TOPICS
TOPICS IN COGNITIVE SCIENCE

Topics in Cognitive Science 5 (2013) 89–110
Copyright © 2013 Cognitive Science Society, Inc. All rights reserved.
ISSN:1756-8757 print / 1756-8765 online
DOI: 10.1111/tops.12001

Complexity in Language Acquisition

Alexander Clark,[a] Shalom Lappin[b]

[a]*Department of Computer Science, Royal Holloway University of London*
[b]*Department of Philosophy, King's College London*

Received 21 September 2011; received in revised form 30 July 2012; accepted 7 August 2012

Abstract

Learning theory has frequently been applied to language acquisition, but discussion has largely focused on information theoretic problems—in particular on the absence of direct negative evidence. Such arguments typically neglect the probabilistic nature of cognition and learning in general. We argue first that these arguments, and analyses based on them, suffer from a major flaw: they systematically conflate the hypothesis class and the learnable concept class. As a result, they do not allow one to draw significant conclusions about the learner. Second, we claim that the real problem for language learning is the computational complexity of constructing a hypothesis from input data. Studying this problem allows for a more direct approach to the object of study—the language acquisition device—rather than the learnable class of languages, which is epiphenomenal and possibly hard to characterize. The learnability results informed by complexity studies are much more insightful. They strongly suggest that target grammars need to be objective, in the sense that the primitive elements of these grammars are based on objectively definable properties of the language itself. These considerations support the view that language acquisition proceeds primarily through data-driven learning of some form.

Keywords: Computational complexity; Language acquisition; Computational learning theory

For many years, arguments from formal learning theory have been applied to the study of language acquisition. Indeed, one of the founding studies of learning theory (Gold, 1967) was concerned in large part with the problem of constructing an adequate model of language acquisition. Negative results derived from this study have been thought to support a view of language acquisition according to which the class of possible human languages is very limited, perhaps even finite (Bertolo, 2001). These arguments focus on what we can loosely call *information theoretic* claims about the paucity of data—the poverty of the stimulus (Clark & Lappin, 2011).[1] These arguments have been criticized

Correspondence should be sent to Alexander Clark, Department of Computer Science, Royal Holloway University of London, Egham, TW20 0EX, UK. E-mail: alexc@cs.rhul.ac.uk

extensively by several authors before (Clark & Lappin, 2011; Johnson, 2004; Lappin & Shieber, 2007). Similar criticisms apply to more recent versions of these arguments that appeal to the more modern theory of PAC learning (Valiant, 1984). We will not discuss these problems in detail in this article.

A second major problem with these information theoretic arguments is that they tend to conflate two independent notions: the idea of a hypothesis class of languages, and the notion of a class of learnable languages. The confusion of these two different ideas leads to the unjustified conclusion that the latter class must be incorporated into the design of the learning algorithm, and explicitly or implicitly represented as a feature of this algorithm. The distinction between the two classes, although it had been recognized in the learning-theory literature for some time (Jain, Osherson, Royer & Sharma, 1999; Zeugmann & Lange, 1995), became much clearer with the advent of learning procedures for context-free languages, where the two classes must, for technical reasons, be distinct. Once these notions are kept apart, it becomes evident why (un)learnability claims based on information theoretic considerations are not well motivated.

The following simple example will bring out the difference between the hypothesis class and the learnable class of languages for a learning algorithm. Consider the most trivial learning algorithm—a rote-learning procedure A, which works as follows. For every string s, which A receives as input it compares it with the set of strings S that it has encountered previously. If $s \in S$, then A does nothing. If $s \notin S$, then A constructs $S' = S \cup \{s\}$. Clearly, this learner can only learn finite languages[2]—it does not generalize beyond the input. Suppose we have a learning model where there is a fixed bound on how many examples that the learner can see: say for concreteness, it is only learning from 1,000 examples. Now regardless of what reasonable measure we put on how this learner converges or how these examples are picked or generated, it is clear that from these examples that the learner cannot learn a language that consists of 1,000,000 strings. If it had a few million examples, then it might be able to learn such a language, but since it only has a few examples it cannot. Such large languages are not therefore in the learnable class under this model. The learnable class will be some set of languages whose size is below some threshold that depends on the precise details of how we define the learning setup, as well as on the number of examples allowed, 1,000 in this case. The hypothesis class though is just the set of all finite languages, which is vastly larger. Crucially, this algorithm does not contain explicitly or implicitly any description of the learnable class. We will return to this point later, and illustrate it with more interesting examples.

This problem does not entail that formal learning theory has nothing to offer the study of language acquisition. On the contrary, it is highly relevant. However, we argue that the crucial problems are not information theoretic, as suggested in the Gold results. Instead, they are complexity theoretic. By modeling the computational complexity of the learning process, we can, under standard assumptions, derive interesting results concerning the types of representations (or grammars) that are efficiently learnable. It is uncontroversial that the human capacity to learn is bounded by the same computational limitations that restrict human abilities in other cognitive domains. The interaction of this condition with the complexity of inducing certain types of representations from available data constitutes a fruitful object of study.

At present, this complexity problem admits only one type of solution; nearly all currently known efficient algorithms for grammar induction apply the same basic principles (Clark, 2010c). The grammars/representations are based on objectively observable features of the language itself. To the extent that we construct learning algorithms that accord with this requirement, we can achieve efficient learning for comparatively rich classes of expressive grammars. This approach offers the prospect of a computationally viable account of language acquisition that is based on general data-driven procedures for grammar induction.

For purposes of this discussion, we restrict ourself to the case of syntactic learning—grammar induction in a narrow sense. This is a significant idealization, as the child does, of course, learn much more than this. He/she learns, *inter alia*, to associate semantic interpretations, morphological structures, and phonological + phonetic representations to input strings. In addition, the child has access to much richer sources of information than the ones we consider here—the child observes utterances in a situational context and crucially the child can interact with his/her caregivers both linguistically and nonlinguistically, as well as with the environment. By limiting ourselves to syntax we can formulate the mathematical problems of complexity in acquisition in a tractable way. We do not have the formal resources to say anything interesting about the larger context of language acquisition. However, we will later briefly consider the problem of learning the syntax–semantics interface.

We will adopt the following standard definitions from formal language theory and (the early years of) generative grammar. We assume that we have a finite set of words, which we denote by Σ. A (formal) language is a subset of strings of finite length over Σ. We denote languages by L, with various subscripts where appropriate. This language will correspond to the set of grammatical sentences from some language. Drawing a sharp line between grammatical and ungrammatical sentences is fraught with methodological issues that we pass over here. These languages are, in general, infinite. Therefore, we need to consider finite representations or generative devices, which we call *grammars*, without thereby committing ourselves to any claims concerning their formal encoding. We are not assuming that these grammar are, for example, phrase structure grammars of a particular type. We consider deterministic finite automata, arbitrary programs, and so on, to be grammars, subject only to the constraint that they define for any string whether it is in, or out of the language being defined. We use G, again with appropriate subscripts, to denote these "grammars." In Chomsky's terms, these grammars might be called I-languages (although this expression is also used in different contexts), whereas what we call languages correspond to E-languages.

We assume that we have a language acquisition device (LAD), which takes as input some data and outputs a grammar of some kind. The input typically consists of a finite sequence of examples drawn from the language L, but we also consider cases where the learner has access to a more powerful source of information.

