Commentary on
"Child language acquisition: Why Universal Grammar doesn't help"
By Ben Ambridge, Julian Pine, and Elena Lieven

Evaluating learning strategy components: Being fair

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

1. Introduction

The basic issue that the authors (**AP&L**) highlight about proposed learning strategies seems exactly right: What will *actually* work, and what exactly makes it work? They note that "…nothing is gained by positing components of innate knowledge that do not simplify the problem faced by language learners" (p.56, section 7.0), and this is absolutely true. To examine how well several current learning strategy proposals work that involve innate, linguistic knowledge, AP&L present evidence from a commendable range of linguistic phenomena, from what might be considered fairly fundamental knowledge (e.g., grammatical categories) to fairly sophisticated knowledge (e.g., subjacency and binding). In each case, AP&L identify the shortcomings of some existing Universal Grammar (**UG**) proposals, and observe that these proposals don't seem to fare very well in realistic scenarios. The challenge at the very end underscores this – AP&L contend (and I completely agree) that a learning strategy proposal involving innate knowledge needs to show "precisely how a particular type of innate knowledge would help children acquire X" (p.56, section 7.0).

More importantly, I believe this should be a metric that *any* component of a learning strategy is measured by. Namely, for any component (whether innate or derived, whether language-specific or domain-general), we need to not only propose that this component could help children learn some piece of linguistic knowledge but also demonstrate at least "one way that a child could do so" (p.57, section 7.0). To this end, I first want to highlight how computational modeling is well suited for doing precisely this: for any proposed component embedded in a learning strategy, modeling allows us to empirically test that strategy in a realistic learning scenario. It's my view that we should test all potential learning strategies, including the ones AP&L themselves propose as alternatives to the UG-based ones they find lacking. An additional and highly useful benefit of the computational modeling methdology is that it forces us to recognize hidden assumptions within our proposed learning strategies, a problem that AP&L rightly recognize with many existing proposals.

This leads me to suggest certain criteria that any learning strategy should satisfy, relating to its utility in principle and practice, as well as its usability by children. Once we have a promising learning strategy that satisfies these criteria, we can then concern ourselves with the components comprising that strategy. With respect to this, I briefly discuss the type of components AP&L find unhelpful, and note that several of the components they would prefer might still be reasonably classified as UG components. The main issue they have is not with components that are innate and language-specific, but rather components of this kind that in addition involve very precise knowledge. This therefore does not rule out UG components that involve more general knowledge, including the components AP&L themselves propose. In addition, AP&L ask for explicit examples of UG components that actually do work. I provide a brief review of one potentially UG component that's part of a successful learning strategy for syntactic islands (described in Pearl & Sprouse 2013) which also satisfies the criteria I suggest for evaluating learning strategies more generally.

2. Metrics for any learning strategy & the power of computational modeling

It's reasonable to question how well any learning strategy works in a realistic learning scenario, where it's constrained to use the data children have access to, to succeed in the time children have to learn, and to be implementable given children's cognitive abilities. This is where computational modeling becomes an incredibly useful tool: a computational model implementing a particular learning strategy can provide an existence proof that the strategy will (or won't) succeed when constrained in realistic ways (see Pearl 2010 for a general overview of how computational modeling can be used this way in language acquisition research).

A special strength of computational modeling is that it forces hidden assumptions to be made explicit, since all aspects of the model need to be concrete before the model can be run. AP&L note that this is often a problem with UG-based learning strategies: "…an additional problem that is common to many UG approaches…it requires a cascade of further assumptions that are rarely made explicit…before it can be said to provide a potentially workable solution" (p.16, section 2.2). However, this is not just a problem of UG-based proposals – it appears in many proposed learning strategies. For example, AP&L

propose a distributional analysis strategy for grammatical categorization (e.g., Noun and Verb, Subject and Object), and note that one concern is the appropriate information to use in that distributional analysis. They suggest that there is “no need to build in innate constraints to rule out every theoretically possible distributional-learning strategy” (p.13, section 2.1) because it is “possible that children track all kinds of semantic and distributional properties that are rapidly discovered to be irrelevant” (p.29, section 3.3). It’s perfectly fine to suggest this, but to really make it a viable option, it should be demonstrated how this plays out in a realistic learning scenario. Some questions, reflecting hidden assumptions, that a computational model would have to answer are these:

(i) What properties does the child try out?
(ii) How many properties are tried out simultaneously?
(iii) How long does it take before the child gives up on any one property, viewing it as irrelevant?
(iv) Does it matter what order properties are tried out in?

