ELSEVIER

Cognition, 61 (1996) 39–91

COGNITION

A computational study of cross-situational techniques for learning word-to-meaning mappings

Jeffrey Mark Siskind*

Department of Electrical Engineering, Technion, Haifa 32000, Israel

Abstract

This paper presents a computational study of part of the lexical-acquisition task faced by children, namely the acquisition of word-to-meaning mappings. It first approximates this task as a formal mathematical problem. It then presents an implemented algorithm for solving this problem, illustrating its operation on a small example. This algorithm offers one precise interpretation of the intuitive notions of cross-situational learning and the principle of contrast applied between words in an utterance. It robustly learns a homonymous lexicon despite noisy multi-word input, in the presence of referential uncertainty, with no prior knowledge that is specific to the language being learned. Computational simulations demonstrate the robustness of this algorithm and illustrate how algorithms based on cross-situational learning and the principle of contrast might be able to solve lexical-acquisition problems of the size faced by children, under weak, worst-case assumptions about the type and quantity of data available.

1. Introduction

Suppose that you were a child. And suppose that you heard the utterance *John walked to school.* And suppose that when hearing this utterance, you saw John walk to school. And suppose, following Jackendoff (1983), that upon seeing John walk to school, your perceptual faculty could produce the expression GO(**John**, TO(**school**)) to represent that event. And further suppose that you would entertain this expression as the meaning of the utterance that you just heard. At birth, you

* Corresponding author: Present address: Department of Electrical Engineering and Computer Science, University of Vermont, Burlington, VT 05405, USA. Fax: 802/656-0696; e-mail: Qobi@emba.uvm.edu.

PII S0010-0277(96)00728-7

could not have known the meanings of the words *John, walked, to*, and *school*, for such information is specific to English. Yet, in the process of learning English, you come to possess a mental lexicon that maps the words *John, walked, to*, and *school* to representations like **John**, GO(x, y), TO(x), and **school**, respectively. This paper explores one way that this might be done.

The above situation is not sufficient for you to arrive at the correct word-to-meaning mappings. On the basis of this situation alone, you could not rule out the mappings *John* → TO(x), *walked* → **school**, *to* → **John**, and *school* → GO(x, y), for these mappings, taken together, are also consistent with the aforementioned utterance–observation pair. Yet, considering multiple situations, and requiring the hypothesized lexicon to be consistent across those situations, might allow you to converge on the correct word-to-meaning mappings.

The situation faced by children is likely to be more complex than the above example suggests. When seeing John walk to school, you also saw him moving his feet and wearing a red shirt. Thus, your perceptual faculty might also produce MOVE(**John**, **feet**) and WEAR(**John**, RED(**shirt**)) as possible meanings of the utterance *John walked to school.* How are you to know that GO(**John**, TO(**school**)) is the correct meaning of that utterance, ruling out the other possibilities? I refer to this problem as *referential uncertainty.* When acquiring word-to-meaning mappings, the learner faces a myriad of problems, among them: (a) deciding how to disambiguate the referential uncertainty to determine the correct meaning of each utterance and (b) deciding how to break that utterance meaning into parts to assign as the correct meaning of each word in the utterance. This paper addresses these two problems.

The general notion of cross-situational learning has been proposed by many authors, among them Pinker (1989) and Fisher et al. (1994). Nonetheless, the following question remains: *How can one render this intuitive idea as a precise algorithm and measure its effectiveness for learning word meanings?* This paper addresses this question. I should stress at the outset that I do *not* claim that children employ the particular algorithm presented in this paper. As a computational model, this work provides an existence proof for an algorithm that solves an approximation of the lexical-acquisition task. It leaves the investigation of what techniques children actually use to solve that task to further empirical study. It is hoped that the algorithm presented here will provide a theoretical framework to support such investigation.

2. Overview

Later in this paper, I present the details of a particular lexical-acquisition algorithm and discuss the assumptions that allow it to function. Since the algorithm encodes some fairly straightforward common-sense principles, it is helpful to first present the intuition behind those principles.

2.1. Constraining hypotheses with partial knowledge

Partial knowledge of word meanings can allow the learner to filter out impossible hypothesized meanings of utterances that contain those words. For example, imagine that the learner heard the utterance *Mary lifted the block.* Further imagine that, when hearing this utterance, the learner entertained three potential meanings for this utterance, given the non-linguistic context: CAUSE(**Mary**, GO(**block**, UP)), WANT(**Mary**, **block**), and BE(**block**, ON(**table**)). On the basis of non-linguistic context alone, the learner could not know which of these hypotheses is correct. Imagine, however, that the learner possessed partial information about the meanings of the words in that utterance. More specifically, imagine that the learner knew that the word *lifted* contained CAUSE as part of its meaning. Given such partial information, and certain assumptions about how the meanings of words combine to form meanings of utterances that contain those words, the learner could rule out WANT(**Mary**, **block**), since that hypothesis does not contain CAUSE as part of its meaning. Similarly, imagine that the learner knew that none of the words *Mary*, *lifted*, *the*, and *block* could contain BE as part of their meaning. Such knowledge would allow the learner to rule out BE(**block**, ON(**table**)), since no word could contribute BE to the target utterance meaning. This intuition motivates the following conjecture: *When learning word meanings, children use partial knowledge of word meanings to constrain hypotheses about the meanings of utterances that contain those words.*

2.2. Cross-situational inference

One way that a learner might determine the meaning of a word is to find something in common across all observed uses of that word. Commonality across observed uses can be elucidated by forming a set of possible meanings for each use, from the non-linguistic context, and intersecting those sets. Such a strategy could potentially be used at two different levels: intersecting sets of entities that represent *portions* of the meaning of a word or intersecting sets of entities that represent the *entire* meaning of a word. As an example of the former, consider the following scenario. Imagine that the learner heard the two utterances *John lifted the ball* and *Mary lifted the block.* And imagine that the learner hypothesized CAUSE(**John**, GO(**ball**, UP)) as the meaning of the former and CAUSE(**Mary**, GO(**block**, UP)) as the meaning of the latter. From the first use of the word *lifted*, the learner could form the set {CAUSE, **John**, GO, **ball**, UP} of meaning fragments. From the second use of the word *lifted*, the learner could form the set {CAUSE, **Mary**, GO, **block**, UP} of meaning fragments. Under certain assumptions, intersecting these sets would allow the learner to infer that the meaning of the word *lifted* cannot contain any meaning fragments except CAUSE, GO, and UP.

As an example illustrating the intersection of sets containing whole-word meanings, consider the same situation again. Given certain compositionality assumptions, from the first use of the word *lifted*, the learner could form the set

$$\left\{\begin{array}{c}
\mathbf{John}, \mathbf{ball}, \mathrm{UP}, \mathrm{GO}(x, y), \mathrm{GO}(\mathbf{ball}, x), \mathrm{GO}(x, \mathrm{UP}), \mathrm{GO}(\mathbf{ball}, \mathrm{UP}),\\
\mathrm{CAUSE}(x, y), \mathrm{CAUSE}(\mathbf{John}, x),\\
\mathrm{CAUSE}(x, \mathrm{GO}(y, z)), \mathrm{CAUSE}(\mathbf{John}, \mathrm{GO}(y, z)),\\
\mathrm{CAUSE}(x, \mathrm{GO}(\mathbf{ball}, y)), \mathrm{CAUSE}(\mathbf{John}, \mathrm{GO}(\mathbf{ball}, x)),\\
\mathrm{CAUSE}(x, \mathrm{GO}(y, \mathrm{UP})),\mathrm{CAUSE}(\mathbf{John}, \mathrm{GO}(x, \mathrm{UP})),\\
\mathrm{CAUSE}(x, \mathrm{GO}(\mathbf{ball}, \mathrm{UP})), \mathrm{CAUSE}(\mathbf{John}, \mathrm{GO}(\mathbf{ball}, \mathrm{UP}))
\end{array}\right\}$$

of potential whole-word meanings. This set contains all subexpressions of CAUSE(**John**, GO(**ball**, UP)), the hypothesized whole-utterance meaning, as well as all variations of those subexpressions with one or more of their subexpressions replaced with variables. Likewise, from the second use, the learner could form the set

$$\left\{\begin{array}{c}
\mathbf{Mary}, \mathbf{block}, \mathrm{UP}, \mathrm{GO}(x, y), \mathrm{GO}(\mathbf{block}, x), \mathrm{GO}(x, \mathrm{UP}), \mathrm{GO}(\mathbf{block}, \mathrm{UP}),\\
\mathrm{CAUSE}(x, y), \mathrm{CAUSE}(\mathbf{Mary}, x),\\
\mathrm{CAUSE}(x, \mathrm{GO}(y, z)), \mathrm{CAUSE}(\mathbf{Mary}, \mathrm{GO}(y, z)),\\
\mathrm{CAUSE}(x, \mathrm{GO}(\mathbf{block}, y)), \mathrm{CAUSE}(\mathbf{Mary}, \mathrm{GO}(\mathbf{block}, x)),\\
\mathrm{CAUSE}(x, \mathrm{GO}(y, \mathrm{UP})), \mathrm{CAUSE}(\mathbf{Mary}, \mathrm{GO}(x, \mathrm{UP})),\\
\mathrm{CAUSE}(x, \mathrm{GO}(\mathbf{block}, \mathrm{UP})), \mathrm{CAUSE}(\mathbf{Mary}, \mathrm{GO}(\mathbf{block}, \mathrm{UP}))
\end{array}\right\}$$

of potential whole-word meanings. Intersecting these sets would allow the learner to restrict the meaning of the word *lifted* to be either UP, GO(x, y), GO(x, UP), CAUSE(x, y), CAUSE(x, GO(y, z)), or CAUSE(x, GO(y, UP)).

Both of these forms of inference are instances of what is commonly known as cross-situational learning. This intuition motivates the following conjecture: *When learning word meanings, children apply cross-situational inference, both at the level of word meaning fragments, and at the level of whole-word meanings, to constrain the possible meanings of words, given their context of use.*

2.3. Covering constraints

Cross-situational inference at the meaning-fragment level is only able to rule out fragments as potential components of a word's meaning. It cannot provide evidence that a particular fragment is essential. For example, applying cross-situational inference to the above situation, the learner could infer that **ball** could not be part of the meaning of the word *lifted*, since it is absent from the situation surrounding the second use of that word. However, using cross-situational inference alone, the learner could not determine that CAUSE must be part of the meaning of *lifted*. To make such an inference, the learner might apply an additional source of constraint. Suppose that human language has the property that all components of the meaning of an utterance must be derived from the meanings of words in that utterance. Imagine that, by applying cross-situational inference to other utterances, the learner could rule out CAUSE as a component of the meanings of the words *John*, *the*, and *ball*. Under the covering constraint, the learner could infer that CAUSE must be a part of the meaning of *lifted*, since it is part of the meaning of *John lifted the ball* yet is not part of the meaning of any other word in that utterance. Cross-situational inference allows one to determine what fragments *can be* part of a word's meaning. In contrast, covering constraints allow one to determine what fragments *must be* part of a word's meaning. This intuition motivates the following conjecture: *When learning word meanings, children apply cross-situational inference and covering constraints in a complementary fashion to progressively reduce the set of fragments that can be in a word's meaning, and increase the set of fragments that must be in that word's meaning, until the two sets are equal, thus identifying a single set of fragments out of which a word's meaning can and must be constructed.*

2.4. Principles of exclusivity

Suppose that human language has the property that the words in an utterance must contribute non-overlapping portions of the utterance meaning. The learner could use this property, via a principle of exclusivity, to perform an additional form of inference in the following fashion. Imagine that a learner heard the utterance *John walked* and hypothesized the meaning WALK(**John**) for that utterance. Further imagine that the learner already determined, via a combination of cross-situational inference and covering constraints, that *John* must mean **John**. Applying only cross-situational inference and covering constraints to this data, the learner could not determine whether WALK(**John**) or WALK(x) is the meaning of *walked*. However, under the stipulation that *John* and *walked* contribute non-overlapping portions of the utterance meaning WALK(**John**), knowing that *John* means **John** rules out WALK(**John**) as a potential meaning of *walked*. This intuition motivates the following conjecture: *When learning word meanings, children apply principles of exclusivity to constrain the possible meanings of some words in an utterance, given knowledge about the meanings of other words in that utterance.*

The above collection of conjectures is intended only as a partial description of the lexical-acquisition strategy used by children. They leave open the possibility that children employ additional information and forms of inference during lexical acquisition beyond these four conjectures.

The ultimate goal of this research is to test the above four conjectures. This paper, however, adopts a much more modest goal, namely testing the effectiveness of these strategies for performing lexical acquisition by way of computational simulation. Computational simulation places strong constraints on theories. It requires that intuitive pre-formal notions be rendered precise and assumptions be made explicit. The remainder of this paper does precisely that. It presents a formal approximation of the lexical-acquisition task faced by children, along with an algorithm for solving that formal problem. It discusses, and attempts to justify, the assumptions that underlie both the formal problem and the implemented algorithm. And it demonstrates, by way of simulation, that the algorithm can solve the formal problem. Given our current understanding of human mental processes, however, some of the assumptions needed to render the pre-formal notions precise cannot be verified. Thus, the question of whether children employ the strategies discussed here is open to further research.

3. The lexical-acquisition task

The mental lexicon presumably contains a variety of different kinds of information about words, including their acoustic, morphological, syntactic, and semantic properties. A full account of lexical acquisition will need to explain how all such information is learned. This paper, however, is concerned with a subtask of lexical acquisition, namely learning word-to-meaning mappings. For the remainder of this paper, I use the term "lexical acquisition" to refer only to this subtask.

In this paper, I adopt a simplified model of interaction between the lexical-acquisition faculty[1] and other cognitive faculties such as speech perception and the perceptual/conceptual faculty. This model is depicted in Fig. 1. In this model, the lexical-acquisition faculty receives two streams of input, one from the speech-

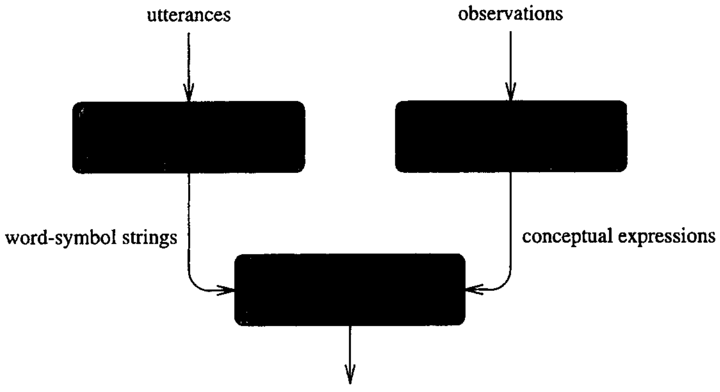

Fig. 1. The simplified model of interaction between the lexical-acquisition faculty and other cognitive faculties that is adopted in this paper.

[1] Through much of this paper, I use the term "faculty" when referring to various facets of the human cognitive capacity such as speech and visual perception, conceptual reasoning, and lexical acquisition. For some, this term might carry implicit connotations of modularity. By using the term "faculty," I do not intend to suggest that such capacities are indeed modular. Rather, for the sake of simplicity, I adopt a modular architecture as a working hypothesis. Without this hypothesis, and the approximations that it affords, it would not be possible to conduct the simulations presented later in this paper. While the modular architecture that I adopt is likely to be only an idealized approximation, I believe that simulations conducted under this idealization, nonetheless, provide useful insight into the lexical-acquisition process. Modularity is thus a weak, worst-case assumption, in that additional information pathways could only help, not hinder, the lexical-acquisition process.

perception faculty and one from the perceptual/conceptual faculty. The speech-perception faculty segments the acoustic stream into utterances and words, and classifies sequences of acoustic segments, such as /æ·pl̩/, into *word symbols*, such as *apple*.[2] Cutler et al. (1983), Norris and Cutler (1985), Cutler and Carter (1987), Jusczyk et al. (1993), Christophe et al. (1994), and Brent and Cartwright (this issue), among others, discuss some different segmentation strategies that children appear to use, depending on their native language. The perceptual/conceptual faculty produces hypothesized meaning representations to associate with each utterance heard. For example, the perceptual/conceptual faculty might produce the conceptual symbol **apple** when seeing an apple, or the conceptual expression CAUSE(**mother**, GO(**ball**, UP)) when seeing Mother lift a ball. Badler (1975), Okada (1979), Borchardt (1985), Hays (1989), and Siskind (1992, 1995), among others, describe computational approaches to visual event perception, while Regier (1992) describes a method for learning such perceptual/conceptual processes. In order to accommodate the acquisition of meanings of words that refer to non-visual stimuli, as well as lexical acquisition by blind children, one can take the perceptual/conceptual faculty to be the combination of all perceptual modalities, not just vision. One can similarly generalize the notion of speech-perception faculty to include visual and haptic input in order to accommodate lexical acquisition of signed languages as well.

