the underlying biases that are required to do so, and what is the nature of those biases?
Now what?

This learning strategy for *wh*-dependencies makes some developmental predictions – can we verify these experimentally?

“*that*-trace” effect prediction:
Children initially disprefer all dependencies containing *that*, even ones adults allow
Now what?

This learning strategy for *wh*-dependencies makes some developmental predictions – can we verify these experimentally?

“*that*-trace” effect prediction:
Children initially disprefer all dependencies containing *that*, even ones adults allow

Subject extraction
*Who do you think *that* __ read the book?
Who do you think __ read the book?

Pearl & Sprouse 2013
Now what?

This learning strategy for *wh*-dependencies makes some developmental predictions – can we verify these experimentally?

“*that*-trace” effect prediction:
Children initially disprefer all dependencies containing *that*, even ones adults allow

**Subject extraction**
*Who* do you think *that* ___ read the book?
*Who* do you think ___ read the book?

**Object extraction**
*What* do you think *that* he read ___ ?
*What* do you think he read ___ ?

*Pearl & Sprouse 2013*
Now what?

How does this learning strategy for *wh*-dependencies measure up cross-linguistically?

**Island effects vary.**
Ex: Italian does not have a subject island effect when the *wh*-dependency is part of a relative clause, though it does when the *wh*-dependency is part of a question. (Sprouse et al. submitted)

Would the input naturally lead our kind of learner to this distinction?
Now what?

Can we extend this learning strategy to create an integrated theory of syntactic acquisition?

Related phenomena: The distribution of gaps

**Parasitic gaps**: Dependencies that span an island (and so should be ungrammatical) but which are somehow rescued by another dependency in the utterance.

*Which book did you laugh [before reading ___]?* 
Which book did you judge ___true [before reading ___parasitic]?
Now what?

Can we extend this learning strategy to create an integrated theory of syntactic acquisition?

Related phenomena: The distribution of gaps

Across-the-board (ATB) extraction: Similar situation.

Which book did you [[read ___ ] and [then review ___]]? dependency for both gaps: IP-VP-VP

*Which book did you [[read the paper] and [then review ___]]? dependency for gap: IP-VP-VP

*Which book did you [[read ___ ] and [then review the paper]]? dependency for gap: IP-VP-VP

Pearl & Sprouse 2013
Now what?

Can we extend this learning strategy to create an integrated theory of syntactic acquisition?

*Semi-related phenomena: Binding dependencies*

There don’t appear to be the same restrictions on binding dependencies that there are on *wh*-dependencies.

The boy thought the joke about himself was really funny.

*Who did the boy think [the joke about ___ ] was really funny? Subject island*
Now what?

Can we extend this learning strategy to create an integrated theory of syntactic acquisition?

Not-so-related phenomena: Distribution of NPs

There are restrictions on where NPs can appear, sometimes based on the lexical item/class of verb or the syntactic construction.

- It seems/*tries/*believes that Jack is clever.
- Jack *seems/*tries/*believes is clever.
- Jack seems/ tries/*believes to be clever.
- It *seems/*tries/*believes Jack to be clever.
- I *seem / *try / believe Jack is clever.
- I *seem / *try / believe Jack to be clever.

Jack climbed the beanstalk.
*It was climbed the beanstalk by Jack.
Take away points from today

Using computational methods to look at two questions about children’s ongoing mental computation during language learning
Take away points from today

Using computational methods to look at two questions about children’s ongoing mental computation during language learning

What learning strategies comprise it?
Looking for strategies that are useful, useable, and work better with limited cognitive resources

Informing us about the learning process, and how children learn language as effectively as they do.
Take away points from today

Using computational methods to look at two questions about children’s ongoing mental computation during language learning

What learning biases do children need to succeed at it?
Understanding the nature of children’s language learning toolkit

Impacts our understanding of the fundamental building blocks children use, and also helps define what is and is not part of Universal Grammar.

Case study: Syntactic Islands
Recap: Understanding children’s ongoing mental computation using computational methods

Computational methods are part of an arsenal of empirical investigation methods that we can use to help us understand language learning. This includes the learning strategies children use, the learning biases children have, the knowledge representations that are learnable, and the time course of language development.
Thank you!