In the real world, the LAD is intended to correspond to some part of the brain/mind of the child. The input is the primary linguistic data, which consists of the sequence of sentences generated by the adults in the child's environment, during the years of language acquisition. The output is a grammar for the language that the child is exposed to. We take the LAD to be an algorithm, and we study the general properties of these algorithms using the mathematical tools of computer science. Clearly, such mathematical results can specify boundary conditions on learning procedures. Demonstrating the existence of algorithms that correctly perform the learning task under specified conditions offers a possible explanation of the way in which children acquire language. Conversely, showing the mathematical impossibility of an algorithm that implements this task will rule out certain types of explanation. Although the nature of language acquisition is clearly an empirical question, this sort of mathematical analysis is helpful—a conclusion that is uncontroversial in cognitive science.

The goal of computational learning theory applied to language acquisition is to apply mathematical analysis to the observed behavior of the LAD to shed light on its formal properties, and, by extension, to provide insight into the process of language acquisition. Notice how different this enterprize is from standard approaches to machine learning, where we are interested in designing algorithms to perform various learning tasks, or we wish to prove learnability results for such algorithms. In the current case, we have a device, the LAD, which we observe achieving a learning task. We are concerned to use our analytical tools to model its internal structure. This is reverse engineering, rather than software engineering. The difference profoundly alters the explanatory task.

1. The classic arguments from Gold-style learning

We first consider the classic Gold model of identification in the limit from positive data alone, as defined in Gold (1967). Gold considered a number of different models of learnability. We address only the one that is most relevant to language acquisition. Learning proceeds incrementally. The LAD is presented with an infinite sequence of unlabeled examples. At each step, the learner receives an example and must output a hypothesis. Informally, we say that the learner succeeds in learning the language if it converges (in a sense to be made precise shortly) to a correct hypothesis for every appropriate sequence. To give this some technical content we need to specify the convergence criteria, the set of appropriate sequences of examples, and a suitable definition of a correct hypothesis. To take the last item first, we say that a hypothesis—in this context, a grammar—is correct if it defines the correct language. It must be extensionally correct. So the Gold model is concerned largely with weak learning—the learning of the set of strings—rather than strong learning. The latter consists in learning the grammar that assigns the correct set of structural descriptions to strings. We return to this point later.

The convergence criterion for a given sequence requires there to be a finite point at which the hypothesis that the algorithm generates is correct, and that after this point the hypothesis does not change. This condition is equivalent to the requirement that the learner must only change its mind a finite number of times, and the final hypothesis must be correct. Note that the hypothesis must be exactly correct. The set of strings defined by the hypothesis needs to be identical to the one generating the examples.

The second crucial element of this model concerns the sequence of examples. Every child receives a different sequence of examples. Clearly, the examples have to depend in some way on the language that the child is trying to acquire. The Gold model uses the following minimal definition: a *presentation* of a language L is a sequence of elements of L (sentences in this case), such that only elements of L occur in the sequence and every element of L occurs at least once in the sequence. In this model, we say that a learner *identifies in the limit* a language if, on every possible presentation of the language, the learner converges to the target. A learner identifies in the limit a class of languages if, for every language in the class and for every presentation of that language, it converges to the right language.

A superfinite class of languages is any class of languages which contains all finite languages and at least one infinite language—that is to say any proper superset of the class of finite languages. The classes of regular languages, context-free languages, and so on, are all, by this definition, superfinite. The most influential result in Gold's article is the following. Gold showed that if a language class is superfinite, then it is not identifiable in the limit from positive data. As a consequence, none of the classes in the Chomsky hierarchy are learnable in this model.

One of the problems of this paradigm (lucidly pointed out by Johnson, 2004, among others) is that the requirement to learn under every possible presentation is too strong. The presentations are unrestricted. The learner must be able to learn even when the presentation is generated by an adversary. Because the adversary can defer key examples indefinitely, it is very hard for the learner to accurately infer which sentences are ungrammatical—although every positive example must appear eventually, there is no limit to when it appears. Thus the absence of an example, or set of examples, from a given finite sequence does not allow the learner to conclude that those examples are ungrammatical. Of course, from the point of view of the Gold learning paradigm, this unrealistic assumption is counterbalanced by another, equally unrealistic assumption, that the learner may learn as slowly as it likes, as long as it eventually converges. The conjunction of these two assumptions means that the Gold model is mathematically not trivial, but it renders it a poor model of language acquisition.

Under a more realistic, probabilistic learning model, this problem disappears.[3] Positive examples alone provide enough information, assuming that they are generated randomly and that the distribution of examples is not pathological. The mathematical results that establish this conclusion range from the famous, but limited early result of Horning (1969) to various extensions and improvements proposed in Angluin (1988) and Chater and Vitányi (2007). Some of these systems use inefficient algorithms for very large classes of languages, while others construct efficient learning algorithms for more restricted classes, as in Clark (2006), Clark and Thollard (2004), Palmer and Goldberg (2007), Ron, Singer, and Tishby (1998). On this approach, the frequency of occurrence of the strings provides an indirect form of negative evidence that greatly enhances the amount of data available to the learner.

We are focusing on another aspect of the argument, which has received much less attention than the problem of positive versus negative evidence. A key point in the Gold paradigm, and in models of learnability generally, is that learnability is a property of classes of languages, rather than of individual languages. Consider the following trivial LAD, which has a particular grammar G_0 encoded within it. This LAD completely ignores the input, and simply outputs G_0 in all cases. Clearly, in a vacuous mathematical sense, this is a learning algorithm for the language $L(G_0)$, although no learning, in the normal sense of that term, has taken place. It is hard to rule out these trivial algorithms when formalizing learnability. Therefore, learnability is always formalized in terms of classes of languages or grammars.

This purely formal idea has led to a concern with classes of languages that are learnable under a particular paradigm, rather than with the LAD itself. However, this perspective requires an additional move that is never made explicit. Suppose we know that $\mathcal{L}$ is the class of languages that is learnable by some LAD A. We need to determine what this fact allows us to infer about A. A naive answer would be that A must have "knowledge" of $\mathcal{L}$. In fact, this inference is unsound, although it is widely accepted.

We see the roots of this misconception even in Gold (1967). Concerning one of his candidate explanations of language acquisition, Gold suggests that it may be the case that

> The class of possible natural languages is much smaller than one would expect from our present models of syntax. That is, even if English is context-sensitive, it is not true that any context sensitive language can occur naturally. Equivalently, we may say that the child starts out with more information than that the language it is presented is context-sensitive.

The final sentence in this quote, is, as we shall see, incorrect. This idea crops up repeatedly in discussions of Gold's theorem in the context of language acquisition. Picking a representative example, Matthews (2001) claims that one of the ways in which a learner can succeed is when the class of natural languages is suitably constrained, and learners are knowledgeable of these constraints and furthermore exploit this knowledge in the course of language acquisition.