Snow (1977) reports that as many as 49% of the utterances heard by children do not refer to the here-and-now. Similarly, Fisher et al. (1994) cites Beckwith et al. (1989) claiming that close to a third of verb uses to young children do not refer to the here-and-now. To partly deal with this problem, I assume that while the perceptual/conceptual faculty primarily produces descriptions of observed objects, states, and events, more generally it can hypothesize what a speaker is likely to have said in a given situation, even though such utterances might have referred to something not directly observable. The perceptual/conceptual faculty need not be telepathic to make such hypotheses. Instead, it can incorporate naive psychological knowledge and use this knowledge to make inferences about the beliefs, goals, desires, and intentions of other agents. For example, the perceptual/conceptual faculty might produce WANT(**mother**, **ball**) when the child observes Mother reach for a ball, or NOT-BE(**father**, AT(**home**)) when the child looks around crying and Mother says that Father isn't home. The model presented here does not require the perceptual/conceptual faculty to produce representations of all true statements about the world, whether positive or negative. Quine (1960) points out that there are an infinite number of true facts about the world that a learner might need to entertain as potential meanings of each utterance. Rather, in the model presented here, the perceptual/conceptual faculty produces a limited number of *hypotheses* about what was *likely* to have been said in a given situation, not everything that *could* have been said. Thus, presumably, in the above situation, the hypothesis NOT-BE(**father**, AT(**home**)) would be more likely than other hypotheses, such as

[2] Throughout this paper, I use *italics*, **bold face**, and CAPITALIZED words to represent word and sense symbols, conceptual constant symbols, and conceptual function symbols, respectively.

NOT-POSSESS(**me**, **giraffe**, IN(**pocket**)), given the child's knowledge of his own mental state, his model of the inferences that his mother can make about his mental state, and his assumptions about how these inferences would influence what his mother might say in that situation. Under such a model, however, it is possible that, at times, the perceptual/conceptual faculty might produce only incorrect hypotheses. This paper presents techniques for dealing with such noisy input data.

I refer to the output of the perceptual/conceptual faculty as *conceptual structure*. I assume that conceptual structure takes the form of *conceptual expressions*, such as GO(**John**, TO(**school**)), that are constructed out of an inventory of *conceptual symbols*, such as GO, **John**, TO, and **school**. For expository purposes, the examples presented in this paper use conceptual expressions reminiscent of those proposed by Jackendoff (1983, 1990). The algorithm presented in this paper is not particular, however, to the choice of conceptual representation. It works for expressions constructed out of any conceptual-symbol inventory, including those proposed by Leech (1969), Miller (1972), Schank (1973), Borchardt (1985), and Pinker (1989). Furthermore, it works regardless of whether word meanings are represented by simple conceptual expressions that contain only a single primary conceptual symbol, as has been proposed by Fodor (1970), or by more complex conceptual expressions that contain multiple conceptual symbols, as has been proposed by Jackendoff and others. Many examples in this paper represent the meanings of words such as *John*, *walked*, *to*, and *school* by simple conceptual expressions, such as **John**, GO(x, y), TO, and **school**, each of which contains a single primary conceptual symbol (namely **John**, GO, TO, and **school**). Other examples, however, represent the meanings of words, such as *lift*, by complex nested conceptual expressions, such as CAUSE(x, GO(y, UP)), that contain several conceptual symbols. This paper presents computer simulations that illustrate the operation of the algorithm on a variety of different input sets, each containing conceptual expressions of varying depths and branching factors, constructed out of different conceptual-symbol inventories of different sizes.

Many languages have words that play a purely syntactic role. Examples of such words are case markers, such as the English word *of*, and complementizers, such as the English word *that*. Such words might not contribute any conceptual symbols to the representations of the meanings of utterances that contain those words. To indicate this, I represent the meanings of such words via the conceptual expression $\perp$. For expedience, I also use $\perp$ to represent the meanings of words that fall outside the semantic space of the chosen conceptual-symbol inventory. Thus, since the Jackendovian notation adopted in this paper does not represent definite and indefinite reference, determiners, such as the English word *the*, will take on $\perp$ as their meaning. This is simply an expository issue and not an inherent limitation of the lexical-acquisition algorithm presented here. Adopting a richer conceptual-symbol inventory would allow the algorithm to represent and learn the meanings of determiners.

The learning algorithm presented here makes no use of the phonology of the word symbols or the semantic truth conditions of the conceptual expressions themselves. As far as it is concerned, these expressions are simply strings of uninterpreted symbols. Symbols such as *apple* and **apple** have no inherent semantic or phonological content. In this view, lexical acquisition is simply a process of learning the mapping between two pre-existing mental representation languages (Fodor, 1975). It is conceivable that children's mental processes might not adopt such a strict partitioning between the speech perception and perceptual/conceptual faculties on one hand and the lexical-acquisition faculty on the other. The word and conceptual symbols that pass between the two might not be totally devoid of phonological and semantic content. The lexical-acquisition faculty might use limited phonological content, say, to distinguish nouns from verbs (Kelly, 1992), and, accordingly, bias words to map to objects versus events. Similarly, the lexical-acquisition faculty might use limited semantic content, say, to distinguish between basic-level and sub- or super-ordinate categories to prefer mappings to conceptual symbols that represent basic-level categories (Horton and Markman, 1980). The work presented here does not question whether children use phonological or semantic information during lexical acquisition or the extent to which they do so. Instead, it makes a weak, worst-case assumption, in that additional sources of information or constraint can only help, not hinder, the lexical-acquisition process.

It is likely that many of the utterances heard by children conform to the syntactic constraints of the language heard. Similarly, the conceptual expressions produced by the perceptual/conceptual faculty might conform to some well-formedness constraints imposed by the "syntax" of conceptual structure, whatever that may be. It is conceivable that children make use of either or both sources of constraint to guide lexical acquisition (Gleitman, 1990; Fisher et al., 1994). The algorithm presented in this paper, however, does not use any syntactic properties of the language heard, or any well-formedness properties of the underlying conceptual structure, beyond the semantic-interpretation rule to be discussed. This again is a weak, worst-case assumption, in that use of such properties can only help, not hinder, the lexical-acquisition process.

4. Why lexical acquisition is difficult

Lexical acquisition is difficult for at least five reasons. First, children hear multi-word utterances; they must figure out which words in an utterance map to which parts of the utterance meaning. Second, children hear utterances in contexts where more than one thing could have been said (Gleitman, 1990); they must figure out which of those things is, in fact, the meaning of the utterance just heard. Third, children must start this task without any prior knowledge that is specific to the language being learned; this is sometimes referred to as the *bootstrapping* problem (Pinker, 1984; Gleitman, 1990). Fourth, the input is noisy; it may contain utterances paired with only incorrect hypothesized meanings. Children must determine which parts of the input to ignore. Finally, many words are homonymous; they can have several different senses. Children must determine which sense of each word is being used at a given time. Each of these difficulties is examined in greater detail below.

4.1. Multi-word utterances

A common conjecture, that dates as far back as St. Augustine (Bruner, 1983) and Locke (1690), is that children hear single-word utterances in a context where the meanings of those words are made clear by ostention. If this were true, lexical acquisition would be trivial to explain. Children would simply be presented with the lexicon as input. An examination of the Nina corpus (Suppes, 1974) in the CHILDES database (MacWhinney and Snow, 1985) that contains a transcript of adult speech to Nina when she was between the ages of 1 year 11 months and 3 years 3 months, however, gives evidence that children receive insufficient input data in the form of single-word utterances to account for lexical acquisition using the above strategy. Only 1913 (5.6%) out of the 34,438 utterances in the corpus are single-word utterances, while only 276 (8.5%) out of the 3246 word types that appear in the corpus appear in single-word utterances. Furthermore, Aslin et al. (1995) report that even when parents were given explicit instructions to teach words to their children, in the data gathered for 13 out of 19 parent–child pairs, fewer than 30% of the parental utterances consisted of isolated words. Even if these cursory estimates are atypically low, there are still whole classes of words, such as obligatorily-transitive verbs, prepositions, quantifiers, and determiners, that would rarely, if ever, appear in isolated-word utterances.

The following question then arises: *Given multi-word input, how do children figure out which words map to which parts of the utterance meaning?* Presumably, children could consider all possible mappings, examine multiple situations, and accept only those mappings consistent with all, or at least many, of the different situations encountered. This basic strategy has been suggested by numerous authors, including Pinker (1989), who called it "event category labeling," and Fisher et al. (1994), who called it "cross-situational learning." This paper attempts to formalize and extend this notion, and to explore its efficacy by way of computational simulations. The algorithm described in this paper also performs an additional form of inference, similar to that proposed by Tishby and Gorin (1994), allowing known meanings of some words in an utterance to constrain the hypotheses about the meanings of other words in that utterance. This paper shows how these techniques can be effective in the presence of multi-word input.

4.2. Referential uncertainty

As discussed in the Introduction, it is likely that children face a learning task that is more complex than the one just described. Not only must they break an utterance meaning into its parts and assign those parts correctly to the individual words in the utterance, they must also figure out the correct utterance meaning from the myriad possible things that could have been said in a given situation. In this paper, such referential uncertainty is modeled by allowing the learner to hypothesize a *set* of possible meanings for each utterance heard. I assume that this set is produced by the perceptual/conceptual faculty. The learner then faces the two-fold task of (a) figuring out which of the meanings hypothesized for a given utterance is correct and (b) figuring out how to break the correct utterance meaning apart and assign fragments of that meaning to the words in the utterance. I refer to the average number of meanings hypothesized for each input utterance as the *degree of referential uncertainty*. This paper presents computer simulations that illustrate that the lexical-acquisition algorithm described in this paper has the same performance across a wide range of degrees of referential uncertainty.

4.3. Bootstrapping

During later stages of lexical acquisition, children might possess information, specific to the language being learned, that can simplify the learning task. For example, a child who hears the utterance *I woke up yesterday, turned off my alarm clock, took a shower, and cooked myself two grimps for breakfast* (Granger, 1977) might infer a lot about the meaning of the word *grimp*, if she already knows the grammar and morphology of English as well as the meanings and lexical categories of the other words in the utterance. She might determine that *grimp* is likely to be a common noun that names a type of food that one might cook and consume two of for breakfast. Furthermore, she might know where in the world to look for likely meanings of the word *grimp*, namely those food items that she saw the speaker cook and consume two of for breakfast the previous day. Granger (1977), Jacobs and Zernik (1988), and Berwick (1983), among others, describe implemented systems that perform just this kind of inference. Such inference, however, can be performed by children only during later stages of lexical acquisition when they already possess substantial information specific to the language being learned. Children must start the lexical-acquisition process without such knowledge. They hear utterances that might initially sound to them like *Foo bar baz quux*. The following question then arises: *How do children start the lexical-acquisition process without any seed information?* This problem has become known as the *bootstrapping* problem (Pinker, 1984; Gleitman, 1990).

The algorithm that I present consists of a collection of inference rules. While all of these inference rules *can make use* of partial information in the lexicon, only some of them *require* such partial information to operate. For example, inference rules that implement principles of exclusivity require partial information in order to operate, while those that implement cross-situational learning do not. Thus, under certain assumptions, a learner hearing the word *shirt* in the absence of red things might infer, using cross-situational techniques, that *shirt* could not mean RED. Such inference could be performed without any prior lexical knowledge. If the same learner later heard the phrase *red shirt*, and somehow could determine that the phrase must refer to a red shirt, the learner could use principles of exclusivity to infer that *red* must mean RED and *shirt* must mean **shirt**. This later inference can be performed only in light of the lexical knowledge obtained by the earlier observation and inference.

The algorithm that I present opportunistically applies whatever inference rules it can to the current observation in the context of the current partial lexicon. Thus learning is slow during the early stages and becomes faster as more information is acquired. This paper presents computer simulations that demonstrate this effect. Ultimately, the algorithm is able to acquire new words with only one or two occurrences. Thus this single algorithm exhibits a range of different behaviors, with bootstrapping at one extreme, and context-based single-occurrence acquisition at the other, purely as a result of data exposure, without any maturational parameter shift.

4.4. Noise

A cross-situational strategy implies that when children hear a word in multiple situations, they choose, as its meaning, something common across those situations. Presumably, hearing a word in enough different situations would reduce the set of possibilities to a single meaning consistent with all observed situations. This strategy, however, breaks down in the presence of noisy input and homonymy. Suppose that a learner heard the word *ball* in many situations, some of which did not contain a ball. In this case, there would be no meaning that is consistent across all of the observed situations. It seems likely that many words are eventually heard in some counterfactual situation. How, then, could the learner determine the meanings of such words?

In this paper, I offer two different solutions to this problem. First, recall that the model that I propose does not require hypothesized utterance meanings to reflect visual observations of the here-and-now. Rather, it assumes that these are the product of a general perceptual/conceptual faculty that has some capacity to model what might be said in each situation. Second, a key feature of the algorithm presented in this paper is that it can learn despite the presence of noise in the input. Noise arises for the following reason: Since there are infinitely many things that could be said in every situation, the perceptual/conceptual faculty must necessarily enumerate only a finite subset of the possible utterance meanings for each situation. Sometimes, this subset will lack the correct utterance meaning. Such cases constitute *noise*. A simple strategy might be to ignore noisy utterances. This is not as easy as it sounds since noisy utterances are not marked in the input.

It is instructive to look at the noise problem in a different light. Another strategy sometimes proposed as a model of lexical acquisition is that children somehow estimate the statistical correlation between words and observations. Thus, a child would learn that *open* means OPEN by seeing numerous situations where *open* and OPEN co-occur. The difficulty with this approach, as pointed out by Gleitman (1990), is that there may be many situations where *open* and OPEN do not co-occur. Since the non-co-occurrence situations might outnumber the co-occurrence situations, the correlation could be low. In keeping with the general strategy adopted in this paper of investigating weak, worst-case assumptions, it is useful to explore how a child might learn despite low correlation between word and observation.

Low correlation itself need not be a problem. Presumably, a learner could adopt a relative metric and decide to pair a word with a meaning when no word correlates better with that meaning and no meaning correlates better with that word. Tishby and Gorin (1994) point out, however, that acquisition techniques based solely on correlations can be unfocused. To paraphrase and extend their example, suppose that the learner received many instances of *red shirt*, paired correctly with RED(**shirt**), and even more instances of *shirt*, paired correctly with **shirt**. Furthermore, suppose that the learner also received a few noisy instances of *red* paired with **shirt**. In this case, the correlation between *red* and **shirt** would be higher than between *red* and RED(x). Yet, given that there is more evidence for associating *shirt* with **shirt** than for any other association, the learner should be able to apply a principle of exclusivity or contrast to determine that *red* means RED(x). Clark (1987) and Markman (1989) suggest an application of exclusivity at the level of the lexicon. In their models, different words should have different meanings. Tishby and Gorin suggest that exclusivity be applied individually to each utterance. In their model, the meaning of an utterance must be equivalent to the sum of its parts. Thus, while in general, nothing prevents two words from having the same meaning, a learner that knows that *shirt* must mean **shirt** could infer that *red* could not also mean **shirt** by hearing *red shirt* paired with RED(**shirt**). Like the algorithm proposed by Tishby and Gorin (1994), the algorithm presented here also applies exclusivity to individual utterances, though it does so in a very different fashion, without computing statistical correlations.

The learner may face other kinds of noise in addition to utterances that do not refer to the here-and-now. Some utterances may be ungrammatical, while others may require, for their comprehension, a theory of compositional semantics that is more elaborate than the simple semantic-interpretation rule embodied in the lexical-acquisition algorithm described in this paper. A single strategy can deal with all such sources of noise. The learner must simply ignore some portion of the input. This paper presents a method for deciding which portion of the input to ignore.

4.5. Homonymy

The fact that words can have multiple senses, either related or unrelated, poses a difficult problem for the learner. The learner must determine which sense is being used at a given time since the input data is not marked with such information. For example, when hearing *Did you remove the band from around the box?* and *Did you hear the band play our song?*, the learner must somehow determine that *band* refers to a fastening device in the former and a group of musicians in the latter. Applying cross-situational techniques, the learner would fail to find a single meaning for *band* that is consistent with all of the observed situations. The algorithm described in this paper treats homonymy and noise with a single mechanism. When cross-situational techniques discover an inconsistency, the algorithm hypothesizes a split in possible word senses. If this split is corroborated by further evidence, the split is adopted as legitimate homonymy. If not, the spurious sense is rejected as noise.