Just to be clear, this isn’t to say that this strategy wouldn’t work – but once it’s been embedded in a computational model, it becomes much clearer how it would work and what’s necessary to make it work. In a similar vein, I want to highlight what seem to me to be hidden assumptions of some of the strategies proposed by AP&L.

A strategy AP&L suggest for learning word order in section 3.3 (p.28) is this: “children could (1) group together items that share certain semantic regularities (e.g., acting as agents) and certain distributional regularities and (2) observe the ordinal positions in which these categories appear…once this has been done, the child has effectively learned the word-order of her language”. For step 1, the questions relating to distributional learning mentioned above apply here with respect to the distributional regularities that are used to cluster items that will serve as the Subjects and Objects. For step 2, how is the observation of the ordinal positions accomplished? How reliable does the distributional clustering in step 1 need to be in order for the learner to make the correct observation of ordinal position? What would make this level of reliability possible in a realistic learning scenario?

Another strategy AP&L suggest, related to learning about structure dependence in syntax in section 4.2 (p.33) involves “evidence that strings of arbitrary length that share distributional similarities can be substituted for one another”, which AP&L describe as “evidence for the structure dependent nature of syntax”. Reiterating the questions above about the reliability of the distributional clustering, how reliable do these derived syntactic categories (i.e., strings of arbitrary length that are substitutable) need to be and what makes this level of reliability possible? Then, once the child has the necessary syntactic categories, how do they yield the rules that create hierarchical clausal structure (e.g., Sentence → NP VP, NP → Det N)? What kind of information is required to make this inference?

A third strategy AP&L suggest in section 5.0 (p.43), related to learning about subjacency, requires children to know "whether or not a particular constitutent falls within the potential focus domain: whether or not it can be denied (without recasting the entire phrase)". How would children develop this sophisticated knowledge about the potential focus domain, and its syntactic implementation? That is, what data would lead them to this knowledge? How is the inference about this knowledge made, based on these data?

In general, each of these proposed learning strategies would benefit from an explicit demonstration of (a) their utility when given realistic child input data, and (b) their useability by children. Given this, the following may be good criteria for any proposed learning strategy, once its utility in principle has been demonstrated for isolated (and perhaps difficult) test cases:

(i) **Utility**: Show it's useful on realistic child input data. An example of this approach is the computational-level Bayesian model of Perfors, Tenenbaum, & Regier (2011), who demostrate how a rational learner capable of optimal inference could learn that the structure of questions (among other things) is best described by hierarchical rules. Their learner draws on overhypotheses about the structure of syntax and learns from child-directed data from the CHILDES database (MacWhinney 2000). This knowledge about hierarchical structure then becomes a reasonable foundation for future knowledge about how syntactic rules could be formed, e.g., a child might favor using hierarchical structure for other aspects of syntactic knowledge.

(ii) **Useability**: Show it's useable by children, who have cognitive limitations such as a limited memory and limited processing resources, as well as a limited time during which they learn. An example of this approach is the algorithmic-level learning model of Pearl & Mis (submitted), who demonstrate how a learner using an online probabilistic learning algorithm could learn to correctly interpret the pronoun *one*, and generate the looking time preferences observed

> in 18-month-olds. The modeled learner not only learns from realistic child-directed data (thus satisfying utility), but also learns the correct interpretation from the same amount of data 18-month-olds would encounter and processes the data incrementally, as children are likely to do.

As is apparent from the descriptions above, a very straightforward way to investigate utility and useability is by creating a computational model that incorporates the proposed learning strategy and testing it out on a specific learning problem. In order to do this, the learning problem itself also has to be precisely specified: what does the learner start with (*initial state*), what data does the learner choose to learn from (*data intake*), how long does the learner have to learn (*learning period*), and where is the learner supposed to end up (*target state*)? Each of these components of the learning problem can be specified by drawing on theoretical, experimental, and computational results (see Pearl & Mis submitted for a more thorough discussion of this).

