Psych 215L: Language Acquisition

Lecture 16
Poverty of the Stimulus: Anaphoric One

Making arguments from acquisition

“One explicit motivation for Universal Grammar (UG) comes from an argument from acquisition: UG allows children to acquire language knowledge as effectively and rapidly as they do...In particular, UG is meant to be one or more learning biases that are part of our biological endowment (innate) and are only used for learning language (domain-specific). These learning biases allow children to solve induction problems, where the available data appear to be compatible with multiple hypotheses about the generalizations for the language.”

The utility of computational modeling

“Computational modeling results based on realistic input allow us to make progress on the debate surrounding UG by providing a formal mechanism for exploring whether learning strategies can solve apparent induction problems (or at least, generate observed behavior). We are then able to compare the types of biases required by each successful strategy, consider whether any are UG, and compare precisely what kinds of UG biases each successful strategy motivates if UG biases are indeed involved.”
Characterizing induction problems

Initial state:
Initial knowledge state + existing learning capabilities
(prior knowledge and learning biases)

data intake (Fodor 1990):
available input that children use to learn
(defined by assumptions and biases in initial state)

Learning period:
how long children have to learn (how many data points)
(usually assessed by experimental studies of children’s behavior)

target state:
knowledge children are trying to attain
(usually a specific representation, theory-dependent)

Induction problems, UG, and informative data

Traditional assumption:
Only directly related data are informative data. These data are often rare, and that’s why induction problems occur

The direct evidence assumption

If you want to learn linguistic knowledge L, you learn it by observing examples of L in your input (and possibly by also being sensitive to indirect negative evidence about what examples are missing from the input.)

Learning complex yes/no questions

Direct evidence L: “Is the boy who is in the corner t₁, happy?”

Possible indirect negative evidence: “Is the boy who t₃ in the corner is happy?”
The direct evidence assumption

If you want to learn linguistic knowledge $L$, you learn it by observing examples of $L$ in your input (and possibly by also being sensitive to indirect negative evidence about what examples are missing from the input.)

Learning the representation of English anaphoric one

Direct evidence $L$:
“Look – a red bottle! Oh look, another one.”

Possible indirect negative evidence:
**“She sat by the side of the river and he sat by the one of the road.”**

The direct evidence assumption

If you want to learn linguistic knowledge $L$, you learn it by observing examples of $L$ in your input (and possibly by also being sensitive to indirect negative evidence about what examples are missing from the input.)

Learning syntactic islands

Direct evidence $L$:
“Who _____ thinks the necklace is expensive?”
“What does Jack think is _____ expensive?”

Possible indirect negative evidence:
**“Who does Jack think [[the necklace for $e_{wa}$] is expensive?”**

A broader set of informative data

Indirect evidence: other kinds of data that may also be relevant, thereby broadening the set of informative data

Recent computational models have been exploring this:
- Complex yes/no questions (Reali & Christiansen 2005, Kam et al. 2008, Pernice, Tennenbaum, & Bregier 2011)
- Anaphoric one (Bregier & Gahl 2004, Pearl & Lida 2009, Foraker et al. 2009)
- Syntactic islands (Pearl & Sprouse forthcoming a, b)

Mapping out UG & the acquisition process

Big questions:
- When induction problems exist, what does it take to solve them?
  - What indirect evidence is available? How might a child leverage this evidence?
  - What learning biases can get the job done, given the available data? Are they necessarily innate and domain-specific (UG)?
Anaphoric One

Look - a red bottle!

Do you see another one?

Process: Because the antecedent ("red bottle") includes the modifier "red", the property RED is important for the referent of one to have.

\( \Rightarrow \text{referent of one} = \text{RED BOTTLE} \)
Anaphoric One: Syntactic Category

Standard linguistic theory (Chomsky 1970, Jackendoff 1977) claims that one in these kind of utterances is a syntactic category smaller than an entire noun phrase, but larger than just a noun (N'). This category is sometimes called N. This category includes strings like "bottle" and "red bottle".

Anaphoric One: Syntactic Category

Importantly, one is not N'. If it was, it could only have strings like "bottle" as its antecedent, and could never have strings like "red bottle" as its antecedent.

Anaphoric One: Interpretations based on Syntactic Category

If one was N', we would have a different interpretation of "Look - a red bottle! Do you see another one?"

Because one's antecedent could only be "bottle", we would have to interpret the second part as "Do you see another bottle?" and the purple bottle would be a fine referent for one.

Since one's antecedent is "red bottle", and "red bottle" cannot be N', one must not be N'.
Anaphoric One: Adult Knowledge

"Look – a red bottle! Look, there’s another one!"

Target state:
Syntactic knowledge: category N
Semantic knowledge: mentioned property ("red") is included in the linguistic antecedent (antecedent = "red bottle")

→ "Look – a red bottle! Look, there’s another red bottle!"

Anaphoric One: Children’s Knowledge

Litz, Waxman, & Freedman (2003) [LWF] found that 18-month-olds have a preference for the red bottle in the same situation.