Lawrence Phillips  Jon Sprouse

Diogo Almeida   Misha Becker   Bob Berwick   Alexander Clark
Bob Frank       Sharow Goldwater Norbert Hornstein Jeff Lidz
Colin Phillips  William Sakas  Mark Steyvers  Virginia Valian
Charles Yang

Audiences at:
CogSci 2012
Workshop on Input & Syntactic Acquisition 2009, 2012
NYU Linguistics Colloquium 2012

This work was supported in part by NSF grant BCS-0843896.
Extra material for word segmentation
Bayesian learners

Constrained learner (Online + Optimal decisions [OnlineOpt]):

For each utterance:

- Use dynamic programming to compute probabilities of all segmentations, given the current lexicon.
- Choose the best segmentation.
- Add counts of segmented words to lexicon.

```
0.33  dId yu wa/n6 sIt dQn
0.21  dId/yu wa/n6 sIt dQn
0.15  dId/yu wa n6 sIt dQn
...
```
Bayesian learners

Constrained learner (Online + Sub-optimal decisions [OnlineSubOpt]):

For each utterance:
- Use dynamic programming to compute probabilities of all segmentations, given the current lexicon.
- Sample a segmentation probabilistically.
- Add counts of segmented words to lexicon.

\[
\begin{align*}
\text{did you wanna sit down} \\
0.33 & \quad \text{dId yu wa/n6 sIt dQn} \\
0.21 & \quad \text{dId/yu wa/n6 sIt dQn} \\
0.15 & \quad \text{dId/yu wa n6 sIt dQn} \\
\ldots & \quad \ldots
\end{align*}
\]
Bayesian learners

**Constrained** learner (Online + Limited Working Memory [OnlineMem]) (using Decayed Markov Chain Monte Carlo):

For each utterance:

- Probabilistically sample $s$ boundaries from all utterances encountered so far.
- $\text{Prob(sample } b) \propto b_a^{-d}$ where $b_a$ is the number of potential boundary locations between $b$ and the end of the current utterance and $d$ is the decay rate (Marthi et al. 2002).
- Update *lexicon* after each boundary sample.

*did you wanna sit down*

Probability of sampling boundary

**Phillips & Pearl 2012, in prep**
Bayesian learners

Constrained learner (Online + Limited Working Memory [OnlineMem]) (using Decayed Markov Chain Monte Carlo):

For each utterance:

• Probabilistically sample s boundaries from all utterances encountered so far.
• \( \text{Prob(sample } b) \propto b_a^{-d} \) where \( b_a \) is the number of potential boundary locations between \( b \) and the end of the current utterance and \( d \) is the decay rate (Marthi et al. 2002).
• Update lexicon after each boundary sample.

Phillips & Pearl 2012, in prep
Bayesian learners

Constrained learner (Online + Limited Working Memory [OnlineMem])
(using Decayed Markov Chain Monte Carlo):

For all DMCMC learners:

\[ d = 1.5 \text{ (~77\% chance of sampling a boundary in the current utterance)} \]
\[ s = 20000 \text{ samples per utterance (78\% fewer samples than ideal learner)} \]

Utterance 1

\[ s \text{ samples} \]

Utterance 2

Did you wanna sit down? That’s okay then.

Boundaries

Phillips & Pearl 2012, in prep
Understanding the impact of cognitive limitations

One effect of the constrained learner’s cognitive limitations is to push the learner away from the very naïve underlying language models (the unigram or bigram assumption).

Bigram syllable-based learners

<table>
<thead>
<tr>
<th>Method</th>
<th>Log Posterior</th>
<th>Token F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>BatchOpt</td>
<td>-552732</td>
<td>77.1</td>
</tr>
<tr>
<td>OnlineOpt</td>
<td>-623216</td>
<td>75.1</td>
</tr>
<tr>
<td>OnlineSubOpt</td>
<td>-631540</td>
<td>77.8</td>
</tr>
<tr>
<td>OnlineMem</td>
<td>-577879</td>
<td>86.3</td>
</tr>
</tbody>
</table>

Log posterior: How close to the underlying naïve model

*Smaller negative numbers = closer (*10\(^{-557232}\) closer than \(*10\(^{-577879}\)*)*

*Phillips & Pearl in prep*
Understanding the impact of cognitive limitations

Observation: **BatchOpt** vs. **OnlineMem**
Being further away from the underlying naïve model = better word segmentation performance.