To give a quick idea of how this works, let's walk through an example of each learning problem component with respect to learning about how to interpret the pronoun *one*. One important aspect of interpretation involves identifying the linguistic antecedent when it's present. For example, in "*Look – a red bottle! Oh look – there's another one.", one*'s linguistic antecedent can be *red bottle*, since the second sentence can mean *"Oh look – there's another red bottle.*". Under a syntactic story, a learner determines the linguistic antecedent for *one* by looking for something that has the same grammatical category as *one*. So, in the initial state, a learner would probably need to know something about the available grammatical categories of words. We can approximate the grammatical categories the learner needs by using current theoretical descriptions of relevant grammatical categories. For the learner's data intake, the learner may be able to use all utterances involving *one* that have a linguistic antecedent, and we can estimate how many of these occur in the input by examining naturalistic child-directed data, such as the datasets in the CHILDES database. For the learning period, experimental results from Lidz, Waxman, & Freedman (2003) suggest 18-month-olds have an adult interpretation of *one* in certain contexts, so the learning period for that aspect of knowledge about *one* must be completed by 18 months. For the target state, theoretical descriptions of adult knowledge of *one*, based on informal or formal experimental judgment data, represent the target knowledge state. The adult behavior, observed experimentally in 18-month-olds, can provide the target behavior state.

Once the learning problem is well-defined, a learning strategy can then be instantiated with respect to that problem. In fact, a learning strategy should be defined as a collection of knowledge, learning capabilities, and/or learning biases in the learner's initial state. For example, a probabilistic learning strategy for learning about *one* based on leveraging any pronoun data that is informative will have several additional things in the initial state of the learner: (i) the knowledge of what pronouns are, (ii) the capability of recognizing pronouns in the input, (iii) a bias to learn from data containing pronouns, (iv) the capability of doing probabilistic inference, and (v) a bias to use this capability when learning instead of using other decision procedures. This specification of the learning strategy again forces us to be explicit about all the pieces that are necessary to make a learning strategy function, and actually implement it in a computational model.

I want to now comment briefly about how we can gauge the success of a learning strategy, once we've implemented it. Given a specified learning problem, the obvious answer seems to be whether that learning strategy allowed the learner to reach the target state. But what target state *is* that, exactly? Does adult knowledge always need to be attained? I'd like to suggest this isn't necessarily true. First, given that we often have experimental data on children's linguistic behavior, and when it appears adult-like, a reasonable aim may simply be to generate that adult-like behavior no matter what knowledge underlies it. It may be that adult behavior is possible without adult knowledge (e.g., see Pearl & Mis submitted), but a learning strategy that can generate the adult behavior might still be considered a pretty good learning strategy.

Second, when the aim is to learn knowledge that serves as the foundation for more sophisticated knowledge, perhaps the goal should be sufficiency instead of perfection. That is, is the output of the learning strategy good enough to use as input for the next stage of learning? For example, even if the grammatical categories derived from distributional clustering aren't perfect, are they good enough to learn word order or phrase structure (or at least get those learning processes started while the grammatical category knowledge is refined further)? If so, I would say the learning strategy should be counted as successful.

3. A closer look at what makes successful learning strategies work

Once we identify a successful strategy, we can then examine all the components that are involved and how a child might come to have those components. This leads naturally to the issue of how to classify learning strategy components – e.g., are they UG or not? If they're UG, then they're by definition innate components, rather than derivable, so the answer is simple: the child comes equipped with these learning components.

This leads me to something that I think should be emphasized about UG learning strategy components: any component that is both innate and used only for learning language (i.e., language-specific) is a UG component. Importantly, this doesn't mean that a UG component has to be about *specific* knowledge – it could be something rather general about language. Again, as long as the component is both innate and language-specific, it's a UG component. AP&L correctly note that many previous proposals about UG components involve specific knowledge, and they take issue with this, which is quite understandable when we worry about how children would come to genetically encode whatever learning components are UG. However, AP&L seem comfortable with the idea of more general UG learning components, as they note in section 1.0 (p.5) that "most – probably all – accounts of language acquisition will invoke at least *some* language-related biases …we do not use the term UG to refer to an initial state that includes only this very general type of knowledge." AP&L simply don't classify this more general innate, language-specific knowledge as UG.

Nonetheless, I reiterate that it's not logically necessary for a UG component to involve specific knowledge – all that matters is that the component is both innate and only used for learning language. So, given this, it's not really fair to castigate UG components as a whole, though perfectly fair to be unhappy with particular UG components that are found to be lacking. To also be fair to AP&L, perhaps there haven't been many proposed UG components that don't involve specific knowledge. Even more importantly, perhaps there haven't been many explicit demonstrations of how any UG components (whether specific or general) actually make learning work for some aspect of acquisition. AP&L note in both the opening and closing of their article that they aren't aware of any (section 1.0, p.7: "...there exists no current proposal for a component of innate knowledge that would be useful to language learners", section 7.0, p.55: "…there are no proposals for components of innate knowledge that would simplify the learning process for the domains considered").