We begin our analysis of the misconception implicit in this view by defining our terms more precisely. Suppose we have some fixed LAD, which we denote by A. This is a function from samples to grammars. The *learnable class* of A, $\mathcal{L}(A)$, is the set of all languages L such that A identifies L in the limit, from every possible presentation of L. We define the *hypothesis class* of A, $\mathcal{H}(A)$, to be the set of all languages that A may hypothesize for any possible input. Clearly, both of these classes are well defined for any fixed A, although it may be difficult (or impossible) to actually determine $\mathcal{L}(A)$ for a particular choice of A. Equally clearly, these two classes are different. If A can learn a language L, then it must be able to output a representation for L, and so the languages defined by $\mathcal{H}(A)$ must include $\mathcal{L}(A)$. But crucially, these classes need not and often will not coincide.

There are two important respects in which these two classes may differ. The first and most direct way is that during the process of learning, the learner may hypothesize languages that it is not guranteed to learn. The second is that the learner may consider languages of unbounded complexity, while it may only be able to learn languages whose complexity is below some threshold. Let us start by giving a very simple artificial example of the first type of difference. We know that the class of context-sensitive languages is not identifiable from positive data alone. We define a simple learner whose hypothesis class is the class of all context-sensitive languages, but where the learnable class is just the class of all finite languages. This learner assumes that the input data are ordered in a helpful way with shortest strings first—that is to say all of the strings (if any) of length 1 occur before the strings of length 2 and so on. Therefore, when the learner sees the first string of length k, where all previous strings have been of length less than k, it assumes that any string that has not occurred so far, of length less than k, is not in the language. At this point, it changes its hypothesis to be the smallest context-sensitive grammar that includes all of the strings of length less than k that it has seen so far, and excludes all of the strings of length less than k that it has not seen. Now of course, the assumption that the learner makes is incorrect—in the Gold model, only some presentations will have this property. So we can alter the learner so that if it sees a shorter string appearing after a longer string, then it will revert to a more primitive algorithm, a rote learner, and output merely a hypothesis that consists just of the list of strings it has seen so far. Now it is clear that the learnable class in this case is the class of finite languages, as the criterion for convergence is that the algorithm must learn for every possible presentation of the language, which will include some that are not length ordered. The hypothesis class is the class of all context-sensitive languages, and it is easy to verify that for every context-sensitive language there are presentations of the language on which the learner will identify that language—namely those which *are* length ordered. Therefore, the hypothesis class—the class of context-sensitive languages—and the learnable class—the class of finite languages—differ significantly, the former properly including the latter.

For a less artifical example, consider the simple learner described in Clark and Eyraud (2007). This learns the class of substitutable context-free grammars (CFGs) from positive data alone. Substitutability is a simple closure property of languages—if lxr and lyr and $l'xr'$ are in the language, then so is $l'yr'$, for any strings l,r,x,y,l',r'. It is possible to prove that this property precisely characterizes the learnable class of languages of that algorithm under the identification in the limit paradigm. However, this property is, in general, undecidable for arbitrary context-free grammars. As far as we know, it is not possible to construct algorithms which limit their hypotheses to this class, while still being efficient. Reasonable algorithms for learning substitutable CFGs must be able to consider at least some candidate CFGs that are not substitutable. These will represent languages that are not learnable in the sense that they can be learned from all presentations, but they will be languages that can be learned from some presentations.

Returning to our main concern, language acquisition, it is appropriate to consider the interpretation of each of these two classes, and the relationship between them. $\mathcal{L}(A)$ depends on the asymptotic behavior of the algorithm under infinitely many data sets. It depends not just on A, but on the learning model defined by Gold. If we consider a different learning model, perhaps one where the examples are generated stochastically, then we might end up with a different class of learnable languages. It is more perspicuous to write $\mathcal{L}(A)$ as $\mathcal{L}(A, \mathrm{Gold})$—the class of languages learnable for A under the Gold paradigm. This class depends not just on A, but also on the modeling assumptions that are imposed by the Gold learning paradigm. But, it is entirely implausible to attribute knowledge of this learning paradigm to the child, even implicitly.

$\mathcal{H}(A)$, on the other hand, depends only on A. It is the range of outputs that A will produce for any input, and so it directly represents the biases of the learner. This is the class that we need to identify to understand the design of A.

Keeping in mind this distinction between the hypothesis class and the learnable class, we reexamine the arguments from Gold's theorem on learnability from positive data. Gold's theorem tells us that the learnable class for a LAD A cannot be superfinite (and similarly, more powerful negative results show that this class must be restricted in some way). However, it tells us nothing at all about the hypothesis class of A. As $\mathcal{H}(A)$ contains $\mathcal{L}(A)$, knowing that the latter is small has no consequences for the former.

In particular, these results do not tell us, contra Gold (1967), that the learner must restrict the hypothesis class in any way. Yang (2008) expresses the standard view, derived from Gold.

> Learning is not possible unless the hypothesis space is tightly constrained by prior knowledge, which can be broadly identified as Universal Grammar.

Yang is correct in suggesting that the restriction on the hypothesis space, $\mathcal{H}(A)$, is our primary concern, but his claim that restricting this class is necessary to achieve learnability is incorrect. The source of this misconception is not hard to identify. When proving the proposition that an algorithm A_1 can learn a class of languages $\mathcal{L}_1$, we need to prove that $\mathcal{L}(A_1) \supseteq \mathcal{L}_1$. In this case, we are not interested in the situation where the learner is given a presentation of a language that is not in $\mathcal{L}_1$. Moreover, to simplify the analysis and make the proof easier, we tend to design an algorithm that is explicitly designed to learn only languages in $\mathcal{L}_1$. In many cases, the learner can exploit this knowledge and restrict the set of hypotheses to $\mathcal{L}_1$. An example is learning reversible languages (Angluin, 1982), where it is easy to discern whether a given deterministic finite automaton is reversible or not. This is a fact that can accelerate the learning process, and lead to a shorter and more intuitive proof of convergence.

The confusion inherent in attributing knowledge of the learnable class to the algorithm is exacerbated by a proof strategy called Identification by Enumeration (IBE). In this proof strategy, we assume that the learner is equipped with an enumeration of the grammars in the learnable class $G_1, G_2,$. The learner then returns the first element in the enumeration that is compatible (in a suitable technical sense) with the data seen so far. This is a useful method for proving learnability of various classes. However, it is just one of several possible approaches, and it works only for some cases.[4] IBE learners are a type of learner where the knowledge of the learnable class is represented in a partcularly explicit way, and is equal to the hypothesis class. The fact that there are some learners for some classes that employ IBE and explicitly restrict their hypothesis class to the learnable class does not motivate the conclusion that all learners operate in this way. Other learners may explicitly represent a hypothesis class that is significantly larger than the learnable class, or implicitly define a larger hypothesis class without explicitly representing it. The same confusion affects other learnability frameworks, specifically probabilistic learning paradigms like PAC learning (Valiant, 1984).

Controlling the size of the hypothesis class, or *capacity control* as it is sometimes called, is one way to achieve generalization, but there are other methods that use much larger and less constrained hypothesis classes. For example, *k*-nearest neighbor classifiers use hypothesis classes which are unbounded in terms of their capacity[5] while being asymptotically optimal in many cases up to a constant factor (Cover & Hart, 1967).

