SEMANTIC BASED LEARNING OF SYNTAX IN AN AUTONOMOUS ROBOT

MATTHEW MCCLAIN
Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
mrmcclai@uiuc.edu

STEPHEN LEVINSON
Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
sel@ifp.uiuc.edu

It is the goal of the Language Acquisition Group at the University of Illinois at Urbana-Champaign (LAR-UIUC) to build a robot that is able to learn language as well as humans through embodied sensori-motor interaction with the physical world. This paper proposes cognitive structures to enable an autonomous robot to learn the syntax of two-word sentences using its understanding of lexical semantics. A production rule of syntax in Chomsky Normal Form will be explicitly represented using a hidden Markov Model. Results of robotic experiments show that these models can learn representations of syntax in this form and that they can be used to produce novel sentences.

Keywords: Machine language acquisition, robotics, syntax, semantics

1. Introduction

For the last 50 years, the search for strong AI (artificial intelligence) has been approached with the idea that computers can use language as well as humans without having any understanding of its meaning. This is the central reason why the pursuit has been unsuccessful. The Language Acquisition and Robotics Group at the University of Illinois at Urbana-Champaign (LAR-UIUC) is attempting to construct robots that are able to use language as well as humans by learning the semantics of language through embodied interaction with the world. We posit that the memory should be associative, as its primary function is to correlate the robot's various sensori-motor inputs into a meaningful model of the world. Reinforcement learning, both supervised and unsupervised, is required to train the associative memory. As well, the robot's sensori-motor capabilities should be complex so that robust models of the world can be developed using them.

LAR-UIUC is using an embodied robotic framework to study intelligence and language from a bottom-up perspective. The current work is another step toward our goal. In this sense, it builds upon a great deal of work done by past members and should easily lend itself to work by future members.

The title of this paper is meant to reflect our belief that semantics plays a central role in language acquisition. We reject the idea that language acquisition is a result of an

innate language faculty in humans. The work in this paper will show that information about syntax can be inferred using semantics, and does not require innate knowledge.

The next section presents related work from various fields. The third section presents the robotic framework developed by LAR-UIUC. The problem that this paper seeks to address is presented in the fourth section, and the fifth section describes the proposed solution. Details of the robotic experiments and the results are presented in the sixth section. The seventh section presents some ideas for continuing this work, and the eighth section contains some concluding remarks.

2. Related Work

2.1. *Syntax acquisition and representation*

The acquisition of syntax by machines in an embodied framework has been studied by Sugita and Tani, and Roy. Sugita and Tani use a pair of recursive neural networks (RNNs) to learn the compositional semantics of two-word command sentences.[1] However, the model used can only understand sentences with words that have been previously used in the same syntactic construct. Thus, the model's representation of lexical categories (also called parts-of-speech, for example "noun" and "verb") is limited to the position of words in a specific syntactic construct. Roy has implemented algorithms that learn adjective-noun phrases using visual information in both simulated visual scenes and a robotic implementation.[2,3] Similar to Sugita and Tani, the model's representation of part of speech is limited to co-occurrence in a syntactic construct. The model learns the visual features that are associated with each word, but this is not used to define the lexical categories (referred to by Roy as "word classes").

Brown has proposed five stages of children's sentence production from the study of the development of children's speech.[4] Brown designates these stages both by the mean length of utterance (MLU, measured in morphemes) and on the syntactic complexity that the children's utterances display. Bloom's study of children's sentence production in speech goes beyond the formal structure of children's sentences and takes into account the context in which the child utters the sentence.[5]

Chang, Dell, and Bock have developed a dual-path connectionist model that is able to learn rules of syntax and lexical categories, and use this information to produce syntactically correct sentences.[6]

Kamp and Reyle give examples of how predicate argument structures have been used to describe the compositional semantics of syntactic constructions.[7] While this provides a representation of the compositional semantics, the lexical semantics (the meanings of the individual words) is ignored.