Bigram syllable-based learners

<table>
<thead>
<tr>
<th></th>
<th>Log Posterior</th>
<th>Token F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BatchOpt</strong></td>
<td>-552732</td>
<td>77.1</td>
</tr>
<tr>
<td><strong>OnlineOpt</strong></td>
<td>-623216</td>
<td>75.1</td>
</tr>
<tr>
<td><strong>OnlineSubOpt</strong></td>
<td>-631540</td>
<td>77.8</td>
</tr>
<tr>
<td><strong>OnlineMem</strong></td>
<td>-577879</td>
<td>86.3</td>
</tr>
</tbody>
</table>

Log posterior: How close to the underlying naïve model
*Smaller negative numbers = closer* ($10^{-557232}$ closer than $10^{-577879}$)

*Phillips & Pearl in prep*
Understanding the impact of cognitive limitations

Observation: OnlineOpt vs. OnlineSubOpt
Being further away from the underlying naïve model = better word segmentation performance.

Bigram syllable-based learners

<table>
<thead>
<tr>
<th></th>
<th>Log Posterior</th>
<th>Token F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BatchOpt</strong></td>
<td>-552732</td>
<td>77.1</td>
</tr>
<tr>
<td><strong>OnlineOpt</strong></td>
<td>-623216</td>
<td>75.1</td>
</tr>
<tr>
<td><strong>OnlineSubOpt</strong></td>
<td>-631540</td>
<td>77.8</td>
</tr>
<tr>
<td><strong>OnlineMem</strong></td>
<td>-577879</td>
<td>86.3</td>
</tr>
</tbody>
</table>

Log posterior: How close to the underlying naïve model
*Smaller negative numbers = closer (10^{-557232} closer than 10^{-577879})*
Understanding the impact of cognitive limitations

Interpretation:
Cognitive limitations seems to push the learner away from the underlying naïve language model, and also in the right direction.

<table>
<thead>
<tr>
<th>Bigram syllable-based learners</th>
<th>Log Posterior</th>
<th>Token F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>BatchOpt</td>
<td>-552732</td>
<td>77.1</td>
</tr>
<tr>
<td>OnlineOpt</td>
<td>-623216</td>
<td>75.1</td>
</tr>
<tr>
<td>OnlineSubOpt</td>
<td>-631540</td>
<td>77.8</td>
</tr>
<tr>
<td>OnlineMem</td>
<td>-577879</td>
<td>86.3</td>
</tr>
</tbody>
</table>

Log posterior: How close to the underlying naïve model
Smaller negative numbers = closer ($10^{-557232}$ closer than $10^{-577879}$)

Phillips & Pearl in prep
Understanding the impact of cognitive limitations

Caveat:
It’s not just about being pushed far away from the underlying naïve language model – it’s important to also be pushed in the right direction (OnlineSubOpt vs. OnlineMem).

Bigram syllable-based learners

<table>
<thead>
<tr>
<th></th>
<th>Log Posterior</th>
<th>Token F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BatchOpt</strong></td>
<td>-552732</td>
<td>77.1</td>
</tr>
<tr>
<td><strong>OnlineOpt</strong></td>
<td>-623216</td>
<td>75.1</td>
</tr>
<tr>
<td><strong>OnlineSubOpt</strong></td>
<td>-631540</td>
<td>77.8</td>
</tr>
<tr>
<td><strong>OnlineMem</strong></td>
<td>-577879</td>
<td>86.3</td>
</tr>
</tbody>
</table>

Log posterior: How close to the underlying naïve model
Smaller negative numbers = closer ($10^{-557232}$ closer than $10^{-577879}$)

Phillips & Pearl in prep
Extra material for syntactic islands
Learn from all *wh*-dependencies

<table>
<thead>
<tr>
<th></th>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>wh</em>-dependencies</td>
<td>?</td>
<td>?</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
Learn from all \textit{wh}-dependencies

<table>
<thead>
<tr>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

Clearly \textit{domain-specific}, since this is language data.

May seem reasonable to attend to \textit{wh}-dependency data when learning about \textit{wh}-dependencies (and so this would be \textit{derived})
Learn from all *wh*-dependencies

<table>
<thead>
<tr>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

Clearly *domain-specific*, since this is language data.

May seem reasonable to attend to *wh*-dependency data when learning about *wh*-dependencies (and so this would be *derived*)

...but then why not attend to *all* dependencies (ex: relative clause dependencies, binding dependencies) since *wh*-dependencies are a kind of dependency?