5. The mapping problem

The algorithm presented in this paper solves a precisely specified formal problem that I call *the mapping problem.* While this formal problem is simplified and abstract, I believe that it provides a reasonable model of many aspects of the lexical-acquisition task faced by children. In this problem, the learner is presented with a sequence of utterances, each being an unordered collection of word symbols. The collection is unordered to keep the lexical-acquisition process independent of word order and allow a pure study of cross-situational learning techniques without the use of syntactic knowledge. Each utterance is paired with a set of conceptual expressions that represent hypothesized meanings for that whole utterance. Sometimes this set will include the correct meaning; at other times it will not. An utterance is considered to be noisy if it is paired with only incorrect meaning hypotheses.

Before proceeding, let me reiterate the representation that is used for meanings. Meaning representations are constructed out of conceptual symbols such as GO and **ball**. Such conceptual symbols are formed into conceptual expressions such as CAUSE(**John**, GO(**ball**, TO(**Mary**))). Conceptual expressions are used to represent the meanings of both words and utterances. Such conceptual expressions can contain variables to denote unfilled argument positions, as in CAUSE(x, GO(y, TO(z))). Those conceptual expressions that represent utterance meanings will not contain variables, since in the model given here, utterances are taken to be sentences and sentences do not contain unfilled argument positions. Those conceptual expressions that represent the meanings of argument-taking words, such as verbs, will contain variables while those that represent the meanings of non-argument-taking words, such as many nouns, will not. Conceptual expressions that represent the meanings of words can contain more than one conceptual symbol, as in CAUSE(x, GO(y, TO(z))), though nothing precludes the degenerate case of representing the meaning of a word with a conceptual expression that consists of a single conceptual symbol, as suggested by Fodor (1970). Finally, an ambiguous word will have several *senses*. The meaning of each sense will be represented by a distinct conceptual expression.

In order to fully specify the mapping problem, one must specify the process by which the meanings of words combine to form the meanings of utterances that contain those words. This paper makes as few assumptions as possible about this semantic-interpretation process. First, it assumes that the lexicon for a given language maps each word to a set of conceptual expressions that represent the meanings of different senses for that word. Thus, the lexical entry for *lift* might be {CAUSE(*x*, GO(*y*, UP)), **elevator**}. Then it assumes that the meaning of an utterance is derived from the meanings of its constituent words by first selecting a sense for each word in the utterance, then finding the conceptual expressions that represent the meaning of that sense, and finally passing the resulting collection of conceptual expressions, one for each word in the utterance, to a semantic-interpretation function. I refer to this semantic-interpretation function as Compose. The input to Compose is an unordered collection of conceptual expressions. The input is unordered to keep the semantic-interpretation process independent of word order. No claim that the actual human semantic-interpretation process ignores word order is intended. This is simply a weak, worst-case assumption, in that allowing the semantic-interpretation process to make use of word order can only help, not hinder, the lexical-acquisition process.

The output of Compose is a set of conceptual expressions that denote possible ways of combining the given word-sense meanings into utterance meanings. For example, the output of

$$\textsc{Compose}(\{\textbf{John}, \mathrm{GO}(x, y), \mathrm{TO}\ (x), \textbf{school}\})$$

might be the set

$$\left\{\begin{array}{l}\mathrm{GO}(\textbf{John}, \mathrm{TO}(\textbf{school})), \mathrm{GO}(\textbf{school}, \mathrm{TO}(\textbf{John})),\\ \mathrm{GO}(\mathrm{TO}(\textbf{John}), \textbf{school}), \mathrm{GO}(\mathrm{TO}(\textbf{school}), \textbf{John}),\\ \mathrm{TO}(\mathrm{GO}(\textbf{John}, \textbf{school})), \mathrm{TO}(\mathrm{GO}(\textbf{school}, \textbf{John}))\end{array}\right\}$$

Given a lexicon, a conceptual expression *m* is a *possible interpretation* of an utterance if one can select, from the lexicon, a sense for each occurrence of each word in the utterance, such that *m* is contained in the output of Compose applied to the conceptual expressions that represent the meanings of those senses. This semantic-interpretation rule allows for both lexical and interpretive ambiguity. Lexical ambiguity is modeled by allowing words to have multiple senses. Interpretive ambiguity is modeled by having the function Compose return a set of interpretations rather than a single interpretation. The lexical-acquisition algorithm described in this paper can successfully learn a lexicon despite such ambiguity.

The function Compose is left unspecified, except for two conditions. First, each conceptual symbol that appears in the conceptual expression that represents the meaning of a whole utterance must appear in the conceptual expression that represents the meaning of at least one word in that utterance. This states that the semantic content of an utterance must be derived from the words in that utterance. Variance in the structure of an utterance can affect only the process of combining those word meanings. Second, a conceptual symbol must appear in the conceptual expression that represents the meaning of the whole utterance at least as many times as the sum of the number of times it appears in the conceptual expressions that represent meanings of words in that utterance.[3] This states that the semantic-interpretation rule cannot delete any symbols when producing utterance-meaning representations. When applying the above two conditions, the distinguished conceptual expression $\perp$ is viewed as not containing any conceptual symbols. Apart from explicit use of the above two conditions, and the RECONSTRUCT procedure, to be described later, the lexical-acquisition process treats COMPOSE as a "black box." That is, it does not use any knowledge of the semantic-interpretation process other than what can be obtained by examining the results of presenting sample inputs to COMPOSE.

Taken together, these two conditions imply that the semantic-interpretation rule must be more-or-less compositional. As with all compositional semantic-interpretation rules, this precludes idioms and metaphor. While the algorithm discussed in this paper can learn in the presence of input that contains some idioms and metaphor, treating such input as noise, it is not able to learn idiomatic or metaphoric meaning.

The mapping problem can now be stated formally as follows. The learner is presented with a corpus of utterances, each paired with a set of conceptual expressions that represent hypothesized utterance meanings. The learner assumes that the corpus was generated by some unknown lexicon shared collectively by the speakers of the language heard by the learner. Each utterance is viewed as an unordered collection of word symbols. The lexicon maps each word symbol that appears in the corpus to a set of conceptual expressions that represent the meanings of different senses of that word. Some of the utterances in the corpus, the non-noisy ones, have the property that at least one of the hypothesized meanings for that utterance is a possible interpretation of that utterance given the unknown lexicon used to generate the corpus. The learner must find that lexicon.

6. The noise-free monosemous case

Before presenting the full lexical-acquisition algorithm, which is capable of dealing with noise and homonymy, I will first present a simplified algorithm that handles only noise-free input under the assumption that all words are monosemous. This algorithm receives, as input, a sequence of utterances, each paired with a set of conceptual expressions that represent hypothesized meanings for that utterance. Each utterance is an unordered collection of word symbols. The

[3] The second condition does *not* state that a conceptual symbol must appear *exactly* as many times as the sum of the number of times that it appears in the conceptual expressions that represent meanings of words in that utterance. This stricter condition would make the semantic-interpretation rule "linear" in the sense of linear logic (Girard, 1987). Tishby and Gorin (1994) adopt such a linear semantic-interpretation rule. A linear semantic-interpretation rule cannot copy information from a word or phrase so that the information appears more than once in the resulting utterance meaning. By adopting a weaker constraint, the algorithm presented here can successfully learn a lexicon even when the interpretation of utterances in the corpus requires copying.

algorithm produces, as output, a lexicon that maps word symbols to conceptual expressions.

The algorithm learns word-to-meaning mappings in two stages. Stage one learns the *set* of conceptual *symbols* used to construct the conceptual expression that represents the meaning of a given word symbol, but does not learn how to assemble those conceptual symbols into a conceptual *expression*. I refer to such a set as the *actual conceptual-symbol set*. For example, when learning the meaning of the word symbol *raise*, the algorithm would first learn the actual conceptual-symbol set {CAUSE, GO, UP} during stage one, and subsequently learn how to compose these conceptual symbols into the conceptual expression CAUSE(x, GO(y, UP)) during stage two. The state of the algorithm's knowledge at the end of stage one is only partial, since many different conceptual expressions could be formed out of the actual conceptual-symbol set {CAUSE, GO, UP}, among them CAUSE(x, GO(y, UP)), GO(CAUSE, UP), and UP(CAUSE(x), GO(x, y)). The algorithm does not determine which of these is, in fact, correct until stage two. These two stages are interleaved. At a given point in the learning process, some of the words in the lexicon might be progressing through stage one, while others are progressing through stage two.

To perform stage one, the algorithm maintains two sets of conceptual symbols for each word symbol. One set, the *necessary conceptual-symbol set*, contains conceptual symbols that the algorithm has determined *must* be part of a word's meaning representation. The other set, the *possible conceptual-symbol set*, contains conceptual symbols that the algorithm has determined *can* be part of a word's meaning representation. The necessary and possible conceptual-symbol sets for a word symbol act as lower and upper bounds, respectively, on the actual conceptual-symbol set for that word symbol. In the absence of noise, the necessary conceptual-symbol set for a word symbol will be a subset of the actual conceptual-symbol set for that word symbol, and the possible conceptual-symbol set will be a superset of the actual conceptual-symbol set. For example, the necessary conceptual-symbol set might be {CAUSE} and the possible conceptual-symbol set {CAUSE, GO, UP}, leaving uncertainty as to whether the actual conceptual-symbol set was {CAUSE}, {CAUSE, GO}, {CAUSE, UP}, or {CAUSE, GO, UP}. At the commencement of stage one for a given word symbol, the necessary conceptual-symbol set for that word symbol is initialized to the empty set, while the possible conceptual-symbol set for that word symbol is initialized to the universal set, thus leaving the actual conceptual-symbol set totally unconstrained.[4]

[4] The initial universal possible conceptual-symbol set need not be instantiated extensionally. It would clearly be impossible to do so if the conceptual-symbol inventory were infinite. Even with an infinite conceptual-symbol inventory, the set of conceptual symbols in any given finite conceptual expression will be finite. Possible conceptual-symbol sets are computed as intersections of such finite sets and are thus always finite, except when they are initialized to the universal set. The universal set can be represented with a finite token, namely $\top$. The initial value of $\top$ is simply a cue to perform the first update (see Rule 2 in the appendix) as $P(s) \leftarrow \cup_{m \in M} F(m)$ instead of as $P(s) \leftarrow P(s) \cap \cup_{m \in M} F(m)$. The token $\top$ can be treated similarly in the remaining rules to allow finite computations with universal sets.

Stage one provides four inference rules, to be described momentarily, that modify the necessary and possible conceptual-symbol set for word symbols that appear in utterances as they are processed. These rules add conceptual symbols to the necessary conceptual-symbol sets and remove conceptual symbols from the possible conceptual-symbol sets, until these two sets become equal. When this happens, the algorithm is said to have *converged on the actual conceptual-symbol set* for the given word symbol. At this point, that word symbol progresses to stage two of the algorithm.

To perform stage two, the algorithm maintains a set of conceptual expressions, called the *possible conceptual-expression set*, for each word symbol. At the commencement of stage two for a given word symbol, this set is initialized to the set of all conceptual expressions that can be formed out of precisely the conceptual symbols that appear in the actual conceptual-symbol set that stage one has converged on for that word symbol. Stage two provides two inference rules, to be described momentarily, that remove conceptual expressions from the possible conceptual-expression set for word symbols that appear in utterances as they are processed, until this set contains only a single conceptual expression. When this happens, the algorithm is said to have *converged on the conceptual expression* that represents the meaning of the given word symbol.

The algorithm is *on line* in the sense that it makes a single pass through the input corpus, processing each utterance in turn, and discarding that utterance before processing the next utterance. The algorithm retains only a small amount of inter-utterance information. This information takes the form of three tables:

1. a possible conceptual-symbol table, $P(w)$, that maps each word symbol w to its possible conceptual-symbol set;
2. a necessary conceptual-symbol table, $N(w)$, that maps each word symbol w to its necessary conceptual-symbol set; and
3. a possible conceptual-expression table, $D(w)$, that maps each word symbol w to its possible conceptual-expression set.

These three tables constitute a model of the mental lexicon. I refer to the collection of $P(w)$, $N(w)$, and $D(w)$, as the *lexical entry* for the word symbol w.

The operation of the algorithm is best illustrated by way of example. The following example was chosen to succinctly illustrate all facets of the algorithm while processing only a single utterance. To do so, it starts out midway through the acquisition process where the algorithm has partial information about the necessary and possible conceptual-symbol sets for the word symbols in the utterance. Given this partial information, the algorithm will converge on the conceptual expression for each word symbol in the example utterance, solely by processing this utterance. In practice, however, the algorithm starts out without any partial information and takes considerably longer to reach convergence. Early on, the algorithm typically converges on the actual conceptual-symbol set for a word symbol, only after several occurrences of that word symbol, and then converges on its conceptual expression, only after several more occurrences. As acquisition progresses, the speed of convergence increases until the algorithm typically converges on the actual conceptual-symbol set and the conceptual expression for a new word symbol from a single word-symbol occurrence. This will be illustrated later in this paper.

Let us proceed with the example. Suppose that the algorithm is part-way through the lexical-acquisition process and already possesses the following lexicon:

	N	*P*
John	{**John**}	{**John**, **ball**}
took	{CAUSE}	{CAUSE, WANT, GO, TO, **arm**}
the	{}	{WANT, **arm**}
ball	{**ball**}	{**ball**, **arm**}

This lexicon constitutes partial information about the meanings of the word symbols *John*, *took*, *the*, and *ball*. The algorithm has not yet converged on the actual conceptual-symbol sets for any of the word symbols *John*, *took*, *the*, or *ball*, since *N*(*John*), *N*(*took*), *N*(*the*), and *N*(*ball*) are all proper subsets of *P*(*John*), *P*(*took*), *P*(*the*), and *P*(*ball*), respectively.

Now suppose that the algorithm receives the utterance

(1) *John took the ball.*

along with the following three hypothesized meanings for this utterance:

(2) CAUSE(**John**, GO(**ball**, TO(**John**)))
(3) WANT(**John**, **ball**)
(4) CAUSE(**John**, GO(PART-OF (LEFT(**arm**), **John**), TO (**ball**)))

Recall that the second condition on the semantic-interpretation rule requires that it cannot delete any conceptual symbols when producing utterance-meaning representations. Since the word symbol *took* must contribute the conceptual symbol CAUSE, because *N*(*took*) contains CAUSE, and since (3) is missing that conceptual symbol, the algorithm could rule out (3) as a possible meaning of (1). Similarly, given the first condition on the semantic-interpretation rule, that the semantic content of an utterance must be derived from the words in that utterance, the algorithm could rule out (4) as a possible meaning of (1), since (4) contains the conceptual symbols LEFT and PART-OF and none of the word symbols in (1) can possibly contribute those conceptual symbols, because neither *P*(*John*), *P*(*took*), *P*(*the*), nor *P*(*ball*) contain LEFT or PART-OF. This inference process can be stated more precisely as follows:

Rule 1 *Ignore those utterance meanings that contain a conceptual symbol that is not a member of P(w) for some word symbol w in the utterance. Also ignore those that are missing a conceptual symbol that is a member of N(w) for some word symbol w in the utterance.*

While, in this case, by applying Rule 1, the algorithm has eliminated the referential uncertainty from this utterance, this will not always be the case. Nonetheless, the remaining inference rules are formulated to tolerate residual referential uncertainty.

At this point, given the second condition on the semantic-interpretation rule, namely, that it cannot delete any conceptual symbols when producing utterance-meaning representations, the algorithm can make the following inference: Since (2), the remaining hypothesized meaning for (1), does not contain the conceptual symbols WANT and **arm**, the word symbols *took*, *the*, and *ball* cannot possibly contain those conceptual symbols as part of their meaning. This inference allows the algorithm to remove the conceptual symbols WANT and **arm** from *P*(*took*), *P*(*the*), and *P*(*ball*), yielding the following lexicon:

	N	*P*
John	{**John**}	{**John**, **ball**}
took	{CAUSE}	{CAUSE, GO, TO}
the	{}	{}
ball	{**ball**}	{**ball**}

This inference process can be stated more precisely as follows:

Rule 2 *For each word symbol w in the utterance, remove from P(w) any conceptual symbols that do not appear in some remaining utterance meaning.*

By applying Rule 2, the algorithm has converged on the actual conceptual-symbol set for the word symbols *the* and *ball*.

At this point, given the first condition on the semantic-interpretation rule, namely, that the semantic content of an utterance must be derived from the word symbols in that utterance, the algorithm can make the following inference: Since (2) contains the conceptual symbols GO and TO, and these conceptual symbols are not possibly part of the meaning of the word symbols *John*, *the*, and *ball*, the algorithm can infer that they must be part of the meaning of the word symbol *took*. This inference allows the algorithm to add the conceptual symbols GO and TO to *N*(*took*), yielding the following lexicon:

	N	*P*
John	{**John**}	{**John**, **ball**}
took	{CAUSE, GO, TO}	{CAUSE, GO, TO}
the	{}	{}
ball	{**ball**}	{**ball**}

This inference process can be stated more precisely as follows:

Rule 3 *For each word symbol w in the utterance, add to N(w) any conceptual symbols that appear in every remaining utterance meaning but that are missing from P(w′) for every other word symbol w′ in the utterance.*

By applying Rule 3, the algorithm has converged on the actual conceptual-symbol set for the word symbol *took*.