“Look – a red bottle! Do you see another one?”

LWF interpretation & conclusion:
Preference for the RED BOTTLE means the preferred syntactic antecedent is "red bottle".

LWF conclude that 18-month-old knowledge = syntactic category of one = N
syntactic antecedent when modifier is present (i.e., property is mentioned) includes modifier (e.g., "red") ≠ referent has modifier property

Learning period = completed by 18 months

Anaphoric One: The induction problem

Initial state:
Knowledge: Syntactic categories exist, in particular N, N', and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.
Bias: Only direct evidence of one is useful.
Bias: Only unambiguous evidence of one is useful (Baker 1978).

Data intake:
All unambiguous one evidence in the input.

Learning period:
Completed by 18 months (LWF 2003)

Target state:
One is category N and its antecedent includes the modifier.
Ex: [a red [a bottle]]
**Anaphoric One: The available data**

Acquisition: Children must learn the right syntactic category for one, and the right interpretation preference for one in situations with more than one option.

Problem: Unambiguous data are rare (<0.25%, based on LWF’s analysis)

Unambiguous (UNAMB) data:

"Look – a red bottle! Hmmm - there doesn’t seem to be another one here, though.”

one’s referent = BOTTLE? If so, one’s antecedent = “bottle”.
But it’s strange to claim there’s not another bottle here.
So, one’s referent must be RED BOTTLE, and one’s antecedent = \([a, \text{red}([a, \text{bottle}])]\).

----------

**Anaphoric One: The available data**

Acquisition: Children must learn the right syntactic category for one, and the right interpretation preference for one in situations with more than one option.

Problem: Most data children encounter are ambiguous.

Syntactically (SYN) ambiguous data:

"Look – a bottle! Oh, look - another one.”

one’s referent = BOTTLE
one’s antecedent = \([a, \text{bottle}]) or \([a, \text{bottle}])?

----------

**Anaphoric One: The available data**

Acquisition: Children must learn the right syntactic category for one, and the right interpretation preference for one in situations with more than one option.

Problem: Most data children encounter are ambiguous.

Semantically and syntactically (SEM-SYN) ambiguous:

"Look – a red bottle! Oh, look – another one.”

one’s referent = RED BOTTLE or BOTTLE?
one’s antecedent = \([a, \text{red}([a, \text{bottle}])]\) or \([a, \text{bottle}]) or \([a, \text{bottle}])?

----------

**Previous learning strategies**

Update the initial state


How then?

Children have innate, domain-specific knowledge restricting the hypotheses about one: one cannot be syntactic category N.

What about when there are multiple N antecedents?

\([a, \text{red}([a, \text{bottle}])]\) or \([a, \text{bottle}])?

(No specific proposal for this.)
Previous learning strategies

Update the initial state

Baker (1978) [Baker]

initial state
Knowledge: Syntactic categories exist, in particular \(N^0, N', \) and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.
Bias: Only direct evidence of one is useful.
Bias: Only unambiguous evidence of one is useful (Baker 1978).
+ (UG) Knowledge: one is not \(N^0\).

Previous learning strategies

Update the initial state

Regier & Gahl 2004 [R&G]

initial state
Knowledge: Syntactic categories exist, in particular \(N^0, N', \) and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.
Bias: Only direct evidence of one is useful.
Bias: Only unambiguous evidence of one is useful (Baker 1978).
+ Bias: Use Bayesian inference.

Previous learning strategies

Update the initial state

Regier & Gahl 2004 [R&G]: Sem-Syn ambiguous data can be leveraged, in addition to using unambiguous data.
"Look – a red bottle! Oh, look – another one!"

How?
Use innate domain-general statistical learning abilities (Bayesian inference) to track how often one's referent has the mentioned property (e.g. red). If the referent often has the property (RED BOTTLE), this is a suspicious coincidence unless the antecedent really does include the modifier ("red bottle") and one's category is \(N^0\).

\([\text{is red}[\text{red bottle}]]\)

Previous learning strategies

Update the initial state

Pearl & Lisl 2009 [P&L]: Syn ambiguous data must not be leveraged, even if Sem-Syn and unambiguous data are used.
"Look – a bottle! Oh, look – another one!"

Why?
These data cause an "equal-opportunity" (EO) probabilistic learner to think one's category is \(N^0\).

\([\text{is not bottle}])

How?
P&L propose a domain-specific learning bias to ignore just these ambiguous data, though they speculate how this bias could be derived from an innate domain-general preference for learning when there is local uncertainty.
Previous learning strategies

Update the initial state

Pearl & Lida 2009 [P&L, R&G intended]

Initial state
Knowledge: Syntactic categories exist, in particular NP, N', and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.
Bias: Only direct evidence of one is useful.
- Bias: Only unambiguous evidence of one is useful (Baker 1979).
- (U?) Bias: Only Sem-Syn ambiguous and unambiguous data are useful.

