When domain general learning fails and when it succeeds: Identifying the contribution of domain specificity.

Lisa Pearl & Jeffrey Lidz
Linguistics Department
1401 Marie Mount Hall
University of Maryland
College Park, MD 20742
llsp@umd.edu, jlidz@umd.edu
Abstract

We identify three components of any learning theory: the representations, the filters on data intake, and the knowledge updating procedure. With these in mind, we model the acquisition of the English anaphoric pronoun one. We show first, that a domain general updating procedure fails to learn anaphoric one. However, when this procedure is paired with a domain specific filter on data intake, then it succeeds. Thus, we argue for a learning theory that is both domain specific and domain general.

Keywords: learnability, representations, filters, Bayesian learning, anaphoric one, domain specific learning, domain general learning
1. Introduction

Vast quantities of ink and hard feelings have been spilt and spent on the nature of learning in humans and other animals. Are there domain specific learning mechanisms or is learning the same across all domains? One of the most frequent battlegrounds in this debate is the case of language learning. Is there a domain specific language acquisition device or does language acquisition rely solely on domain general learning mechanisms? We believe that the phrase “domain specific learning” can be and has been interpreted in several distinct ways, leading to the illusion of disagreement. However, by examining these interpretations, we believe we can find points of reconciliation for these differing viewpoints.

There are three pieces to any learning theory. First, learners must have a way of representing the data to be learned from. In the domain of language learning, these would be the linguistic representations, such as phonemes, morphemes and phrase structure trees. If learners come equipped with a space of possible linguistic representations, then we have a domain specific representational format in our learning theory. On the other hand, if learners represent the information in the linguistic signal in terms of cooccurrence probabilities between properties of the acoustic signal, then we do not have a domain specific representational format in our learning theory. Of course, it is possible that domain specific representations can be constructed out of the domain general representations of the input. In this case, then, we do not have domain specific representations as part of the learning theory. Instead, we would have domain specific representations as the output of learning.

Second, learners must decide which data to learn from. In the domain of language learning, one might propose that only some data are usable by the learner. For example, Lightfoot (1991) proposes that main clause data are privileged for the learner. Data in embedded clauses are initially ignored for the purposes of grammar learning. Because such filters are defined over the linguistic representations, they instantiate domain specific filters on data intake. On the other hand, if what looks like a constraint against embedded clause data were in fact due to learners only having a finite amount of working memory, causing them to use, say, only the first 4 words of an utterance, this would be a domain general filter. Of course, it is also possible that learners treat all of their linguistic input equally and that there is no constraint from the learning mechanism on what data are relevant for learning.

Third, learners must have a way of updating their knowledge on the basis of the selected data. Any learning algorithm that is used only for language learning (e.g., Gibson and Wexler 1994) would count as an example of a domain specific learning procedure. On the other hand, if the same learning algorithm is used across different domains (e.g., Bayesian learning, Tenenbaum and Griffiths, 2001), then the learning algorithm is domain general.

In principle, it is an independent question for each aspect of the learning theory whether it is domain specific or domain general. Although these aspects have typically been equated, they are, in fact, separate and should be addressed independently. Any one of these components might be domain general while the others are domain specific. This is how we reconcile the opposing viewpoints on linguistic nativism. In the current paper we provide a case study in which we show that language acquisition can take advantage of a domain general learning procedure. However, we also show that this learning procedure can only work when paired with domain specific filters on data intake.¹

¹ The overall message of the paper is unaffected by the question of whether the representations are built in or derived by a domain general learning procedure. In order for the learning
A. Why a domain general learning procedure?

Probabilistic reasoning has been shown to be the optimal strategy for solving problems and making decisions given noisy or incomplete information (J. Pearl, 1996). But it is important to keep in mind that reasoning (probabilistically or otherwise) requires an adequate understanding of the representations that are used in the relevant mental computations. A probabilistic learner takes probabilistically available information to derive a conclusion about a discrete representation from a range of antecedently available options (cf. Shannon, 1948). The appeal of probabilistic learning is that one can apply this domain general learning procedure to representations in any particular domain, allowing for the same domain general learning procedure to apply independent of the character of the representations that are to be learned. For language learning, a probabilistic approach allows us to marry a domain general learning theory with domain specific representations and filters. We argue below that the domain specific filters on the learner play a vital role in constraining the usefulness of a domain general (probabilistic) learning procedure.

The phenomenon under investigation is the interpretation of the anaphoric element one in English. In previous work, we argued that infants’ knowledge of anaphoric one could not be derived from their experience with this form (Lidz, Waxman and Freedman, 2003; Lidz and Waxman, 2004). Instead, we argued, the learner must be equipped with constraints on the interpretation of pronouns. These constraints, we argued, were a part of the domain specific representational format for language learning. Regier and Gahl (2004), however, replied that a Bayesian (i.e. probabilistic) learner could acquire this knowledge with a domain general learning algorithm that lacked such constraints.

Importantly, Regier and Gahl (henceforth, R&G) provided their learning model with a small set of candidates to choose from that were derived from domain specific representational content. However, they argued that this candidate set does not include constraints on the interpretations of pronouns, contra Lidz, Waxman and Freedman (2003) (henceforth LWF). Here, we argue that R&G’s conclusion was too quick. In particular, their learning model considered only a restricted source of evidence, which inflated their estimate of the learner’s success. By doing so, this model implemented two domain specific filters on the learner’s data intake. When the full array of evidence is considered, i.e., without their domain specific filters, a Bayesian learning model fails in this domain.

The problem with R&G’s model goes beyond its implicit filtering of the data. For the problem under consideration, i.e, the interpretation of anaphoric one, and in language learning more generally, learners must come to align representations across domains. When we consider the inherent multidimensionality of linguistic representations, and the consequences of correspondences across levels of representation, we find that the Bayesian learning algorithm fares even worse. This conclusion casts doubt on Bayesian learning as the primary source of constraints on learners. In other words, the domain specific theory of representations and a set of domain specific filters on data intake are required to overcome the overly general nature of the domain general learning procedure.

For the case study of anaphoric one, there are two different levels of representation that must be considered: the syntactic level and the semantic level. At the syntactic level, the infant must learn what the linguistic antecedent of one is; at the semantic level, the infant must
determine what object in the world an NP containing one refers to. Both of these levels contribute to the information a Bayesian learner would use when converging on the correct representation of one. A linguistic antecedent (syntax) can be translated into a reference to an object in the world (semantics) and so both syntactic and semantic representations are implicated in knowledge of one. As we will see below, the correct syntactic representation for English adults is that the linguistic antecedent of one is a string classified as N’. This syntactic knowledge has semantic consequences, which are what LWF used to determine if 18-month olds had that specific syntactic representation. In this way, the knowledge that one refers to N’ strings traverses both the syntax domain and the semantics domain.

Acquisition of anaphoric one is an interesting learning problem because the data that would lead a learner to the correct representation is quite sparse. In particular, LWF showed that less than 0.3% of the child’s input containing anaphoric one provided unambiguous evidence for the correct representation. Moreover, the rate of ungrammatical sentences containing anaphoric one was twice this amount, making the occurrence of useful (unambiguous) data below noise level. Given this pattern of data, LWF argued (following Baker (1979) and Hornstein & Lightfoot (1981)) that constraints on the representation of anaphoric one must be built into the learner’s domain specific representations.

R&G countered that a learner using a domain general Bayesian learning procedure could converge on this knowledge by using ambiguous data with certain properties. Using this ambiguous data, they argued, would make the proposed constraint on the linguistic representations unnecessary. However, R&G’s model made use of only some of the available ambiguous data and of only semantic data to converge on the syntactic representation. This decision implements two domain specific filters on the learner’s data intake. In this article, we investigate the results of a probabilistic Bayesian learning procedure that removes these filters.

The procedure we develop uses all the available ambiguous data as well as both syntactic and semantic data to converge on the probabilities of competing representations. We will show that, even under the most generous estimates of the various parameters involved in such a model, a Bayesian learner lacking domain specific filters on data intake will not only fail to converge on the syntactic knowledge that one is anaphoric to N’ strings, but will also fail to produce the results reported in the LWF experiment.

The paper proceeds as follows. In section 2, we briefly describe the Bayesian learning framework that we will be adopting and detail how the learning procedure operates in the face of variously structured hypothesis spaces. In section 3, we describe the grammar of anaphoric one, the linguistic phenomenon we use as our case study. We also describe the behavioral evidence indicating that 18-month-olds have acquired the adult representation of anaphoric one, and repeat the argument from LWF that the input available to children is too sparse to support acquisition of this knowledge. In section 4, we address various proposals to circumvent the sparse data problem. We argue that prior proposals about a domain-general solution to this problem in fact implement implicit domain specific filters on data intake. In section 5, we describe a Bayesian learning model that is truly domain general, in that it removes all implicit filtering on the data. We show that such a model fails to acquire the adult representations of anaphoric one. In section 6, we also show that such a model would fail to behave as infants do in the LWF experiment. In addition, we describe how under a set of less charitable parameter values, the Bayesian learning model would perform even more poorly. In section 7, we identify the source of the model’s failure. One contributing factor to the spectacular failure of the model derives from the link between syntax and semantics. A second contributor to this failure is the abundance of ambiguous data, which given Bayesian learning techniques causes to the learner to misconverge.
We argue that successful acquisition depends on a domain specific filter on the data. Finally, in section 8, we speculate on the origin of the necessary domain-specific filter, tentatively suggesting that its roots may lie in an innate predisposition towards the correct syntactic representation.

2. Bayesian Learning: How It Works

Bayesian learning is a probabilistic learning procedure for choosing one hypothesis from a predefined hypothesis space, given a series of examples as input. There are several ways the hypothesis spaces can be constructed, and the layout of the hypothesis space affects how the Bayesian learner converges on the correct alternative. We examine several instances of hypothesis spaces below.

A. A Simple Case: Two Hypotheses, Equally Likely

Suppose there are two nonoverlapping hypotheses in the set: A and B. Suppose also that the learner has no reason to be biased towards one hypothesis, so the initial probabilities the learner assigns to both A and B are 0.50.

![Two Non-Overlapping Hypotheses, Equally Probable Initially](image)

Figure 1. Two non-overlapping hypotheses, equally probable initially. The shading reflects how much probability is associated with each hypothesis.

The learner then receives input (say, \(d_i\) data points) and tries to shift the probability mass between A and B to reflect the distribution in the input. If the input consists only of examples of A, the learner will eventually shift the probability so A is 1.0 and B is 0.0. Conversely, if the input consists only of examples of B, the learner will eventually shift the probability so A is 0.0 and B is 1.0. If the learner receives a mixed distribution as input, the learner will shift the probability to match the perceived distribution – thus, if the input is consistently 30% A examples, the learner will eventually shift A to 0.30 and B to 0.70.
Two non-overlapping hypotheses with equal initial probability after seeing various types of input. The shading reflects how much probability is associated with each hypothesis.

B. Two Hypotheses, with a Bias for One Hypothesis

Suppose the hypothesis space again has two nonoverlapping hypotheses, A and B, but this time the learner is biased towards A. For example, let the initial probability assigned to A be 0.7, and the initial probability assigned to B be 0.3.

Figure 3. Two non-overlapping hypotheses, with initial bias towards hypothesis A. The shading reflects how much probability is associated with each hypothesis.
Again, the learner receives input and tries to shift the probability mass between A and B to reflect the perceived distribution in the input. As before, a learner receiving all A or all B examples will eventually shift the probability so that one hypothesis is 1.0 while the other is 0.0 – however, it will take longer for B to reach 1.0 (requiring more than $d_1$ data points) since B begins at a probability below that of A. The same is true for a mixed distribution in the input: the learner will eventually shift to the correct probability distribution, but may take longer if B is the one with the higher probability in the input.

Figure 4. Two non-overlapping hypotheses with an initial bias for hypothesis A after seeing various types of input. The shading reflects how much probability is associated with each hypothesis.

C. A Less Simple Case: Two Hypotheses in a Subset Relation, Equally Likely

Suppose the hypothesis space again consists of two hypotheses, but one hypothesis, A, is a subset of the other hypothesis, B – thus all examples of A are also examples of B (Pinker, 1979; Berwick, 1985; Berwick & Weinberg, 1984; Manzini & Wexler, 1987; Tenenbaum & Griffiths, 2001). The learner is then trying to decide if the subset A or the superset B is the correct hypothesis. Suppose the initial probabilities assigned to both A and B are 0.5.
Figure 5. Two overlapping hypotheses in a subset relation, with equal probability initially. The shading reflects how much probability is associated with each hypothesis.