At this point, given the second condition on the semantic-interpretation rule, namely, that it cannot delete any symbols when producing utterance-meaning representations, the algorithm can make the following inference: Since the conceptual symbol **ball** appears only once in (2), and the word symbol *ball* necessarily contributes this conceptual symbol, the word symbol *John* cannot also contain **ball** as part of its meaning. This inference allows the algorithm to remove the conceptual symbol **ball** from *P*(*John*), yielding the following lexicon:

	N	*P*
John	{**John**}	{**John**}
took	{CAUSE, GO, TO}	{CAUSE, GO, TO}
the	{}	{}
ball	{**ball**}	{**ball**}

This inference process can be stated more precisely as follows:

Rule 4 *For each word symbol w in the utterance, remove from P(w) any conceptual symbols that appear only once in every remaining utterance meaning if they are in N(w′) for some other word symbol w′ in the utterance.*

By applying Rule 4, the algorithm has converged on the actual conceptual-symbol set for the word symbol *John*.

At this point, the algorithm has converged on the actual conceptual-symbol set for all of the word symbols that appear in (1). It can thus move from stage one, discovering the actual conceptual-symbol set of each word symbol, to stage two, discovering the conceptual expression of each word symbol. The algorithm can first initialize the possible conceptual-expression set for each of the word symbols in (1) to the universal set. Then it can remove from this set any conceptual expression not composed from the actual conceptual-symbol sets that have been inferred for those word symbols, in a way consistent with (2).[5] For example, the only conceptual expression that matches some subexpression of (2), and contains precisely the single conceptual symbol **John**, is, in fact, the conceptual expression

[5] Like before, the initial universal possible conceptual-expression set need not be instantiated extensionally. It would clearly be impossible to do so, since there are infinitely many conceptual expressions. Even so, possible conceptual-expression sets are computed as intersections of sets of fragments of the conceptual expressions that represent hypothesized utterance meanings. With a finite degree of referential uncertainty, these sets of fragments will always be finite, and thus the possible conceptual-expression sets will always be finite, except when initialized to the universal set. Like before, the universal set can be represented with a finite token, namely $\top$. The initial value of $\top$ is simply a cue to perform the first update (see Rule 5 in the Appendix) as $D(s) \leftarrow \bigcup_{m \in M}$ RECONSTRUCT(m, $N(s)$) instead of as $D(s) \leftarrow D(s) \cap \bigcup_{m \in M}$ RECONSTRUCT(m, $N(s)$). The token $\top$ can be treated similarly in the remaining rules to allow finite computations with universal sets.

John. Therefore, the word symbol *John* must mean **John**.[6] Similarly, the only conceptual expression that matches some subexpression of (2), and contains precisely the single conceptual symbol **ball**, is the conceptual expression **ball**. Therefore, the word symbol *ball* must mean **ball**. Likewise, since the meaning of the word symbol *the* does not contain any conceptual symbols, it must mean $\perp$. There are, however, two conceptual expressions that contain precisely the conceptual symbols CAUSE, GO, and TO, and can match some subexpression of (2), namely CAUSE(x, GO(y, TO(z))) and CAUSE(x, GO(y, TO(x))). Therefore, the best the algorithm can do, at this point, is to infer that the word symbol *took* must take on one of these two conceptual expressions as a representation of its meaning. In other words, by examining (2), the algorithm can determine that the first argument of CAUSE in the meaning of the word symbol *took* must be different from the remaining arguments. In contrast, the algorithm cannot determine whether the remaining arguments are necessarily, or just incidentally, the same. Thus, this technique will allow the algorithm to converge on the conceptual expressions that represent the meanings of the word symbols *John*, *the*, and *ball*, but leave some uncertainty as to the argument structure of the word symbol *took*.

Sometimes, this uncertainty can be resolved by intersecting the possible conceptual-expression sets derived in this manner from several different utterances that contain the same word symbol. Thus, the possible conceptual-expression sets can be updated in a manner analogous to the way Rule 2 updates the possible conceptual-symbol sets. This inference process can be stated more precisely as follows:

> **Rule 5** *Let* RECONSTRUCT*(m, N(w)) be the set of all conceptual expressions that unify (*Robinson, 1965*) with m, or with some subexpression of m, and that contain precisely the set N(w) of non-variable conceptual symbols. For each word symbol w in the utterance that has converged on its actual conceptual-symbol set, remove from D(w) any conceptual expressions not contained in* RECONSTRUCT*(m, N(w)), for some remaining utterance meaning m.*

By applying Rule 5, the algorithm has converged on the conceptual expressions that represent the meanings of the word symbols *John*, *the*, and *ball*.

In our example, Rule 5 alone cannot resolve all of the uncertainty. To remove the remaining uncertainty, the algorithm can observe that there is no way to take the word symbol *took* to mean CAUSE(x, GO(y, TO(z))) and consistently produce (2) as the meaning of (1), given the meanings that have been inferred so far for the word symbols *John*, *the*, and *ball*. Thus, the algorithm can rule out CAUSE(x,

[6] This is not as trivial as it appears. The fact that $P(John) = N(John) = \{$**John**$\}$ does not, in itself, constrain the meaning of *John* to be **John**. Since the algorithm is not given conceptual-symbol arity as explicit input, were it not for applying this inference rule to the hypothesized meaning representations, the algorithm could not rule out **John**(x), **John**(x, y), **John**(x, x), **John**(x, y, z), **John**(x, y, x), ... as potential meanings for *John*.

GO(y, TO(z))) as a possible meaning of the word symbol *took*, leaving CAUSE(x, GO(y, TO(x))) as the sole remaining alternative. This inference process can be stated more precisely as follows:

> **Rule 6** *If all word symbols in the utterance have converged on their actual conceptual-symbol sets, for each word symbol w in the utterance, remove from D(w) any conceptual expressions t, for which there do not exist possible conceptual expressions for the other word symbols in the utterance that can be given, as input, to* COMPOSE, *along with t, to yield, as its output, one of the remaining utterance meanings. This is a generalized form of arc consistency (*Mackworth, 1992*).*

By applying Rule 6, the algorithm has converged on the conceptual expression that represents the meaning of the word symbol *took*.

Appendix A contains formal statements of Rules 1–6. The noise-free monosemous algorithm essentially applies these rules repeatedly to each utterance, as it is received, until no change is made to the lexical entries of the word symbols that appear in the utterance. Then the utterance is discarded and the algorithm proceeds to the next utterance. While Rules 1–4 always terminate quickly, Rules 5 and 6 can potentially take a long time. Thus, a time limit is enforced whereby Rules 5 and 6 are aborted if they take too long. In practice, this time limit is exceeded only on a small fraction of the utterances, usually the long ones, and does not appear to adversely affect the convergence properties of the algorithm.

Nominally, the algorithm is not affected by utterance length. It can acquire partial information from utterances of any length, both during early stages of acquisition, when there is little or no partial information in the lexicon, as well as during later stages, when there is a more complete lexicon. The time limit on Rules 5 and 6 typically comes into play only during intermediate stages of acquisition. During the early stage of acquisition, most of the lexical entries are in stage one where Rules 5 and 6 do not yet apply. During the later stage of acquisition, the lexicon already possesses the meanings of most words in most utterances, so Rules 5 and 6 do not become combinatorially explosive and the time limit does not apply. The time limit only affects utterances that contain many words in stage two of the convergence path that have large possible conceptual-expression sets. Thus, the algorithm need not start learning with particularly short utterances. This paper presents computer simulations that demonstrate this capability of the algorithm.

7. Extensions to handle noise and homonymy

The algorithm described in the previous section runs into difficulty with noise and homonymy. This is illustrated by the following two simple examples. First, suppose that the learner heard the utterance *John lifted the ball* and paired this utterance with the single (correct) hypothesized utterance-meaning representation

CAUSE(**John**, GO(**ball**, UP)). Applying Rule 2, the learner would form the possible conceptual-symbol set {CAUSE, **John**, GO, **ball**, UP} for the word symbol *lifted*. Now suppose that the learner heard a second utterance *Mary lifted the ball*, but this time paired this utterance with the single incorrect hypothesized utterance meaning WANT(**Mary**, **ball**). This second utterance constitutes noise. Applying Rule 2, the learner would incorrectly update the possible conceptual-symbol set for the word symbol *lifted* to the set {**ball**}. Processing this utterance corrupts the possible conceptual-symbol set for the word symbol *lifted*, since it now lacks the conceptual symbols CAUSE, GO, and UP needed to represent the correct meaning of that word symbol. Second, suppose that the learner heard the utterance *Mary left school* and paired this utterance with the single hypothesis GO(**Mary**, FROM(**school**)). Now suppose that the learner heard a second utterance, *John hit Mary's left arm*, and paired this utterance with the single hypothesis HIT(**John**, PART-OF(LEFT(**arm**), **Mary**)). In this case, neither utterance is noisy, but the word symbol *left* is used in a different sense in the first utterance than in the second. Thus, applying Rule 2, the learner would form the possible conceptual-symbol set {GO, **Mary**, FROM, **school**} for the word symbol *left* after processing the first utterance and incorrectly update this possible conceptual-symbol set to {**Mary**} after processing the second utterance. The possible conceptual-symbol set for the word symbol *left* is now corrupted, since it lacks the conceptual symbols needed to represent the meanings of either of the two senses.

All of the rules described in the previous section are monotonic. They always add elements to the necessary conceptual-symbol sets and remove elements from the possible conceptual-symbol sets and possible conceptual-expression sets. When an impossible conceptual symbol is added to a necessary conceptual-symbol set, a necessary conceptual symbol is removed from a possible conceptual-symbol set, or a necessary conceptual expression is removed from a possible conceptual-expression set, I say that the resulting lexical entry is *corrupted*. So far, there is no way to recover from corruption due to noise and homonymy. Furthermore, corruption tends to spread through the lexicon, since Rules 3, 4, and 6 allow the lexical entries of words to be affected by the lexical entries of other words in the same utterance. Thus, a single noisy utterance or a single homonymous word can wreak havoc in the lexicon.

There is no simple way for the learner to determine when a lexical entry has been corrupted. It is possible, however, to determine a weaker property. In the absence of noise and homonymy, the algorithm described in the previous section maintains two invariants for each lexical entry: The necessary conceptual-symbol set will be a subset of the possible conceptual-symbol set and the possible conceptual-expression set will be non-empty. When either of these invariants is violated, I will say that a lexical entry is *inconsistent*. An inconsistent lexical entry is necessarily corrupted though the inverse might not be true. In practice, however, corrupted lexical entries tend to become inconsistent fairly quickly. This allows inconsistency to be used as an indicator of corruption, and ultimately of noise and homonymy.

There is an additional form of inconsistency that the algorithm can discover. In the absence of noise and homonymy, the set of hypothesized meanings associated with an utterance must contain the correct meaning. That meaning should not be eliminated by Rule 1. Thus, an inconsistency is detected whenever Rule 1 eliminates all of the hypothesized meanings associated with some utterance as it is processed.

Detecting an inconsistency when processing an utterance is indicative of one or more of the following situations:

- the current utterance is noisy;
- a word in the current utterance is homonymous; or
- the lexical entry for some word in the current utterance has been corrupted by processing a previous utterance.

I will now describe an extended algorithm for learning in the presence of noise and homonymy that provides a uniform method for dealing with each of these situations.

The extended algorithm represents the lexicon as a two-level structure that first maps word symbols to *sense symbols*, and then maps sense symbols to conceptual expressions. Sense symbols are simply atomic tokens, such as s_1, s_2, ..., that are used to name senses. For example, the lexicon might map the word symbol *ball* to the two sense symbols $ball_1$ and $ball_2$, and then map these sense symbols to the conceptual expressions **spherical-toy** and **formal-dance-party**, respectively. I refer to the set of sense symbols associated with a word symbol as the *sense-symbol set* of that word symbol. The lexicon has the property that no two word symbols can map to sense-symbol sets that contain the same sense symbol. Thus, sense symbols can be viewed as homonymous sense indices created on-the-fly when a new sense is hypothesized.

In the extended algorithm, the possible conceptual-symbol table $P(s)$, the necessary conceptual-symbol table $N(s)$, and the possible conceptual-expression table $D(s)$ all map sense symbols, rather than word symbols, to lexical entries. The extended algorithm makes use of two additional tables as part of its model of the mental lexicon:

1. a sense-symbol table, $L(w)$, that maps each word symbol w to its sense-symbol set; and
2. a confidence-factor table, $C(s)$, that maps each sense symbol s to a *confidence factor*, a non-negative integer.

The sense-symbol table $L(w)$ and the possible conceptual-expression table $D(s)$ constitute the two-level output lexicon produced by the extended algorithm. I refer to the collection of $P(s)$, $N(s)$, $D(s)$, and $C(s)$ as the *lexical entry* for the sense symbol s and to the collection of lexical entries for all of the sense symbols in $L(w)$ as the lexical entry for the word symbol w. The confidence-factor table is used to handle noise and homonymy and will be described momentarily. Briefly, the confidence factor of each sense is initially zero and increases as the algorithm gathers more evidence that it has not mistakenly hypothesized that sense to explain a noisy utterance.

The extended algorithm is best described as the combination of several general principles. First, let us momentarily make the simplifying assumption, as before, that all word symbols map to a single sense symbol, i.e. that there is no homonymy in the lexicon. In this case, one can determine whether processing an utterance would result in an inconsistency, without actually letting such an inconsistency corrupt the lexicon, simply by saving the state of the lexical entries under consideration before processing an utterance and restoring them should an inconsistency arise during processing. Second, let us now relax the monosemy constraint and allow the lexicon to contain multiple senses per word. In this case, one can decide whether an utterance is inconsistent by testing the consistency of each element in the cross product of the sense-symbol sets of the word symbols in the utterance. Each element in such a cross product is termed a *sense assignment.* If no sense assignment in the cross product is consistent, then treat the utterance as inconsistent. I will explain how to deal with such inconsistent utterances momentarily. If exactly one sense assignment in the cross product can be processed without inconsistency, then assume that the sense symbols contained in that sense assignment denote, in fact, the intended senses for each word symbol and permanently update the lexical entries of those sense symbols using Rules 1–6. If more than one sense assignment in the cross product can be processed without inconsistency, then some metric is used to select the best sense assignment and that sense assignment is processed as before. The sum of the confidence factors for each sense symbol in a sense assignment is currently used as the selection metric, though presumably other selection metrics could be used as well.

The above strategy has two objectives: to perform sense disambiguation on the words of incoming utterances and to prevent corruption. This strategy meets these objectives only partially. It is possible, particularly during early stages of learning, for a noisy utterance to corrupt the lexicon without being detected as an inconsistency. It is also possible for the selection metric to incorrectly disambiguate word senses and cause the wrong lexical entries for some word symbol to be processed and thus corrupted. Nonetheless, such situations occur much less frequently than would otherwise be the case if consistency were to be ignored. Techniques that I will describe momentarily can handle such residual cases of incorrect sense disambiguation and corruption.

The question then remains as to what to do when processing an inconsistent utterance. The strategy adopted here is to incrementally add newly created sense symbols to the sense-symbol sets of the word symbols that appear in the utterance, until the utterance is no longer inconsistent, and then process that utterance as usual. The lexical entries of the newly added sense symbols are initially unconstrained, that is, they have empty necessary conceptual-symbol sets, universal possible conceptual-symbol sets, and universal possible conceptual-expression sets. Clearly, it is always possible to render an utterance consistent simply by adding a single new sense symbol to the sense-symbol set for each word-symbol occurrence in the utterance. The algorithm finds the smallest number of new sense symbols that need to be added, in order to process the utterance without detecting an inconsistency, and adds only those new sense symbols.

New sense symbols can be added in this fashion for several reasons:

1. The current utterance is noise. In this case, the new sense symbols are spurious. They are only created to explain a noisy utterance. It is unlikely that the lexical entries of such sense symbols will converge and be selected to explain a future utterance. Such sense symbols will be filtered out by a sense-pruning process to be described momentarily.
2. The newly created sense symbols do indeed represent new senses (potentially for words that already posses other senses) that have not been heard before. The lexical entries of these new sense symbols will begin traversing the convergence path and will hopefully converge to the correct meaning representation.
3. The lexical entries for some of the word symbols in the current utterance have previously been corrupted and thus can no longer account for the current utterance. The lexical entries of these new sense symbols are intended to replace and repair the corrupted lexical entries of the old sense symbols. The lexical entries of these new sense symbols will begin traversing the convergence path and will hopefully converge to the correct meaning representation. It is unlikely that the corrupted lexical entries of the old sense symbols will converge and be selected to explain a future utterance. Such sense symbols will be filtered out by a sense-pruning process to be described momentarily.