To remedy this, here is a recent example of a learning strategy component that is of this more general flavor, and which forms part of a successful learning strategy described by Pearl & Sprouse (2013) for the acquisition of constraints on *wh*-dependencies, sometimes referred to as knowing about syntactic islands. One explanation from a syntactic standpoint is that *wh*-dependencies that cross syntactic islands are ungrammatical. As an example of this, an English speaker will find the following utterance involving an island-crossing *wh*-dependency ungrammatical:

(1a) * What did Jack think the story about __ was written by Lily?

In contrast, an English speaker would find the following utterance significantly better-sounding, as the *wh*-dependency does not cross a syntactic island:

(1b) Who did Jack think the story about penguins was written by __?

A successful learning strategy for acquiring knowledge of four different syntactic islands involves tracking trigrams of the phrase structure nodes containing the dependency (called *container nodes* by Pearl & Sprouse 2013):

(2) Who did Jack think the story about penguins was written by __?

(i) Phrase structure containing the *wh*-dependency
Who did $[_{\text{IP}}$ Jack $[_{\text{VP}}$ think $[_{\text{CP}}$ $[_{\text{IP}}$ the story about penguins $[_{\text{VP}}$ was written $[_{\text{PP}}$ by __]]]]]]?

(ii) Container node characterization of *wh*-dependency
start-IP-VP-CP_{null}-IP-VP-PP-end

(iii) trigrams of container nodes $\in Trigrams_{start-IP-VP-CP_{null}-IP-VP-PP-end}$
start-IP-VP
IP-VP-CP_{null}
VP-CP_{null}-IP
CP_{null}-IP-VP
IP-VP-PP
VP-PP-end

The perceived grammaticality of any *wh*-dependency is determined by its probability, calculated as the smoothed product of its container node trigrams:

(3) p(*Who did Jack think the story about penguins was written by __?*) =

$$\prod_{trigram \in Trigrams_{start-IP-VP-CP_{null}-IP-VP-PP-end}} p(trigram)$$

This successful strategy requires the learner to characterize *wh*-dependencies at a particular level of granularity, namely what might be considered "standard" phrasal nodes (e.g., IP, VP, NP, etc.), with the exception of CP, which is subcategorized by the lexical item in complementizer position (e.g., CP_{null}, CP_{that}, etc.). Notably, without the CP subcategorization, the strategy fails to work for two of the four syntactic islands examined by Pearl & Sprouse (2013). Thus, this component of the successful learning strategy is fairly particular, because the learner must characterize *wh*-dependencies at the right level of granularity. The learner should not characterize *wh*-dependencies more precisely, such as subcategorizing other phrasal nodes (e.g., VP_{think}) or using other intermediate phrasal categories (e.g., vP), and the learner should not characterize *wh*-dependencies less precisely (e.g., CP instead of CP_{null} and CP_{that}).

Pearl & Sprouse (2013) suggest that knowing to use this characterization of *wh*-dependencies may be a UG learning component, as it is unclear why this level of granularity should be selected from all the other ways that exist of characterizing *wh*-dependencies. This leads to two important points. First, something like AP&L's suggestion for distributional learning strategies may apply here: Perhaps the learner tries out a whole variety of different *wh*-dependency characterizations, realizes somehow that they don't seem to be working, and settles on this one that does end up working. In that case, perhaps this isn't a UG component after all, if it is in fact derivable. However, this leads to the same questions that occurred before, relating to how this would operate in practice: What options does the learner try? Does it matter what order they're tried in? How many are tried at once? How long does the learner try one before giving up on it? And most importantly, if it is in fact possible to derive the correct *wh*-dependency characterization this way, what learning components were needed to do so and what kind of components are they (e.g., UG or not)? The second point is that, even if it turns out that this characterization of *wh*-dependencies can't be derived and so must be a UG component, it is a more general type of UG knowledge than explicit knowledge about syntactic island structures or subjacency. Thus, it would be a UG component, but a UG component that involves knowledge that is more general than, say, explicit constraints on *wh*-dependencies or subjacency.