Formal grammars have been developed by Chomsky to provide a mathematical representation of the syntax of natural language.[8] This representation contains four elements: V_T, a set of terminal symbols (words, in natural language); V_N, a set of non-terminal symbols (lexical or phrase categories, such as N for noun or NP for noun phrase); S, a special non-terminal symbol that represents a well-formed sentence of the grammar; and R, a set of production rules that dictate how the symbols of the grammar replace each other. The Chomsky Hierarchy defines different classes of grammars based on the forms of the production rules.[9] Of relevance to this paper are context-free

grammars, in which the production rules can be written in Chomsky Normal Form (CNF). In CNF, the production rules can be one of two types: X→x or X→YZ, where the lower-case letters are terminal symbols and upper case letters are non-terminals. The first type of rule gives lexical categorization and the second type gives sentence production rules. While context-free grammars cannot represent all of the complexity of natural language, they are able to capture a significant amount of the structure of natural language with relatively simple production rules.

2.2. *Hidden Markov models*

Hidden Markov models (HMMs) have been shown to be able to learn aspects of the structure of language without prior knowledge. The Cave and Neuwirth experiment gives a significant example of this.[10] An HMM is a stochastic model in which the underlying structure is composed of unobservable discrete states which generate observable outputs, and the next-state behavior is Markovian (the next-state is only dependent on the previous state). When an HMM is trained with a sequence of observations, the parameters converge to a local maximum in the model's parameter space, which represents the most likely model to have generated the observation sequence.

2.3 *Developmental robotics*

New approaches to artificial intelligence are being explored that deal with the issue of embodiment. The new field has been called "Developmental Robotics" or "Epigenetic Robots", and is characterized by using robots embodied with sensori-motor capabilities, to discover for themselves the structure in their environment and their own embodiment instead of having most of their functionality pre-programmed.

Rodney Brooks was one of the first researchers to articulate the failures of traditional AI as a paradigmatic failure.[11] He argues that the foundations of human intelligence are in how we evolved to survive in the world, and so attempting to create intelligent machines that do not interact with the world is doomed to failure.

Weng of Michigan State University focuses on what he calls "autonomous mental development.[12]" This approach is meant to parallel human mental development in that the robots learn by interacting with their environment, without task-specific goals. Weng has implemented his ideas in the robots SAIL-1, SAIL-2, and Dav, and has been able to demonstrate the ability to learn complex tasks such as obstacle avoidance in autonomous navigation (with Zeng) and auditory development (with Zhang).[13,14]

Sporns and Alexander of the University of Indiana have conducted research in neurophysiological modelling using a system of rewards to modulate how their robots interact with their environment.[15,16]

3. Robotic Framework

3.1. *Embodied cognitive framework*

The embodied cognitive framework in Figure 1 is used by LAR-UIUC to guide our implementation. Central to this cycle are the associative and working memories, which comprise the noetic system. This is where the robots learn their models of the world based upon their sensori-motor experience, and make decisions based on these models.

The cognitive cycle is completed by the outside world, in which the robot is able to perceive the effects of its own actions. As well, proprioceptive feedback is available to help make robust measurements of the outside world.

3.2 *Robotic implementation*

This section presents details of the robotic implementation used by LAR-UIUC.

3.2.1 *Hardware*

Humans are our only examples of natural language users, and so we would like the physical implementation of our robots to be as anthropomorphic as possible. Our robots'

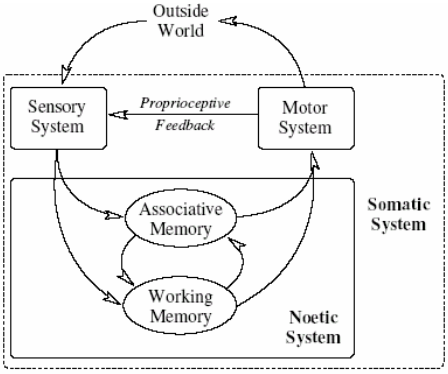


Figure 1: Embodied Cognitive Framework

motor abilities must allow them to move about their environment and interact with the environment through articulated movement (for example, grasping at objects and outputting speech). The sensory system must provide the robot with enough information to build useful models of its environment. We have chosen to focus on visual and audio inputs, as these are the main sources of sensory information in humans and (possibly not coincidentally) have the most readily available physical implementations as cameras and microphones. As well, our robots have some touch sense, implemented by various sensors. A picture of LAR-UIUC's robots is shown in Figure 2.