This strategy for handling noise and homonymy is best illustrated by the following example. Like before, suppose that the algorithm is part-way through the lexical-acquisition process and already possesses the following lexicon:[7]

John$_1$	$D=\{$**John**$\}$, $C=1000$
saw$_1$	$N=\{\}$, $P=\{$**wood-cutting-tool**, **hammer**$\}$
saw$_2$	$N=\{\}$, $P=\{$SEE, GO$\}$
had$_1$	$N=\{\}$, $P=\{$POSSESS, WANT$\}$
had$_2$	$N=\{\}$, $P=\{$CONDUCT, CAUSE$\}$
Mary$_1$	$D=\{$**Mary**$\}$, $C=1000$
arrive$_1$	$D=\{\text{GO}(x, \text{TO}(\text{BE}(x, y)))\}$, $C=10$
at$_1$	$D=\{\text{AT}(x)\}$, $C=1000$
the$_1$	$D=\{\perp\}$, $C=10\,000$
a$_1$	$D=\{\perp\}$, $C=10\,000$
ball$_1$	$D=\{$**formal-dance-party**$\}$, $C=10$
ball$_2$	$D=\{$**spherical-toy**$\}$, $C=1000$
party$_1$	$D=\{$**political-organization**$\}$, $C=10$
Susan$_1$	$N=\{\}$, $P=\{$DANCE, **Mary**, **Betty**, **Susan**$\}$
with$_1$	$D=\{\text{WITH}(x)\}$, $C=1000$

[7] In this example, the confidence factor for a sense symbol is incremented each time that sense symbol is contained in a preferred sense assignment. The actual procedure used for updating the confidence factors is somewhat more complex and is presented in the Appendix A.

This lexicon is homonymous and contains two senses for each of the word symbols *saw*, *had*, and *ball*.

Now suppose that the algorithm receives the utterance

(5) *John saw Mary arrive at the ball.*

in a context where John saw Mary arrive at a formal dance party. The algorithm would thus obtain the following hypothesized meaning for this utterance:

SEE(**John**, GO(**Mary**, TO (BE(**Mary**, AT(**formal-dance-party**)))))

Since this utterance contains two homonymous word symbols, each with two senses in the current lexicon, the algorithm would examine four possible sense assignments. Of these, only one is consistent with the lexicon.

$John_1$ saw_2 $Mary_1$ $arrive_1$ at_1 the_1 $ball_1$.

Note that the algorithm can determine this, even though it has yet to converge on the conceptual expressions, or even the actual conceptual-symbol sets, for the two senses of the word symbol *saw*. In this case, the algorithm assumes that the above sense assignment is the intended one and updates the lexical entries for the sense symbols in this sense assignment using Rules 1–6. This allows the algorithm to converge on the conceptual expression SEE(x, y) for the sense symbol saw_2, yielding the following modified lexical entry:

saw_2 | $D=\{\text{SEE}(x, y)\}$, $C=1$

The confidence factors of the sense symbols $John_1$, $Mary_1$, $arrive_1$, at_1, the_1, and $ball_1$ are incremented as well.

Processing (5) illustrates how the set of hypothesized utterance meanings can often be used to disambiguate lexical ambiguity. This is handled by Step 2a of the algorithm that will be described momentarily. Using the set of hypothesized utterance meanings to disambiguate lexical ambiguity is only a heuristic, however. While not shown in this example, it is possible for this heuristic to select an incorrect disambiguation. This might corrupt some lexical entries. An example that illustrates recovery from such corruption will be shown momentarily.

Now suppose that the algorithm receives the utterance

(6) *John had a ball.*

in a context where this could refer either to the fact that John owned a spherical toy or to the fact that John had conducted a formal dance party. The algorithm would thus obtain the following two hypothesized meanings for this utterance:

POSSESS(**John**, **spherical-toy**)
CONDUCT(**John**, **formal-dance-party**)

Again, there are two homonymous word symbols in this utterance, each with two senses in the current lexicon. Thus, the algorithm would examine four possible sense assignments. This time, however, there are two sense assignments that are consistent with the current lexicon.

$John_1$ had_1 a_1 $ball_2$.
$John_1$ had_2 a_1 $ball_1$.

In this case, the algorithm computes a *selection metric* for each consistent sense assignment and prefers the one with the highest selection metric. The algorithm maintains a *confidence factor* for each sense in the lexicon that counts the number of times that that sense was used to process an utterance. The confidence factor takes on a zero value until a sense has converged on the conceptual expression that represents the meaning of that sense. The selection metric is taken to be the sum of the confidence factors of the sense symbols in a sense assignment. Thus, the algorithm computes the following selection-metric values:

$$C(John_1) + C(had_1) + C(a_1) + C(ball_2) = 12\,001$$

$$C(John_1) + C(had_2) + C(a_1) + C(ball_1) = 11\,012$$

and prefers the first sense assignment since it has the highest selection-metric value. It then updates the lexical entries for the sense symbols in this sense assignment using Rules 1–6. This allows the algorithm to converge on the conceptual expression POSSESS(x, y) for the sense symbol had_1, yielding the following modified lexical entry:

had_1 | $D = \{\text{POSSESS}(x, y)\}$, $C = 1$

The confidence factors of the sense symbols $John_1$, a_1, and $ball_2$ are incremented as well.

Processing (6) illustrates how the set of hypothesized utterance meanings might not fully disambiguate the lexical ambiguity and how a selection metric based on confidence factors can be used to further disambiguate the lexical ambiguity. This is handled by Step 2b of the algorithm that will be described momentarily. Confidence factors are a reasonable selection metric since they are a rough measure of the relative frequency of occurrence of different word senses. Using this selection metric to disambiguate lexical ambiguity is only a heuristic, however. While not shown in this example, it is possible for this heuristic to select an incorrect disambiguation and corrupt some lexical entries. An example that illustrates recovery from such corruption will be shown momentarily.

Now suppose that the algorithm receives the utterance

(7) *Mary had a party.*

in a context where Mary conducts a celebration. The algorithm would thus obtain the following hypothesized meaning for this utterance:

CONDUCT(**Mary**, **celebration**)

There is one homonymous word symbol, *had*, in this utterance, with two senses in the current lexicon. Notice that the current lexicon has only one sense for the word symbol *party*, namely, the sense that denotes a political organization. Thus, there are two possible sense assignments. Neither of these, however, is consistent with the current lexicon. Thus, the algorithm tries to find the smallest set of word symbols from the current utterance for which it can add a new sense to alleviate the inconsistency. In this case, it is possible to add a new sense, $party_2$, for the word symbol *party* and alleviate the inconsistency. Initially, the necessary conceptual-symbol set for this sense is set to the empty set and the possible conceptual-symbol set for this sense is set to the universal set. Application of Rules 1–6, however, to the sense assignment

$Mary_1$ had_2 a_1 $party_2$.

allow this new sense $party_2$ to converge on the conceptual expression **celebration**, and also allow the existing sense had_2 to converge on the conceptual expression CONDUCT(x, y), yielding the following new and modified lexical entries:

had_2	$D = \{\text{CONDUCT}(x, y)\}$, $C = 1$
$party_2$	$D = \{\textbf{celebration}\}$, $C = 1$

The confidence factors of the sense symbols $Mary_1$ and a_1 are incremented as well.

Processing (7) illustrates how the algorithm can learn new senses for words that are already in the lexicon. This is handled by Step 2c of the algorithm that will be described momentarily. In this case, the new senses correspond to legitimate homonymy. As the next example will show, this is not always the case.

Now suppose that the algorithm receives the utterance

(8) *John had a ball at the party.*

in a context where John dances with lots of different women at the celebration. The algorithm might thus obtain the following three hypothesized meanings for this utterance:

DANCE(**John**, WITH(**Mary**), AT(**celebration**))
DANCE(**John**, WITH(**Betty**), AT(**celebration**))
DANCE(**John**, WITH(**Susan**), AT(**celebration**))

This utterance contains three homonymous word symbols, *had*, *ball*, and *party*, each with two senses in the current lexicon. Thus, there are eight possible sense assignments. None of these, however, are consistent with the current lexicon and set of hypothesized utterance meanings. Here again, the algorithm tries to find the smallest set of word symbols from the current utterance for which it can add a new sense to each word symbol in that set to alleviate the inconsistency. In this case, there is no single word symbol for which the addition of a new sense would alleviate the inconsistency. It is possible, however, to alleviate the inconsistency by adding new senses for the two word symbols *had* and *party*. Initially, the necessary and possible conceptual-symbol sets for the new sense symbols had_3 and $party_3$ are set to the empty set and universal set, respectively. Application of Rules 1–6, however, to the sense assignment

$John_1$ had_3 a_1 $ball_3$ at_1 the_1 $party_2$.

yields the following new lexical entries:

had_3	N={DANCE, WITH}, P={DANCE, WITH, **Mary**, **Betty**, **Susan**}
$ball_3$	N={DANCE, WITH}, P={DANCE, WITH, **Mary**, **Betty**, **Susan**}

The confidence factors of the sense symbols $John_1$, at_1, the_1, a_1, and $party_2$ are incremented as well.

Processing (8) illustrates how the algorithm can learn in the presence of noise. This is also handled by Step 2c of the algorithm that will be described momentarily. In this case, the new senses are spurious. They are postulated solely to account for a noisy utterance. It is unlikely, however, that the algorithm will receive sufficient further evidence to allow the sense symbols had_3 and $ball_3$ to converge. They will ultimately be removed from the lexicon by a periodic pruning process.

Now suppose that the algorithm receives the utterance

(9) *John danced with Susan at the party.*

in a context where John danced with Betty at the celebration. The algorithm would thus obtain the following hypothesized meaning for this utterance:

DANCE(**John**, WITH(**Betty**), AT(**celebration**))

This utterance contains the homonymous word symbol *party*, as well as a new word symbol, *danced*, that is not currently in the lexicon. Because of the new word symbol, this utterance is, by definition, inconsistent. The algorithm must, at least, create the new sense $danced_1$. In this case, it turns out that creating just this one new sense is sufficient to alleviate the inconsistency. Doing so leaves two possible sense assignments. Of these, only one is consistent.

John$_1$ *danced*$_1$ *with*$_1$ *Susan*$_1$ *at*$_1$ *the*$_1$ *party*$_2$.

The algorithm thus assumes that this sense assignment is the intended one and updates the lexical entries for the sense symbols in this sense assignment using Rules 1–6, yielding the following new and modified lexical entries:

danced$_1$	$N=\{\}$, $P=\{$DANCE, **Betty**$\}$
Susan$_1$	$N=\{\}$, $P=\{$DANCE, **Betty**$\}$

The confidence factors of the sense symbols *John*$_1$, *at*$_1$, *the*$_1$, *party*$_2$, and *with*$_1$ are incremented as well.

Processing (9) illustrates how the disambiguation strategies might incorrectly choose a word sense, might fail to notice the need to postulate a new word sense, or might fail to notice a noisy utterance. Such mistakes can corrupt a lexical entry such as the entry for the sense symbol *Susan*$_1$. The next example will show how to recover from such corruption. Processing (9) also illustrates how the algorithm can begin learning new words like *dance*, even from a noisy utterance.

Now suppose that the algorithm receives the utterance

(10) *John kissed Susan at the party.*

in a context where John kisses Susan at the celebration. The algorithm would thus obtain the following hypothesized meaning for this utterance:

KISS(**John**, **Susan**, AT(**celebration**))

This utterance contains the homonymous word symbol *party*, as well as a new word symbol, *kissed*, that is not currently in the lexicon. Because of the new word symbol, this utterance is, by definition, inconsistent. The algorithm must, at least, create the new sense *kissed*$_1$. In this case, it turns out that creating just this one new sense is not sufficient to alleviate the inconsistency. Both of the following possible sense assignments are still inconsistent:

John$_1$ *kissed*$_1$ *Susan*$_1$ *at*$_1$ *the*$_1$ *party*$_1$.
John$_1$ *kissed*$_1$ *Susan*$_1$ *at*$_1$ *the*$_1$ *party*$_2$.

Thus, the algorithm must create additional new senses to alleviate the inconsistency. In this case, this can be done be creating a new sense symbol *Susan*$_2$ for the word symbol *Susan*. After doing so, the algorithm must then examine four possible sense assignments. Of these, only one is consistent.

John$_1$ *kissed*$_1$ *Susan*$_2$ *at*$_1$ *the*$_1$ *party*$_2$.

The algorithm thus assumes that this sense assignment is the intended one and updates the lexical entries for the sense symbols in this sense assignment using Rules 1–6, yielding the following new lexical entries:

Susan$_2$	$N=\{\}$, $P=\{$KISS, **Susan**$\}$
kissed$_1$	$N=\{\}$, $P=\{$KISS, **Susan**$\}$

The confidence factors of the sense symbols *John*$_1$, *at*$_1$, *the*$_1$, and *party*$_2$ are incremented as well.

Processing (10) illustrates how the algorithm can recover from a corrupt lexicon by creating a new sense *Susan*$_2$ to supersede a previously corrupted lexical entry *Susan*$_1$. It is unlikely that corrupted lexical entries like *Susan*$_1$ will progress far along the convergence path and thus they will ultimately be pruned.

The strategy illustrated in the above series of examples can be stated more precisely as the following algorithm:

> The input to the algorithm consists of a sequence of utterances, each being an unordered collection $w_1,\ldots,w_n$ of word symbols. Each utterance is paired with a set of conceptual expressions that represent hypothesized meanings of that utterance. The sense-symbol table $L(w)$ initially maps each word symbol w to the empty set of sense symbols. Apply the following steps to each utterance as it is processed:

1. Consider all unordered collections $s_1,\ldots,s_n$ of sense symbols in the cross product $L(w_1)\times\cdots\times L(w_n)$. Each such unordered collection is taken to be a sense assignment. Apply Rules 1–6 to each sense assignment to determine which ones lead to inconsistencies. Save the lexical entries of $s_1,\ldots,s_n$ before applying Rules 1–6 and restore these saved lexical entries after the rule applications.
2. One of the following three situations will now exist:
   - 2.1. *Exactly one sense assignment in the cross product is consistent.* In this case, apply Rules 1–6 permanently to this sense assignment and proceed to the next utterance.
   - 2.2. *More than one sense assignment in the cross product is consistent.* In this case, choose the sense assignment that maximizes the selection metric $C(s_1)+\cdots+C(s_n)$, apply Rules 1–6 permanently to this sense assignment, and proceed to the next utterance.
   - 2.3. *No sense assignment in the cross product is consistent.* In this case, find the smallest subset of word symbols in the current utterance such that if a new sense symbol would be added to the sense-symbol set for each word symbol in that subset, Step (1) would not lead to an inconsistency. Add a new sense symbol to each of the sense-symbol sets of the word symbols in that minimal subset and reprocess this utterance starting with Step (1).

This algorithm will not enter an infinite loop, since once control passes through Step (2c), it must pass through either Step (2a) or Step (2b) on the second pass.

The above strategy makes use of a number of heuristics, among them, using consistency to approximate corruption, and the selection metric used to perform sense disambiguation. These heuristics are imperfect. At times, they let consistent but corrupt lexical entries pass unnoticed. It is unlikely, however, that such lexical entries would be used to explain an utterance, that is, to account for how one of the hypothesized meanings for that utterance is derived from the meanings of the words in that utterance. This leads to the following simple sense-pruning strategy: Every so often, discard sense symbols that have not been used to explain many utterances. This is implemented by means of the confidence factor. Roughly speaking, the confidence factor is the number of utterances that a given sense symbol has been used to explain. It is an approximate measure of the relative frequency of occurrence of a sense and is used both to govern the sense-pruning strategy as well as to compute the selection metric for word-sense disambiguation. Senses with sufficiently high confidence factors are immune from pruning and are said to be *frozen*. A more precise definition of the pruning strategy and the method for determining confidence factors is included in Appendix A.

Parts of this extended algorithm can be time-consuming to compute, particularly analyzing all sense assignments in a cross product or finding the minimal number of new senses to add. Thus, a time limit is enforced whereby an utterance is discarded if it takes too long to process. Like the earlier time limit on Rules 5 and 6, this time limit is exceeded only on a small fraction of the utterances, usually the long ones. Again, it does not appear to adversely affect the convergence properties of the algorithm.