More broadly, I suspect this kind of issue occurs for many current learning strategies that work (and especially for learning strategies that are proposed but not yet explicitly tested). That is, it may be unclear whether a given language-specific component of a working strategy is innate or derived. So, we may use "innate" (and thus "UG") as a placeholder until we can demonstrate how that component can be derived. If we can't show it's possible to derive the component, then the "innate" label is no longer just a placeholder.

I think some of the components proposed by AP&L also illustrate this issue quite well. One example relates to hierachical structure, where AP&L contend in section 4.2 (p.35) that "[h]ierarchical syntactic structure is a reflection of hierarchical conceptual structure". This is certainly a viable hypothesis, where the hierarchical structure is not really syntactic at its core, but rather conceptual. However, I don't believe the issue of a UG component automatically goes away, since we must ask where this conceptual structure comes from: Why is it hierarchical? Why is it hierarchical in this particular way, e.g., with events comprising the frame and participants comprising the arguments that slot into that frame? Does the learner ever consider a hierarchical hypothesis that is "participant-centric", with events acting as arguments instead? Does the learner ever consider a non-hierarchical hypothesis? If the learner does consider other conceptual hypotheses, how quickly is the correct hypothesis selected and how is this done? In contrast, if the learner does not consider other conceptual hypotheses (or in general if the hypothesis space is constrained in some useful way), where does this precise knowledge about conceptual structure come from? In effect, the origin of just the right conceptual structure may well involve innate, language-specific components.

As another example, AP&L suggest that a good learning strategy for subjacency involves fairly rich knowledge of informational structure, including knowledge of "discourse pragmatics and focus structure" (section 5.0, p.45) and the links between that informational knowledge and observable language use, such as "syntactic devices that distinguish background information from the central assertaion of an utterance" (section 5.0, p.45). The potential benefit of this strategy is that it may have wider empirical coverage than a purely syntactic approach, which would be great from a learning standpoint. However, given how sophisticated the necessary knowledge is, the origin of this knowledge is an important question. It seems that there may be many ways that aspects of informational structure could be linked to observable language use. How does the learner come to know the precise relationships that lead to subjacency? What is the hypothesis space, and how does the learner sort through it effectively? If the learner's hypothesis space is constrained in some useful way, where does that knowledge come from? Is the necessary knowledge language-specific? Can it be derived from other knowledge the learner already has? Given the complexity of the knowledge, I'm tempted to think that there's some UG component involved somewhere – though it may very well be more general knowledge, rather than specific knowledge.

4. Conclusion

I think it's an excellent idea to have rigorous criteria for learning strategy performance so that we can really tell what works and what doesn't. In this commentary, I've suggested some metrics that any proposed learning strategy can be evaluated against, relating to its utility in practice and its useability by children. Importantly, these metrics apply no matter what components comprise the learning strategy – all we're trying to figure out first and foremost is what actually works. Once we have a strategy that satisfies these metrics (e.g., the strategy for learning about syntactic islands described in Pearl & Sprouse 2013), that strategy can become the focus for more targeted investigation, especially with respect to the nature of the learning strategy components. It may be that some components are indeed UG components, but they involve more general knowledge rather than specific knowledge. No matter what kind of components are involved in a successful learning strategy, it's important to understand what they are and where they come from, and this is why computational modeling can be a handy methodology for figuring out exactly how children come to have the knowledge that they do.

References

Lidz, J., Waxman, S., & Freedman, J. 2003. What infants know about syntax but couldn't have learned: Experimental evidence for syntactic structure at 18 months. *Cognition*, *89*, B65–B73.

MacWhinney, B. 2000. *The CHILDES Project: Tools for Analyzing Talk*. Mahwah, NJ: Lawrence Erlbaum Associates.

Pearl, L. 2010. Using computational modeling in language acquisition research, In E. Blom & S. Unsworth (eds). *Experimental Methods in Language Acquisition Research*, John Benjamins, 163-184.

Pearl & Mis. Submitted. Knowing where to look: Identifying how children learning syntactic knowledge. *lingbuzz*: http://ling.auf.net/lingbuzz/001922.

Pearl, L., & Sprouse, J. 2013. Syntactic islands and learning biases: Combining experimental syntax and computational modeling to investigate the language acquisition problem. *Language Acquisition*, *20*, 19–64.

Perfors, A., Tenenbaum, J., & Regier, T. 2011. The learnability of abstract syntactic principles. *Cognition*, *118*, 306–338.