If the input (say $d_2$ data points) consists only of B examples, these are unambiguous examples for B (since they cannot be examples of A). The learner will therefore quickly converge on B with probability 1.0.

Figure 6. Two overlapping hypotheses in a subset relation with equal probability initially, after seeing $d_2$ data points that are unambiguous for hypothesis B. The shading reflects how much probability is associated with each hypothesis.

However, suppose the input consists only of examples of A. Because all of these A examples are also examples of B, they are equally compatible with both hypotheses. Thus, it is impossible to receive any unambiguous data for A. Since the learner in this situation is not receiving any unambiguous B examples, the learner might be doomed to remain at probability 0.5 forever for both hypotheses.

Fortunately, the size principle of Tenenbaum & Griffiths (2001) is an implementation of Bayesian learning that allows the learner in this situation to converge on the subset (A), using the following logic: if the superset B were really the correct hypothesis, there should be some unambiguous B examples in the input. The continuing absence of unambiguous superset examples allows the learner to eventually shift all the probability to the subset A. Note that this
convergence will take longer than if the input consisted only of unambiguous superset B examples and the learner needed to shift all the probability to B.

![Diagram](image1)

**Figure 7.** Two overlapping hypotheses in a subset relation with equal probability initially, after seeing more than $d_2$ data points that are examples of A. The learner uses the size principle to converge on hypothesis A. The shading reflects how much probability is associated with each hypothesis.

In the case of a mixed distribution, the unambiguous superset B examples will have more of an effect on the learner’s probability distribution than the ambiguous A examples – though both will contribute to the final probability distribution the learner converges on.

![Diagram](image2)

**Figure 8.** Two overlapping hypotheses in a subset relation with equal probability initially, after seeing more than $d_2$ data points that are a mix of examples of A and B. The learner uses the size principle to converge on the probability that reflects the distribution observed in the input. The shading reflects how much probability is associated with each hypothesis.

**D. Summary**

Having given a brief overview of how Bayesian learning can be used to choose from a set of predefined hypotheses in the general case, we turn now to the specific case of anaphoric *one*. As we will see, the hypothesis space for anaphoric *one* is an instantiation of the subset-
superset scenario just described. Before detailing the specifics of Bayesian learning for anaphoric one, we first describe the correct representation for anaphoric one.

3. Anaphoric One

A. Adult Knowledge: Grammar

For English adults, the element one is anaphoric to strings that are classified as N’ (i.e., the antecedent for one is an N’ string), as in example (1) below. The structures for the N’ strings are represented in figure 9.

(1a) One is anaphoric to N’ (ball is antecedent)
   “Jack likes this ball and Lily likes that one.”
(1b) One is anaphoric to N’ (red ball is antecedent)
   “Jack likes this red ball and Lily likes that one.”

These representations encode two kinds of information: constituency structure and category structure. The constituency structure tells us that in a Noun Phrase (NP) containing a determiner (det), adjective (adj) and noun (N⁰), the adjective and noun form a unit within the larger Noun Phrase. The fact that one can be interpreted as a replacement for those two words (as in (1b)), tells us that those two words form a syntactic unit. The category-structure tells us which pieces of phrase structure are of the same type. That is, both ball and red ball are of the type N’. The following argument explains how we know this.

Consider the following examples in which one cannot be anaphoric to a noun (cf. Baker (1979)):

(2i) a. I met the member of Congress…
    b. * …and you met the one of the Society for Creative Anachronism.
    c. [NP the [N’ [N⁰ member] [PP of Congress]]]

2 Note that the precise labels of the constituents here are immaterial. If the structure is [DP this [NP red [NP ball]]], the conclusions reached in this paper would not be changed.
(2ii)  
  a.  I reached the conclusion that syntax is innate…
  b.  * …and you reached the one that learning is powerful.
  c.  [NP the [N’ [N⁰ conclusion] [CP that syntax is innate]]]

These contrast with cases in which what follows the head noun is an adjunct/modifier. Here, one can substitute for what appears to be only the head noun.

(2iii)  
  a.  I met the student from Peoria…
  b.  … and you met the one from Podunk.
  c.  [NP the [N’ [N⁰ student]] [PP from Peoria]]

(2iv)  
  a.  I met the student that Lily invited to the party
  b.  … and you met the one that Jack invited.
  c.  [NP the [N’ [N⁰ student]] [CP that Lily invited to the party]]

These cases differ with respect to the status of what follows the noun. In (2i) and (2ii) what follows the noun is a complement, but in (2iii) and (2iv) what follows the noun is a modifier. We can see that one can take a noun as its antecedent only when that noun does not take a complement. We represent this by saying that one must take N’ as its antecedent and that in cases in which there is no complement, the noun by itself is categorized as both N⁰ and N’. In other words, in cases like (1a), it must be the case that ball = N’, as in the structure in Figure 1. If it weren’t, we would have no way to distinguish this case from one in which one cannot substitute for a single word, as in (i) and (ii).

B. Pragmatics

In addition, when there is more than one N’ to choose from (as in (1b) above), adults prefer the N’ corresponding to the longer string (red ball). For example, in (1b) an adult (in the null context) would often assume that the ball Lily likes is red – that is, the referent of one is a ball that has the property red (cf. Akhtar et al. (2004)). This semantic consequence is the result of the syntactic preference for the larger N’ red ball. If the adult preferred the smaller N’ ball, the semantic consequence would be no preference for the referent of one to be red, but rather for it to have any property at all. Importantly, though, this preference is not categorical. It is straightforward to find cases where it is overridden, as in (3):

(3)  I like the yellow bottle but you like that one.

Here, it is quite easy to take one to refer to bottle and not yellow bottle.

C. Children’s Knowledge

But do children believe one can be anaphoric to an N⁰ string? If so, the semantic consequence would be readily apparent: the antecedent for one would never be phrasal, and hence the referent of one would be indifferent to properties mentioned by modifiers in the antecedent. LWF conducted an intermodal preferential looking paradigm experiment (Golinkoff et al., 1987; Spelke, 1979) to see if infants did, in fact, have a preference for the referent of one to have properties mentioned by the modifier in the antecedent (i.e., for a red bottle if a potential antecedent of one is red bottle).
Figure 10. LWF experiment. The infant is first shown a bottle of one color while several utterances of the form “Look! An adjective bottle.” are played simultaneously. Then, in the test stage, two bottles are shown – one of the adjective color and one of another color. The utterance “Do you see another one?” is played simultaneously and the infant’s looking preferences are recorded.

The 18-month olds demonstrated a significant preference for looking at the bottle that had the same property mentioned in the N’ string – e.g. the bottle that was red when the N’ string red bottle was a potential antecedent. LWF explained this behavior as a semantic consequence of the syntactic knowledge that one is anaphoric to the larger N’ string (red bottle). Since infants preferred the larger N’ string (as adults do) and this larger N’ string could not be classified as N⁰, LWF concluded that the 18-month olds have the syntactic knowledge that one is anaphoric to N’ strings in general.

D. Sparse Data

In order to determine whether children’s knowledge could have been acquired on the basis of the relevant forms and structures, LWF conducted a corpus analysis on child-directed speech. The important empirical question was how frequently data appeared in child-directed speech that signaled that one was anaphoric to N’ instead of N⁰. If the data were not frequent, learning this syntactic knowledge would be difficult. The distribution LWF found is displayed in table 1 below.

<table>
<thead>
<tr>
<th>Data Type</th>
<th># of data points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unambiguous</td>
<td>2</td>
</tr>
<tr>
<td>“I have a red ball, but Jack doesn’t have one.”</td>
<td>792</td>
</tr>
</tbody>
</table>

Total Data in Corpus | Total # with anaphoric one
54,800                  | 792

3 If the children had allowed one to be anaphoric to N⁰ (bottle), they would have had no preference for which bottle to look at.

4 The looking preferences recorded for this experiment were identical to those recorded for a later experiment that had the same set-up but used the utterance “Do you see another red bottle?” in the test phase. Thus, LWF concluded that 18-month olds did, in fact, interpret one as anaphoric to the string red bottle in the original experiment. See Lidz and Waxman (in prep.) for more empirical data supporting this conclusion.
Table 1. The distribution of utterances in the corpus examined by LWF.

<table>
<thead>
<tr>
<th>Type I Ambiguous</th>
<th>36</th>
</tr>
</thead>
<tbody>
<tr>
<td>“I have a red ball, and Jack has one, too.”</td>
<td></td>
</tr>
<tr>
<td>(Jack has a ball, but it does not have the property red.)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type II Ambiguous</th>
<th>750</th>
</tr>
</thead>
<tbody>
<tr>
<td>“I have a ball, and Jack has one, too.”</td>
<td></td>
</tr>
<tr>
<td>(Jack has a ball, and it has any number of properties.)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ungrammatical</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>“…you must be need one.” (Adam19.cha, line 940)</td>
<td></td>
</tr>
</tbody>
</table>

All data are defined by a pairing of utterance and environment. We now elaborate on the pairings for each data type. Unambiguous antecedent data have the following form:

(4) Unambiguous antecedent example
Utterance: “I have a red ball, but Jack doesn’t have one.”
Environment: Jack has a ball, but it does not have the property red.

Because Jack does indeed have a ball, the antecedent of one cannot be ball. However, Jack’s ball is not red, so the antecedent of one can be red ball. Since red ball can only be classified as N’, these data are unambiguous evidence that one can be anaphoric to N’.

An example of this type taken from the Adam corpus in CHILDES (MacWhinney, 19xx) is given here. (Adam40.cha, line 890)

(5) CHI: Do you have another flat tire?
   MOT: No. I don’t think I have one.

In this context, the mother had a tire, but not a flat tire, so the antecedent of one is unambiguously flat tire.

Type I ambiguous antecedent data have the following form:

(6a) Type I ambiguous antecedent example
Utterance: “I have a red ball, and Jack has one, too.”
Environment: Jack has a ball, and it has the property red.

(6b) Type I ambiguous antecedent example
Utterance: “I have a red ball, but Jack doesn’t have one.”
Environment: Jack does not have a ball.

For data of the form in (2a), Jack has a ball, so the antecedent of one could be ball. However, Jack also has a ball that is red, so the antecedent of one could be red ball. Because ball could be classified as either N’ or N⁰, these data are ambiguous between one anaphoric to N’ and one anaphoric to N⁰.

An example of this type taken from the Adam corpus in CHILDES (MacWhinney (2000)) is given here (Adam01.cha, line 291).

(7) MOT: That’s a big truck.
   MOT: There goes another one.

In this context, one could be taken to refer to either truck or big truck.
For data of the form in (6b), Jack does not have a ball – but it is unclear whether the ball he does not have has the property red. For this reason, the antecedent of one is again ambiguous between red ball and ball, and one could be classified as either N’ or N⁰. There were no examples in either Adam or Nina’s corpus of this form.

Type II ambiguous antecedent data have the following form:

(8a) Type II ambiguous antecedent example
Utterance: “I have a ball, and Jack has one, too.”
Environment: Jack has a ball, and it has various properties.

(8b) Type II ambiguous antecedent example
Utterance: “I have a ball, but Jack doesn’t have one.”
Environment: Jack does not have a ball.

For both forms of type II ambiguous data, the antecedent of one must be ball. However, since ball can be classified as either N’ or N⁰, such data is ambiguous with respect to what one is anaphoric to.

An example of this type taken from the Adam corpus of CHILDES (MacWhinney (2000)) is given here (Adam01.cha, line 566).

(9) CHI: my pillow my
MOT: That’s a good one to jump on.
Because there are no modifiers in the antecedent, my pillow, this data is uninformative about the structure of one.

There were no examples in either Adam or Nina’s corpus of the form (8b).

Ungrammatical data involve a use of anaphoric one that is not in the adult grammar, such as in (10):

(10) Ungrammatical antecedent example
Utterance: “…you must be need one.”

Since the utterance is already ungrammatical, it does not matter what environment it is paired with. Such data is noise in the input.