This returns us to a point that we made earlier. The main goal of computational learning theory as it is standardly deployed is to understand concepts like generalization so that they can be used to design better learning algorithms (e.g., the development of Support Vector Machines, Cristianini & Shawe-Taylor, 2000). These tools cannot be applied in reverse to draw strong conclusions about the nature of the hypothesis class. So, the theoretical analysis of PAC learning produced the result that concept classes can be learned if and only if they have finite VC dimension, one of the great achievements of computational learning theory. But, this result does not support an inference to the conclusion that the hypothesis class must have finite VC dimension.[6] A particularly clear illustration of this point is the class of finite languages, where the representation of each such language is simply a finite list of its grammatical sentences. A trivial rote-learning algorithm exists for this class—one that simply memorizes each sentence that it observes. Such an algorithm need have no prior bound on the cardinality of the language. Note, however, that the class of finite languages has infinite VC dimension, rendering it a PAC-unlearnable class. But, for any bounded subclass of the finite languages which has finite VC dimension and is therefore learnable, a trivial rote learner will succeed. In this case, the hypothesis class of the learner is clearly the class of all finite languages, and the learnable class must be very much smaller. It follows that any inference from the boundedness of the learnable class to the putative boundedness of the hypothesis class does not hold.

The missing premise needed to sustain this inference is the claim that the hypothesis class is restricted to the learnable class. Learners with this property have been studied before. In the inductive inference community, they are known as *prudent* learners (Fulk, 1990). We formulate this as the *Prudent Learner Premise* (PLP): the LAD is a prudent learner. It only considers hypotheses that represent languages within the learnable class under the Gold paradigm. In this case, we do obtain the identity $\mathcal{L}(A) = \mathcal{H}(A)$.[7] Although the PLP is ultimately an empirical premise, it would be difficult to find experimental evidence for it. To support the PLP we need to show that under every possible input, the hypothesis considered by the child is in the learnable class. This involves demonstrating that all of the intermediate states of the child's knowledge represent languages that could be learned under all presentations. Simply looking at patterns of child speech will not do: child language production is not a direct reflection of competence as measured for example in comprehension (Benedict, 1979; Hendriks & Koster, 2010).

Moreover, the PLP implicitly incorporates the assumptions of the Gold model. As we noted above, the reality of the language-acquisition environment falsifies these assumptions. So, for example, if the input is the sort of pathological presentation allowed by the "adversarial" Gold model, then it is exceedingly rare and unlikely to be produced from the sorts of natural distributions that children are typically exposed to.[8] Therefore, evidence from the behaviour of actual children in natural learning environments is insufficient to support the PLP.[9]

From a formal point of view, there are also good reasons for thinking that the PLP does not hold. As we observed previously, in many algorithms for context-free learning, the learnable class is undecidable (Clark & Eyraud, 2007). For an arbitrary context-free grammar it is not, in general, possible to compute whether it is in the learnable class or not, and as a result it may not be possible for the learner to restrict the hypothesis class to the learnable class.

It should be clear, then, that arguments which depend on the PLP do not yield significant insights into the actual process of language acquisition. To obtain such insight, we need to turn to another branch of learning theory for constraints that may help us to understand the boundary conditions on the design of the learner. We need to examine the computational complexity of learning.

2. Computational complexity

The study of computational complexity supplies a second source for negative conclusions concerning learnability. Fairly soon after the study of complexity began, results appeared suggesting serious computational problems with learning (Gold, 1978), and work since then has shown significant difficulty with learning classes of representations in the Chomsky hierarchy, under a variety of learning models (Abe & Warmuth, 1992; Angluin & Kharitonov, 1995; Kearns & Valiant, 1994). These results indicate that even in situations where there is sufficient data for the learner to identify the correct hypothesis, finding it may be a computationally intractable task (i.e., NP-hard or intractable under other complexity assumptions, typically derived from cryptography).

It is uncontroversial that humans, adults or infants, have limited computational resources. Rooij (2008) state this as the *Tractable Cognition Thesis*:

> Human cognitive capacities are constrained by the fact that humans are finite systems with limited resources for computation.

This restriction on computational power applies to every aspect of cognition. In the domain of language, it will limit the power of the formalisms that we can acquire or use. Giving this thesis technical content involves making assumptions about an appropriate model of computational tractability. Rooij (2008) argues for fixed parameter tractability

(FPT), a more sophisticated and nuanced version of the standard asymptotic worst-case analysis used for the last 50 years. Combining the Tractable Cognition Thesis, and the variety of negative results based on computational complexity for the learnability of various representations, we arrive at a serious constraint on possible models of language acquisition, that is very different from the information theoretic constraints we examined earlier.

But, if learning classes of representations such as the class of deterministic finite state automata (DFAs) in Kearns and Valiant (1994) is so hard, then we find ourselves in an apparent paradox. Children do, in fact, learn target representations that are much more powerful than DFAs. Indeed, the Kearns and Valiant (1994) result concerns acyclic DFA—which generate only finite languages!

This paradox is only apparent. The fact that a problem is computationally hard does not entail that all instances of the problem are hard. Often, only a small proportion of the class actually presents computational difficulties. For example in 3-SAT, many problems either have so many constraints that they are clearly unsolvable, or they have so few that they admit of multiple solutions. It is only when the number of constraints falls within a narrow range that the problem is hard. As Impagliazzo (1995) says, in a well known overview article:

> There is a large gap between a problem not being easy and the same problem being difficult. A problem could have no efficient worst-case algorithm but still be solvable for "most" instances, or on instances that arise in practice. Thus a conventional completeness result can be relatively meaningless in terms of the "real life" difficulty of the problem, since two problems can both be NP-complete, but one can be solvable quickly on most instances that arise in practice and the other not.

These considerations suggest a solution to our conundrum. We need to identify the subset of instances (grammars, say) for which the learning problem is tractable. The negative learnability results that we discussed at the beginning of the section do not, in themselves, indicate whether this subset is large or small. Is it the case that nearly all DFAs are learnable, with only a few pathological exceptions? Or are DFAs in general hard to learn? The main issue that we are addressing here is how to answer this question for richer families of representations that are suitable for natural language description (as it is clear that DFAs are inadequate models of the syntax of natural languages).

Let us begin with the learnability of regular languages, as this part of the theory of learning is well understood (although far from complete). There are, broadly speaking, two kinds of representation for this class of languages, where these have sharply different learnability properties. The first is the class of deterministic finite state automata (DFA) and the second is the larger class of general nondeterministic finite state automata (NFA). Every DFA is also an NFA, but NFAs are more expressive. Although both kinds of automata define the same class of languages, the regular languages, the class of NFAs often allow for a more compact representation (Meyer & Fischer, 1971).

The learnability properties of the two classes are very different. For DFAs, if we have a sufficiently good source of information—a minimally adequate teacher—then efficient learning algorithms can be constructed, that are polynomial in the size of the representation and the data (Angluin, 1987). In this learning paradigm, the learner is not purely passive, but it can interact with the teacher by asking membership queries. The learner can ask whether a particular string is grammatical or not. Without such a source of information, if the learner is merely passively observing sentences, then the Kearns and Valiant (1994) results show that learning is hard, even when the data consists of both positive and negative examples.

By contrast, even with a teacher who can answer these queries, the class of NFAs is not efficiently learnable (Angluin & Kharitonov, 1995). We can now see one of the advantages of this complexity-based analysis. Rather than concerning ourselves with broad and abstract criteria for learnability, specified in terms of classes of languages, we focus on the choice of representation class and the consequences of this choice for possible learning algorithms. This approach yields direct insight into the nature of the LAD. Specifically, it illuminates the types of representations that the LAD might produce, and the nature of the algorithms it might employ.