Using indirect positive evidence

Pearl & Mis (2011, 2012 ms) [+OtherPro]: Other pronouns in the language can also be used anaphorically: him, her, it, ...

Look at the cute penguin. I want to hug him/her/it.
\[[c, cute \{\}penguin\} \rightarrow [c, him/hers/it]\]

Look! A cute penguin. I want one.
\[[c, cute \{\}penguin\} \rightarrow \{c, one\}\]

Note: The issue of one's category only occurs when one is used in a syntactic environment that indicates it is smaller than an NP (<NP).

Using indirect positive evidence

Pearl & Mis (2011, 2012 ms) [+OtherPro]: Track how often the referent of the anaphoric element (one, him, her, it, etc.) has the property mentioned in the potential antecedent, using innate domain-general statistical learning abilities (Bayesian inference).

Important: This applies, even when the syntactic category is known.

Look at the cute penguin. I want to hug him/her/it.
Look! A cute penguin. I want one.

Is the referent cute? Yes! So the antecedent includes the modifier "cute".
Using indirect positive evidence

Pearl & Mis [2011, 2012 ms] [•OtherPro]

initial state

Knowledge: Syntactic categories exist, in particular N, N′, and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.

+ Bias: Only direct evidence of one is useful.
+ Bias: Only unambiguous evidence of one is useful (Baker 1998).
+ (UG?) Bias: Use Bayesian inference.
+ (UG?) Bias: Learn from other pronoun data.

Data set comparisons: Learners using syntactic and semantic information

Unamb <NP
"Look – a red bottle! Hmm – there doesn’t seem to be another one here, though."
Learners: Baker, R&G, P&L’s EO, +OtherPro

Sem-Syn Amb
"Look – a red bottle! Oh, look – another one!"
Learners: R&G, P&L’s EO, +OtherPro

Syn Amb
"Look – a bottle! Oh, look – another one!"
Learners: P&L’s EO, +OtherPro

Unamb NP
"Look – a red bottle! I want one/it."
Learners: +OtherPro

Information in the data

Understanding a referential expression
Includes both syntactic and semantic/referential information, since both are used to determine the linguistic antecedent.

Information in the data

"Look, a red bottle! Look, another one!"

Syntactic information

R = referential expression used
ex: "another one"

Pron = pronoun used in referential expression
ex: "one"

Em = smaller than NP?
ex: yes
Information in the data

"Look, a red bottle! Look, another one!"

**Syntactic information**
- C: syntactic category of pronoun used (= syntactic category of linguistic antecedent)
  - ex: N'
- det: antecedent includes determiner(s)?
  - ex: no
- mod: antecedent includes modifier(s)?
  - ex: yes

**Semantic/referential information**
- m: property mentioned in potential antecedent
  - ex: yes
- o-m: referent (object) in current context has mentioned property
  - ex: yes
- i: mentioned property is included in antecedent?
  - ex: yes

Information in the data

"Look, a red bottle! Look, another one!"

A: antecedent
- ex: "red bottle"
  (depends on both syntactic information of det and mod, and semantic/referential information from 1.)

O: intruded object (learner can usually observe this)
- ex: RED BOTTLE

Information in the data: Unamb <NP>

"Look, a red bottle! Hmm – there isn’t another one here thought!"

A: antecedent
- ex: "another one"

O: intruded object (learner can usually observe this)
- ex: RED BOTTLE
**Information in the data: Sem-Syn ambiguous**

"Look, a red bottle! Look ~ another one!"

- **R = “another one”**
- **Pro = “one”**
- **env = <NP** (0-m = yes)
- **C = N’ or NP?** (det = no)
- **mod = yes or no?** (i = yes or no?)

- **A = “red bottle” or “bottle”?**
- **O = yes bottle**

---

**Information in the data: Syn ambiguous**

"Look, a bottle! Look ~ another one!"

- **R = “another one”**
- **Pro = “one”**
- **env = <NP** (0-m = yes)
- **C = N’ or NP?** (det = no)
- **mod = no** (i = N/A)

- **A = “bottle”**
- **O = bottle**

---

**Information in the data: Unamb NP**

"Look, a red bottle! I want it."

- **R = “it”**
- **Pro = “it”**
- **env = NP** (0-m = yes)
- **C = NP** (det = yes)
- **mod = yes** (i = yes)

- **A = “a red bottle”**
- **O = yes bottle**

---

**The online probabilistic framework**

When an object has the property mentioned in the potential antecedent (o-m=yes), tracking the probability that the property is included in the antecedent (i=yes):

- \( p_{\text{incl}} = p[i=\text{yes} \mid o-m=\text{yes}] \)
- Two values: (i=eyes or i=ins)

When the syntactic environment indicates the category is smaller than NP, tracking the probability that the syntactic category is N’ (C=N’):

- \( p_{\text{env}} = p[C=N' \mid \text{env}=<NP] \)
- Two values: (C=N’ or C=N’)