The vast majority of the anaphoric one input consists of type II ambiguous data (750 of 792, 94.7%). Type I ambiguous data makes up a much smaller portion (36 of 792, 4.5%). Ungrammatical data are quite rare (4 of 792, 0.5%), and unambiguous data rarer still (2 of 792, 0.25%). Since LWF considered unambiguous data as the only informative data, they concluded that such data seemed far too sparse to definitively signal to a learner that one is anaphoric to N’. Their experimental results, however, suggested that 18-month olds know that one is anaphoric to N’. LWF therefore claimed that such knowledge does not need to be learned. Instead, such knowledge would either be specified innately or the learner would have other innate biases that would allow this knowledge to be derived from the data available. One possibility (cf. Baker (1979), Hornstein & Lightfoot (1981)) would be that the child is constrained only to hypothesize phrasal antecedents for pronouns. Thus, once the child identified one as a pronominal form, the possibility that it was anaphoric to N° would simply never be considered, as it is not included in the hypothesis space.

4. Learning Anaphoric One
A. Suggestions for Learning that One is Anaphoric to N’

Two replies to LWF made suggestions for how this syntactic knowledge could be learned from the available data. Akhtar et al. (2004) noted that even if the percentage of unambiguous data is quite small, 18-month olds have still been exposed to an estimated 1,000,000 utterances; this should yield a larger number of unambiguous data than the LWF corpus analysis obtained. However, it is unlikely that this is a fair estimate of the amount of data that the child has been exposed to. This is because much of the first year of life is spent learning phonological and lexical properties of the language which would be prerequisites to learning syntax. To derive a fairer estimate of the amount of relevant data an 18-month old might have been exposed to, we assume that learning the syntactic and semantic properties of one can only commence once the child has some repertoire of syntactic categories. Thus, we posited that the learning period begins at 14 months because there is experimental data supporting infant recognition of the category Noun and the ability to distinguish it from other categories such as Adjective at this age (Booth & Waxman, 2003). If learners hear approximately 1,000,000 sentences from birth until 18 months, they should hear approximately 278,000 sentences between 14 months and 18 months. The adjusted expected distribution of anaphoric one data is displayed in table 2.

<table>
<thead>
<tr>
<th>Total Data before 18 months</th>
<th>Total # with anaphoric one</th>
</tr>
</thead>
<tbody>
<tr>
<td>~278,000</td>
<td>4017</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Type</th>
<th># of data points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unambiguous</td>
<td>10</td>
</tr>
<tr>
<td>“I have a red ball, but Jack doesn’t have one.” (Jack has a ball, but it does not have the property red.)</td>
<td>183</td>
</tr>
<tr>
<td>Type I Ambiguous</td>
<td>3805</td>
</tr>
<tr>
<td>“I have a ball, and Jack has one, too.” (Jack has a ball, and it has the property red.)</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 2. The expected distribution of utterances in the input to learners between 14 and 18 months.

Perhaps the most striking feature of this distribution is that there are still pitifully few unambiguous data points available. With only 10 chances to hear unambiguous data (on this estimate), a learner could well miss out due to fussiness, distraction, or other vagaries of toddler life.

R&G offer a solution: make use of the type I ambiguous data as well, which gives 183 additional data points (on this estimate). Using a Bayesian learning model that implements the size principle of Tenenbaum & Griffiths (2001), R&G demonstrate how a learner could use both unambiguous and type I ambiguous data to converge on the correct representation. We review their learning model in the next section.

B. A Regier & Gahl Bayesian Learner
The power of R&G’s model comes from using indirect evidence available in the type I ambiguous data. This is an attractive strategy, since there are nearly 20 times as many type I ambiguous data as there are unambiguous data (183 to 10). The indirect evidence itself is derived solely from the environment in which type I ambiguous data are uttered – specifically, by the learner examining the distribution of the referents of one. For example, suppose the learner hears type I ambiguous data such as the example in (6a) (repeated below as (11)):

(11) Type I ambiguous
Utterance: “I have a red ball, and Jack has one, too.”
Environment: Jack has a ball, and it has the property red.

Since the adult preference is to choose the larger N’ as the antecedent, the antecedent of one will nearly always be red ball and the referent of the NP containing one will have the property red. The learner is able to observe the simultaneous presence of the N’ as potential antecedent (red ball) and a referent in the world of one with the property mentioned in the N’ (red). We note that this observation requires the learner to have a very abstract notion of what to generalize over. It is insufficient to generalize over a single property such as “red” or “behind his back”; instead, the learner must generalize over “property mentioned in the N’ antecedent”.

Now, the connection between the N’ antecedent and a referent with the property mentioned in the N’ will be true for some portion of the type I ambiguous data. Crucially, for R&G’s model, it is never true that the referent of one definitively lacks the property mentioned in the N’ antecedent (i.e. the referent of one is definitively not red when the antecedent is red ball). The Bayesian learner is very sensitive to this fact in the following way:

(12) Bayesian Learner Logic
(a) For type I ambiguous data, suppose that the referent of one could have any property, and not necessarily have the property mentioned in the larger N’ antecedent. Suppose also that the set of potential referents for an utterance like (11) is represented in figure 11.

---

5 This reasoning will not work for type I ambiguous data of the form in (2b): “I have a red ball, but Jack doesn’t have one”, where Jack does not have a ball. This is because the learner cannot tell whether or not the ball Jack doesn’t have has the property red. These data are therefore not useful as indirect evidence. Such data did not occur in the Adam and Nina corpora from which our estimates are derived.
Figure 11. The set of potential referents for *one* in the world when an utterance such as “I have a red ball, and Jack has one, too” is heard.

(b) The actual distribution of referents observed by the learner, however, is only a particular subset of all the possible referents.

![Diagram of referents](image)

“…red ball…one…”

Figure 12. The observed set of referents for *one* when an utterance such as “I have a red ball, and Jack has one, too” is heard.

(c) It is highly unlikely that the referent of *one* is only ever a member of the subset if the referent could be any member of the superset. The Bayesian learner will therefore consider a restriction to the subset to be more and more probable as time goes on. This is the size principle of Tenenbaum & Griffiths (2001): if there is a choice between a subset and the superset, and only data from the subset is seen, the learner will be most confident that the subset is the correct hypothesis. Thus, the learner uses the lack of data for the superset as indirect evidence that the subset is correct.

(d) Once the learner is biased to believe that there is a restriction to the subset of referents described by the property mentioned in the N’ (*red* in *red ball*), the learner then assumes that the correct antecedent is, in fact, the larger N’. Since the larger N’ cannot be classified as N^0, the learner then knows that *one* always has an N’ antecedent.

(e) For the LWF experiment, a Bayesian learner would have converged on the subset of red bottles as the potential referents of *one* in the test utterance. Given a choice between a red and a non-red bottle, the Bayesian learner therefore looks at the bottle that belongs to the correct subset: the red bottle.

A great strength of the R&G model is that the bias to choose the subset, given indirect evidence, does not need to be explicitly assumed. Instead, it falls out neatly from the mathematical implementation of the Bayesian learning procedure itself – the size principle of Tenenbaum & Griffiths (2001).

---

6 R&G’s model demonstrates how this could happen after very few type I ambiguous data.
However, the R&G model still harbors two implicit biases about domain specific data filters on the learner’s intake. The first bias is that only unambiguous and type I ambiguous data are used; type II ambiguous data are ignored even though they may also provide indirect evidence to a Bayesian learner. The second bias is that only semantic data (the referents of one) are used to converge on the syntactic knowledge of what one is anaphoric to; syntactic data are ignored.

In the remaining sections of the paper, we will see that stripping away these two biases leads to markedly different results from those of R&G. Specifically, once we remove these two biases, we discover that a Bayesian learner cannot learn that one is anaphoric to N’ and will not behave as the 18-month olds did in the LWF experiment. The benefit that comes from using indirect negative evidence to shift the majority of the probability to the subset in the hypothesis space is tempered by the link between the two levels of representation. The indirect learning that characterizes the Bayesian approach leads to conflicting predictions in the syntactic and semantic domains. Thus, the existence of multiple levels of representation in language reduces the efficacy of this kind of learner.

5. An Equal-Opportunity Bayesian Learner

We have named our learning model the Equal-Opportunity (EO) Bayesian Learner since it removes the two implicit biases of R&G’s Bayesian learner and therefore gives equal treatment to all data. First, it denies privileged status to a subset of the data and instead uses all the data available: unambiguous, type I ambiguous, and type II ambiguous. Second, it denies privileged status to semantic data – syntactic and semantic data are both used to shift probability between opposing hypotheses. 7 There is an intuitive logic to using both types of data, since one should presumably use syntactic data (among other kinds of data) to converge on syntactic knowledge. 8 This syntactic knowledge then has semantic consequences, which are displayed in the LWF experiment. If a Bayesian learning procedure, unconstrained by domain specific filters, is to be an effective domain general learning solution, it should correctly acquire knowledge that spans domains such as syntax and semantics as well as knowledge contained completely within these domains.

A. The Hypothesis Space

The hypothesis space is defined for both the syntactic and semantic domains. In both domains, there are two hypotheses to choose from. Each hypothesis makes predictions about the data that will be encountered and, consequently, the elements that will be analyzable under that

7 R&G highlight the benefits of using a domain general learning procedure to solve the problem of anaphoric one. The appeal of a domain general learning procedure without domain specific filters resides in the lack of biases found inside the learner. Thus, the EO Bayesian learner was designed in this spirit. No data is excluded from consideration.
8 Note that even if we believed the knowledge about one was stated purely in semantic terms, the data that any grammar predicts will include both syntactic data (i.e. what the linguistic antecedent for one is) and semantic data (what the referent of one is). So, excluding either kind of data is an arbitrary restriction on the learner that would need to be justified. For this reason, the hypothesis to include both syntactic and semantic data does not rely on a particular specification of knowledge about anaphoric one.].
hypothesis. So, for each case, the elements analyzable by one hypothesis are a subset of the elements analyzable by the other. For syntax, the hypotheses under consideration are that one is anaphoric to strings that are classified as $N^0$ or one is anaphoric to strings that are classified as $N'$. Every string in $N^0$ can also be classified as $N'$ but there are strings in $N'$ that cannot be classified as $N^0$. Therefore, the strings that comprise the $N^0$ set are a subset of the strings that comprise the $N'$ set.

![Diagram of syntax hypothesis space, $N^0$ vs. $N'$.]

Figure 13. The syntax hypothesis space, $N^0$ vs. $N'$. All the elements in the sets are strings that are possible antecedents of one. Every string classified as $N^0$ can be classified as $N'$. In addition, there are strings in $N'$ that are not in $N^0$, and so the $N^0$ set is a subset of the $N'$ set.

For semantics, the referents of one could have the restriction that they must have the property named by the modifier; alternatively, the referents of one could have no restriction on what property they have. Since the modifier is linguistically not part of the $N^0$ (recall figure 1) and instead is part of the $N'$ phrase, we will refer to the property named by the modifier as the $N'$-property. We will refer to referents with no restrictions as being any-property referents, since these referents can have any property.

Just as in the syntactic domain, the semantics hypothesis space consists of two hypotheses. And, just as in the syntactic domain, the elements predicted by one hypothesis are a subset of the elements predicted by the other (see figure 14). Every referent that has the $N'$-property (say red for red ball) is a member of the $N'$-property set. By definition, every member of the $N'$-property set is also a member of the any-property set, since the $N'$-property is one of the properties available for objects to have. However, there are members of the any-property set (say green balls for the linguistic antecedent red ball) that do not have the $N'$-property (red). So, since all the members of the $N'$-property set are members of the any-property set, the $N'$-property set is a subset of the any-property set. Moreover, some members of the any-property set are not members of the $N'$-property set. So, the any-property set is a superset of the $N'$-property set in the semantic domain.
Figure 14. The semantic hypothesis space, N'-property vs. any-property. Any-property is a superset of N'-property. Note that in order to define the sets (N'-property vs. any-property), the utterance must be used to determine the salient property that the referent of the antecedent has. The salient property can be determined from the linguistic antecedent of one.

The difficulty for a Bayesian learner becomes apparent when we examine how the two prediction spaces defined by the hypotheses are connected. Specifically, the subset in the syntax is linked to the superset in the semantics, and the subset in the semantics is linked to the superset in the syntax, as shown in Figure 15. This is due to the compositional property of syntactic representations: larger syntactic constituents have meanings that are restrictions on the meanings (and so the referents) of their constituent subparts. The strings in the superset of the syntax (e.g. red ball) designate a subset of referents in the semantics (e.g. the red balls), while the strings in the subset of the syntax (e.g. ball) designate the superset of referents in the semantics (e.g. all balls).