Empirical necessity of just using *wh*-dependency data:
There are different island effects for relative clauses (*Sprouse et al. submitted*) and no island effects for binding dependencies, so **the learner needs to know to pay attention just to *wh*-dependencies.**

*Pearl & Sprouse 2013*
Parse data into phrase structure trees

<table>
<thead>
<tr>
<th></th>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

Pearl & Sprouse 2013
Clearly domain-specific, since the structure is specific to language.

May be possible to bootstrap this information (acquiring syntactic categories: Mintz 2003, 2006; acquisition of hierarchical structure given syntactic categories as input: Klein & Manning 2002). If so, this would be derived...
Clearly *domain-specific*, since the structure is specific to language.

May be possible to bootstrap this information (acquiring syntactic categories: Mintz 2003, 2006; acquisition of hierarchical structure given syntactic categories as input: Klein & Manning 2002). If so, this would be *derived*...

...but it’s *currently unclear* if all the necessary phrase structure knowledge can be bootstrapped.

**Important:**
The need for this capability is not specific to learning islands – it’s (presumably) needed for learning any kind of syntactic knowledge.

*Pearl & Sprouse 2013*
<table>
<thead>
<tr>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

**Attend to container nodes & subcategorize by CP**

*Pearl & Sprouse 2013*
Identifying container nodes
- applies to language data: domain-specific
- derived from ability to parse utterances
Identifying container nodes
- applies to language data: domain-specific
- derived from ability to parse utterances

Attending to container nodes (among all the other data out there)
- applies to language data: domain-specific
- innate vs. derived?
  • could be specified innately (like bounding nodes)
  • could be derived from a bias to use representations that are already being used for parsing
Attend to container nodes & **subcategorize by CP**

<table>
<thead>
<tr>
<th></th>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

*Jaeger 2010*

*Pearl & Sprouse 2013*
Attend to container nodes & subcategorize by CP

<table>
<thead>
<tr>
<th></th>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

About a linguistic representation: **domain-specific**

**Innate** vs. **derived**?
- Could be specified **innately**
Attend to container nodes & **subcategorize by CP**

<table>
<thead>
<tr>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>?</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

About a linguistic representation: **domain-specific**

**Innate vs. derived?**

- Could be specified **innately**

- Could be **derived** from prior linguistic experience:
  
  - Uncontroversial to assume children learn to distinguish different types of CPs since the lexical content of CPs has substantial consequences for the semantics of a sentence.

  - Also, adult speakers are sensitive to the distribution of *that* versus null complementizers (Jaeger 2010).

  ...but still have to know this is the right thing to subcategorize.
Extract & track container node trigrams

Pearl & Sprouse 2013
Extract & track container node trigrams

Applied in different cognitive domains: **domain-general**


...though why trigrams instead of some other n-gram?
Why learning from container node trigrams works

For each island-spanning dependency, there is at least one extremely low probability container node trigram in the dependency.

Complex NP island
\[\text{start-IP-VP-NP-CP}_{\text{that}}-\text{IP-VP-end}\]

Subject island
\[\text{start-IP-VP-CP}_{\text{null}}-\text{IP-NP-PP-end}\]

Whether island
\[\text{start-IP-VP-CP}_{\text{whether}}-\text{IP-VP-end}\]

Adjunct island
\[\text{start-IP-VP-CP}_{\text{if}}-\text{IP-VP-end}\]

These trigrams are never observed in the input – which is crucially different than being observed rarely. Thus, these islands are worse than dependencies involving trigrams that are rarely seen (e.g., dependencies with CP_{that}) and even longer dependencies that involve more frequent trigrams (e.g., triply embedded object dependencies using CP_{null}).

Pearl & Sprouse 2013
The empirical necessity of trigrams

Not unigrams
A unigram model will successfully learn Whether and Adjunct islands, as there are container nodes in these dependencies that never appear in grammatical dependencies (CP_{whether} and CP_{if})....but it will fail to learn Complex NP and Subject islands, as all of the container nodes in these islands are shared with grammatical dependencies.

Complex NP: *IP-VP-NP-CP_{that}-IP-VP
Subject: *IP-VP-CP_{null}-IP-NP-PP
Whether: IP-VP-CP_{whether}-IP-VP
Adjunct: IP-VP-CP_{if}-IP-VP
The empirical necessity of trigrams

Not bigrams
At least for Subject islands, there is no bigram that occurs in a Subject island violation but not in any grammatical dependencies. The most likely candidate for such a bigram is IP-NP...However, sentences such as What, again, about Jack impresses you? or What did you say about the movie scared you? suggest that a gap can arise inside of NPs, as long as the extraction is of the head noun (what), not of the noun complement of the preposition.