Information theoretic results of the Gold type, when applied to the the classes of the Chomsky hierarchy—the regular, context-free, and context-sensitive languages—either imply that all three of them are learnable, as in the case for learning from positive and negative data, or that none of them are learnable, as in the case of positive-only learning. Similarly, any finite collection of languages can be learned from positive-only data (Gold, 1967). But, this result tells us nothing about what these languages are or how they might be represented. Other infinite learnable classes, however, may run orthogonally to the Chomsky hierarchy, consisting of some but not all finite languages, and some infinite languages. Shinohara (1994) is perhaps the most powerful of these results, but again the class of representations[10] is so large that it gives no guidance into what they might be. It is important to note that these results are not based on computationally efficient algorithms.

It is only when they are augmented with constraints on computational complexity that they start to provide some more detailed insight. The result in Angluin (1982) offers a more interesting perspective. It tells us that learnability can depend on particular properties of the representations themselves. Determinism is the basis for learnability of DFAs. This idea is developed in a slightly different way for richer formalisms. In contrast, the negative results for NFAs (Angluin & Kharitonov, 1995) seem to indicate that the greater compactness of nondeterministic representations causes complexity problems.

Given the limited descriptive power of DFAs, it is important to see how these insights can be generalized to more powerful formal systems. The crucial learnability feature of DFAs, we argue, is not their determinism *per se*, but the deeper property of their *objectivity*. An automaton is objective in this sense if its states satisfy certain criteria of identifiability and distinguishability relative to distributional patterns in a data set. The states in the minimal DFA for a regular language correspond to equivalence classes of strings under the Myhill–Nerode congruence. Determinism follows from this property. We can generalize objectivity to achieve important positive learning results for more powerful grammar formalisms (Clark, 2006, 2010b; Clark & Eyraud, 2007; Yoshinaka, 2011).

Objectivity is naturally extended to context-free grammars whose nonterminals correspond to congruence classes under a different congruence relation, the relation of complete mutual substitutability, first studied by distributional linguists in the American structuralist tradition (Chomsky, 1959; Harris, 1954; Wells, 1947). This has given rise to a variety of simple algorithms using this basic representational assumption (Clark, 2010a; Clark & Eyraud, 2007). As DFAs and NFAs can be converted easily to context-free grammars, we can consider the relationship more precisely. The MAT learner in Clark (2010a) can learn all regular languages, so we can consider the context-free grammars output by this algorithm as another representation for regular languages. These grammars are deterministic in a particular sense—for any string, there is only one nonterminal that can generate that string, and there cannot be two productions with the same right-hand side. Thus, the grammars have a certain bottom-up determinism. Moreover, the congruence classes can be used to define an automaton, and this automaton will be deterministic.

Yoshinaka (2011) uses this approach to move to the next stage, when he demonstrates that a class of mildly context-sensitive grammars can be efficiently learned by algorithms designed on the basis of objectively identifiable nonterminals. This is a subclass of Multiple Context-Free Grammars (Seki, Matsumura, Fujii & Kasami, 1991), which have been shown to be equivalent to Minimalist Grammars (Michaelis, 2001; Stabler, 1997) under certain conditions. While the subclass learnable by the algorithm in Yoshinaka (2011) is extremely limited, this is a highly suggestive result. It points to an interesting confluence between the types of grammars considered to be adequate by linguists, and the classes of representations learnable through distributional means.

The use of congruence classes in these algorithms, although mathematically convenient and close to the original conceptions of structuralism (Harris, 1954, 1955) is linguistically inadequate, as (Chomsky, 1955, 1959) realized early on. The obvious fact is that words are ambiguous and differ from each other in numerous subtle ways. Thus, the requirement of exact identity of distribution is far too strict to impose on the syntactic categories of a natural language. It entails that the congruence classes consist only of individual words or sequences.

It is possible to modify the models. Rather than partitioning words and strings into separate nonoverlapping classes, it is more natural to separate them into classes that may overlap and include one another, an assumption that more faithfully represents the reality of syntactic and lexical categories. Clark (2010c) shows that this general approach can also be applied to a family of lattice-based grammars, which use symbols (nonterminals) that are no longer atomic, as in CFGs, but are specified as sets of features arranged in a hierarchy (a lattice in this case). This gives them something of the flavor of feature-based representations such as GPSG (Gazdar, Klein, Pullum & Sag, 1985) and HPSG (Pollard & Sag, 1994). The features used in the lattice-based grammars are very simple shallow distributional features. They are different in kind from the standard notion of a syntactic feature. However, traditional syntactic features can be modeled as bundles of these more primitive distributional features. These approaches, while no longer deterministic, are still learnable because the primitive elements of the model correspond to a canonical structure of the language, a structure that rather than being arbitrarily imposed on the language, is unique and well defined. It is based on intrinsic properties of the language itself.

Again the common theme here is *objectivity*. As has been noted by many linguists over the years, grammars are radically undetermined both by the primary and secondary linguistic data (Chomsky, 1966, p. 22). By adopting objective representations (in the narrow sense intended here), we at least partially circumvent the problem of underdetermination that infects linguistic theory. In restricting ourselves to representations that are based directly on the observable data, we constrain the process of theory formation significantly.

What conclusions about the LAD can we draw from these positive learning results? One claim that is not justified by these results, and which we should therefore resist, is that the learner must have knowledge of which languages will be learnable. For reasons similar to those that we gave in our criticism of the Gold paradigm, we have no grounds for attributing to the LAD knowledge of the properties of the target class that guarantee its learnability.

We can relate this debate to the Bayesian paradigm, which enjoys considerable popularity within current work in cognitive science (Griffiths, Kemp & Tenenbaum, 2008). In Bayesian modeling, a prior over a hypothesis space is explicitly specified for the learner, who uses a general-purpose inference scheme (typically a Markov Chain Monte Carlo method) to infer either the most likely grammar, given a data set, or, alternatively, a posterior distribution over the set of possible grammars. These learners are to a certain extent *ideal* learners—under mild assumptions they are guaranteed to converge optimally, given unlimited computation, to the right answer for any grammar in the hypothesis class. In this case, we can identify the hypothesis class as the set of all hypotheses with nonzero prior probabilities.

In many cases, the hypothesis class is very large, generally corresponding to a given level of the Chomsky hierarchy (Johnson, Griffiths & Goldwater, 2007). We know perfectly well, from the negative learnability results cited above, that the learning algorithms will not converge within some fixed time limit for every problem in the given class. For the hard problems, they will fail to converge. Fixing a limit on the amount of computational time that the learner is allowed to use, thereby converting the ideal learner into a real computational system, will create a learnable subclass of the hypothesis class. Exactly which subclass depends on the details of the particular sampling algorithm that is used, and studying the implications of particular choices could be a promising area of future research. So with Bayesian methods applied in this way, nonideal learners again have a learnable class that is significantly smaller than the hypothesis class which is explicitly represented. The learnable class here arises out of the interaction between a soft set of constraints on hypotheses, in the form of the prior, and computational constraints on the inference procedure. The end result will be a class of languages that may and indeed must be very constrained and limited—but those limitations need not be part of the prior constraints themselves, and as a result cannot legitimately be considered as a property of the learner itself.