---

11/20/12
The online probabilistic framework

General form of update equations for \( p \), [adapted from Chew 1971]:

\[
p = \frac{\alpha + \text{data}}{\alpha + \beta + \text{total data}} \quad \text{after every informative data point encountered:}
\]

\[
\text{data} = \text{data} + \alpha \quad \text{incremented by probability that data point suggests x is true}
\]
\[
\text{total data} = \text{total data} + 1 \quad \text{One informative data point seen}
\]

The online probabilistic framework: Updating \( p_{incl} \)

**Explanation**

<table>
<thead>
<tr>
<th>( \phi_{incl} )</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unamb &lt;NP</td>
<td>1</td>
</tr>
<tr>
<td>Unamb NP</td>
<td>1</td>
</tr>
<tr>
<td>Syn Amb</td>
<td>N/A</td>
</tr>
<tr>
<td>Sem-Syn Amb</td>
<td>( \frac{\text{rep}}{\text{rep} + \text{rep}} )</td>
</tr>
<tr>
<td>Category = N, choose N with modifier; property is included</td>
<td></td>
</tr>
<tr>
<td>Category = N, choose N without modifier; property is not included, choose object with property by chance</td>
<td></td>
</tr>
<tr>
<td>Category = N, property is not included, choose object with property by chance</td>
<td></td>
</tr>
</tbody>
</table>

The online probabilistic framework: Updating \( p_N \)

**Explanation**

<table>
<thead>
<tr>
<th>( \phi_N )</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unamb &lt;NP</td>
<td>1</td>
</tr>
<tr>
<td>Unamb NP</td>
<td>N/A</td>
</tr>
<tr>
<td>Syn Amb</td>
<td>( \frac{\text{rep}}{\text{rep} + \text{rep}} )</td>
</tr>
<tr>
<td>Category = N', choose N' with modifier, property is included</td>
<td></td>
</tr>
<tr>
<td>Category = N', choose N' without modifier, property is not included, choose object with property by chance</td>
<td></td>
</tr>
<tr>
<td>Category = N', property is not included, choose object with property by chance</td>
<td></td>
</tr>
</tbody>
</table>

| Sem-Syn Amb | \( \frac{\text{rep}}{\text{rep} + \text{rep}} \) | Probability category is N' |
| Category = N', choose N' without modifier |
| Category = N' |

\( \text{rep} = \frac{m}{m+n} \quad \text{rep} = (1 - \text{p(N)}) \times (1 - \text{p(N)}) \times \frac{1}{2} \quad \text{rep} = (1 - \text{p(N)}) \times (1 - \text{p(N)}) \times \frac{1}{2} \quad \text{rep} = (1 - \text{p(N)}) \times (1 - \text{p(N)}) \times \frac{1}{2} \)
Example updates

Start with $p_u = p_{out} = 0.50, m = 1, n = 2.9, s = 10$

One Unamb <NP data point: $p_u = 0.67, p_{out} = 0.67$

One Unamb NP data point: $p_u = 0.50, p_{out} = 0.67$

One Sem-Syn Amb data point: $p_u = 0.59, p_{out} = 0.53$

One Syn Amb data point: $p_u = 0.48, p_{out} = 0.50$

Corpus Analysis & Learner Input

Brown/Eve corpus (CHILDES: MacWhinney 2000): starting at 18 months
17,521 utterances of child-directed speech, 2,874 referential pronoun utterances

<table>
<thead>
<tr>
<th></th>
<th>Baker</th>
<th>R&amp;G</th>
<th>P&amp;L's EO</th>
<th>P&amp;M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unamb &lt;NP</td>
<td>0.00%</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Sem-Syn Amb</td>
<td>0.66%</td>
<td>0</td>
<td>242</td>
<td>242</td>
</tr>
<tr>
<td>Syn Amb</td>
<td>7.52%</td>
<td>0</td>
<td>2743</td>
<td>2743</td>
</tr>
<tr>
<td>Unamb NP</td>
<td>8.42%</td>
<td>0</td>
<td>0</td>
<td>3073</td>
</tr>
<tr>
<td>Uninformative</td>
<td>83.4%</td>
<td>36500</td>
<td>36258</td>
<td>33515</td>
</tr>
</tbody>
</table>

Pearl & Lidz (2009): Children learn one's representation between 14 and 18 months.

Based on estimates of the number of utterances children hear from birth until 18 months (Akbartar et al., 2004), we can calculate the data distribution in their input (36,500 referential pronoun utterances total).
Measures of Success:
LWF children's behavior

In addition to directly assessing $p_{D_1}$ and $p_{D_2}$, we can measure how often a learner would reproduce the behavior in the LWF experiment ($p_{rep}$):

- Look - a red bottle!
- Do you see another one?

Testing LWF’s assumption about what behavior means

In addition to directly assessing the learner’s behavior, we can assess LWF’s assumption that target behavior indicates the children have the target representation for one.