Figure 15. The superset of the syntax is linked to the subset in the semantics, while the subset of the syntax is linked to the superset in the semantics.

Because the syntactic and semantic representations are linked in this fashion, a Bayesian learner that relies on indirect evidence to shift probability towards the subset will receive conflicting data when combining the information from syntax and semantics. The learner must somehow reconcile this data to decide how much probability to give each hypothesis in each domain. Specifically, if the Bayesian learner shifts probability on the syntactic subset, it is forced to shift the same amount of probability to the semantic superset – the wrong answer for English anaphoric one. Yet, the learner shouldn’t ignore syntactic data since anaphoric one has a representation at the syntactic level. Thus, the problem for a Bayesian learner that uses all available data (syntactic and semantic) is particularly severe here.

It is important to recognize that the problem of linked hypothesis spaces extends far beyond the particular case of anaphoric one. Because syntactic structures are semantically...
compositional, this problem will persist across the acquisition of any aspect of the grammar that depends on the link between syntax and semantics.

B. EO Bayesian Learning

The model we are about to describe uses Bayesian reasoning to update the learner’s confidence in each of two alternative hypotheses. We detail the learning process independently for each of the two domains (syntax and semantics) that are relevant for determining the appropriate structure of anaphoric one. We then describe how we implement the updating algorithm given that these two domains are linked.

1. Updating the Syntax Hypotheses

Recall that there are two hypotheses under consideration in the syntactic domain: the N’ hypothesis and the N⁰ hypothesis. The N’ hypothesis takes the antecedent of one to be a constituent of the category N’; the N⁰ hypothesis takes the antecedent of one to be a constituent of the category N⁰.

We represent the probability that the N’ hypothesis is correct with p_{N’}. Because there are only two hypotheses in the hypothesis space, and because probabilities range from 0 to 1, the probability that the N⁰ hypothesis is correct is 1 – p_{N’}. We set the initial value of p_{N’} before the learner has observed any data to 0.5 as an instantiation of the assumption that both hypotheses are equiprobable.

The update function requires a single parameter t, which represents the total amount of data expected during the learning period. While it might seem unrealistic to expect a real learner to already know how much data will be available, t can easily be mapped to the amount of change the real learner’s brain is allowed to undergo before settling into the final state. In the simulated learner here, we simply quantify that amount of change as the total estimated amount of data available during the learning period (4017 data points). The model uses t to determine how much shifting should be done, given a single piece of data. If t is small, only a small number of changes are allowed and each piece of data shifts the probability quite a lot; conversely, if t is large, a large number of changes are allowed and each piece of data shifts the probability a smaller amount.

The exact update functions for p_{N’} depend on the data type observed – unambiguous, type I ambiguous, or type II ambiguous. Unambiguous and type I ambiguous data cause the learner to use the function in (13a), which is essentially an implementation of the indirect negative evidence update function used by the R&G model. Type II ambiguous data, which was not considered by R&G, cause the learner to use the function in (13b). These functions differ because of the linguistic characteristics of the input, not because they differ in their use of Bayesian reasoning.

---

9 The implementation we use differs from the R&G Bayesian learner by being more conservative about updating the probabilities of the competing hypotheses. The R&G learner is quite liberal about shifting probability to the superset hypothesis – in fact, a single piece of data for the superset is enough to shift all the probability to that hypothesis. See Appendix B for details.

10 Of course, the value of t is essentially arbitrary, but in order to model this learning process, it needed to be estimated. While the estimate presented here seems fair, we present a range of possible t-values in Appendix C. What we see there is that the size of t does not influence the final probability of the correct grammar.

11 For details of how these functions are derived, see appendix A.
The update function for unambiguous and type I ambiguous data (which comprise 193 of the data points) depends only on the prior probability that N’ is the correct hypotheses \( \begin{equation} p_{N'} = \frac{p_{N'} \text{old} \times t + 1}{t + 1} \end{equation} \) and \( t \). An intuitive interpretation of the update function is that the numerator represents the learner’s confidence that the observed utterance-world pairing \( u \) is a result of the N’ hypothesis being correct; the denominator represents the total data observed so far. Thus, 1 is added to the numerator because the learner is fully confident that \( u \) indicates that the N’ hypothesis is correct; and, 1 is added to the denominator because a single data point has been observed.

Unambiguous data signal that the N’ hypothesis is correct (in that only the N’ hypothesis could have produced \( u \)) and so should be treated with full confidence by the learner. The type I ambiguous data, in contrast, do not indicate that only the N’ hypothesis could have produced \( u \) – these data are ambiguous between the N’ and N’ hypotheses. Thus, a smaller value should be added to the numerator for such data to indicate less than full confidence that only the N’ hypothesis could have produced \( u \). However, we will allow the Bayesian learner to treat the type I ambiguous data with full confidence in the N’ hypothesis. We make this allowance for two reasons. First, we know of no way to reasonably estimate how much confidence should be associated with a type I ambiguous data point. Second, this allowance is most generous towards the Bayesian learner because it overestimates the confidence the learner has in the N’ hypothesis. If we were less generous and lessened the confidence in the type I ambiguous data, the probability of N’ would only be lower than what we present here. As we will see below, even with this generous estimate, the learner will fail to assign sufficient probability to the N’ hypothesis.

The update function for type II ambiguous data (which comprise 3805 of the data points) depends on the prior probability that N’ is the correct hypotheses \( \begin{equation} p_{N'} = \frac{p_{N'} \text{old} \times t + p_{N'\mid a}}{t + 1} \end{equation} \) and \( t \). The intuitive interpretation for this function remains the same as the interpretation for the function in (13a): the numerator represents the learner’s confidence that the observed ambiguous utterance-world pairing \( a \) is a result of the N’ hypothesis being correct; the denominator represents the total data observed so far. Thus, a value less than 1 \( (p_{N'\mid a}) \) is added to the numerator because the learner is only partially confident that ambiguous data point \( a \) indicates the N’ hypothesis is correct; and, 1 is added to the denominator because a single data point has been observed. The partial confidence value \( p_{N'\mid a} \) depends on the likelihood that the
utterance in a, which has only a noun string as the antecedent of one (ex: “…ball…one…”), would be produced if any N’ string could have been chosen from the set of N’ strings. We refer to this as $p_n$ from N’. See appendix A for details about how we derive this value.

The likelihood value $p_n$ from N’ is what allows the learner to retrieve information from the type II ambiguous data. Remember that noun-only strings are compatible with either the N$^0$ hypothesis or the N’ hypothesis. But, the Bayesian learner uses the size principle to bias the learner towards the smallest set that will cover the available data. Since the N$^0$ hypothesis consists entirely of noun-only strings, such strings will bias the Bayesian learner towards this hypothesis. The more unbalanced the ratio of noun-only strings to other strings in the N’ set, the stronger the effect of the size principle will be. Example (14) displays how much biasing occurs after a single piece of type II ambiguous data.

(14) Updated $p_{N'}$ after a single type II ambiguous data point a
Let $p_{N'} = 0.5$, $p_n$ from N’ = 0.25, and $t = 4017$.
Updated $p_{N'} = .499925$ (a very slight bias for N$^0$)

While the amount of bias towards the N$^0$ hypothesis is quite small, keep in mind that the majority of the data is type II ambiguous and so these small biases will add up over time.

2. Updating the Semantics Hypotheses
Recall that there are two hypotheses under consideration in the semantic domain: the N’-property hypothesis and the any-property hypothesis. The N’-property hypothesis requires the referent of one to have the property mentioned in the N’ antecedent; the any-property hypothesis allows the referent of one to have any property. In this case, it’s the N’-property hypothesis that represents the subset hypothesis. Thus, as above, the size principle will favor this hypothesis for any data that is compatible with both hypotheses.

We represent the probability that the N’-property hypothesis is correct with $p_{N'}$-prop. Because there are again only two hypotheses in the hypothesis space, the probability that the any-property hypothesis is correct is 1 - $p_{N'}$-prop. We set the initial value of $p_{N'}$-prop before the learner has observed any data to 0.5 as an instantiation of the assumption that both hypotheses are equiprobable.

The update function requires two parameters: $t$ and $c$. As before, $t$ represents the total amount of data expected during the learning period and is instantiated in our model as 4017, the estimated amount of data available during the learning period. The parameter $c$ represents the number of properties (or categories of referents) in the world that the learner is aware of (e.g. red, striped, behind his back, etc.).

For the semantic domain, the data are divided according to how the properties of the referent of one compare to the salient property in the N’ antecedent. The data types, representing the utterance-world pairings, are same-property, different-property, and unknown-property.

Same-property examples are those in which the antecedent of one mentions some property and the referent of one also has that property. Some of the data analyzed as type I ambiguous in the syntactic domain are same-property data. There are 183 or less data points of this form (because some portion of type I ambiguous are unknown-property data points).

(15a) Example of same-property data (syntax: type I ambiguous)
Utterance: “I have a red ball, and Jack has one, too.”
World: Jack has a red ball.
The referent of *one* (the ball that Jack has) has the same property mentioned in the N’ antecedent (red).

The data analyzed as unambiguous in the syntactic domain are also same-property data in the semantic domain. There are 10 data points of this form. Because these data necessarily include negation, seeing why they are same-property data is a bit complicated. Consider the example in (15b).

(15b) Example of same-property data (syntax: unambiguous)

Utterance: “I have a red ball, but Jack doesn’t have one.”

World: Jack has a non-red ball.

The speaker in this situation is asserting the absence of a red ball. The referent of *one* is a red ball that is not present in the situation. Thus, the meaning of *one* includes the property mentioned in the antecedent.

A portion of the data analyzed as type II ambiguous in the syntactic domain are also same-property data in the semantic domain.

(15c) Example of same-property data (syntax: type II ambiguous)

Utterance: “I have a ball, and Jack has one, too.”

World: Jack has a ball with the same salient property as the antecedent referent (i.e. the referent of *ball* is a red ball, the referent of *one* is a red ball, and the learner chooses red as the salient property).

Even though no property is mentioned in the N’, the referent of the antecedent has several properties. Since the N’ property is unspecified, the learner chooses at random (or perhaps derives from context) which property of the antecedent referent is salient. Given that there are *c* properties in the learner’s world, let there be a 1/*c* chance that the referent of *one* coincidentally has the same salient property as the antecedent referent. There are then 3805*1/*c* or less data points of this form (because a portion of type II ambiguous data are different-property and unknown-property data points).

A different-property example is given in (16). A portion of the data analyzed as type II ambiguous in the syntactic domain is different-property.

(16) Example of different-property data (syntax: type II ambiguous)

Utterance: “I have a ball, and Jack has one, too.”

World: Jack has a ball without the same salient property as the antecedent referent. (i.e., the referent of *ball* is a red ball, the referent of *one* is a striped, green ball, and the learner chooses green as the salient property).

---

12 Note that this assumes properties are mutually exclusive (as if they were colors and a referent could only have one color). There could be a higher chance of the referent of *one* having the same salient property if a referent could have 2 or more properties (such as being red and striped, etc.). The mutual exclusivity assumption allows us to make a conservative estimate of the number of properties under consideration at any one time.
As before, even though no property is mentioned in the N’, the referent of the antecedent has several properties. Since the N’ property is unspecified, the learner chooses at random (or derives from context) which property of the antecedent referent is salient. Given that there are $c$ properties in the learner’s world, let there be a $(c-1)/c$ chance that the referent of one coincidentally does not have the same salient property as the antecedent referent. There are then $3805*(c-1)/c$ or less data points of this form (because a portion of type II ambiguous are same-property and unknown-property data points).

Finally, we come to the unknown-property data, with examples given in (17).

(17a) Example of unknown-property data (syntax: type I ambiguous)

   **Utterance:** “I have a red ball, but Jack doesn’t have one.”
   **World:** Jack has no ball.

(17b) Example of unknown-property data (syntax: type II ambiguous)

   **Utterance:** “I have a ball, but Jack doesn’t have one.”
   **World:** Jack has no ball.

In both the examples in (17), the speaker is asserting the absence of a ball. The referent of one is a ball, with some unknown properties, that is not present in the situation. Thus, the meaning of one may or may not include the property chosen as salient by the learner.