The standard solution to learnability problems within mainstream generative grammar has been the Principles and Parameters model (Chomsky, 1981), which hypothesized that the learner had innate knowledge of a finite class of grammars parameterized by a small number of binary parameters. It was thought that the finiteness of the set of possible grammars trivialized the learning problem. In the light of the complexity problems discussed here, where the problematic classes are typically finite classes of simple grammars, specified by a finite set of binary parameters, it should be clear that such finiteness assumptions do not completely solve the learning problem (Niyogi & Berwick, 1996).[11] Fodor and Sakas (2004) remark cogently that

> parameter setting as a process has become a source of problems rather than solutions.

The computational arguments make it clear that something else other than finiteness is necessary for learnability; finiteness, which was thought to be of central importance for learnability turns out to be neither necessary nor sufficient.[12]

These considerations strongly motivate a shift in emphasis away from efforts to define the class of possible human languages through identification of learnable language classes, to the exploration of types of representations that are efficiently learnable because of their objectivity. This approach also offers a methodological advantage. Regardless of how one regards the status of specific linguistic theories and grammar formalisms, it is clear that we lack the necessary evidence to determine the syntactic structures that are actually encoded in our brains (and we will continue to be in this position for the foreseeable future).[13]

Before concluding, we should address at least one factor that we have neglected so far —the issue of bounds on memory. Just as the learner has limited computational resources in terms of the number of computational steps that it can apply, so too does the learner have limited storage capacity. Considerable research has been directed at the effect in various contexts of limiting the number of examples that a learner can store (Fodor & Sakas, 2005; Nowak, Komarova & Niyogi, 2001; Wexler & Culicover, 1980). Indeed, in many such models a completely memoryless learner is considered—a learner that cannot remember any preceding utterance. First, is important to distinguish two senses of the word "memory." In the abstract computational sense, this merely refers to the storage ability of the abstract computer; the infinite tape of the Turing machine, whereas in the psychological sense it tends to be used for short-term or long-term declarative memory that is accessible to consciousness. While it is certainly true that children do not memorize the complete linguistic data that they are exposed to, there is ample evidence that children are extremely sensitive to the frequencies of occurrences of words and fragments of words in the input. Furthermore, the total size of the linguistic data that the child is exposed to, considered as a sequence of words or even phonemes, is tiny compared with the estimated information storage capacity of the brain. A model that places some significant bounds on the storage capacity of the algorithm seems for these reasons to be empirically unmotivated. For more detailed discussion see (Clark & Lappin, 2011, pp. 144–155).

3. Weak and strong learning

One might object to the line of research we have been considering here as, at best, tangentially relevant to linguistics. It is concerned largely with weak learnability, the task of learning the set of strings in a language, rather than the acquisition of the mapping between syntax and semantics mediated by appropriate structural descriptions (Berwick, Pietroski, Yankama & Chomsky, 2011). We describe the latter task as strong learning, on analogy with the distinction between weak and strong generative capacity (Miller, 1999).

The issue is rendered even more complicated by the fact that regular languages, especially as represented by DFAs, have no associated concept of strong learning. It was only after positive learning results were achieved for classes beyond the regular languages that we could properly formulate the question of strong learning. It is natural that we should make progress in weak learning before strong learning, as strong learning is harder than weak learning.

There are a number of different models for learnability in this context that depend on the inputs that are available to the child. We assume a fairly standard architecture, where a grammar generates structural descriptions that then generate independently pairs of surface strings and semantic interpretations. Therefore, we have three classes of objects that we might consider as input to the learner, either separately or in combination: the structural descriptions, the surface strings, and finally the semantic interpretations. These differ in the degree of accessibility both to the linguist/native speaker, and to the child. The child can observe the surface strings, imperfectly to be sure, because of limitations in the perceptual system, noise and so on, but nonetheless directly. The child has, at least in the early phases, no direct information about the meaning of a string. As its cognitive abilities mature, and as its knowledge of the language increases, it becomes able to infer the meaning of utterances from the situational context, partial knowledge of the syntax of the sentence, and some knowledge of the speaker's intentions. However, this is clearly not happening until comparatively late in the language-acquisition process. On the other hand, linguists and adult speakers know what the available interpretations for a given sentence are, albeit without conscious access to the exact details of its semantic representation. Finally, neither the child nor the adult (nor the linguist) has knowledge of the structural descriptions. They are theoretical entities that we hypothesize to explain the relation between sound and meaning. Therefore, there are two relevant models for the input to the child. Either the child receives only the surface string of words or sounds, or the child receives a pair of the surface string and its meaning.[14] When we consider the output of the learning, algorithm again there are several different possibilities. One sensible, but limited model is the classic weak learning model, as discussed earlier—the learner is required to learn a grammar that generates the correct set of surface strings.

It may seem more plausible to consider a learning model where the learner is presented with pairs of surface strings and meanings or semantic interpretations, something closer to the model considered in Wexler and Culicover (1980). The output will then be a grammar, which can generate the set of grammatical sentences together with their semantic interpretations. Yoshinaka and Kanazawa (2011) take the first steps in showing how distributional learning could operate in this model, using a very general formalism called Abstract Categorial Grammars. While again still limited, this shows that algorithms using the same approach proposed here can also apply in this richer framework.

It seems unlikely that the same assumptions will be appropriate for modeling the entire process of language acquisition from birth to linguistic competence. Rather it is more insightful to assume that later on in the developmental process, the learner starts to be able to extract more semantic information from the environment, while at the beginning, this ability is limited or nonexistent. There is a third possibility, which is that a learner using only weak inputs, may nonetheless be able to infer some structures that are suitable for providing an interface to semantics. This could be called *weak–strong* learning—the input to the learner is weak (only surface strings), but the output must be strong (in the sense of a grammar that defines an appropriate set of structural descriptions. To formalize this, we need to define an appropriate type of structural description. Some research along these lines is presented in Clark (2011).

4. Conclusion

We have argued that there are two main problems to be addressed in modeling the task of language acquisition using the tools of learning theory. One is an information theoretic problem concerning the available data that is closely related to the question of the poverty of the stimulus, as it has traditionally been discussed. The second is a complexity theoretic issue, which, by contrast, has largely been neglected. The basic difficulty with classical formulations of the first problem is that they do not tell us anything of genuine substance concerning the computational processes involved in actual human language learning. Positive examples give us enough information to support learning, unless the distribution of data samples is pathological. But, there is no reason to think that it is. Moreover, information theoretic arguments for the poverty of the stimulus suffer from a serious conceptual confusion. They systematically confound the hypothesis space with the class of learnable languages. Consequently, they fail to provide insight into the properties of the learner.

Results derived from complexity considerations offer a more promising source of insight into the boundary conditions of learning. The positive results that we have surveyed here indicate that we should be modeling the target of language acquisition as an objective representation—a grammar whose primitive symbols can be directly identified from the measurably observable properties of the language itself. Such representations are efficiently learnable.[15] Therefore, these results suggest that the types of grammars that have been posited within different variants of the Principles and Parameters program are not plausible candidates for learnable representations. This is because the values of the proposed parameters are remote from the data of natural language, and so they cannot be efficiently estimated or learned. In positive terms, although the most elementary formalisms initially considered on the objective representation approach are too weak, more sophisticated grammars seem to achieve the right level of expressive power for capturing the properties of natural language syntax, while remaining efficiently learnable.