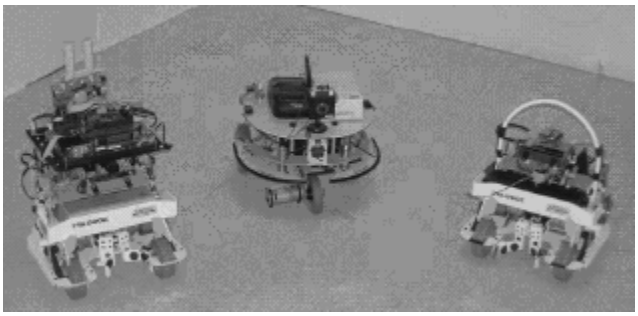


Figure 2: LAR-UIUC's Robots: Illy, Alan, and Norbert

As the base of our robotic implementation we have chosen Arrick Robotic's Trilobot. This provides our robots with the ability to move about their environment, a front arm and gripper for grasping objects, and a head with pan and tilt ranges of motion. As well, sensors around the perimeter of the robot and in the gripper provide some basic tactile sense. To this platform we have added cameras for stereo vision and microphones for binaural hearing. An on-board computer has been added to handle some of the computational load of the robots' cognition. The robots have also been equipped with wireless internet to distribute information to desktop computers.

3.2.2 *Software*

Various software programs have been developed for the robots to implement their cognitive abilities. A visual processing program has been implemented by Lin to perform object segmentation and provide shape, color, and position information for each object.[17] The color information is computed as the proportion of the object's pixels that are in a set of color histogram bins. The shape information is the ratio of the object's width to height and a shape moment, which measures the average squared distance of the pixels from the object's center. This visual information is the robots' visual feature set, and is the basis for the robots' visual representation of their world. Kleffner has implemented algorithms to process speech using warped linear prediction.[18] Programs for distributing raw sensory information amongst computers have been developed by Squire.[19] A working memory has been implemented by McClain which provides the robot with the ability to explore its environment.[20]

3.3 *Semantic associative memory*

Squire has implemented an associative long-term memory which allows the robots to learn the semantics of words based on sensory experience.[19] The model used for the semantic associative memory has three components: a visual model, an auditory model, and a concept model.

Each of the components of the associative memory were implemented by Squire using an HMM. In the visual model, each state corresponds to the visual representation of one object as a joint probability distribution over the visual features. As well, the auditory model uses the each state of an HMM to represent one word using a joint probability distribution over the auditory features. The states of the concept model HMM have outputs that are distributions over the states of the visual and auditory models, which can be interpreted as a naming of objects.

4. Semantic Based Learning of Syntax

4.1. *Background*

The associative memory developed by Squire can be viewed as an implementation of a single-word learning stage for our robots.[19] Bloom notes that, "Many investigators have reported that the child's single-word utterances, before the development of syntax, seem to function as one-word sentences.[21]" (p10) While there is a difference between the meanings of single words and one-word sentences in human language, this difference is negligible in our robotic implementation. Then, according to Brown, "At about eighteen

months children are likely to begin constructing two-word utterances; such a one, for instance as Push car…A construction such as Push car is not just two single-word utterances spoken in a certain order.[22]" (p77) Thus, using childhood language development as a guide, the next step for LAR-UIUC is the task of enabling our robots to learn to use two-word sentences.

4.2. *Proposed research*

We propose to enable our robots learn the syntax of noun-verb sentences, using the production rule (in Chomsky Normal Form) S→NV as a basis for representing the syntactic information. Once a production rule has been learned, the robot should be able to use this information in new contexts. The information that is to be learned is the ordering of the lexical categories and the compositional semantics of the production rule.