A portion of type I ambiguous and type II ambiguous data consists of unknown-property data. Such data cannot be used for updating the probabilities of the opposing semantic hypotheses. However, we will be generous and allow R&G’s assumption to hold true: none of the type I ambiguous data are of this form. In addition, we will assume none of the type II ambiguous data are of this form. Therefore, we will allow all type I ambiguous data to be of the form in (15a), which is an example of same-property data. This gives an overestimation of $p_{N’-prop}$, which is the subset in the semantic hypothesis space. Consequently, this will bias the learner towards the superset in the syntactic hypothesis space, N’. Thus, the model here will overestimate the amount of probability the learner will assign to the correct hypothesis for the structure and meaning of anaphoric one. We also allow the type II ambiguous data to be proportionally distributed between same-property $(1/c)$ and different-property $(c-1)/c$ utterance-world pairings.

Table 3 represents the expected distribution of data for updating the semantic hypotheses in our model.

<table>
<thead>
<tr>
<th>Total Data before 18 months</th>
<th>Total # with anaphoric one</th>
</tr>
</thead>
<tbody>
<tr>
<td>~278,000</td>
<td>4017</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Type</th>
<th># of data points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same-Property</td>
<td>$10 + 183 + 3805*1/c$</td>
</tr>
<tr>
<td>“I have a red ball, and Jack has one, too.” (Jack has a red ball.)</td>
<td></td>
</tr>
<tr>
<td>“I have a red ball, but Jack doesn’t have one.” (Jack has a non-red ball.)</td>
<td></td>
</tr>
<tr>
<td>“I have a ball, and Jack has one, too.” (Jack has a ball with the salient property that the antecedent referent has.)</td>
<td></td>
</tr>
<tr>
<td>Different Property</td>
<td>$3805*(c-1)/c$</td>
</tr>
</tbody>
</table>
**Table 3.** The expected distribution of utterances in the input to the Bayesian learner for updating the semantics hypotheses.

The exact update functions for \( p_{N'}\text{-prop} \) depend on the data type observed. The semantic update functions are similar to their syntactic update counterparts. Same-property data cause the learner to use the function in (18a) while different-property data cause the learner to use the function in (18b).^{13}

(18a) Update function for same-property data

\[
p_{N'}\text{-prop} = \frac{p_{N'}\text{-prop} \cdot \text{old} \cdot t + p_{N'}\text{-prop} \cdot | s}}{t + 1}
\]

(18b) Update function for different-property data

\[
p_{N'}\text{-prop} = \frac{p_{N'}\text{-prop} \cdot \text{old} \cdot t + 0}{t + 1}
\]

The update function for both same-property data and different-property data depends only on the prior probability that the \( N' \)-property hypothesis is correct (\( p_{N'}\text{-prop}\cdot\text{old} \)) and \( t \). An intuitive interpretation of the update functions is that the numerator represents the learner’s confidence that the observed utterance-world pairing \( s \) is a result of the \( N' \)-property hypothesis being correct; the denominator represents the total data observed so far. For the different-property data, 0 is added to the numerator because the learner has no confidence that the \( N' \)-property hypothesis is correct. The same-property data behave as the type II ambiguous data in the syntactic domain: a value less than 1 (\( p_{N'\text{-prop} \cdot | s} \)) is added to the numerator since the same-property data point is consistent with both the \( N' \)-property hypothesis and the any-property hypothesis (the \( N' \)-property is, by definition, a member of the any-property set). The learner is therefore not fully confident that \( s \) indicates the \( N' \)-property hypothesis. The partial confidence value \( p_{N'\text{-prop} \cdot | s} \) depends on the likelihood that the referent of \( \text{one} \) in \( s \), which has the same property mentioned in the \( N' \) antecedent (ex: “…red ball…one…”, referent has property ‘red’), would have been chosen if a referent with any property could have been chosen from the set of potential referents. This value is \( 1/c \), given \( c \) properties in the world. See appendix A for details about how we derive the partial confidence value \( p_{N'\text{-prop} \cdot | s} \). As for the denominator, we add 1 for both the same-property data point and the different-property data points because a single data point has been observed.

---

^{13} For details of how these functions are derived, see appendix A.
3. The Updating Algorithm for Linked Domains

Recall that there is an inherent connection between the syntax and the semantics. In particular, the subset hypothesis in the syntax corresponds to the superset hypothesis in the semantics, and vice versa. That is, the N' hypothesis in the syntax, which represents the superset in this domain, is connected to the N'-property hypothesis in the semantics, which represents the subset in that domain. Similarly, the N^0 hypothesis in the syntax, which represents the subset in this domain, is connected to the any-property hypothesis in the semantics, which represents the superset in that domain. See figure {subset-superset figure above} for a reminder. Given this arrangement of hypothesis spaces, any piece of data impacting a hypothesis in one domain should impact the corresponding hypothesis in the other domain by the same amount. We now provide a description of how we model this process.

First, suppose the learner receives a data point. This data point can be analyzed in either domain. So, the learner chooses which one to analyze it in first. Then, the update functions described above are employed to determine the amount the probability should be shifted within that domain. Next, the probability is shifted in the other domain by the same amount. See figure 16, which shows the learner analyzing the data in syntax and updating both syntax and semantics. Now, the learner analyzes the data point in the other domain, applies the update functions described above to determine the amount the probability should be shifted within this domain. Next, the probability is shifted in the other domain by the same amount. See figure 17, which shows the learner analyzing the data in the semantics and updating both semantics and syntax.
Figure 16. The learner encounters a data point (a) and analyzes it first in the syntactic domain (b), and then updates the probability of the syntax hypotheses (c) and the probability of the linked semantics hypotheses (d).
Figure 17. After analyzing the data point in the syntax domain and updating the probabilities across the domains, the learner then starts at the state in (a) and analyzes the data point in the semantics domain (b). Then, the learner updates the probability of the semantics hypotheses (c) and the probability of the linked syntax hypotheses (d).

C. What Good Learning Would Look Like

In the model, the learner initially assigns equal probability to the two hypotheses in each of the two domains. The probability of choosing the correct grammar is the product of the probability of choosing the correct hypothesis in the syntax and that of choosing the correct hypothesis in the semantics: \(0.500 \times 0.500 = 0.250\). Given that the end state should be a probability of 1 (or nearly 1), a good learning algorithm should have a trajectory like that illustrated in figure 18. In other words, the learner should steadily increase the probability of choosing the correct grammar.

![Idealized trajectory of probability of correct grammar for anaphoric one](image)

Figure 18. The idealized trajectory of the probability of the correct grammar for anaphoric one as a function of the data points encountered by the learner.

D. Simulating an EO Bayesian Learner

Now that we have established how an EO Bayesian Learner learns and what the ideal learning outcome would be, we can simulate learning over our estimate of the set of data that 18-month olds have been exposed to. Each data point is analyzed in both the syntax and semantics domains; and, each data point is classified for both syntax (unambiguous, type I ambiguous, or type II ambiguous) and semantics (same-property, different-property). The final probabilities in each domain, which will be the same, because they are updated in lock step, are then used in two ways. First, we use the final probabilities to estimate the probability of the EO Bayesian learner converging on the correct grammar of anaphoric one. This probability corresponds to the likelihood of the learner accessing the correct grammar when encountering an instance of anaphoric one. Second, we use these probabilities to predict the EO Bayesian Learner’s behavior in the LWF experiment.

1. Syntax

The probability \(p_N\) is updated as each data point is observed. The model requires a value for \(p_N\) from \(N\), the probability of producing a noun-only string from the N’ string set. This requires that we determine how many strings are in the N’ set. There are two ways of doing this. First,
we could allow a string to consist of individual vocabulary items (“bottle”, “ball”, “ball behind his back”, etc.). Alternatively, we could allow a string to consist of individual categories (Noun, Noun PrepositionalPhrase, etc.). Recall that as $p_{n^\prime}$ increases, the bias towards the $N^\prime$ hypothesis increases. Therefore, to be generous and maximize the model’s estimate of $p_{N^\prime}$, we minimize the ratio of $N^\prime$ strings to $N^0$ strings in $N^\prime$ set, and allow the strings in the $N^\prime$ set to consist of individual categories instead of vocabulary items. The number of categories is necessarily smaller than the number of vocabulary items in those categories, and so this yields a larger value for $p_{n^\prime}$.

Let the set of strings in $N^\prime = \{\text{Noun, Adjective Noun, Noun PrepositionalPhrase, Adjective Noun PrepositionalPhrase}\}$. The probability of producing a Noun string from this $N^\prime$ string set is $1/4$ or 0.25. We can now look at the semantic domain.

2. Semantics
The probability $p_{N^\prime \text{-prop}}$ is updated as each data point is observed. The model requires a value for $c$, the number of properties in the learner’s world. Recall that as $c$ decreases, the number of type II ambiguous data (as analyzed by the syntax) that are same-property data points increases. Therefore, to be generous and maximize $p_{N^\prime \text{-prop}}$, we allow $c$ to be 5, a very conservative estimate.

3. Crosstalk Updating
Recall that the update algorithm analyzes each data point in two domains and shifts the probability between the opposing hypotheses within each domain and across domains accordingly. As you can see in figure 19, the learning trajectory as a function of the amount of data seen does not match our ideal learning outcome. In fact, as the learner encounters more data, the probability of the correct grammar steadily drops to a final value of 0.0361. This final value represents the product of the probability of the correct syntactic hypothesis ($p_{N^\prime}$) and that of the correct semantic hypothesis ($p_{N^\prime \text{-prop}}$); both values are 0.190 (1000 simulations, sd= 0.00382). Thus, based on the data observed, the learner is extremely unlikely to access the correct grammar (i.e., that one is anaphoric to strings described by $N^\prime$, and that referent of one must have the $N^\prime$ property).

---

\(^{14}\) This is still a conservative estimate – there are likely to be additional category strings in $N^\prime$, such as Adjective Adjective Noun, which would again lower $p_{n^\prime}$ from $N^\prime$.

\(^{15}\) The properties could be described by the set {red, behind his back, little, purple, nice}, for instance.

\(^{16}\) Note that this value is obtained using the procedure in which the learner chooses at random whether to analyze the data point in the syntax first or in the semantics first. The same value is obtained if the learner always analyzes the data point in the syntax first and if the learner always analyzes the data point in the semantics first.
6. The Outcome of an EO Bayesian Learner

To summarize, even with conservative estimates of various parameters, the EO Bayesian learner is heavily biased against the correct grammar. In fact, the probability of converging on the correct grammatical representation of anaphoric *one* is quite small (0.0361). In short, there is less than a one in twenty-five chance of an EO Bayesian learner converging on the correct grammar for anaphoric *one*.

This result is strikingly different from that reported in R&G. What is the source of this difference? Recall that R&G’s model made use of only a subset of the available data and gave priority to semantic data over syntactic data. However, if a Bayesian learner is unconstrained in its data intake, then we would expect that it does not favor one type of data over any other. Favoring one type of data over another represents a domain specific filter. Our EO Bayesian model, on the other hand, lacks any domain specific filter on data intake. It uses all the available data (unambiguous, type I ambiguous, and type II ambiguous) and treats syntactic and semantic data on a par as equally relevant to the learner. As we can see, such an unconstrained domain general learning procedure on its own fails to acquire the correct grammar of anaphoric *one*.

This failure is especially striking because of how generous we were regarding the data available to the EO Bayesian learner and how the learner interpreted that data. In section XX, we highlight where we were generous and see that revoking that generosity only pushes the final probability of choosing the correct grammar closer to zero.

Before turning to alternative treatments of the input data, however, we can also examine how the EO Bayesian learner would perform in an experimental set up like the one in LWF. We do this because it is possible that a learner could display behavior consistent with the correct grammar without actually having the correct grammar. In the next section, we explain how this could occur. But we also show that our EO Bayesian learner does not display the behavior exhibited by human learners. Thus, we conclude that unconstrained Bayesian learning by itself is not sufficient to model human learning or behavior in this domain.

A. The EO Bayesian Learner in the LWF experiment

Recall that a learner in the LWF experiment must make a choice between a red and a non-red bottle. First, the learner is shown a red bottle paired with the utterance “Look! A red
bottle.”; then, a red bottle and non-red bottle are shown and paired with the utterance “Do you see another one?” The learner must then choose whether to look at the red bottle or the non-red bottle. Figure 20 displays the decisions the EO learner could make and what bottle this learner would look at after each decision. The total probability of looking at the red bottle is made by summing the probabilities of all the decisions that lead to the learner looking at the red bottle.

![Decision Tree](image)

Figure 20. The decision tree for determining the probability that the EO Bayesian learner will look at the red bottle in the LWF experiment. The total probability of looking at the red bottle is calculated by summing all the probabilities of the decisions that lead to looking at the red bottle.