Acknowledgments

We thank the participants in our 2007 and 2011 Linguistics Summer Institute courses, and in our NASSLLI 2010 course, as well as the audiences who attended numerous talks in which we presented ideas contained in this article, for many helpful comments and suggestions. We would also like to thank Nick Chater, Makoto Kanazawa, and an anonymous reviewer for helpful suggestions that have greatly improved the article.

Notes

1. On the relation between information theory and machine learning see (MacKay, 2003).
2. By finite language, we mean merely a language that consists of a finite number of strings; we use the term regular language to refer to finite-state languages which may be infinite
3. See Clark and Lappin (2011) for detailed discussion of this point, and suggestions for alternative learning models.
4. IBE is sometimes claimed to be maximally powerful. In fact, there are classes that are not learnable through IBE, but are learnable through other procedures. However, there are more elaborate learning models that are enumerative and can work in more general settings (De Jongh & Kanazawa, 1996).
5. They have infinite VC dimension.
6. See for example Lemma 3.10 of (Haussler & Kearns, 1991).
7. Under certain conditions, it can be shown that every learnable class of languages can be learned by a learner that is prudent in this sense (Fulk, 1990). These results do not concern efficient learning. This does not, of course, mean that all learners are prudent.
8. We can observe the properties of naturally occurring samples of the primary linguistic data, and from these deduce properties of the distributions from which they are drawn. For example, we can have good estimates of the distributions of lengths of utterances, of lexical frequencies, and, from annotated corpora, of the relative frequency of various construction types, morphosyntactic features and so on (Cohen & Smith, 2012; MacWhinney, 1995; Sagae, Davis, Lavie, MacWhinney & Wintner, 2007).
9. We would like to thank Robert Matthews for helpful discussion of this point.
10. It coincides with the class of context-sensitive languages.

11. In recent articles, proving the hardness of certain learning problems, the proof is based on constructing a finite set of 2^n grammars, parameterized by n binary parameters, and showing that learning this class would involve solving some cryptographic problem. Works using this technique include (Kearns et al., 1994). Alternatively, we have some finite class of automata generating binary strings that can be trivially encoded by a small set of binary parameters.
12. See (Clark & Lappin, 2011; Lappin & Shieber, 2007) for detailed discussion of this point.
13. Although there has been a great deal of interesting work in the last decade on the neural infrastructure for language processing, the results obtained so far are still very far from shedding direct light on the cognitively real representations that emerge in the brain (Gouvea, Phillips, Kazanina & Poeppel, 2010; Marantz, 2005).
14. Models where the learner receives access to the trees are widely used in computational linguistics, but they are irrelevant here.
15. There are a number of subtle details involved in precisely defining the notion of efficient learnability. See (Case & Kötzing, 2009; Higuera, 1997; Pitt, 1989) for discussion.

References

Abe, N., & Warmuth, M. K. (1992). On the computational complexity of approximating distributions by probabilistic automata. *Machine Learning*, *9*, 205–260.

Angluin, D. (1982). Inference of reversible languages. *Journal of the ACM*, *29*(3), 741–765.

Angluin, D. (1987). Learning regular sets from queries and counterexamples. *Information and Computation*, *75*(2), 87–106.

Angluin, D. (1988). *Identifying languages from stochastic examples (Tech. Rep. No. YALEU/ DCS/RR-614)*. New Haven, CT: Yale University, Dept. of Computer Science.

Angluin, D., & Kharitonov, M. (1995). When won't membership queries help? *Journal of Computer and System Sciences*, *50*(2), 336–355.

Benedict, H. (1979). Early lexical development: Comprehension and production. *Journal of Child Language*, *6*(2), 183–200.

Bertolo, S. (Ed.) (2001). *Language acquisition and learnability*. Cambridge: Cambridge University Press.

Berwick, R., Pietroski, P., Yankama, B., & Chomsky, N. (2011). Poverty of the stimulus revisited. *Cognitive Science*, *35*, 1207–1242.

Case, J., & Kötzing, T. (2009). Difficulties in forcing fairness of polynomial time inductive inference. In R Gavaldà et al. (Ed.), *20th international conference on algorithmic learning theory* (pp. 263–277). Berlin: Springer-Verlag.

Chater, N., & Vitányi, P. (2007). Ideal learning of natural language: Positive results about learning from positive evidence. *Journal of Mathematical Psychology*, *51*(3), 135–163.

Chomsky, N. (1955). *The logical structure of linguistic theory*. Unpublished doctoral dissertation, MIT, Cambridge, Massachusetts.

Chomsky, N. (1959). Review of Joshua Greenberg's essays in linguistics. *Word*, *15*, 202–218.

Chomsky, N. (1966). *Topics in the theory of generative grammar*. Mouton: Berlin.

Chomsky, N. (1981). *Lectures on government and binding*. Dordrecht: Foris Publications.

Clark, A. (2006). PAC-learning unambiguous NTS languages. In *Proceedings of the 8th International Colloquium on Grammatical Inference (ICGI)* (pp. 59–71). Berlin: Springer.

Clark, A. (2010a, September). Distributional learning of some context-free languages with a minimally adequate teacher. In J. Sempere & P. Garcia (Eds.), *Grammatical Inference: Theoretical Results and Applications. Proceedings of the International Colloquium on Grammatical Inference* (pp. 24–37). Valencia, Spain: Springer.

Clark, A. (2010b, July). Efficient, correct, unsupervised learning of context-sensitive languages. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning* (pp. 28–37). Uppsala, Sweden: ACL.

Clark, A. (2010c, October). Towards general algorithms for grammatical inference. In M. Hutter et al. (Ed.), *Proceedings of the Conference on Algorithmic Learning Theory* (pp. 11–30). Canberra, Australia: Springer. (Invited Paper)

Clark, A. (2011). A language theoretic approach to syntactic structure. In M. Kanazawa et al. (Ed.), *Proceedings of the 12th Meeting on the Mathematics of Language (MOL)*. Nara, Japan: Springer.

Clark, A., & Eyraud, R. (2007, August). Polynomial identification in the limit of substitutable context-free languages. *Journal of Machine Learning Research*, *8*, 1725–1745.

Clark, A., & Lappin, S. (2011). *Linguistic nativism and the poverty of the stimulus*. Oxford: Wiley-Blackwell.

Clark, A., & Thollard, F. (2004). PAC-learnability of probabilistic deterministic finite state automata. *The Journal of Machine Learning Research*, *5*, 473–497.

Cohen, S., & Smith, N. (2012). Empirical risk minimization for probabilistic grammars: Sample complexity and hardness of learning. *Computational Linguistics*, *38*(3), 1–48.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, *13*(1), 21–27.

Cristianini, N., & Shawe-Taylor, J. (2000). *Support vector machines*. Cambridge: Cambridge University Press.

De Jongh, D., & Kanazawa, M. (1996). Angluin's theorem for indexed families of re sets and applications. In *Proceedings of the Ninth Annual Conference on Computational Learning Theory* (pp. 193–204). ACM.