Is it possible to get target behavior in the LWF experiment without having the target representation for one in general (as measured by $p_{D_1}$ and $p_{D_2}$)?

Is it possible to get target behavior in the LWF experiment without having the target representation for one at the time the behavior is being produced?

- $p_{rep/bak} = \frac{rep_1}{rep_1 + rep_2 + rep_3}$ the probability the look to the red bottle is because the learner has the target representation (N, “red bottle”)

Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

$\mathbf{s} = 2, 5, 7, 10, 20, 49$

<table>
<thead>
<tr>
<th>Baker</th>
<th>$\mathbf{P_{D_1}}$</th>
<th>$\mathbf{P_{D_2}}$</th>
<th>$\mathbf{P_{rep/bak}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.50 (±0.01)</td>
<td>0.56 (±0.01)</td>
<td>0.23 (±0.01)</td>
</tr>
</tbody>
</table>

Since the input data include no Unambiguous NP data, and those are the only data the Baker learner learns from, it learns nothing.

It is at chance for having the target syntactic and semantic representation.

It is only slightly above chance at producing the observed toddler behavior, and when it does, it unlikely to have the target representation when doing so.

Implication: This is an induction problem if only unambiguous NP data are relevant.
Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

\( s = 2, 5, 7, 10, 20, 49 \)

<table>
<thead>
<tr>
<th></th>
<th>Baker</th>
<th>R&amp;G</th>
<th>+OtherPro</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_{\text{ant}} )</td>
<td>0.50 (±0.01)</td>
<td>0.91-0.97 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{SN}} )</td>
<td>0.50 (±0.01)</td>
<td>0.98-0.99 (±0.01)</td>
<td>0.37-0.38 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{indep}} )</td>
<td>0.56 (±0.01)</td>
<td>0.88-0.99 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{indep}} )</td>
<td>0.23 (±0.01)</td>
<td>0.87-0.99 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
</tbody>
</table>

The learner robustly decides the antecedent should include the mentioned property. However, the learner has a moderate dispreference for believing one is \( N \) when it is smaller than \( N \).

This is therefore not the target representation.

Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

\( s = 7, 10, 20, 49 \)

<table>
<thead>
<tr>
<th></th>
<th>Baker</th>
<th>R&amp;G</th>
<th>+OtherPro</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_{\text{ant}} )</td>
<td>0.50 (±0.01)</td>
<td>0.91-0.97 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{SN}} )</td>
<td>0.50 (±0.01)</td>
<td>0.98-0.99 (±0.01)</td>
<td>0.37-0.38 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{indep}} )</td>
<td>0.56 (±0.01)</td>
<td>0.88-0.99 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{indep}} )</td>
<td>0.23 (±0.01)</td>
<td>0.87-0.99 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
</tbody>
</table>

Other learning strategies: R&G (P&Ls filtered learner)

Variability, depending on the value of \( q \), which determines how suspicious a coincidence is that the intended object just happens to have the mentioned property.

When \( q > 7 \) or above, this learner believes a mentioned property should be included in the antecedent and one is \( N \) when it is smaller than \( N \) which is similar to previous findings by R&G & P&L. In addition, it is likely to generate the observed toddler behavior and have the target representation when doing so.

Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

\( s = 5 \)

<table>
<thead>
<tr>
<th></th>
<th>Baker</th>
<th>R&amp;G</th>
<th>+OtherPro</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_{\text{ant}} )</td>
<td>0.50 (±0.01)</td>
<td>0.68 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{SN}} )</td>
<td>0.50 (±0.01)</td>
<td>0.79 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{indep}} )</td>
<td>0.56 (±0.01)</td>
<td>0.38 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>( p_{\text{indep}} )</td>
<td>0.23 (±0.01)</td>
<td>0.58 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
</tbody>
</table>

Other learning strategies: R&G (P&Ls filtered learner)

Variability, depending on the value of \( q \), which determines how suspicious a coincidence it is that the intended object just happens to have the mentioned property.

However, when \( s = 5 \), the learner is less sure the mentioned property should be included in the antecedent, which causes the learner to be less likely to generate the observed toddler behavior and only slightly above chance at having the target representation when generating that behavior.
Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

$s = 2$

<table>
<thead>
<tr>
<th>Baker</th>
<th>R&amp;G</th>
<th>P&amp;L's EO</th>
<th>+OtherPre</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.02 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.34 (±0.01)</td>
<td>0.34 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.50 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.23 (±0.01)</td>
<td>&lt;0.01 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
</tbody>
</table>

Other learning strategies: R&G (P&L's filtered learner)

Variability, depending on the value of $s$, which determines how suspicious a coincidence is that the intended object just happens to have the mentioned property.

When $s = 2$, the learner is sure the mentioned property should not be included in the antecedent, and prefers one to be N when it is smaller than N. This causes the learner to be at chance for generating the observed toddler behavior, and very unlikely to have the target representation when generating that behavior.

Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

$s = 2, 5, 10$

<table>
<thead>
<tr>
<th>Baker</th>
<th>R&amp;G</th>
<th>P&amp;L's EO</th>
<th>+OtherPre</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.02 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.34 (±0.01)</td>
<td>0.34 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.50 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.23 (±0.01)</td>
<td>&lt;0.01 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
</tbody>
</table>

Other learning strategies: P&L's EO learner

Variability, depending on the value of $s$, which determines how suspicious a coincidence it is that the intended object just happens to have the mentioned property.

When $s$ is less than 10, the learner does not believe the mentioned property should be included in the antecedent, and prefers one to be N when it is smaller than N. This causes the learner to be at chance for generating the observed toddler behavior, and unlikely to have the target representation when generating that behavior.

Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

$s = 20, 49$

<table>
<thead>
<tr>
<th>Baker</th>
<th>R&amp;G</th>
<th>P&amp;L's EO</th>
<th>+OtherPre</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.99 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.99 (±0.01)</td>
<td>0.34 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.50 (±0.01)</td>
<td>0.99 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
<tr>
<td>$p_{int}$</td>
<td>0.23 (±0.01)</td>
<td>0.98 (±0.01)</td>
<td>&gt;0.99 (±0.01)</td>
</tr>
</tbody>
</table>

Other learning strategies: P&L's EO learner

Variability, depending on the value of $s$, which determines how suspicious a coincidence it is that the intended object just happens to have the mentioned property.

However, when $s$ is 20 or 49, the learner strongly believes the mentioned property should be included in the antecedent, though it still prefers one to be N when it is smaller than N. This causes the learner to be likely to generate the observed toddler behavior, and likely to have the target representation when generating that behavior.

This is more like the +OtherPre learner results.
### Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

<table>
<thead>
<tr>
<th>s</th>
<th>Baker R &amp; G</th>
<th>P&amp;L’s ED</th>
<th>+OtherPre</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.50 (+0.21)</td>
<td>0.99 (+0.01)</td>
<td>0.97-0.99 (0.01-0.01)</td>
</tr>
<tr>
<td>5</td>
<td>0.50 (+0.21)</td>
<td>0.99 (+0.01)</td>
<td>0.34-0.37 (0.03-0.03)</td>
</tr>
<tr>
<td>10</td>
<td>0.56 (+0.05)</td>
<td>0.98-0.99 (0.01-0.01)</td>
<td>0.79-0.94 (0.02-0.07)</td>
</tr>
<tr>
<td>20</td>
<td>0.23 (+0.01)</td>
<td>0.98-0.99 (0.01-0.01)</td>
<td>0.72-0.94 (0.02-0.11)</td>
</tr>
</tbody>
</table>

**What’s going on?**

The flip side of what we saw with the R&G learner. If the suspicious coincidence is very strong, Sem-Syn ambiguous data help the learner increase $p_{n1}$ (and $p_{n2}$) – in fact, they become almost as powerful as Unambiguous <NP> data. Because both $p_{n1}$ and $p_{n2}$ are used to calculate $\Phi_{n1}$ and $\Phi_{n2}$, a very high $p_{n1}$ can bolster $p_{n2}$ and overpower the effect of the troublesome Syn ambiguous data.

### Learner results: Strategy comparison

Averages over 1000 simulations, standard deviations in parentheses.

<table>
<thead>
<tr>
<th>s</th>
<th>Baker R &amp; G</th>
<th>P&amp;L’s ED</th>
<th>+OtherPre</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.50 (+0.01)</td>
<td>0.02-0.99 (0.01)</td>
<td>&lt;0.01-0.99 (0.01-0.01)</td>
</tr>
<tr>
<td>5</td>
<td>0.50 (+0.01)</td>
<td>0.34-0.99 (0.01-0.01)</td>
<td>0.14-0.37 (0.03-0.03)</td>
</tr>
<tr>
<td>10</td>
<td>0.56 (+0.01)</td>
<td>0.50-0.99 (0.01-0.01)</td>
<td>0.50-0.94 (0.01-0.07)</td>
</tr>
<tr>
<td>20</td>
<td>0.23 (+0.01)</td>
<td>&lt;0.01-0.95 (0.01-0.01)</td>
<td>&lt;0.01-0.94 (0.01-0.11)</td>
</tr>
</tbody>
</table>

**Recap: Learning strategies & induction problems**

“...using a learning strategy that draws on indirect positive evidence, a child would be able to produce the behavior at 18 months that was thought to indicate the target knowledge state, presumably solving the induction problem.”

“However, surprisingly, this behavior can be produced without reaching the target state - in fact, a child with an immature context-dependent representation of one could produce the observed behavior. This suggests that the link between observed behavior, interpretation, and representation may not be as clear cut as once thought. Even though children demonstrate they have the adult interpretation some of the time (by displaying adult-like behavior), this does not necessarily mean they have the adult representation all of the time.”
Recap: Learning strategies & induction problems

“...the learning period may not be restricted to 18 months. Instead, it could be that children achieve the target knowledge state later on. If so, this means they may have access to additional data, knowledge, and learning capabilities to solve the induction problem that we did not allow the learners modeled here...More generally, it would suggest a two-stage acquisition trajectory for anaphoric one, with the first stage completed by 18 months and the second stage completed sometime afterwards.”