8. Simulations

An attempt was made to assess the efficacy of the learning algorithm presented here. Four studies were performed.[8] First, an attempt was made to determine how well the algorithm scales as the complexity of the learning task varied along five independent axes. This is important because there are many parameters of the learning task, such as the degree of referential uncertainty, the noise rate, the conceptual-symbol inventory size, and the homonymy rate, that depend on the form of mental representations about which we currently know very little. This first series of studies attempted to determine which of these parameters materially affect the efficacy of the learning algorithm and which do not. Second, the growth in size of the vocabulary attained by the algorithm was measured as a function of its exposure to a simulated training corpus. It is commonly believed that lexical acquisition in children starts off slowly, for the first 50 or so words, then proceeds at a rapid pace, and ultimately tapers off as the child attains fluency. This second series of simulations was performed to see if the algorithm exhibits the same

[8] The programs and data used in these studies are available from http://www.emba.uvm.edu/~qobi.

behavior. Third, the number of exposures to a new word that is required to learn that word was measured as a function of the amount of the corpus already processed at the time of the new word occurrence. Carey (1978) has observed that older children learn at least part of the meaning of many words from a single exposure. This third series of simulations was performed to see if the algorithm exhibits this same behavior. Finally, a fourth simulation was performed to determine whether the algorithm could solve a very large learning task whose complexity approaches the complexity of the task faced by children. Before presenting the results of each of these studies, I will first present the experimental method used to conduct the simulations.

8.1. Method

The algorithm presented here learns from utterances paired with hypothesized utterance meanings, but there are no corpora of naturally-occurring utterances paired with such meaning representations. As a consequence, the algorithm has been tested on synthetic corpora generated randomly according to controllable distributional parameters. The simulations were conducted according to the following general strategy. For each simulation, a random lexicon was constructed that maps simulated words to simulated meanings. This *original* lexicon was then fed into a process that generates a potentially unlimited stream of random utterances paired with sets of meaning hypotheses. The lexical-acquisition algorithm was then applied in an on-line fashion to this stream to produce a *reconstructed* lexicon without benefit of access to the original lexicon. During the simulation, however, the reconstructed lexicon was continually compared with the original lexicon by a mechanism distinct from the acquisition algorithm. Each simulation was terminated when the reconstructed lexicon contained a target fraction of the correct word-to-meaning mappings from the original lexicon. A target *convergence goal* of 95% was used for all of the simulations reported in this paper. The reason that the simulations were terminated prior to achieving total convergence is that, as will be discussed shortly, a lexical-choice rule based on Zipf's Law was used to generate the stream of random utterances. Because Zipf's Law implies that many words occur very infrequently, convergence on the last 5% of the lexicon proceeds very slowly. Computer resource limitations prevented running the simulations with a higher convergence goal.

The following procedure was used to generate the lexicon for each simulation. This procedure was driven by three independent parameters: the *vocabulary size*, the *homonymy rate*, and the *conceptual-symbol inventory size*. Given a vocabulary size of n and a homonymy rate of r, the generated lexicon would map n words to rn senses. The "words" in this lexicon were simply the symbols $w_1,..., w_n$. Given a conceptual-symbol inventory size of m, the "meanings" of the senses in this lexicon were represented with randomly-constructed conceptual expressions over the conceptual symbols $f_1,..., f_m$. A uniform distribution was used to select the conceptual symbols when constructing the random conceptual expressions. The meanings of 47.5% of the senses were represented with variable-free conceptual expressions. These had a maximal depth of 2 and a maximal branching factor of 3 and were intended to model noun-like meanings. A typical conceptual expression of maximal depth 2 and maximal branching factor 3 used to model a noun-like meaning would look like $f_1(f_6, f_2(f_4, f_3), f_2(f_7, f_8))$. The conceptual expressions used to represent the meanings of another 47.5% of the senses contained from 1 to 3 variables to denote open argument positions. These were intended to model verb-like meanings and had the same maximal depth and branching factor. A typical 2-variable conceptual expression of maximal depth 2 and maximal branching factor 3 used to model a verb-like sense would look like $f_5(x, f_9(x, y, f_{10}))$. A uniform distribution was used to control the choice of depth and branching factor when constructing the random conceptual expressions.[9] The meaning representations of the final 5% of the senses were taken to be $\perp$ to model function words. All of these senses were distributed uniformly among the words. Some words contained only a single sense while others contained several. A given word could have a mixture of senses with noun-like, verb-like, and function-word-like meanings.

Given a lexicon, random utterance–meaning pairs were constructed by applying the following grammar in a top-down fashion, starting with category S:

$$\begin{aligned}&\mathrm{S} \rightarrow \mathrm{XP}\\&\mathrm{XP} \rightarrow \mathrm{NP}|\mathrm{VP}\\&\mathrm{NP} \rightarrow \{\mathrm{F}\}\mathrm{N}\\&\mathrm{VP} \rightarrow \{\mathrm{F}\}\mathrm{V}\ \mathrm{XP}^{+}\end{aligned}$$

In other words, each utterance is a phrase, a phrase is either a noun phrase or a verb phrase, a noun phrase consists of a noun and an optional function word, and a verb phrase consists of a verb, an optional function word, and a complement phrase to fill each argument position. Since the word order of each utterance was randomized before being presented to the acquisition algorithm, the order of the categories in the right-hand sides of the above rules is unimportant. Furthermore, single-word utterances, utterances that contained more than 30 words, and utterances paired with meaning expressions that contained more than 30 conceptual symbols were discarded.

When generating random utterance–meaning pairs, all rules in the above grammar were taken to be equiprobable. The terminal categories N, V, and F were filled randomly with noun-like, verb-like, and function-word-like entries from the lexicon, respectively. The number of XP complements associated with each V node was chosen to match the number of arguments required by the meaning of the verb-like lexical entry selected to fill that V node. The entries used to fill the

[9] Since we know very little about the actual shape and size of human conceptual representations, it is not possible to justify the choice of maximum depth or branching factor made here. The maximal depth of 2 and maximal branching factor of 3 were chosen simply to allow the use of representations like AND(**round**, COLOR(**surface**, RED), COLOR(**inside**, WHITE)) and CAUSE(*x*, GO(*x*, *y*, **rollingly**)) for words like *apple* and *roll*, respectively. Representations of similar complexity have been proposed by many authors including Leech (1969), Miller (1972), Schank (1973), Jackendoff (1983, 1990), Borchardt (1985), and Pinker (1989).

terminal categories were selected using a distribution based on Zipf's Law. Under Zipf's Law, the occurrence frequency of a word is inversely proportional to its rank. Zipf (1949) argues that such a relationship fits empirical word-frequency measurements. A lexical-choice rule based on Zipf's Law should make learning word-to-meaning mappings difficult since many words will occur very infrequently.

The above procedure pairs each utterance with its correct meaning representation. The following extended procedure was used to generate utterances paired with sets of hypothesized meaning representations to model referential uncertainty. This extended procedure was driven by four independent parameters: the *noise rate*, the *degree of referential uncertainty*, the *cluster size*, and the *similarity probability*. The noise rate p specified the probability of generating a noisy utterance. The degree of referential uncertainty l specified the number of meaning hypotheses paired with each utterance. Noisy utterances were paired with l semantic representations corresponding to l other randomly-generated utterances. Non-noisy utterances were paired with their correct semantic representation, along with the semantic representations of $l-1$ other randomly-generated utterances.

It is unlikely that the meaning hypotheses that children formulate for utterances are distributed uniformly in semantic space. They are likely to be divided into clusters, where each cluster differs significantly in meaning from each other cluster, but where the hypotheses in each cluster are relatively similar. To model this possibility, the following procedure was used to generate the alternate meaning hypotheses associated with each utterance. A cluster size of k specified that the l hypotheses associated with each utterance were grouped into l/k clusters, each containing k hypotheses. Given a similarity probability q, each cluster was generated in the following fashion. First, an initial seed semantic representation was generated randomly by the aforementioned grammar. Then, additional semantic representations were generated using the same parse tree, but where lexical entries at the leaf nodes in the parse tree were randomly replaced, with probability $1-q$, with alternate lexical entries. Noun-like entries were used to replace noun-like entries and verb-like entries were used to replace verb-like entries that shared the same number of unfilled argument positions. When generating a noisy utterance, the l hypotheses were associated with a randomly-generated utterance. When generating a non-noisy utterance, the l hypotheses were associated with the utterance used to generate one of the seed semantic representations for one of the clusters. Thus, non-noisy utterances were associated with a correct meaning hypothesis, a cluster of near misses, and several clusters of incorrect hypotheses, while noisy utterances were associated only with clusters of incorrect hypotheses. A cluster size of 5 and a similarity probability of 0.75 was used for all of the simulations reported in this paper.

8.2. Sensitivity analysis

A number of simulations were performed to determine the sensitivity of the algorithm to the various corpus-construction parameters. For these simulations, a baseline run was performed with the following parameters: a vocabulary size of 1000 words, a degree of referential uncertainty of 10, a noise-rate of 0%, a conceptual-symbol inventory size of 250, and a homonymy rate of 1.0 (no homonymy). Then, three additional runs were performed for each of the five parameters, varying that parameter independently while keeping the remaining parameters at their baseline values. The varying corpus-construction parameters for the different simulation runs are summarized in the following table:

Parameter	Baseline			
Vocabulary size	1000	2500	5000	10,000
Degree of referential uncertainty	10	25	50	100
Noise rate	0%	5%	10%	20%
Conceptual-symbol inventory size	250	500	1000	2000
Homonymy rate	1.00	1.25	1.50	2.00

For each simulation, the number of utterances needed to achieve the target 95% convergence goal was measured. The results of these simulations are summarized in Figs. 2–6. The algorithm appears to scale linearly in the vocabulary size and

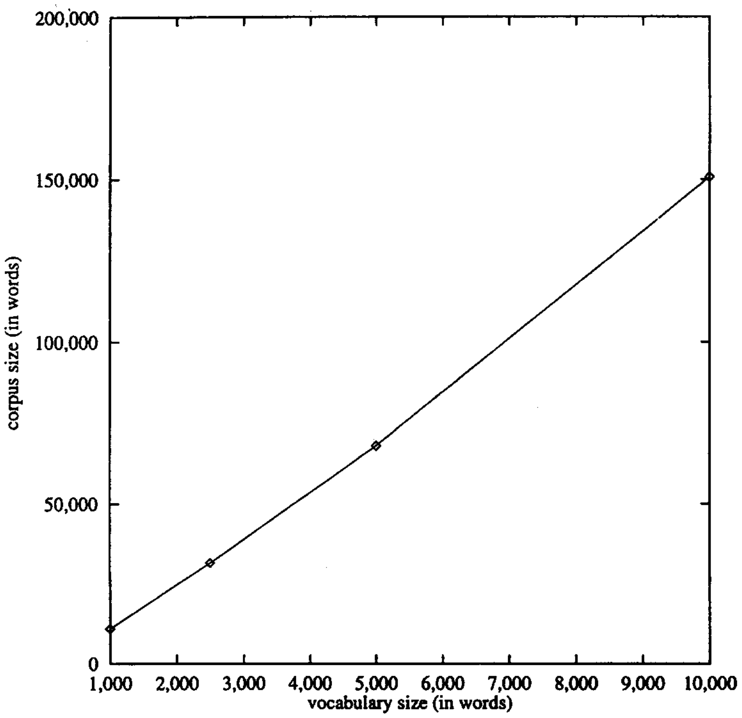

Fig. 2. Corpus size needed for 95% convergence as a function of the vocabulary size.

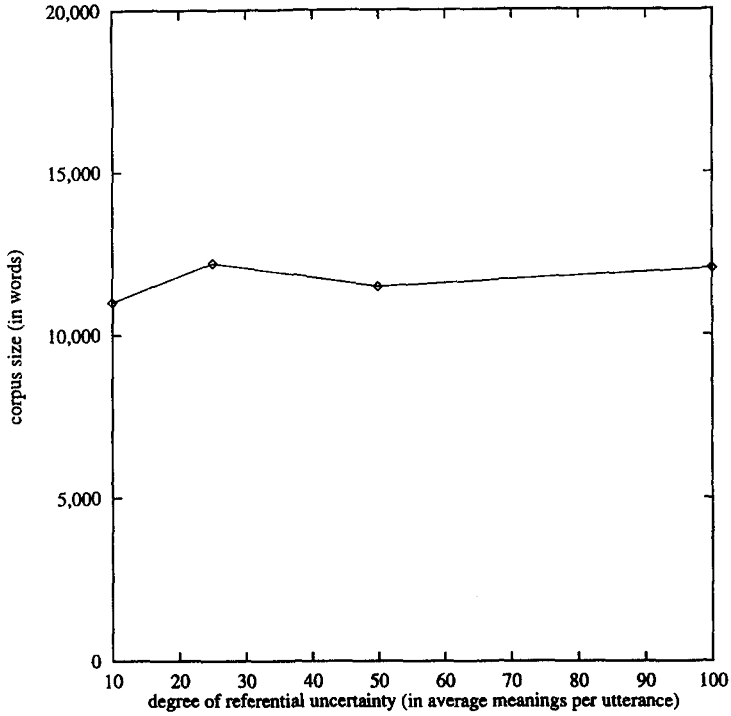

Fig. 3. Corpus size needed for 95% convergence as a function of the degree of referential uncertainty.

appears to be insensitive to the degree of referential uncertainty and the conceptual-symbol inventory size. The problem grows more difficult with increasing noise and homonymy.

The length of utterances processed during each of these simulations ranged from 2 to 29 words. The mean utterance length (MLU) varied from simulation run to simulation run and ranged from 4.99 to 6.29. Since all of these simulations were performed with a convergence goal of 95%, each of the reconstructed lexicons was missing 5% of the word-to-meaning mappings contained in the corresponding original lexicon. These constitute false negatives. In the absence of noise or homonymy, the reconstructed lexicon never contained false positives, that is, word-to-meaning mappings not contained in the original lexicon. The number of false positives produced in the presence of noise or homonymy is summarized by the following table:

Noise rate	0%	5%	10%	20
False positives	0	9	32	49
Homonymy rate	1.00	1.25	1.50	2.00
False positives	0	11	10	20

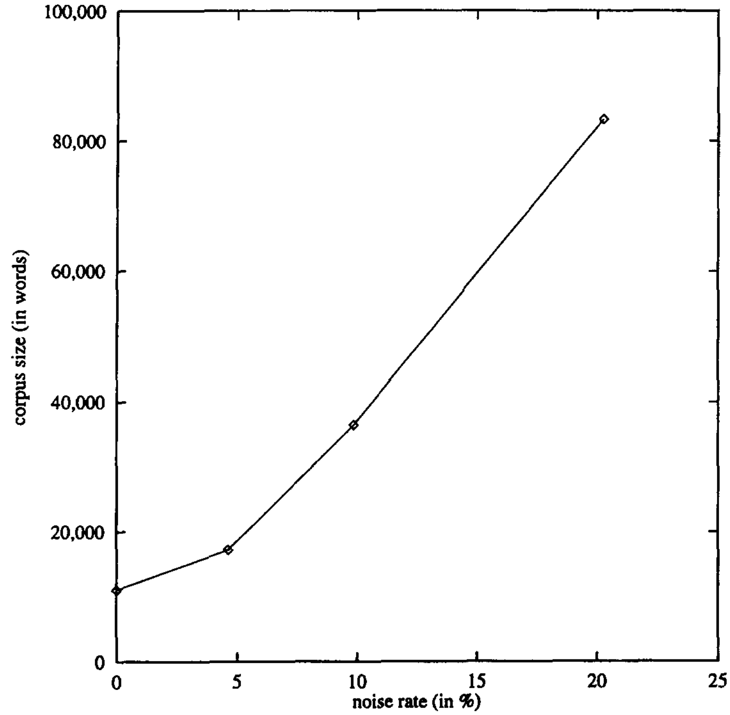

Fig. 4. Corpus size needed for 95% convergence as a function of the noise rate.

8.3. Vocabulary growth

Fig. 7 shows the vocabulary growth as a function of the number of utterances processed during the baseline run. Since convergence on actual conceptual-symbol sets was very nearly identical to convergence on conceptual expressions, only the latter is plotted. Furthermore, since the baseline run did not contain any noise or homonymy, no spurious senses were hypothesized. Thus, the sense convergence rate was identical to the word convergence rate and only the latter is plotted. Note that the simulation exhibits behavior similar to children. The learning rate is slow for the first 25 words or so, then proceeds rapidly, and ultimately tapers off as the algorithm nears convergence.