Complex NP: \(\text{IP-VP-NP-CP}_{\text{that}}\text{-IP-VP}\)
Subject: \(\*\text{IP-VP-CP}_{\text{null}}\text{-IP-NP-PP}\)
Whether: \(\text{IP-VP-CP}_{\text{whether}}\text{-IP-VP}\)
Adjunct: \(\text{IP-VP-CP}_{\text{if}}\text{-IP-VP}\)

Pearl & Sprouse 2013
Calculate dependency probability from trigrams

<table>
<thead>
<tr>
<th></th>
<th>Innate</th>
<th>Derived</th>
<th>Domain-specific</th>
<th>Domain-general</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Calculate dependency probability from trigrams

Applied in different cognitive domains: **domain-general**

Likely **innate**

Pearl & Sprouse 2013
Complementizer *that*

*that*-trace effects

*Who* do you think that ___ read the book?

*Who* do you think ___ read the book?

The current learning strategy captures this distinction.
Complementizer *that*

*that*-trace effects

...but the current learning strategy will also generate a preference for object gaps without *that* compared to object gaps with *that*. (object *that*-trace effect)

What do you think that he read __?  
What do you think he read __? [prefers this one]

Interestingly, Cowart 1997 finds an object *that*-trace effect, but it is much smaller than the subject *that*-trace effect

The model generates an asymmetrical dispreference when using adult-directed corpora, which contain more instances of *that* (5.40 versus 2.81). This could be taken to be a developmental prediction of the current algorithm: Children may disprefer object gaps in embedded *that*-CP clauses more than adults, and this dispreference will weaken as they are exposed to additional tokens of *that* in utterances containing dependencies.

*Pearl & Sprouse 2013*
Some cross-linguistic issues

High probability trigrams that may be ungrammatical

Rizzi (1982) reports situations in Italian where simply doubling a grammatical sequence of trigrams leads to ungrammaticality…

\[
\text{IP-VP-CP}_{\text{wh}} \text{-IP-VP}
\]

but

\[
*\text{IP-VP-CP}_{\text{wh}} \text{-IP-VP-CP}_{\text{wh}} \text{-IP-VP-IP-VP}
\]

But these involve the same trigrams, so the learner in Pearl & Sprouse (2013) will treat both the same (either grammatical or ungrammatical). If humans do have different judgments of these, then this cannot be accounted for by this learning algorithm.

Pearl & Sprouse 2013
Parasitic gaps

The learner can’t handle parasitic gaps, which are dependencies that span an island (and so should be ungrammatical) but which are somehow rescued by another dependency in the utterance.

*Which book did you laugh [before reading __]?  
Which book did you judge ___true [before reading ___parasitic]?

Adjunct island

*What did [the attempt to repair __] ultimately damage the car?  
What did [the attempt to repair ___parasitic] ultimately damage ___true?

Complex NP island

Pearl & Sprouse 2013
Parasitic gaps

Why not? The current learner would judge the parasitic gap as ungrammatical since it is inside an island, irrespective of what other dependencies are in the utterance.

*Which book did you laugh [before reading __]?
Which book did you judge ___true [before reading ___parasitic]?

Adjunct island

*What did [the attempt to repair ___] ultimately damage the car?
What did [the attempt to repair ___parasitic] ultimately damage ___true?

Complex NP island

This may be able to be addressed in a learner that is able to combine information from multiple dependencies in an utterance (perhaps because the learner has observed multiple dependencies resolved in utterances in the input).

Pearl & Sprouse 2013
Across-the-board constructions

A similar problem occurs for across-the-board constructions.

*Which book did you [[read ___] and [then review ___]]?*
dependency for both gaps: IP-VP-VP

*Which book did you [[read the paper] and [then review ___]]?*
dependency for gap: IP-VP-VP

*Which book did you [[read ___] and [then review the paper]]?*
dependency for gap: IP-VP-VP

Again, this may be able to be addressed in a learner that is able to combine information from multiple dependencies in an utterance (perhaps because the learner has observed multiple dependencies resolved in utterances in the input).

Pearl & Sprouse 2013