Recap: Motivating UG

What kind of biases does the +OtherPro learner use?

initial state: Two new biases

Knowledge: Syntactic categories exist, in particular N0, N', and NP.

Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.

+ (non-UG) Bias: Use Bayesian inference.

+ Bias: Learn from other pronoun data.

Bias to learn from other pronoun data: concerns language data, so clearly domain-specific.

innate or derived?

If innate, then this is a UG bias.

Could be derived from prior linguistic experience with pronouns (and noticing overlapping syntactic environments for “one” and other referential pronouns.)
### Other induction problem characterizations

A different target state


**target state**

One is category \( N' \) and its antecedent includes the modifier.

Just learning about the syntactic representation of *one* when it is smaller than \( NP \).

Baker’s original proposal:
initial state includes UG knowledge that one is not \( N^0 \).

---

### Other induction problem characterizations

A different target state


**target state**

One is category \( N' \) and its antecedent includes the modifier.

Just learning about the syntactic representation of *one* when it is smaller than \( NP \).

Foraker et al’s proposal:
Use Bayesian inference on the available syntactic data only, given domain-specific knowledge of complements and modifiers.

---

### Modifiers & Complements

**Syntactic modifier:** not "conceptually evoked by its head noun";
indicates noun string is \( N' \)
Ex: “the ball with dots” (I like the *one with dots*.)

**Syntactic complement:** "conceptually evoked by its head noun";
indicates noun string is \( N^0 \)
Ex: “the side of the road” (*I waited by the one of the road.*)

---

### The Foraker et al. learning strategy

Foraker et al. 2009

**initial state**
Knowledge: Syntactic categories exist, in particular: \( N^0 \), \( N' \), and \( NP \).
Knowledge: Anaphoric elements like *one* take linguistic antecedents of the same category.
+ Bias: Only syntactic data are useful.
+ Bias: Use Bayesian inference.
+ Bias: Learn from all linguistic elements that take complements or modifiers.
+ Knowledge: Complements conceptually evoke their head noun while modifiers do not.
+ Knowledge: Syntactic category \( N^0 \) is sister to a complement, not a modifier.

This strategy was successful at learning one is category \( N' \) (not \( N^0 \)) from child-directed speech data.
Foraker et al. bias types

Foraker et al. 2009
initial state
Knowledge: Syntactic categories exist, in particular N0, N1, and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the
same category.
+ Bias: Only syntactic data are useful.

This bias could be derived from the target knowledge only pertaining to the
syntactic representation.

Foraker et al. bias types

Foraker et al. 2009
initial state
Knowledge: Syntactic categories exist, in particular N0, N1, and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the
same category.
+ Bias: Only syntactic data are useful.
+ (non-G) Bias: Use Bayesian inference.

This bias is likely innate and domain-general.

Foraker et al. bias types

Foraker et al. 2009
initial state
Knowledge: Syntactic categories exist, in particular N0, N1, and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the
same category.
+ (non-G) Bias: Only syntactic data are useful.
+ (UG) Bias: Use Bayesian inference.
+ (UG) Bias: Learn from all linguistic elements that take complements or
modifiers.
+ Knowledge: Complements conceptually evoke their head noun while modifiers
do not.

Knowing complements evoke their head noun while modifiers do not is
domain-specific knowledge that is not obviously derivable.

Foraker et al. bias types

Foraker et al. 2009
initial state
Knowledge: Syntactic categories exist, in particular N0, N1, and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the
same category.
+ (non-G) Bias: Only syntactic data are useful.
+ (UG) Bias: Use Bayesian inference.
+ (UG) Bias: Learn from all linguistic elements that take complements or
modifiers.
+ Knowledge: Complements conceptually evoke their head noun while modifiers
do not.

Knowing complements evoke their head noun while modifiers do not is
domain-specific knowledge that is not obviously derivable.
Foraker et al. bias types

Foraker et al. 2009

initial state

Knowledge: Syntactic categories exist, in particular N₀, N', and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.

+ (non-UG) Bias: Only syntactic data are useful.
+ (UG) Bias: Use Bayesian inference.
+ (UG) Bias: Learn from all linguistic elements that take complements or modifiers.
+ (UG) Knowledge: Complements conceptually evoke their head noun while modifiers do not.

Knowledge: Syntactic category N₀ is sister to a complement, not a modifier.

Knowing N₀ is sister to complement is also domain-specific knowledge that is not obviously derivable.

Foraker et al. bias types

Foraker et al. 2009

initial state

Knowledge: Syntactic categories exist, in particular N₀, N', and NP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.