Summing the probabilities of looking at the red bottle using the values $p_N = 0.190$ and $p_{N\cdot\text{prop}} = 0.190$ yields a probability of 0.518 (standard deviation = .001). This is only very slightly above chance behavior for looking at the red bottle. We now compare this probability against the behavior exhibited by human learners.

Consider the graph in Figure 21, taken from LWF. We can see in the control condition that infants have a baseline preference for the novel object, leading to their looking at the red bottle 42% of the time. Because the EO Bayesian learner has a baseline preference of 50%
(which happens when $p_{N'}=0$)$^{17}$, we cannot directly compare the looking preference probabilities derived from our model against the looking time proportions in the behavioral experiments.

![Figure 21](image-url) Mean looking time (in seconds) to the test images in each condition, taken with permission from LWF. The familiar object is a red bottle while the novel object is a non-red bottle.

We can, however, compare the amount by which the model deviates from its baseline against the amount by which the human learner deviates from its baseline. As we saw above, the EO Bayesian learner chooses the red bottle 51.8% of the time, given the experimental stimulus. This is a change of less than 2%, or about 1/25 of the baseline. In contrast, the human learners chose the red bottle 57% of the time in the experimental condition, which is a change of roughly 15%, which is over 1/3 of the baseline. The amount of change for the humans is substantially greater than the amount of change for the EO Bayesian learning model, suggesting that the EO Bayesian learning model is a poor representation of human learning in this domain.

B. Generosity to the Bayesian Learner

We have now seen two measures of the inadequacy of the EO Bayesian learner as a model of human syntactic and semantic acquisition in this domain. As noted above, however, there were several places in the construction of the model where we biased the learner towards the correct grammar. We did this in two ways. First, we gave a generous interpretation of the available data by providing a liberal estimate of the amount of informative data in the environment. Second, we made conservative assumptions about the learner’s understanding of the environment. Even in the face of this generosity, the EO Bayesian learner failed.

In the first case, we were unable to determine a fair estimate of the amount of informative data in the environment. Consequently, we maximized the size of this data set in order to get an upper bound on the probability of converging on the correct grammar. In what follows, we leave this assumption as is. In the second case, however, we show several ways in which we can relax the conservative assumptions about the learner’s understanding of the environment to make these assumptions more realistic. As we will see, the results reported above represent an upper bound on the probability of converging on the correct grammar. Changing the relevant assumptions only decreases this probability further.

$^{17}$ Recall the first branch of the decision tree in figure 20. If $p_{N'}=0$, then the learner assumes the antecedent is bottle, and hence is at chance for looking at the red bottle.
1. Generous with the value of $c$ and $p_n$ from $N'$

The first conservative assumption we will examine concerns the value of $c$, which represents the total number of properties the learner is aware of in the world. We used a conservative estimate of 5, but data from the MacArthur CDI (Dale & Fenson, 1996) suggest that 14-16 months olds know at least 49 adjectives. Therefore, an 18-month old learner should be aware of at least 49 properties in the world.\(^1\) Recall that the larger the value of $c$, the fewer type II ambiguous data will coincidentally be same-property data points – and the more the learner will be biased towards the any-property (superset) hypothesis in the semantic domain.

We were also generous regarding the value of $p_n$ from $N'$, which is the probability of observing a noun-only string from all the $N'$ strings. We previously described the elements of the $N'$ string set as category strings, such as Noun and Adjective Noun. However, if we describe the elements of the $N'$ string set as strings consisting of vocabulary items, such as “bottle” and “red bottle”, the probability of observing a noun-only string is much smaller: it is the number of noun-only strings divided by the total number of $N'$ strings in the learner’s language. The MacArthur CDI (Dale & Fenson, 1996) suggests that 14-16 month olds know about 247 nouns. Therefore, the total number of $N'$ strings for an 18-month old learner consists of at least all the adjective and noun combinations, which is $49 \times 247 = 12103$.\(^2\) Using these (still somewhat conservative) estimates, $p_n$ from $N'$ is 0.02041. This is considerably smaller than the previous value of 0.25. Recall that the smaller the value of $p_n$ from $N'$, the smaller the value added in the numerator of the type II ambiguous update equation – and the more the learner is biased towards the $N^0$ hypothesis in the syntactic domain.

Using these less generous values of $c$ (49, instead of 5) and $p_n$ from $N'$ (0.02041, instead of 0.25), the probability of the EO Bayesian learner looking at the red bottle in the LWF experiment is 0.507 – even closer to chance than the previous probability of 0.518. And, the probability of converging on the correct grammar is the product of the probability of the correct syntactic hypothesis and the probability of the correct semantic hypothesis (both are 0.118, standard deviation = .00320), which is 0.0139. On the current, more realistic, estimates of the model’s parameters the learner now has less than a one in seventy chance of converging on the correct grammar of anaphoric one.

7. On the Necessity of Domain Specific Filters on Data Intake

We began our discussion with the observation that a learning theory can be divided into three components: the representational format, the filters on data intake, and the learning procedure. The EO Bayesian learner attempted to solve the problem of anaphoric one using a prespecified representational format\(^2\), but no domain specific filters or learning procedures. In

\(^1\) In reality, there are still more properties due to the combination of adjectives (nice red, big striped) and prepositional phrases (nice…behind his back, big striped…in the corner). We will not consider the consequences of recursive modification.

\(^2\) Again, this is a conservative estimate since there are still more $N'$ strings from combinations of prepositional phrases as well as adjectives with prepositional phrases, for instance – e.g. “bottle in the corner”, “big striped ball behind his back”, etc. As should be obvious, the effects of recursive modification only exacerbate the problem.

\(^2\) Although our model requires antecedent knowledge of X-bar theoretic structures, it is an independent question whether these are innate or derived from experience.
contrast, the model presented by R&G, which also used a prespecified representational format and a domain general learning procedure, used two domain specific filters on data intake. This model succeeded. We can now examine (a) whether both of these filters are necessary to converge on the correct grammar, and (b) whether we can derive the necessary filters in a principled fashion.

As noted above, R&G’s model implemented two filters on data intake. First, R&G’s learner considers only semantic data. That is, alternative syntactic hypotheses were evaluated only with respect to the predictions they made about the referents of phrases containing anaphoric one. These are the semantic consequences of the syntactic hypotheses. However, these hypotheses were not evaluated with respect to the predictions they made about the set of possible strings that would be available as antecedents for anaphoric one. These are the syntactic consequences of the syntactic hypotheses, and, as just noted, these were not considered. Second, R&G’s learner systematically excluded type II ambiguous data. These are examples in which the antecedent for anaphoric one is an NP containing no modifiers.

We can now ask what happens to our EO Bayesian learner if we use these filters, separately and together. First, consider a variant of our learner that learns only from the semantic consequences of its syntactic hypotheses. In the semantic domain, our learner maintained two hypotheses: the N’-property hypothesis and the any-property hypothesis. The probabilities of these two hypotheses are updated on the basis of semantic data. Moreover, these hypotheses are linked directly to the syntactic hypotheses. The N’-property hypothesis is linked to the N’ hypothesis; and, the any-property hypothesis is linked to the N_0-hypothesis. Consequently, by updating the probabilities of the semantic hypotheses, we also update the probabilities of the syntactic hypotheses. If we ignore the syntactic consequences of our hypotheses, then the only way to update the syntactic hypotheses is via the link to the semantic hypothesis space.

If we simulate an EO Bayesian learner that only learns via the semantic analysis of the data, the final probability for p_{N'} and p_{N'-prop} is 0.296 (standard deviation = .00352). The final probability of converging on the correct grammar is the product of these probabilities, which is .0877. While this is an improvement over the 0.0361 probability found in the filter-free variant of our model, it is still extremely poor performance. Thus, analyzing the data only in terms of the semantics is not sufficient to lead to the model’s convergence on the correct grammar.

The second filter that R&G’s model used was the exclusion of type II ambiguous data. We now ask what happens if we follow R&G in excluding this data. This variant of the model will, like the original EO Bayesian learner, take into account both the semantic and syntactic consequences of its hypotheses. To do this, we considered only the unambiguous and type I ambiguous data points (193, by our estimate). Because of our assumption that all of these were maximally informative, each of these data points favors the correct hypothesis (N’ in the syntax, and N’-property in the semantics). Moreover, there are no countervailing data points for the alternative hypotheses (N_0 in the syntax and any-property in the semantics). When we run our model on this data set, the final probability for the N’ hypothesis in the syntax and the N’-property hypothesis in the semantics is 0.913. The product of these two, which represents the probability of converging on the correct grammar for anaphoric one is 0.834. This is a vast improvement over the filter-free variant of our model (over 20 times more likely to converge on the correct grammar), as well as over the variant that uses only the semantic analysis to update its hypotheses in both domains (over 9 times more likely to converge on the correct grammar).

We now consider the consequences of using both of these filters simultaneously. Recall that the effect of filter 1, which restricted the learner to using only the semantic analysis, was that only semantic data could impact the hypotheses. Thus, if we use this filter in concert with filter
2, which ignores type II ambiguous data, we get a final $p_{N'}$ and $p_{N'-prop}$ of 0.755. So, the probability of converging on the correct grammar is the product of these two probabilities: 0.570. This result is still substantially better than the filter free variant of the model, as well as the variant that uses only filter 1. Nonetheless, using both filters yields a result that is worse than using only filter 2. Clearly, the benefit gained from filter 2, which is the restriction to ignore type II ambiguous data, is weakened in the context of filter 1. It is therefore in the interest of the learner to apply only filter 2.

To summarize, the EO Bayesian learner shows us that a learner not equipped with domain specific filters on data intake cannot converge on the correct grammar for anaphoric one. Figure 22 displays the learning trajectories and outcomes for the full set of simulations: no filter, semantic filter, type II ambiguous data filter, both filters. As we can see, using the type II ambiguous data filter by itself yields the highest probability for the correct grammar. In other words, the ideal learner must use both syntactic and semantic evidence, but be restricted in which sentences it takes as opportunities to learn from.

Figure 22. The Bayesian Learner’s trajectory as a function of the amount of data encountered: no filters, semantic data only filter, type II ambiguous data filter, and both semantic and type II ambiguous data filters.

8. Deriving the Filter

The necessity of a filter on data intake now raises an important question. Where does this filter come from? It seems fairly obvious that the learner cannot come equipped with a filter that says “ignore type II ambiguous data” without some procedure for identifying this data. What we really want to know is whether there is a principled way to derive the existence of this filter. Specifically, we want the filter to ignore type II ambiguous data to be a consequence of some other principled learning strategy.

Suppose that there is a general principle that learning occurs only in cases of uncertainty, because it is only in cases of uncertainty that information is conveyed (Shannon 1948; cf. Gallistel 2001, forthcoming). Now we describe a situation in which the filter to ignore type II ambiguous data could be derived from this principle.
Suppose that the learner comes equipped with a constraint against anaphora to \( X^0 \) categories, as originally proposed by Hornstein & Lightfoot (1981). The syntactic hypothesis space is reduced to a single hypothesis: \( \text{one} = N' \). In this situation, the learner needs only to solve a different problem in the syntax domain, namely, which \( N' \) is the appropriate antecedent in cases in which there are multiple \( N' \)’s available. For example, if the learner hears “Here’s a red ball. Give me another one, please,” there are two \( N' \)’s available, \textit{red ball} and \textit{ball}. These two different antecedents have different effects in the semantic domain: \textit{red ball} is restricted to red balls whereas \textit{ball} is not. In other words, the \( N' \)-property hypothesis is linked to the larger \( N' \) \textit{red ball}, whereas the any-property hypothesis is linked to the smaller \( N' \) \textit{ball}. Choosing an antecedent can be achieved using the update function in the semantic domain described above (which is done in the spirit of the technique proposed by R&G).

Now, in cases in which there is only one \( N' \) available, there are no choices to be made in finding an antecedent. That is, if the learner hears, “Here’s a ball. Give me another one, please,” the only possible antecedent is the \( N' \) \textit{ball}. Consequently, the learner has no uncertainty about the meaning of the expression and so does not invoke the learning algorithm. The learning algorithm engages only when there is uncertainty about the identity of the antecedent.