4.3. *Lexical categories*

We propose that in the initial stages of language development, it is unnecessary for a language user to have knowledge of lexical categories like nouns and verbs. Tomasello notes, "It is important to conceptualize the child's early cognition not solely in terms of objects and properties, as many theories do, but rather in terms of event structures, with objects being no more prominent in the child's conception of the world than the activities and events in which they are embedded.[23]" (p137). Words are more reliably assigned to lexical categories based on how they are used a sentence than by the semantics of the words. This can easily been seen with the word "throw" in the sentences "Throw the ball" and "The throw to home plate was in time," where it is used as both a verb and a noun. Jurafsky and Martin note that "Traditionally the definition of parts-of-speech has been based on morphological and syntactic function…While word classes do have tendencies toward semantic coherence, this is not necessarily the case, and in general we don't use semantic coherence as a definitional criterion for parts-of-speech.[24]" (p289). Thus there appears to be a paradox: knowledge of syntax is required to learn lexical categories and vice-versa. The solution to this paradox is semantic bootstrapping, in which lexical categories are initially learned using semantic regularities of words. This understanding of lexical categories can be used to learn some syntax, which can in turn be used to learn more abstract definitions of lexical categories. This concept of semantic bootstrapping has been developed by Pinker.[25] While Pinker gives a theoretical accounts of semantic bootstrapping in childhood language acquisition, the work in this paper will give it a real-world implementation.

The semantics used by LAR-UIUC's robots are grounded in the robots' sensori-motor experience, which are encoded as perceptual features. Thus, the semantic understanding of lexical categories will be based on these same perceptual inputs. Specifically, a lexical category will be defined by the subset of the robot's perceptual inputs that members of that category describe. The perceptual inputs will be organized into perceptual modalities, which are sets of related perceptual features. For example, all perceptual features that describe aspects of an object's shape will be members of the shape modality.

4.4. *Compositional semantics*

The meaning of a sentence is derived from two sources of information – lexical semantics and compositional semantics. The lexical semantics is the meanings of the individual words. The compositional semantics of a sentence is the structural information from the ordering of the words. For example, the sentence "ball move" does not just mean "there is a ball' and "something is moving." As well, "ball move" means something than "move ball." The compositional semantics of the production rule S→NV that is to be learned is that the noun and the verb describe perceptual information about the same object. In LAR-UIUC's robotic implementation, perceptual information from each object that the robot is observing is associated into a single input feature vector. Thus, the robot's understanding of compositional semantics will be based on whether or not both words in the two-word sentence describe perceptual information from the same perceptual feature vector. It should be noted here that by defining the compositional semantics in this way, learning the compositional semantics becomes somewhat trivial. A more general approach is discussed in Section 7.2.

5. Implementation

The models proposed in this work will learn the syntactic information of noun-verb sentences by hearing example sentences that describe what the robot is observing. It is assumed that the lexical semantics of the words used are previously learned, as they will be used in inferring the syntactic information. The models will assume that only two-word sentences will be used.

5.1. *Cognitive model*

The proposed cognitive model for syntax acquisition can be seen in Figure 3. This framework extends Squire's semantic associative memory described in section 3.3 by

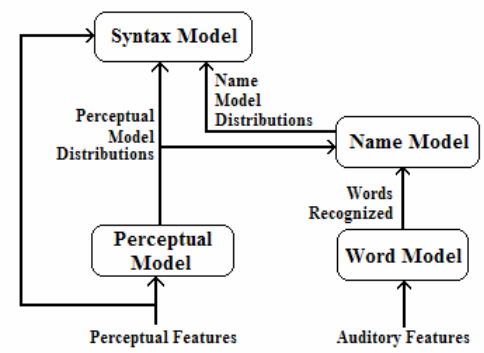


Figure 3: Cognitive model for syntax acquisition

adding a syntax model and renaming the concept, visual, and auditory models as the name, perceptual, and word models, respectively, to better reflect their role in the syntax acquisition framework.

5.2. *Inferring syntactic information*

The syntax model in Figure 3 infers the syntactic information using the perceptual information that the robot is observing and knowledge of the lexical semantics of the words uttered to describe the perceptual information. The inference will take place by observing which of the perceptual modalities each of the words best describe. The modalities that are described by each of the words will be used as the estimates of the lexical categorization parameters and whether the words describe features from the same object will be used as the estimate of the compositional semantics parameter. These parameters will be defined later in Section 5.4. These estimates will then be used to train the syntax model.