+ (non-UG) Bias: Only syntactic data are useful.
+ (UG) Bias: Use Bayesian inference.
+ (UG) Bias: Learn from all linguistic elements that take complements or modifiers.
+ (UG) Knowledge: Complements conceptually evoke their head noun while modifiers do not.

Knowledge: Syntactic category N₀ is sister to a complement, not a modifier.

Upshot: This form of the induction problem leads to a different proposal for the contents of UG, even when Bayesian inference is used.

Other induction problem characterizations

A different initial & target state: Alternate theoretical representations

N₀, N', and NP vs. N₀, N', NP, and DP

Other induction problem characterizations

A different initial & target state: Syntactic categories N₀, N', NP, DP

initial state

Knowledge: Syntactic categories exist, in particular N₀, N', NP, and DP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.

Bias: Only direct evidence of one is useful.
Bias: Only unambiguous evidence of one is useful.

target state

Knowledge: In utterances like “Look, a red bottle! Look, another one!”, one is category NP and so its antecedent includes the modifier (“red”).
Other induction problem characterizations

A different initial & target state: Syntactic categories N0, N', NP, DP
What an indirect positive evidence strategy like +OtherPro would do

initial state
Knowledge: Syntactic categories exist, in particular N0, N', NP, and DP.
Knowledge: Anaphoric elements like one take linguistic antecedents of the same category.
Bias: Only direct evidence of one is useful.
Bias: Only unambiguous evidence of one is useful.
+ (non-IG) Bias: Use Bayesian inference
+ (IG) Bias: Learn from other pronoun data.

Other induction problem characterizations

A different initial & target state: Syntactic categories N0, N', NP, DP
What an indirect positive evidence strategy like +OtherPro would do

(1) Syn ambiguous data still ambiguous between two categories (N0 and N'), and Bayesian inference causes learner to prefer the hypotheses that includes fewer strings, which is still the N0 category. (N' includes noun+complement strings)
Syn ambiguous data still cause p90 to drop, though perhaps not as fast.

Other induction problem characterizations

A different initial & target state: Syntactic categories N0, N', NP, DP
What an indirect positive evidence strategy like +OtherPro would do

(2) Sem-Syn ambiguous data still ambiguous between three antecedents. When n is high enough (>5), the suspicious coincidence still causes the learner to increase p107.
Sem-Syn ambiguous data still cause p107 to increase when the suspicious coincidence is strong enough.

Other induction problem characterizations

A different initial & target state: Syntactic categories N0, N', NP, DP
What an indirect positive evidence strategy like +OtherPro would do

(3) Unambiguous +DP data still indicate antecedent that includes modifier – it’s just that the category label is NP (rather than N').
p201 and p202 both increase.
Unambiguous +DP data still cause p201 and the category that includes the modifier (NP) to increase.
Other induction problem characterizations

A different initial & target state: Syntactic categories N^0, N', NP, DP

What an indirect positive evidence strategy like +OtherPro would do

(4) Unambiguous DP data still indicate antecedent that includes modifier - it’s just that the category label is DP (rather than NP).

\( p_{nol} \) still increases.

Unambiguous DP data still cause \( p_{nol} \) to increase.

Other induction problem characterizations

A different initial & target state: Syntactic categories N^0, N', NP, DP

What an indirect positive evidence strategy like +OtherPro would do

Given that the updates from the different data types are effectively the same, the overall outcome should be similar: \( p_{nol} \) should be high while \( p_{ol} \) should be low. (Note: \( p_{ol} \) should also be very low, since no data cause it to increase.)

Non-target context-dependent representation.

\( p_{nol} = \text{high}, p_{ol} = \text{low} \)

LWF experiment: target behavior (and target representation when displaying that behavior) because of \( p_{nol} \).

Back to the big picture

Motivating UG through the existence of induction problems & the solutions to those induction problems

- **Existence**
  - Requires a specific characterization that defines initial state, data intake, learning period, and target state

  Here: Specifying different forms of the induction problem for learning anaphoric one, drawing on theoretical and experimental data

  ...so we can compare different solutions

- **Solutions**
  - Exploring an indirect positive evidence learning strategy...

    - Can’t reach the target state but can generate observed behavior, so maybe we need to redefine the induction problem (learning period is longer than 18 months).
Back to the big picture

About acquisition

- Solving induction problems in language may proceed in stages
- Anaphoric one knowledge, pre- and post-18 months

- The link between observed behavior and underlying knowledge is not always straightforward
- Target behavior can be produced even without the target knowledge for anaphoric one when learning from both syntactic and semantic information. Specifically, context-dependent representations can result, even though the target representation is (purportedly) context-independent.

Take-home points

- Indirect evidence does not necessarily mean indirect negative evidence – it can come from considering a broader pool of informative data
- Indirect evidence does not necessarily negate the need for learning biases (of whatever kind)
- Considering indirect evidence and its impact on acquisition can help define concrete proposals about what is necessarily innate and domain-specific, and thus what is in Universal Grammar
- Knowing the impact of the necessary learning biases on acquisition may also inform us about the acquisition trajectory