This last point is critical for motivating the learner’s choice to ignore type II ambiguous data. As noted above, having a range of available antecedents causes uncertainty about the antecedent. It is this uncertainty that triggers the learning algorithm. It is important to see at this point that this approach is highly syntactocentric. That is, it is only syntactic uncertainty that invokes the learning algorithm. Suppose, however, that both syntactic and semantic uncertainty could invoke the learning algorithm. Then, type II ambiguous data would not be ignored for the following reason. Even if there were only one possible syntactic antecedent, e.g. \textit{ball}, we would still be left with two semantic hypotheses: same property vs. any-property. Following the strategy that learning occurs under conditions of uncertainty, these data would be used, leading the learner to favor the any-property hypothesis, as detailed in sec. 7 (using the semantic data only filter). Thus, in order to derive our filter from the general learning strategy described above, the learner must only be concerned with syntactic uncertainty. Moreover, the kind of uncertainty that we are talking about here concerns not the category of the antecedent (\( N' \) vs. \( N^0 \)), but rather the identity of antecedent when there are two or more \( N' \)’s to choose from.

We are aware that many readers may be uncomfortable with this approach to deriving the filter. After all, what we’ve found is that a domain specific filter requires a domain specific constraint on representation. It may turn out that there is an alternative derivation of this filter. We would be delighted with such a result, but at the moment, we cannot see what this would look like.

9. Conclusion

In examining whether a given learning problem requires domain specific guidance from the learner, it is important to separate three ways that a learner can exhibit domain specificity. Learners may be constrained in the representations of the domain, the data that they deem relevant, or in the procedures used for updating their knowledge. Any one of these by itself represents a kind of domain specific constraint on learning, and solutions to learning problems may be only partially domain specific. For example, a learner may have domain specific representations that are used along with domain general mechanisms of data uptake and knowledge updating. Alternatively the learner may have no domain specific representations, but be constrained to only consider a subset of the data in that domain.
The division of the learning theory into distinct components, that can in their own right be domain specific or domain general, is important. The debate about linguistic nativism typically takes it as an all or nothing proposition. Put bluntly, the standard arguments hold either that language learning is the consequence of general purpose learning mechanisms or that it isn’t. This is a false dichotomy. What we have shown here is that you can have your cake and eat it too. A domain general learning procedure can be successful, but crucially, only when paired with domain specific filters on data intake. We have suggested, moreover, that the domain specific filter we needed for our problem can plausibly be derived from domain specific constraints on representation. The moral of the story is that domain specificity and domain generality can work together in the realm of grammar acquisition.

In addition, we have tried to highlight the consequences associated with the existence of multiple, connected levels of representation in language. Because the levels of representation are connected to each other, conclusions drawn by the learner in one domain also ramify in other domains. When our learner used both syntactic and semantic information, with no filters, the result was very poor learning. When the learner used both syntactic and semantic information, in concert with the filter on Type II ambiguous data, the result was very good learning. However, when we disconnected the two domains, as when we learned only from semantic data, the result was less extreme, but still in the same direction. Specifically, using only semantic information but no filter on Type II ambiguous data, the learner did not succeed, but it also did not fail as severely as when the domains were connected. Also, when using only semantic information in concert with the filter on Type II ambiguous data, the learner did succeed, but not nearly as well as when the domains were connected. Thus, the connection between domains magnifies any benefit or penalty associated with a particular learning regimen. We believe that this lesson can be generalized to learning in any situation involving multiple linked levels of representation.

Finally, it is important to recognize that we have simulated learning only for one very specific case of grammar acquisition. However, the inherent semantic compositionality of syntactic representations provides a severe hurdle for Bayesian learning techniques that are biased towards the most restrictive hypothesis. As we have noted, as the syntactic structure grows, the set of referents in the semantics shrinks. Consequently, the most restrictive hypothesis in the syntax corresponds to the least restrictive hypothesis in the semantics, and vice versa. This makes it impossible to define a “most restrictive hypothesis” across both domains.

The existence of multiple, linked levels of representation in language, and presumably elsewhere in cognition, has important consequences for learning. A link between domains can amplify the positive effects that come from using data from multiple sources. Nonetheless, this link can structure the data in such a way as to nullify the essential advantage of Bayesian learning techniques.

Acknowledgement

This paper has benefited from the insightful comments of Amy Weinberg, Norbert Hornstein, Colin Phillips, Sandy Waxman, Bill Idsardi, Terry Regier, Charles Yang and the Psycholinguistics-Acquisition group at the University of Maryland. Needless to say, any remaining errors are our own. This work was supported by an NSF graduate fellowship to LP, NSF grant BCS-0418309 to JL and NIH grant R03-DC006829 to JL.

Appendix A
We demonstrate in this section how we derive the update functions for the hypotheses in the syntactic and semantic domains. The syntactic update function changes $p_{N'}$, which is the probability that the $N'$ hypothesis is correct (that the linguistic antecedent of *one* is an $N'$ constituent). The semantic update function changes $p_{N'\text{-}\text{prop}}$, which is the probability that the $N'$-property hypothesis is correct (that the referent of *one* has the property mentioned in the linguistic antecedent).

1. Syntax

   The update function for $p_{N'}$ depends on the data type observed: unambiguous, type I ambiguous, or type II ambiguous. We derive the update function for each data type below.

   a. Unambiguous Data

   Because there are only 2 hypotheses in the syntactic domain ($N'$ and $N^0$), we use a binomial distribution to approximate a learner’s expectation of the distribution of the data to be observed. The binomial distribution is centered at $p_{N'}$, so the learner’s expectation is about how many $N'$ data points should be observed.

   The binomial distribution is normally used to represent the likelihood of seeing $r$ data points out of $t$ total with some property. There are only two choices for each data point: the property is either present or absent. For the syntactic domain, the “property” is being an $N'$ data point (as opposed to being an $N^0$ data point). The highest confidence is assigned to the distribution where $r$ $N'$ data points are observed out of $t$ total. We calculate $r$ by multiplying $t$ by $p_{N'}$, the probability that the binomial distribution is centered at: $r = t \cdot p_{N'}$.

   As an example, suppose $p_{N'}$ is 0.5, as it is in the initial state where the learner assigns equal probability to both the $N'$ and the $N^0$ hypothesis. The binomial distribution is centered at 0.5, and the learner is most confident that $r = 0.5t$ data points of those observed will be $N'$ data points. Thus, with $p_{N'} = 0.5$, the learner expects half the total data points to be $N'$ data points.

   To update $p_{N'}$ after seeing a single unambiguous data point $u$, we follow Manning & Schütze’s (1999) Bayesian updating algorithm and calculate the maximum of the a posteriori (MAP) probability. We begin with the a posteriori probability of $p_{N'}$, which is the probability of $p_{N'}$ after seeing an unambiguous data point $u$. We represent this as $\text{Prob}(p_{N'} | u)$, and calculate it using Bayes’ rule:

   $$(A1) \quad \text{Prob}(p_{N'} | u) = \frac{\text{Prob}(u | p_{N'}) \cdot \text{Prob}(p_{N'})}{\text{Prob}(u)}$$

   We now describe the individual pieces of the right hand side of the equation in (A1). $\text{Prob}(u | p_{N'})$ is the probability of observing the unambiguous $N'$ data point $u$, given the expected probability of observing an $N'$ data point. The expected probability is $p_{N'}$, so the probability of observing $u$ is simply $p_{N'}$. Therefore, $\text{Prob}(u | p_{N'}) = p_{N'}$.

   $\text{Prob}(p_{N'})$ is the probability that $p_{N'}$ is the correct probability to center the binomial distribution at, i.e. that $p_{N'}$ is the correct probability that an $N'$ data point will be observed. Recall that the binomial distribution centered at $p_{N'}$ will assign the highest confidence to the

---

21 We note that this implementation assumes the learner only extracts information from the data point at hand, rather than storing the data points individually and conducting a collective analysis on them later.
situation where \( r = (p_{N'}^*t) \) N’ data points are seen out of \( t \) total. We instantiate \( \text{Prob}(p_{N'}) \) as the probability of observing \( r \) N’ data points out of \( t \) total in a binomial distribution for all values of \( r \), from 0 to \( t \).

\[
\text{Prob}(p_{N'}) = \binom{t}{r} p_{N'}^r (1- p_{N'})^{t-r} \quad \text{(for each } r, \ 0 \leq r \leq t) 
\]

(A2)

Substituting these pieces back into equation (A1) for the a posteriori probability, we obtain the equation in (A3).

\[
\tag{A3} \text{Prob}(p_{N'} \mid u) = \frac{p_{N'}^* \binom{t}{r} p_{N'}^r (1- p_{N'})^{t-r}}{\text{Prob}(u)} \quad \text{(for each } r, \ 0 \leq r \leq t) 
\]

We can now calculate the MAP probability, by finding the maximum of this equation. To do this, we take the derivative with respect to \( p_{N'}^* \), set it equal to 0, and solve for \( p_{N'}^* \).

(A4) Calculating the MAP probability

\[
\frac{d}{dp_{N'}}(\text{Prob}(p_{N'} \mid u)) = \frac{\frac{d}{dp_{N'}}(p_{N'}^* \binom{t}{r} p_{N'}^r (1- p_{N'})^{t-r})}{\text{Prob}(u)} = 0 
\]

\[
\frac{d}{dp_{N'}}(p_{N'}^* \binom{t}{r} p_{N'}^r (1- p_{N'})^{t-r}) = 0 \quad \text{(since } \text{Prob}(u) \text{ is a constant w.r.t. } p_{N'}) 
\]

\[
p_{N'}^* = \frac{r + 1}{t + 1} 
\]

Recall that \( r \) is the previous expected number of N’ data points observed out of \( t \) data points total. Hence, \( r = p_{N'}^\text{old}*t \). Therefore, we write the update function for \( p_{N'}^* \) after observing unambiguous N’ data point \( u \) as (A5).

(A5) Unambiguous data update function

\[
p_{N'} = \frac{p_{N'}^\text{old}*t + 1}{t + 1} 
\]

An intuitive interpretation of the update function is that the numerator represents the learner’s confidence that the observed unambiguous N’ data point \( u \) is a result of the N’ hypothesis being correct; the denominator represents the total data observed so far. Thus, 1 is added to the numerator because the learner is fully confident that \( u \) indicates the N’ hypothesis is correct; 1 is added to the denominator because a single data point has been observed.

b. Type I Ambiguous Data

The derivation for the type I ambiguous data update function is identical, since we allow the Bayesian learner to treat these data as unambiguous for the N’ hypothesis even though they are, in fact, ambiguous. This will lead to an overestimation of the probability an EO Bayesian
learner would assign the N’ hypothesis. As we mentioned before, even with this overestimation, the learner will fail to assign sufficient probability to the N’ hypothesis.

c. Type II Ambiguous Data
The type II ambiguous data update function is quite similar, with the exception that a value smaller than 1 is added to the numerator. Intuitively, this smaller value represents the learner’s smaller confidence that the ambiguous data point a indicates that the N’ hypothesis is correct. We call this smaller value the partial confidence value, and represent it as \( p_{N'|a} \).

(A6) Type II ambiguous data update function
\[
p_N = \frac{p_{N\text{ old}} \cdot t + p_{N'|a}}{t + 1}
\]

The partial confidence value is the probability that one is anaphoric to N’ in a. This is equivalent to the probability that one is anaphoric to N’ in general, given that a has been observed. We write it as \( \text{Prob}(N' | a) \) and calculate it by using Bayes’ rule.

(A7) \( \text{Prob}(N' | a) = \frac{\text{Prob}(a | N') \cdot \text{Prob}(N')}{\text{Prob}(a)} \)

We now describe the individual pieces of the right hand side of the equation in (A7). \( \text{Prob}(a | N') \) is the probability of observing a type II ambiguous data point a, given that the N’ hypothesis is true. Recall that a type II ambiguous data point has an utterance with a noun-only antecedent, such as “…ball…one…”. The N’ hypothesis states that the linguistic antecedent of one must be an N’ constituent.

It is possible for a noun-only string to be an N’ constituent: a noun-only string is chosen from the set of N’ constituents, which consists of both noun-only strings (“ball”, “bottle”, etc.) and other strings that include modifiers (“red ball”, “bottle in the corner”, etc.). The probability we want is the probability of choosing a noun-only linguistic antecedent for one (such as in type II ambiguous utterance a), given the entire set of N’ constituents. Suppose there are \( n \) noun-only strings and \( o \) other strings in the N’ constituent set. We refer to the probability of choosing a noun-only string (such as “ball”) as \( p_{n\text{ from } N'} \), and it is calculated below in (A8).