Figure 4 shows an example of the inference. Here, the perceptual modalities are shape, color, and change in position. Prior to this, the associative memory has learned the names of two objects, "cat" and "ball" (defined by shape and color), and two actions, "stay" and "move" (defined by the observed change in position). Note, however, that since the lexical categorization information is not innate, the perceptual representations of each of these words use all of the perceptual features – that is, the probabilistic models used to describe each of the words will have distributions over all of the perceptual modalities. It is assumed that the unrelated features (for example, the shape feature in the perceptual representation for "stay") will have high variance in the perceptual representations as a result of learning the words in a variety of contexts. The lexical semantics is represented in Figure 4 by the connections between the word, name, and perceptual models, shown for the two words recognized. In this example, the robot observes a cat remaining still and a ball moving and the sentence "ball move" is spoken to describe this. Then, the syntax model infers that that the first word ("ball") best describes the shape and color of one of the objects, and that the second word ("move") best describes the change in position of the same object.

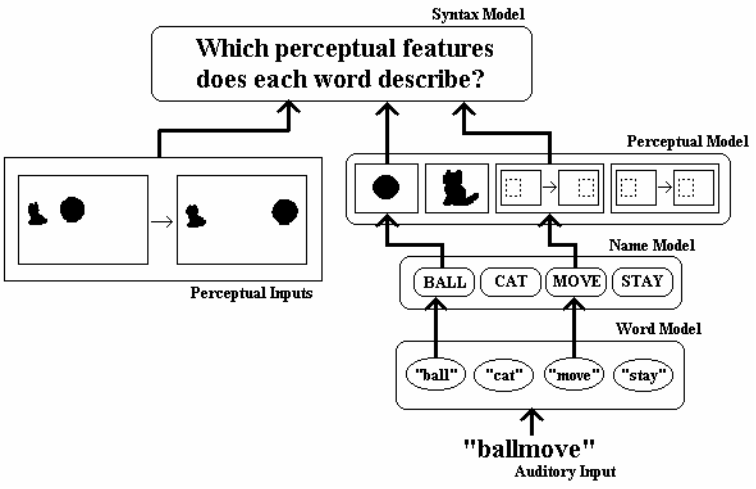


Figure 4: Example of Inferring Syntactic Information

5.3. *Syntax model implementation*

A hidden Markov model (HMM) will be used to represent the production rule to be learned in the syntax model. The experiment by Cave and Neuwirth shows that HMMs can learn categorization and sequence information in an unsupervised manner, which is required for learning the lexical categories.[10] The HMM will have three states, where the lexical categorization information will be the observable outputs of each state. Each state of the HMM is meant to represent a lexical category, with the intention that after training, two of the states will represent the non-terminal symbols N and V, and the third state will represent a delimiter. The state transition probabilities of the HMM will then represent the ordering information of the production rule.

It is noted here that choosing three states for the HMM introduces some prior knowledge to the syntax learning task. While ideally this would not be necessary, some number of states needs to be chosen for the HMM. The syntax model assumes that it hears two-word sentences, and so the choice of three states is reasonable in order to represent each of the words as well as a delimiter.

The model for the compositional semantics will be a single parameter outside of the HMM, β_{syn}, which signifies the probability that the two words spoken describe features from the same object in the robot's visual field. This information is not represented by any part of the HMM because it is associated with the whole production rule, and not any single lexical category. From the example in Figure 4, β_{syn} should be estimated to be 1 because the words "ball" and "move" both describe features of the ball - the first word describes the shape and the second word describes its change in position.

5.4. *Likelihood function*

The syntactic information will be inferred by maximizing the likelihood that the two-word sentence describes the perceptual information, given a set of parameters that represent the syntactic information. This likelihood function will sum the contributions from each possible interpretation of how the two-word sentence could describe the perceptual information. Each of these interpretations will contain a hypothesis of the lexical categorization and a hypothesis of the compositional semantics.

Each lexical categorization hypothesis will be represented by a binary vector whose length is the number of perceptual modalities. The value of an element in the vector gives the word (0 or 1) that is hypothesized to describe the perceptual modality (grouping of perceptual features) that corresponds to the vector's index where that element exists. For example, hypothesis vector [1 0 1] means that the word in position 1 describes the first and third perceptual modalities and the word in position 0 describes the second perceptual modality. With the constraint that each word must describe at least one perceptual modality, the number of lexical categorization hypotheses is 2^n-2, where n is the number of perceptual modalities. The lexical categorization parameters will be represented by a matrix α, where α_{ij} is the probability that the perceptual representation of word j describes perceptual modality i. Thus, a lexical categorization hypothesis h gives an assignment of each perceptual modality to one of the two words. The probability of a hypothesis h will correspond to a product of α_{ij} values where $j_h(i)$, the i^{th}

element of the lexical categorization hypothesis vector j_h, is used to determine the value of j.