8.4. Learning rate

Fig. 8 -shows the number of occurrences needed to learn a word meaning (measured as convergence on conceptual expression) as a function of the number of utterances that have been processed so far. Each data point in this scatter plot

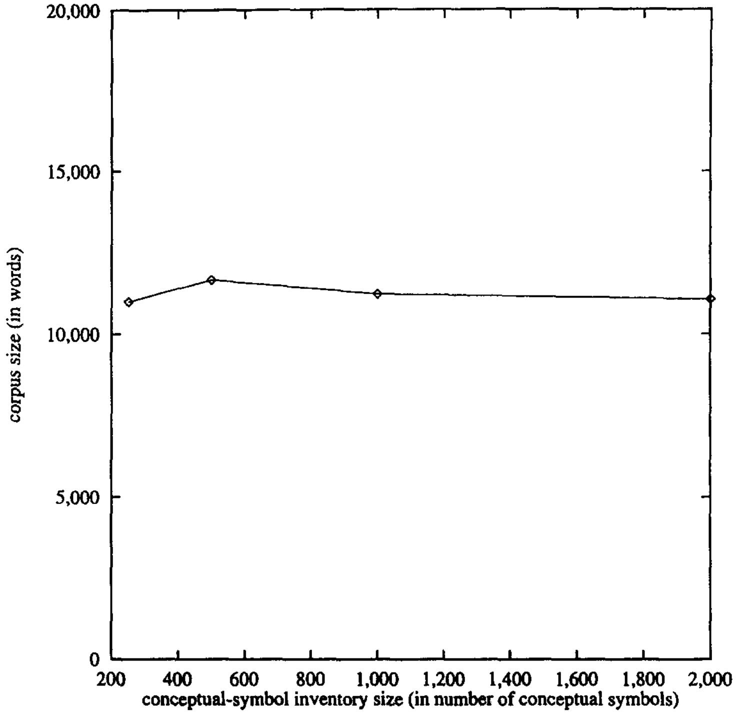

Fig. 5. Corpus size needed for 95% convergence as a function of the conceptual-symbol inventory size.

depicts the number of occurrences of a single word that are needed for convergence as a function of the number of utterances that have already been heard when the first occurrence of that word is heard. As expected, the average number of occurrences needed for convergence on a new word decreases with corpus exposure. In fact, after about 4000 utterances, most words are acquired after being exposed to only one or two occurrences. This concords with the observation made by Carey (1978) that older children learn at least part of the meaning of many words from a single exposure.

8.5. Stressing all parameters simultaneously

Each of the above simulations independently stresses a single corpus-construction parameter. One additional simulation was performed to simultaneously stress the three sensitive parameters, namely vocabulary size, noise rate, and homonymy rate. For this simulation, the baseline parameter values were used for the degree of referential uncertainty and the conceptual-symbol inventory size, while the vocabulary size was set to 10,000, the noise rate was set to 5%, and the

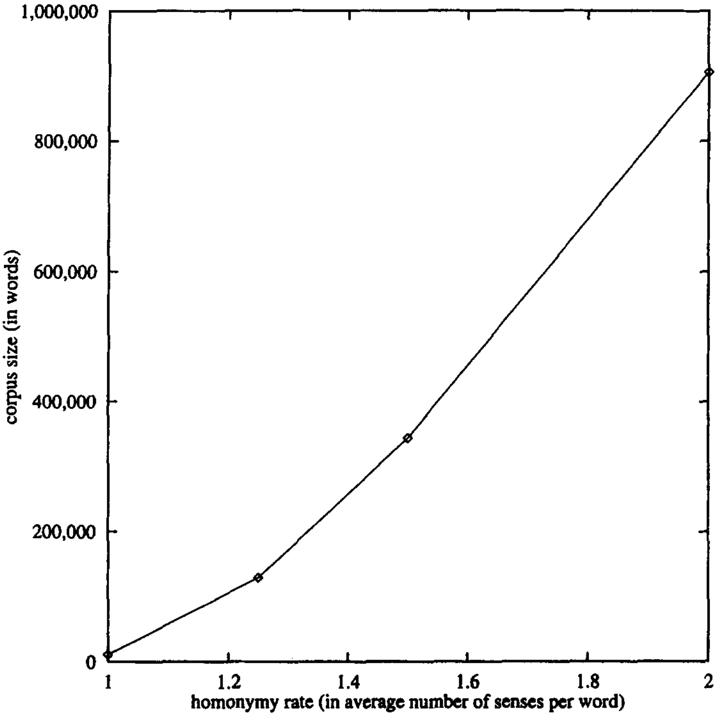

Fig. 6. Corpus size needed for 95% convergence as a function of the homonymy rate.

homonymy rate was set to 1.68. For these corpus-construction parameters, after processing 1,440,945 utterances, the algorithm had correctly converged on 13,560 (80.7%) of the 16,800 senses, producing only 2052 (12.2%) false positives and leaving only 3240 (19.2%) false negatives.[10] Computer resource limitations precluded running this simulation to 95% convergence.

No claim is intended that these simulations reflect all of the complexities that children face when learning their native language. First of all, it is unclear how to select appropriate values for some of the corpus-construction parameters such as noise rate, homonymy rate, and degree of referential uncertainty. In the final simulation, the noise rate of 5% and the value of 10 for the degree of referential uncertainty were chosen arbitrarily, purely to test the acquisition algorithm. Our current impoverished level of understanding of how conceptual representations are constructed from perceptual input, either by adults or by infants, makes it difficult to select a more motivated noise rate or degree of referential uncertainty. It is also difficult to accurately assess the homonymy rate in a given language, as that

[10] This simulation differed from all of the other simulations reported in this paper in that the lexicon was constrained to contain at most one non-frozen sense for each word at a given time. This modified strategy was used for this large simulation as it appears to converge more quickly. Computer resource limitations precluded rerunning all of the remaining simulations with this new strategy.

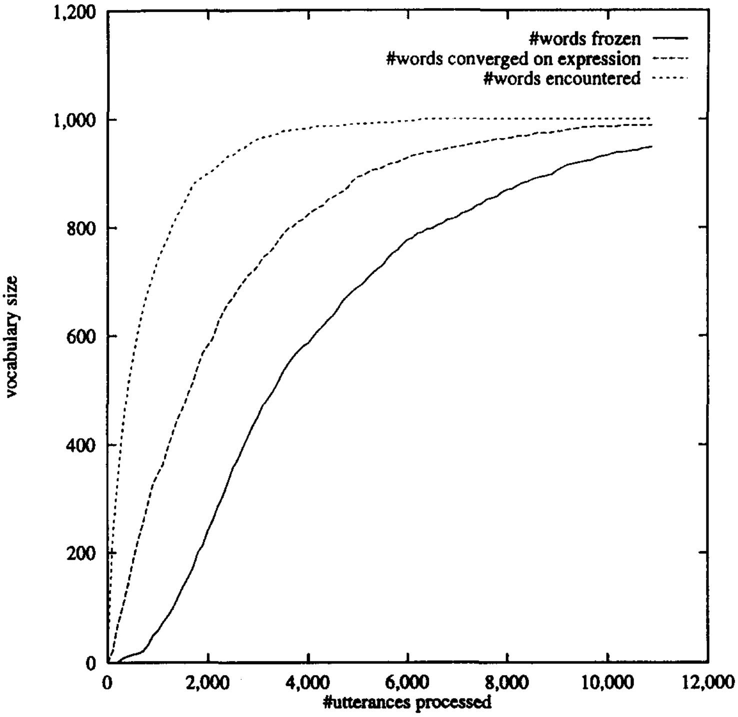

Fig. 7. Vocabulary growth as a function of corpus exposure for the baseline corpus-construction parameters.

depends on how one decides when two senses differ. The homonymy rate of 1.68 senses per word was chosen for the final simulation since the WordNet database (Beckwith et al., 1991) exhibits a homonymy rate of 1.68.

9. General discussion

9.1. Cross-situational learning

The intuitive notion of cross-situational learning has been around for a long time. Many authors (e.g., Pinker, 1989; Fisher et al., 1994) either implicitly or explicitly propose a learning strategy of finding word meanings that are consistent across multiple situations. Roughly speaking, this strategy can be stated as follows: Find a set of possible meanings in each situation and intersect those sets across all situations in which a word occurs to determine the meaning for that word. But the intuitive notion is underspecified. Not only does it ignore problems of referential uncertainty, noise, and homonymy, it leaves a more basic question unanswered: What are the entities to be intersected? As this paper shows, the cross-situational

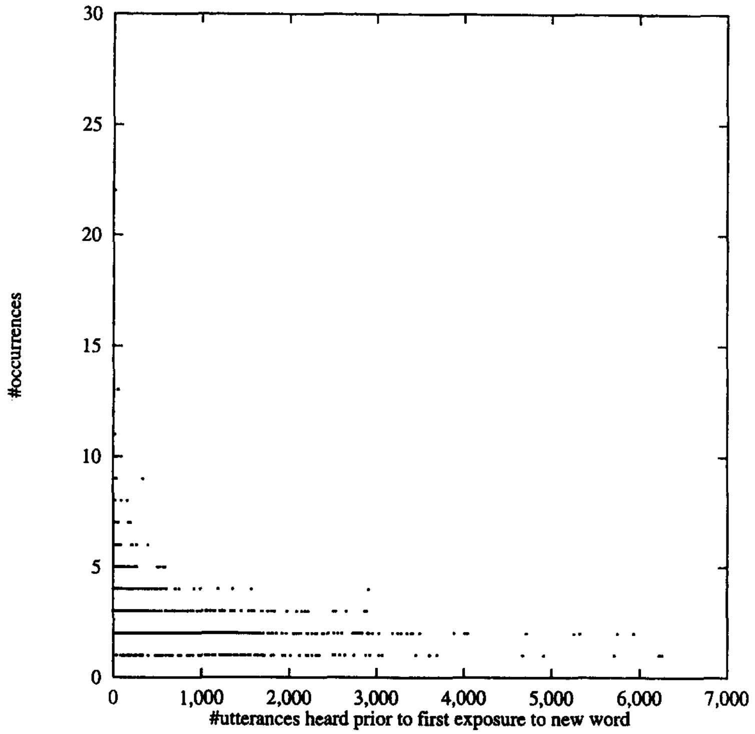

Fig. 8. Number of occurrences needed for convergence on conceptual expression as a function of corpus exposure.

strategy can be applied to at least two different kinds of entities: conceptual-symbol sets, as is done by Rule 2, and conceptual-expression sets, as is done by Rule 5. Both of these forms of inference can be useful for a learner. But the comprehensive learning strategy described in this paper includes more than these two rules. Rules 3, 4, and 6 allow an additional form of inference. Using principles of exclusivity, partial knowledge about the meanings of some words in an utterance can be used to infer knowledge about the meanings of other words in that utterance. The strategy presented here, like that of Tishby and Gorin (1994), incorporates such inference mechanisms.

9.2. Cognitive plausibility

In order for the algorithm presented here to be a plausible model of lexical acquisition in children, one must show that the information needed by the algorithm is available to children and the information inferred by children is produced by the algorithm. Much of the information, however, is in the form of mental representations, about which we currently can say very little. Thus, we cannot reasonably compare the lexicon attained by children with the lexicon produced by the algorithm presented here, nor can we realistically assess whether the assumptions made about the nature of conceptual structure are plausible. Nonetheless, the linguistic data used for lexical acquisition are, in fact, observable. One can ask whether the quantity and length of utterances used by the algorithm to learn a given-sized lexicon are compatible with what is known about the linguistic input to children. In order for the algorithm to be a plausible model of lexical acquisition in children, it must operate on utterances that are no simpler than those available to children and must require the same or fewer utterances for convergence.

Numerous researchers have measured the mean utterance length (MLU) of speech to children and the quantity of speech heard by children per unit time. (See, for example, Schachter et al., 1976, Snow, 1977, Kaye, 1980, Moerk, 1983, and Wells, 1986.) Furthermore, Bernstein-Ratner (personal communication) provided me with an unpublished analysis of her corpus in the CHILDES database (MacWhinney and Snow, 1985). These studies measured mostly short sample periods. Table 1 normalizes these results to utterances per hour.

Children attain basic fluency by age three Carey (1978) reports that children learn approximately 10 words per day during that period. This corresponds to a lexicon of approximately 10,000 words, acquired in approximately 1000 days. If we conservatively take a child's waking day to consist of 10 hours, the data in Table 1 imply that children learn 10,000 words by hearing between 800,000 and 12,600,000 utterances, assuming that throughout this period children hear utterances at this average rate. Thus,the input needs of the algorithm presented here, as illustrated by Figs. 2–6, and the final large simulation, appear to be within the data available to children when acquiring a similar-sized lexicon. Furthermore, the MLU and range of utterance lengths on which the algorithm has been tested indicate that these utterances are of approximately the same complexity as those heard by children.

9.3. Worst-case assumptions

Our current level of understanding of the internal workings of the various faculties is very limited. We have only weak hypotheses as to what form

Table 1
Various measurements of the quantity of speech to children normalized to utterances per hour

Source		Minimum	Average	Maximum
Schachter et al. (1976)	Toddlers		239	
	3-year-olds		245	
	4-year-olds		219	
Snow (1977)		504		1197
Kaye (1980)	Infants		1260	
	2-year-olds		870	
Moerk (1983)			283	
Wells (1986)		80		800
Bernstein-Ratner (pers. comm.)			1089	

conceptual structure takes, what the conceptual-symbol inventory is, what the size of that inventory is, what the size and shape of conceptual expressions might be, and what mechanisms are used to compose word-level conceptual representations to form utterance-level conceptual representations. Similarly, we have no way to estimate the degree of referential uncertainty, the noise rate, or the homonymy rate faced by children, as these are properties of internal representations to which we have no access at the present time. Furthermore, we have only rudimentary evidence of the kinds of information that children use when acquiring language, including, inter alia, the degree to which they employ syntactic constraints, principles of exclusivity, part/whole distinctions, and distinctions between basic-level and sub- or super-ordinate categories to determine word-to-meaning mappings. This paper suggests that acquisition of word-to-meaning mappings might be possible despite weak, worst-case assumptions along several fronts:

- Acquisition of word-to-meaning mappings might be possible with a modular language faculty that is separated into speech perception, perceptual/conceptual, and lexical-acquisition components, with information flowing unidirectionally from the former two to the latter, without any feedback in the learning process.
- Acquisition of word-to-meaning mappings might be possible without using the phonological content of word symbols or the semantic content of conceptual symbols, solely by a process of learning the mapping between two internal mental representations, without reference to properties of those representations other than co-occurrence.
- Acquisition of word-to-meaning mappings might be possible without using knowledge of syntax, word order, or well-formedness constraints on conceptual expressions.
- Acquisition of word-to-meaning mappings might be possible without requiring high co-occurrence correlation between words and their contingencies of use.
- Acquisition of word-to-meaning mappings might be possible without using any information about the semantic-interpretation rule except for two relatively weak properties of compositionality.

9.4. Limitations

The abstract model of language acquisition adopted in this paper makes a number of assumptions that might not be worst case. First, the model assumes that, with sufficient regularity, children can include the correct utterance meaning among the set of referentially uncertain meaning hypotheses. This might not be possible for abstract terms. In the model presented here, utterances that contain such terms are likely to be more noisy than utterances that do not. Since the simulations presented assume a uniform distribution of noise, and do not model differential noise rates based on the classes of words contained in an utterance, they might make optimistic assumptions about the noise rate for utterances that contain abstract terms. Second, the model assumes that homonymy can be modeled by pairing words with small sets of distinct unrelated semantic representations. The acquisition algorithm learns each distinct word sense independently. Human languages, however, exhibit polysemy in addition to homonymy. In other words, words can have multiple distinct senses where each sense is, in turn, a cluster of multiple related subsenses. Treating such polysemy as homonymy would suffer from two problems: (a) the homonymy rate would be too large for the current algorithm to handle and (b) the current algorithm would not use knowledge of one subsense to guide the acquisition of a related subsense. Third, while the model can learn in the presence of idioms and metaphorical meaning, treating such utterances as noise, it cannot learn the meanings of idioms and metaphors themselves. Since a large portion of language use is idiomatic or metaphorical (Lakoff and Johnson, 1980; Lakoff, 1987), some method must be devised for learning the meanings of such expressions. Fourth, the model assumes a simple linking rule that treats compositional semantics purely as argument substitution. The inference rules used by the algorithm, in their strict form, are sound only with this linking rule. Adopting a more complex and realistic linking rule would require a reformulation of the inference rules. Finally, the inference rules assume a strict correspondence between the semantic content of an utterance and the meaning hypothesized for that utterance. Sentential utterances must be paired with sentential meaning hypotheses while phrasal utterance fragments must be paired with fragmentary meaning hypotheses. Since much language use is fragmentary, some method must be devised for associating a fragmentary utterance with only a portion of a hypothesized semantic representation. Because of these limitations, the precise algorithm as presented in this paper cannot be a full account of lexical acquisition in children. The hope is, however, that it can be the basis for additional research to find ways of addressing these limitations.

9.5. Relation to syntactic and semantic bootstrapping

When learning their native language, children must acquire all of the knowledge that is specific to that language. This includes, inter alia, components of both its syntax and its lexicon. It is conceivable that children use information about one to help acquire the other. This raises a central question: *Does the acquisition strategy used by children rely on a particular ordering and flow of information?* There is a range of possibilities with three distinct extremes:

- The process of syntactic acquisition relies on prior lexical knowledge obtained without the use of syntactic knowledge.
- The process of lexical acquisition relies on prior syntactic knowledge obtained without the use of lexical knowledge.
- The processes of syntactic and lexical acquisition are interleaved, each using partial information provided by the other.