Fodor, J., & Sakas, W. (2004). Evaluating models of parameter setting. In *Proceedings of the 28th Annual Boston University Conference on Language Development* (pp. 1–27).

Fodor, J., & Sakas, W. (2005). The subset principle in syntax: Costs of compliance. *Journal of Linguistics*, *41*(3), 513–570.

Fulk, M. (1990). Prudence and other conditions on formal language learning. *Information and Computation*, *85*(1), 1–11.

Gazdar, G., Klein, E., Pullum, G., & Sag, I. (1985). *Generalised phrase structure grammar*. Oxford: Basil Blackwell.

Gold, E. M. (1967). Language identification in the limit. *Information and Control*, *10*, 447–474.

Gold, E. M. (1978). Complexity of automaton identification from given data. *Information and Control*, *37*(3), 302–320.

Gouvea, A., Phillips, C., Kazanina, N., & Poeppel, D. (2010). The linguistic processes underlying the p600. *Language and Cognitive Processes*, *25*(2), 149–188.

Griffiths, T., Kemp, C., & Tenenbaum, J. (2008). Bayesian models of cognition. In R. Sun (Ed.), *Cambridge handbook of computational cognitive modeling* (pp. 59–100). Cambridge: Cambridge University Press.

Harris, Z. (1954). Distributional structure. *Word*, *10*(2–3), 146–162.

Harris, Z. (1955). From phonemes to morphemes. *Language*, *31*, 190–222.

Haussler, D., & Kearns, M. (1991). Equivalence of models for polynomial learnability. *Information and Computation*, *95*(2), 129–161.

Hendriks, P., & Koster, C. (2010). Production/comprehension asymmetries in language acquisition. *Lingua*, *120*(8), 1887–1897.

de la Higuera, C. (1997). Characteristic sets for polynomial grammatical inference. *Machine Learning*, *27*(2), 125–138.

Horning, J. J. (1969). *A study of grammatical inference*. Unpublished doctoral dissertation, Computer Science Department, Stanford University, California.

Impagliazzo, R. (1995). A personal view of average-case complexity. In *Proceedings of Tenth Annual IEEE Structure in Complexity Theory Conference* (pp. 134–147). IEEE.

Jain, S., Osherson, D., Royer, J. S., & Sharma, A. (1999). *Systems that learn: An introduction to learning theory* (2nd ed.). Cambridge, Massachusetts: MIT Press.

Johnson, K. (2004). Gold's theorem and cognitive science. *Philosophy of Science*, *71*(4), 571–592.

Johnson, M., Griffiths, T., & Goldwater, S. (2007). Bayesian inference for PCFGs via markov chain monte carlo. In *Proceedings of NAACL-HLT* (pp. 139–146). ACL.

Kearns, M., & Valiant, G. (1994, January). Cryptographic limitations on learning boolean formulae and finite automata. *Journal of the ACM*, *41*(1), 67–95.

Kearns, M., Mansour, Y., Ron, D., Rubinfeld, R., Schapire, R., & Sellie, L. (1994). On the learnability of discrete distributions. In *Proceedings of the 25th Annual Acm Symposium on Theory of Computing* (pp. 273–282). ACM.

Lappin, S., & Shieber, S. (2007). Machine learning theory and practice as a source of insight into univeral grammar. *Journal of Linguistics*, *43*, 393–427.

MacKay, D. (2003). *Information theory, inference, and learning algorithms*. Cambridge: Cambridge University Press.

MacWhinney, B. (1995). *The CHILDES project: Tools for analyzing talk*. Hillsdale, NJ: Lawrence Erlbaum.

Marantz, A. (2005). Generative linguistics within the cognitive neuroscience of language. *Linguistic review*, *22*(2/4), 429.

Matthews, R. (2001). Cowie's anti-nativism. *Mind & Language*, *16*(2), 215–230.

Meyer, A., & Fischer, M. (1971). Economy of description by automata, grammars, and formal systems. In *Proceedings of the IEEE Twelfth Annual Symposium on Switching and Automata Theory* (pp. 188–191).

Michaelis, J. (2001). Transforming linear context-free rewriting systems into minimalist grammars. In P. de Groote, G. Morrill, & C. Retoré (Eds.), *Logical aspects of computational linguistics* (pp. 228–244). Berlin: Springer.

Miller, P. (1999). *Strong generative capacity: The semantics of linguistic formalism*. Stanford: CSLI Publications.

Niyogi, P., & Berwick, R. (1996). A language learning model for finite parameter spaces. *Cognition*, *61*(1–2), 161–193.

Nowak, M., Komarova, N., & Niyogi, P. (2001). Evolution of universal grammar. *Science*, *291*(5501), 114–118.

Palmer, N., & Goldberg, P. (2007). PAC-learnability of probabilistic deterministic finite state automata in terms of variation distance. *Theoretical Computer Science*, *387*(1), 18–31.

Pitt, L. (1989). Inductive inference, DFAs, and computational complexity. In K. P. Jantke (Ed.), *Analogical and inductive inference* (pp. 18–44). Berlin: Springer-Verlag.

Pollard, C., & Sag, I. (1994). *Head driven phrase structure grammar*. Chicago: University of Chicago Press.

Ron, D., Singer, Y., & Tishby, N. (1998). On the learnability and usage of acyclic probabilistic finite automata. *Journal of Computer and System Sciences*, *56*(2), 133–152.

van Rooij, I. (2008). The tractable cognition thesis. *Coginitive Science: A Multidisciplinary Journal*, *32*(6), 939–984.

Sagae, K., Davis, E., Lavie, A., MacWhinney, B., & Wintner, S. (2007). High-accuracy annotation and parsing of childes transcripts. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition* (pp. 25–32). ACL.

Seki, H., Matsumura, T., Fujii, M., & Kasami, T. (1991). On multiple context-free grammars. *Theoretical Computer Science*, *88*(2), 229.

Shinohara, T. (1994). Rich classes inferable from positive data. *Information and Computation*, *108*(2), 175–186.

Stabler, E. (1997). Derivational minimalism. In C. Retoré (Ed.), *Logical aspects of computational linguistics (LACL 1996)* (pp. 68–95). Berlin: Springer.

Valiant, L. (1984). A theory of the learnable. *Communications of the ACM*, *27*(11), 1134–1142.

Wells, R. S. (1947). Immediate constituents. *Language*, *23*(2), 81–117.

Wexler, K., & Culicover, P. W. (1980). *Formal principles of language acquisition*. Cambridge, MA: MIT Press.

Yang, C. (2008). The great number crunch. *Journal of Linguistics*, *44*(01), 205–228.

Yoshinaka, R. (2011). Efficient learning of multiple context-free languages with multidimensional substitutability from positive data. *Theoretical Computer Science*, *412*(19), 1821–1831.

Yoshinaka, R., & Kanazawa, M. (2011). Distributional learning of abstract categorial grammars. In S. Pogodalla & J.-P. Prost (Eds.), *Lacl* (Vol. 6736, pp. 251–266). Berlin: Springer.

Zeugmann, T., & Lange, S. (1995). A guided tour across the boundaries of learning recursive languages. In K. Jantke & S. Lange (Eds.), *Algorithmic learning for knowledge-based systems* (Vol. 961, pp. 190–258). Berlin/Heidelberg: Springer.