(A8) \( \text{Prob}(a | N') = \frac{n}{n + o} = p_{n\text{ from } N'} \)

\( \text{Prob}(N') \) is the current probability that the N’ hypothesis is correct. This is simply \( p_{N'} \).

\( \text{Prob}(a) \) is the probability of observing a type II ambiguous utterance a, no matter which hypothesis is correct. To calculate this value, we sum the conditional probabilities of observing a for each hypothesis. If N’ is the correct hypothesis, the probability of observing a is \( \text{Prob}(a | N') \) from above. If N’ is the correct hypothesis, then the linguistic antecedent of one is an N’ constituent, which is always a noun. Thus, the probability of observing a noun-only linguistic antecedent (such as in a) is 1. We calculate \( \text{Prob}(a) \) in (A9).

(A9) \( \text{Prob}(a) = \sum_{\text{hypothesis}} p_{\text{hypothesis}} \cdot p(a | p_{\text{hypothesis}}) \)
\[
= p_{N'} \cdot p(a | p_{N'}) + p_{N0} \cdot p(a | p_{N0})
\]
\[ = p_{N'} \cdot \frac{n}{n + o} + (1 - p_{N'}) \cdot 1 \]

Substituting these pieces back into the right hand side of the equation in (A7), we obtain (A10).

\[
(A10) \text{Prob}(N' | a) = \frac{(n/(n + o))^*p_N}{p_N^*(n/(n + o)) + (1 - p_N)^*1} = \frac{p_{\text{from } N'}^*p_N}{p_{\text{from } N'}^*p_{\text{from } N'} + (1 - p_N)^*1} = p_{N' | a}
\]

As we can see, the partial confidence value \( p_{N' | a} \) depends only on \( p_{\text{from } N'} \) and \( p_N \). This partial confidence value, which will be less than 1, is added to the numerator of the type II ambiguous data update function instead of 1. The larger \( p_{\text{from } N'} \) is, the higher the learner’s confidence in the \( N' \) hypothesis when a type II ambiguous data point is observed. Thus, the more likely it is that a noun-only string could be chosen from the \( N' \) constituent set, the more the \( N' \) hypothesis is rewarded when this type of data is seen.

2. Semantics

The update function for \( p_{N' - \text{prop}} \) also depends on the data type observed: same-property, different-property, or unknown-property. We derive the update function for each data type below.

a. Same-Property Data

Because there are only 2 hypotheses in the semantic domain (\( N' \)-property and any-property), we use a binomial distribution to approximate a learner’s expectation of the distribution of the data to be observed. The binomial distribution is centered at \( p_{N' - \text{prop}} \), so the learner’s expectation is about how many \( N' \)-property data points should be observed.

Again, the binomial distribution is normally used to represent the likelihood of seeing \( r \) data points out of \( t \) total with some property. In the semantic domain, the “property” is being an \( N' \)-property data point (as opposed to being an any-property data point). The highest confidence is assigned to the distribution where \( r \) \( N' \)-property data points are observed out of \( t \) total. We calculate \( r \) by multiplying \( t \) by \( p_{N' - \text{prop}} \), the probability that the binomial distribution is centered at: \( r = t^*p_{N' - \text{prop}} \).

As an example, suppose \( p_{N' - \text{prop}} = 0.5 \), as it is in the initial state where the learner assigns equal probability to both the \( N' \)-property and the any-property hypothesis. The binomial distribution is centered at 0.5, and the learner is most confident that \( r = t^*0.5 \) data points of those observed will be \( N' \)-property data points. Thus, with \( p_{N'} = 0.5 \), the learner expects half the total data points to be \( N' \)-property data points.

To update \( p_{N'} \) after seeing a single same-property data point \( s \), we again follow Manning & Schütze’s (1999) Bayesian updating algorithm and calculate the maximum of the a posteriori (MAP) probability. Like the type II ambiguous data update function in the syntactic domain, however, we will add a value smaller than 1 to the numerator. Intuitively, this smaller value represents the learner’s smaller confidence that the same-property data point \( s \) indicates that the \( N' \)-property hypothesis is correct. We call this smaller value the partial confidence value, and represent it as \( p_{N' - \text{prop} | s} \).

\[
(A11) \text{Same-property data update function}
\]
The partial confidence value is the probability that the referent of one has the N'-property mentioned in s. This is equivalent to the probability that the referent of one has the N'-property in general, given that s has been observed. We write it as Prob(N'-prop | s) and calculate it by using Bayes’ rule.

\[
(A12) \quad \text{Prob}(N'-\text{prop} | s) = \frac{\text{Prob}(s | N'-\text{prop}) \ast \text{Prob}(N'-\text{prop})}{\text{Prob}(s)}
\]

We now describe the individual pieces of the right hand side of the equation in (A12). Prob(s | N'-prop) is the probability of observing a same-property data point s, given that the N'-property hypothesis is true. Recall that in a same-property data point, the referent of the antecedent of one has the same salient property that the referent of one has. The N'-property hypothesis states that the referent of the antecedent of one must have the property described by the linguistic antecedent of one. Therefore, if the N'-property hypothesis is true, the probability of observing a same-property data point is 1.

\[
(A13) \quad \text{Prob}(s | N'-\text{prop}) = 1
\]

Prob(N'-prop) is the current probability that the N'-property hypothesis is correct. This is simply \(p_{N'-\text{prop}}\).

Prob(s) is the probability of observing a same-property utterance s, no matter which hypothesis is correct. To calculate this value, we sum the conditional probabilities of observing s for each hypothesis. If N'-property is the correct hypothesis, the probability of observing s is Prob(s | N') from above. If any-property is the correct hypothesis, then there is no restriction on what property the referent of the linguistic antecedent of one has. The probability of that referent having the same property as the referent of one is simply \(1/c\), where there are \(c\) properties in the world. We calculate Prob(s) in (A14).

\[
(A14) \quad \text{Prob}(s) = \sum_{\text{hypotheses}} \text{p}_{\text{hypothesis}} \ast \text{p}(s | \text{p}_{\text{hypothesis}})
\]

\[
= p_{N'-\text{prop}} \ast \text{p}(s | p_{N'-\text{prop}}) + p_{\text{any}-\text{prop}} \ast \text{p}(s | p_{\text{any}-\text{prop}})
\]

\[
= p_{N'-\text{prop}} \ast 1 + (1 - p_{N'-\text{prop}}) \ast \frac{1}{c}
\]

Substituting these pieces back into the right hand side of the equation in (A12), we obtain (A15).

\[
(A15) \quad \text{Prob}(N'-\text{prop} | s) = \frac{1 \ast p_{N'-\text{prop}}}{p_{N'-\text{prop}} \ast 1 + (1 - p_{N'}) \ast \frac{1}{c}} = \frac{p_{N'-\text{prop}}}{p_{N'-\text{prop}} \ast 1 + (1 - p_{N'}) \ast \frac{1}{c}} = p_{N'-\text{prop} | s}
\]

As we can see, the partial confidence value \(p_{N'-\text{prop} | s}\) depends only on \(c\) and \(p_{N'-\text{prop}}\). This partial confidence value, which will be less than 1, is added to the numerator of the same-property data update function instead of 1. The larger \(c\) is, the higher the learner’s confidence in
the N'-property hypothesis when a same-property data point is observed. Thus, the more properties there are in the learner’s world, the more the N'-property hypothesis is rewarded when this type of data is seen.

\[ b. \ \text{Different-Property Data} \]

Different-property data could not possibly be observed if the N'-property hypothesis was correct, since the N'-property hypothesis requires the referent of one to have the property mentioned in the N' constituent. When seeing a different-property data point \( d \), the learner has no confidence that the N'-property hypothesis is correct; thus, a different-property data point \( d \) is unambiguous data against the N'-property hypothesis.

The derivation of the different-property update equation is identical to the derivation of the unambiguous data update equation in the syntactic domain, except that the value added to the numerator is 0. This intuitively reflects the learner’s lack of confidence that the N'-property hypothesis is indicated by \( d \).

\[
(A16) \quad p_{N'\text{-prop}} = \frac{p_{N'\text{-prop} \cdot \text{old}} \cdot t + 0}{t + 1}
\]

\[ \text{Appendix B} \]

The R&G learner is quite liberal about shifting probability to the superset hypothesis: a single piece of unambiguous data for the superset will shift all the probability to the superset hypothesis. However, the correct hypothesis for English anaphoric one is the subset in the semantic domain. The success of this learner for converging on the correct semantic hypothesis for anaphoric one relies on the assumption that there will never be unambiguous data for the semantic superset.

Recall that the semantic superset hypothesis is that one refers to an object that does not need to have the property mentioned in the linguistic antecedent. This is the any-property hypothesis. Unambiguous data for the superset would be an utterance where one refers to an object that does not have the property mentioned in the antecedent. For instance, if the utterance is “…red ball…one…”, unambiguous superset data would be the situation where the referent of one does not have the property ‘red’.

It is crucial for R&G’s model that this type of data never occurs. If the referent of one in the above utterance was a purple ball (perhaps by accident), the new probability for the subset hypothesis (the N'-property hypothesis) in the semantic domain would be 0. We detail why this occurs below.

Recall that we refer to the probability that the N'-property hypothesis is correct as \( p_{N'\text{-prop}} \). Though it is initially 0.5 before any data is observed, it increases as each piece of ambiguous data is observed. This is due to the size principle which biases the learner in favor of the subset hypothesis if ambiguous data is observed.

Let \( u \) be a piece of unambiguous data for the superset hypothesis, where the utterance is “…red ball…one…” and the referent of one is a non-red ball. The learner now calculates the updated probability that the N'-property hypothesis is correct, using Bayes’ rule. The updated \( p_{N'\text{-prop}} \) given the observation of \( u \) is represented as the conditional probability \( p(N'\text{-prop}| u) \). To calculate this probability, we use Bayes’ rule.

\[
(A17) \quad \text{Calculating the conditional probability } p(N'\text{-prop}| u) \text{ using Bayes’ rule}
\]
\[ p(N'\text{-prop}|u) \propto p(u|N'\text{-prop}) \times p(u) \]

The probability \( p(u|N'\text{-prop}) \) is the probability of observing the unambiguous superset data \( u \), given that the \( N' \)-property hypothesis is true. In this case, the referent of \textit{one} in \( u \) specifically doesn’t have the \( N' \)-property (‘red’). Therefore, it could not possibly be generated if the \( N' \)-property hypothesis was true, since the \( N' \)-property hypothesis requires the referent of \textit{one} to have the property mentioned in the linguistic antecedent. So, the probability of observing \( u \) if the \( N' \)-property hypothesis is true (\( p(u|N'\text{-prop}) \)) is 0.

We substitute this value into the equation in (A17) to get \( p(N'\text{-prop}|u) \propto 0 \times p(u) = 0 \). Therefore, the updated probability for \( p_{N'\text{-prop}} \) after seeing a single piece of unambiguous superset data \( u \) is 0, no matter what the previous probability of \( p_{N'\text{-prop}} \) was.

Since this is not terribly robust behavior for a learner, we have adopted the more conservative Bayesian updating approach described by Manning & Schütze (1999). Unlike the liberal R&G model, the Manning & Schütze learner shifts probability much more slowly between hypotheses. Only after observing a vast majority of evidence for one hypothesis would a Manning & Schütze learner shift the vast majority of the probability into that hypothesis.

\textbf{Appendix C}

Recall that our model contains a parameter, \( t \), which represents the amount of change the learner can undergo in the course of learning. We quantify this parameter as the number of data points the learner can use to update its probabilities. In our simulation, this was 4017. However, one might be concerned that the value of \( t \) might play a critical role in determining the final probability of converging on the correct grammar of anaphoric \textit{one}. Below, we show the final probability of converging on anaphoric \textit{one} as a function of the size of \( t \). As we can see, the final value does not appreciably alter based on the size of \( t \). The reason for this stability is that the behavior of the learner is dependent on the probability distribution of the data. In case \( t \) is small, each data point has a larger impact. In case \( t \) is large, each data point has a smaller impact. But, because the probability distribution is always the same, the learner always ends up with the same value. Moreover, if the learner encounters data after having seen \( t \) amount of data, this data cannot be used to update the probabilities.

![Graph showing final probability of correct grammar of anaphoric one for different values of \( t \), no filters](image.png)
Figure A1. Final probability of the correct grammar, given different values of $t$. All values are approximately 0.0361.

References