The first alternative has been proposed by Grimshaw (1979, 1981) and Pinker (1984, 1989), among others, and has become known as the "semantic bootstrapping hypothesis." The second alternative has been proposed by Gleitman (1990) and Fisher et al. (1994), among others, and has become known as the "syntactic bootstrapping hypothesis." In prior work (Siskind, 1990, 1991, 1992), I discussed how the third alternative, interleaving the processes of syntactic and lexical acquisition, can allow a learner to acquire language faster than either the semantic or syntactic bootstrapping approaches. Using an interleaved strategy, the learner can acquire a given amount of information with less input data than a more sequential strategy would require. The particular interleaved strategies explored in this prior work, however, were computationally intensive. They could not scale to process the amount of input available to children and produce a lexicon of the size learned by children. Thus, in more recent work (Siskind, 1993a,b), I have adopted a sequential approach that is more aligned with the semantic bootstrapping hypothesis. This is the approach taken in this paper. To date, only this sequential approach has been shown to scale to larger problems. More research needs to be done to determine whether the other alternatives can be made to scale as well.

10. Conclusion

In this paper, I have presented a precise, implemented algorithm for solving an approximation of the lexical-acquisition task faced by children. Unlike prior theories of lexical acquisition, the fact that this theory has a precise formulation allows it to be tested and its efficacy to be measured. The algorithm makes reasonable assumptions about the length and quantity of utterances needed to successfully acquire a lexicon of word-to-meaning mappings. Furthermore, it addresses five central problems in lexical acquisition that were previously considered difficult: (a) learning from multi-word input, (b) disambiguating referential uncertainty, (c) bootstrapping without prior knowledge that is specific to the language being learned, (d) noisy input, and (e) homonymy.

Until we can gain a better understanding of the size and contents of the conceptual-symbol inventory, the size and shape of conceptual expressions, the semantic-interpretation rule used to compose word meanings to form utterance meanings, and the perceptual/conceptual processes used to hypothesize utterance meanings from observational input, it will not be possible to get realistic estimates of the remaining parameters of the input to lexical acquisition, namely, the degree of referential uncertainty, the noise rate, and the homonymy rate. Thus, serious understanding of conceptual representation, and how it is grounded in perception, lies on the critical path to understanding language acquisition. This realization has motivated my own research (Siskind, 1992, 1995), as well as that of others such as Feldman et al. (1990), Suppes et al. (1991), and Torrance (1994), to study language acquisition computationally in the context of perception and action, and to focus on the requisite conceptual representations involved. Such a holistic computational approach should lead to a better understanding of the language acquisition process.

Acknowledgments

This research was supported, in part, by an AT&T Bell Laboratories Ph.D. scholarship to the author, by a Presidential Young Investigator Award to Professor Robert C. Berwick under National Science Foundation Grant DCR-85552543, by a grant from the Siemens Corporation, and by the Kapor Family Foundation. This research was also supported, in part, by ARO grant DAAL 03-89-C-0031, by DARPA grant N00014-90-J-1863, by NSF grant IRI 90-16592, by Ben Franklin grant 91S.3078C-1, and by the Natural Sciences and Engineering Research Council of Canada. Part of this research was performed while the author was a graduate student at the Massachusetts Institute of Technology Artificial Intelligence Laboratory, while the author was visiting the Xerox Palo Alto Research Center as a research intern and as a consultant, while the author was a postdoctoral fellow at the University of Pennsylvania Institute for Research in Cognitive Science, and while the author was at the University of Toronto Department of Computer Science. I wish to thank Michael Brent, Graeme Hirst, Gary Marcus, Mark Steedman, and the three anonymous referees for giving me numerous helpful comments on an earlier draft of this paper, Carl de Marcken for providing me with a cleaned-up version of the Nina corpus from CHILDES, and Neil Bohannon, Robin Chapman, Annick DeHouwer, Antje Meyer, Ernst Moerk, Barbara Pearson, Nan Bernstein-Ratner, Catherine Snow, Gordon Wells, and Grover Whitehurst for sending me informative responses to a query I posted to info-childes pertaining to an estimate of total daily linguistic input to children. This paper is an extension of work previously reported in the *Proceedings of the Twelfth National Conference on Artificial Intelligence.*

Appendix A

In the following description, S denotes a set of sense symbols, one for each word symbol in an utterance, M denotes the set of conceptual expressions that represent hypothesized meanings of that utterance, $F(m)$ denotes the set of all conceptual symbols that appear in the conceptual expression m, and $F_1(m)$ denotes the set of all conceptual symbols that appear only once in m. Both $F(\bot)$ and $F_1(\bot)$ yield the empty set.

Rule 1

$$M \leftarrow \{m \in M | \bigcup_{s \in S} N(s) \subseteq F(m) \wedge F(m) \subseteq \bigcup_{s \in S} P(s)\}$$

Rule 2

for $s \in S$ do $P(s) \leftarrow P(s) \cap \bigcup_{m \in M} F(m)$**do**

Rule 3

```
for $s \in S$ do $N(s) \leftarrow N(s) \cup \left[\left(\bigcap_{m \in M} F(m)\right) \setminus \bigcup_{s' \in S, s' \neq s} P(s')\right]$ do
```

Rule 4

```
for $s \in S$ do $P(s) \leftarrow P(s) \setminus \left[\left(\bigcap_{m \in M} F_1(m)\right) \cap \bigcup_{s' \in S, s' \neq s} N(s')\right]$ do
```

Rule 5

```
for $s \in S$
do if $N(s) = P(s)$
then $D(s) \leftarrow D(s) \cap \bigcup_{m \in M}$ Reconstruct$(m, N(s))$ fi od
```

Rule 6

```
if $(\forall s \in S) N(s) = P(s)$
then for $s \in S$
  do if $(\forall s' \in S)[s' \neq s \rightarrow D(s') \neq \bot]$
  then $D(s) \leftarrow \{t \in D(s) |$
    $\underbrace{(\exists t_1 \in D(s_1)) \cdots (\exists t_n \in D(s_n))}_{\{s, s_1, \ldots, s_n\} = S}$
    $(\exists m \in M) m \in$ Compose$(\{t, t_1, \ldots, t_n\})\}$ fi od fi
```

Procedure for incrementing the confidence factors

If all of the sense symbols that have been selected for all of the word symbols in the utterance have converged on a conceptual expression, then increment the confidence factor of those sense symbols that mean $\bot$, if all of the sense symbols that do not mean $\bot$ are frozen, and likewise increment the confidence factor of those sense symbols that do not mean $\bot$, if all of the sense symbols that do mean $\bot$ are frozen.

```
if $(\forall s \in S)\ |D(s)| = 1$
  then for $s \in S$
  do if $[D(s) = \{\bot\} \wedge (\forall s \in S)(D(s) \neq \{\bot\} \rightarrow C(s) \geq \mu)] \vee$
        $[D(s) \neq \{\bot\} \wedge (\forall s \in S)(D(s) = \{\bot\} \rightarrow C(s) \geq \mu)]$
    then $C(s) \leftarrow C(s) + 1$ fi od fi
```

The integer constant μ denotes a *freezing point.* A sense symbol s is frozen if it has converged on a conceptual expression and $C(s) \geq \mu$. Discard senses, unless they are frozen, after processing every k utterances. For the simulations in this paper, $\mu = 2$ and $k = 500$.

References

Aslin, R.N., Woodward, J.C., LaMendola, N.P., & Bever, T.G. (1995). Models of word segmentation in fluent maternal speech to infants. In J. Morgan & K. Demuth (Eds.), *Signal to syntax: Bootstrapping from speech to grammar in early acquisition.* Hillsdale, NJ: Erlbaum.

Badler, N.I. (1975). *Temporal scene analysis: Conceptual descriptions of object movements* (Tech. Rep. 80). University of Toronto Department of Computer Science.

Beckwith, R., Fellbaum, C., Gross, D., & Miller, G. (1991). WordNet: A lexical database organized on psycholinguistic principles. In U. Zernik (Ed.), *Lexical acquisition: Exploiting on-line resources to build a lexicon* (pp. 211–232). Hillsdale, NJ: Erlbaum.

Beckwith, R., Tinkler, E., & Bloom, L. (1989). *The acquisition of non-basic sentences.* Paper presented at the Boston University Conference on Language Development.

Berwick, R.C. (1983). Learning word meanings from examples. *Proceedings of the Eighth International Joint Conference on Artificial Intelligence* (pp. 459–461). Karlsruhe.

Borchardt, G.C. (1985). Event calculus. *Proceedings of the Ninth International Joint Conference on Artificial Intelligence* (pp. 524–527). Los Angeles, CA.

Brent, M.R., & Cartwright, T.A. (1996). Distributional regularity and phonotactic constraints are useful for segmentation. *Cognition, 61,* 93–124.

Bruner, J. (1983). *Child's talk.* New York, NY: Norton.

Carey, S. (1978). The child as word learner. In M. Halle, J. Bresnan, & G.A. Miller (Eds.), *Linguistic theory and psychological reality.* Cambridge, MA: MIT Press.

Christophe, A., Dupoux, E., Bertoncini, J., & Mehler, J. (1994). Do infants perceive word boundaries? An empirical study of the bootstrapping of lexical acquisition. *Journal of the Acoustical Society of America, 95*(3), 1570–1580.

Clark, E.V. (1987). The principle of contrast: A constraint on language acquisition. In B. MacWhinney (Ed.), *Mechanisms of language acquisition* (pp. 264–293). Hillsdale, NJ: Erlbaum.

Cutler, A., & Carter, D.M. (1987). The predominance of strong initial syllables in the English vocabulary. *Computer Speech and Language, 2,* 133–142.

Cutler, A., Mehler, J., Norris, D., & Segui, J. (1983). A language-specific comprehension strategy. *Nature, 304,* 159–160.

Feldman, J.A., Lakoff, G., Stolcke, A., & Weber, S.H. (1990). Miniature language acquisition: A touchstone for cognitive science. *Proceedings of the 12th Annual Conference of the Cognitive Science Society* (pp. 686–693). Massachusetts Institute of Technology, Cambridge, MA.

Fisher, C., Hall, G., Rakowitz, S., & Gleitman, L. (1994). When it is better to receive than to give: Syntactic and conceptual constraints on vocabulary growth. *Lingua, 92*(1), 333–375.

Fodor, J.A. (1970). Three reasons for not deriving kill from cause to die. *Linguistic Inquiry, 1,* 429–438.

Fodor, J.A. (1975). *The language of thought.* Hove, UK: Harvester Press.

Girard, J.-Y. (1987). Linear logic. *Theoretical Computer Science, 50,* 1–102.

Gleitman, L. (1990). The structural sources of verb meanings. *Language Acquisition, 1*(1), 3–55.

Granger, Jr., R.H. (1977). FOUL-UP: A program that figures out meanings of words from context. *Proceedings of the Fifth International Joint Conference on Artificial Intelligence* (pp. 172–178). Cambridge, MA.

Grimshaw, J. (1979). Complement selection and the lexicon. *Linguistic Inquiry, 10,* 279–326.

Grimshaw, J. (1981). Form, function, and the language acquisition device. In C.L. Baker & J.J. McCarthy (Eds.), *The logical problem of language acquisition.* Cambridge, MA: MIT Press.

Hays, E.M. (1989). On defining motion verbs and spatial prepositions. In C. Freksa & C. Habel (Eds.), *Repräsentation und Verarbeitung räumlichen Wissens* (pp. 192–206). Berlin: Springer-Verlag.

Horton, M., & Markman, E.M. (1980). Developmental differences in the acquisition of basic and superordinate categories. *Child Development, 51,* 708–719.

Jackendoff, R. (1983). *Semantics and cognition.* Cambridge, MA: MIT Press.

Jackendoff, R. (1990). *Semantic structures.* Cambridge, MA: MIT Press.

Jacobs, P., & Zernik, U. (1988). Acquiring lexical knowledge from text: A case study. *Proceedings of the Seventh National Conference on Artificial Intelligence,* St Paul, MN (pp. 739–744).

Jusczyk, P.W., Cutler, A., & Redanz, N.J. (1993). Infants' preference for the predominant stress patterns of English words. *Child Development, 64*, 675–687.

Kaye, K. (1980). Why we don't talk "baby talk" to babies. *Journal of Child Language, 7*(3), 489–507.

Kelly, M.H. (1992). Using sound to solve syntactic problems: The role of phonology in grammatical category assignment. *Psychological Review, 99*(2), 349–364.

Lakoff, G. (1987). *Women, fire, and dangerous things.* Chicago, IL: University of Chicago Press.

Lakoff, G., & Johnson, M. (1980). *Metaphors we live by.* Chicago, IL: University of Chicago Press.

Leech, G.N. (1969). *Towards a semantic description of English.* Bloomington, IN: Indiana University Press.

Locke, J. (1690). *An essay concerning human understanding.* Republished by Oxford: Clarendon (1975).

Mackworth, A.K. (1992). Constraint Satisfaction. In S.C. Shapiro (Ed.), *Encyclopedia of artificial intelligence* (2nd ed., pp. 285–293). New York, NY: Wiley.

MacWhinney, B., & Snow, C. (1985). The child language data exchange system. *Journal of Child Language, 12*, 271–296.

Markman, E.M. (1989). *Categorization and naming in children: Problems of induction.* Cambridge, MA: MIT Press.

Miller, G.A. (1972). English verbs of motion: A case study in semantics and lexical memory. In A.W. Melton & E. Martin (Eds.), *Coding processes in human memory* (Ch. 14, pp. 335–372). Washington, DC: Winston.

Moerk, E.L. (1983). *The mother of Eve – As a first language teacher.* Norwood, NJ: Ablex.

Norris, D., & Cutler, A. (1985). Juncture detection. *Linguistics, 23*, 689–705.

Okada, N. (1979). SUPP: Understanding moving picture patterns based on linguistic knowledge. *Proceedings of the Sixth International Joint Conference on Artificial Intelligence* (pp. 690–692). Tokyo.

Pinker, S. (1984). *Language learnability and language development.* Cambridge, MA: Harvard University Press.

Pinker, S. (1989). *Learnability and cognition.* Cambridge, MA: MIT Press.

Quine, W.V.O. (1960). *Word and object.* Cambridge, MA: MIT Press.

Regier, T.P. (1992). *The acquisition of lexical semantics for spatial terms: A connectionist model of perceptual categorization.* Ph.D. thesis, University of California at Berkeley.

Robinson, J.A. (1965). A machine-oriented logic based on the resolution principle. *Journal of the Association for Computing Machinery, 12*(1), 23–41.

Schachter, F.F., Fosha, D., Stemp, S., Brotman, N., & Ganger, S. (1976). Everyday caretaker talk to toddlers vs. threes and fours. *Journal of Child Language, 3*(2), 221–245.

Schank, R.C. (1973). *The fourteen primitive actions and their inferences.* Memo AIM-183, Stanford Artificial Intelligence Laboratory.

Siskind, J.M. (1990). Acquiring core meanings of words, represented as Jackendoff-style conceptual structures, from correlated streams of linguistic and non-linguistic input. *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics* (pp. 143–156). Pittsburgh, PA.

Siskind, J.M. (1991). Dispelling myths about language bootstrapping. *AAAI Spring Symposium Workshop on Machine Learning of Natural Language and Ontology.* Paulo Alto, CA (pp. 157–164).

Siskind, J.M. (1992). *Naive physics, event perception, lexical semantics, and language acquisition.* Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA.

Siskind, J.M. (1993a). *Lexical acquisition as constraint satisfaction* (Tech. Rep. IRCS-93-41). University of Pennsylvania Institute for Research in Cognitive Science.

Siskind, J.M. (1993b). Solving a lexical acquisition task via an encoding as a propositional satisfiability problem. *Proceedings of the 6th Annual CUNY Sentence Processing Conference.* Amherst, MA (p. 82).

Siskind, J.M. (1995). Grounding language in perception. *Artificial Intelligence Review, 8*, 371–391.

Snow, C.E. (1977). The development of conversation between mothers and babies. *Journal of Child Language, 4*(1), 1–22.

Suppes, P. (1974). The semantics of children's language. *American Psychologist, 29*, 102–114.

Suppes, P., Liang, L., & Böttner, M. (1991). Complexity issues in robotic machine learning of natural language. In L. Lam & V. Naroditsky (Eds.), *Modeling complex phenomena*. Berlin: Springer-Verlag.

Tishby, N., & Gorin, A. (1994). Algebraic learning of statistical association for language acquisition. *Computer Speech and Language*, *8*(1), 51–78.

Torrance, M.C. (1994). *Natural communication with mobile robots*. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA.

Wells, C.G. (1986). *The meaning makers: Children learning language and using language to learn*. Portsmouth, NH: Heinemann.

Zipf, G. (1949). *Human behavior and the principle of least effort*. New York, NY: Addison-Wesley.