The compositional semantics will be represented in the likelihood function by the m by m square matrix $\boldsymbol{\beta}$, where m is the number of objects present in the robot's visual field. The elements of $\boldsymbol{\beta}$ represent every possible way that the two words can correspond to the objects in the robot's visual field. Since each object will be have a corresponding perceptual feature vector (described in Section 6.1.1), β_{xy} is the probability that the word 0 describes perceptual features in vector x and word 1 describes features in vector y.

The likelihood function is:

$$L(\alpha(\beta,v,\vec{w}) = \sum_{h=0}^{2^n-2}\sum_{x=1}^{m}\sum_{y=1}^{m}\beta_{xy}\prod_{i=1}^{n}\alpha_{ij_h(i)}[\prod_{j_h(i)=0}P(v_{xi}\mid w_0)]*[\prod_{j_h(i)=1}P(v_{yi}\mid w_1)] \quad (1)$$

where m is number of objects present in the robot's visual field, each of which has an associated perceptual feature vector, v_{xi} represents feature i from feature vector x and w_i is word number $i \in \{0,1\}$ of the two word sentence. Included in this likelihood function is the assumption that if a word describes more than one feature, those features are present in the same perceptual feature vector.

The probabilities in Eq. (1) must be computed using the learned lexical semantics from the perceptual and name models in Figure 3, using:

$$P(v_{xi}\mid w_k) = \sum_i\sum_j P(n_i\mid w_k)P(p_j\mid n_i)P(v_{xi}\mid p_j) \quad (2)$$

where v_{xi} is perceptual modality i in feature vector x, w_k is word k, n_i is representation i of the name model, and p_j is representation j of the perceptual model. The probability of a perceptual modality given a perceptual representation will be computed using:

$$P(v_{xi}\mid p_j) = \prod_{a=0}^{d_i}P(v_{xi}(a)\mid p_j) \quad (3)$$

where d_i is the number of perceptual features in modality i and $v_{xi}(a)$ is feature a of modality i in feature vector x.

5.5 *Estimating the syntactic parameters*

The parameter matrices $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ will be estimated in an expectation-maximization (E-M) fashion, starting with the estimation of the $\boldsymbol{\alpha}$ values. To estimate the values for $\boldsymbol{\alpha}$, Eq. (1) will be maximized with respect to the lexical categorization hypotheses with the constraint that for each perceptual modality i,

$$\sum_j \alpha_{ij} = 1 \quad (4)$$

Eq. (5) will be used to compute the most likely lexical categorization hypothesis. Here, the previously learned value of β_{syn} will be used to create the matrix $\boldsymbol{\beta}$ by making the values on the diagonal of $\boldsymbol{\beta}$ equal to β_{syn} and the values off of the diagonal equal to 1-β_{syn}.

$$\hat{h} = \underset{h}{argmax}(\sum_{x=1}^{m}\sum_{y=1}^{m}\beta_{xy}[\prod_{j_h(i)=0}P(v_{xi}\mid w_0)][\prod_{j_h(i)=1}P(v_{yi}\mid w_1)]) \quad (5)$$

The values of $\boldsymbol{\alpha}$ that correspond to the hypothesis $j_{\hat{h}}$ are estimated to be 1, and the rest of the values in $\boldsymbol{\alpha}$ are estimated to be 0. The syntax HMM is then trained with these

estimates. This training takes three iterations, one for the each word using the corresponding α_{ij} values (j=0 for the first word and j=1 for the second) and a third to mark the end of the utterance. Thus, the HMM will first be trained with the vector of estimates that correspond to the features described by the first word, [α_{00} α_{10} …], then the estimates that correspond to the second word, [α_{01} α_{11} …], and finally with [0 0 …] to signify that the end of the sentence does not describe any perceptual information. For example, let i=0 represent the color modality in α_{ij}, i=1 represent the shape modality, and i=2 represent the change in position modality. In the "ball move" example shown in Figure 4, the correct hypothesis is j_h = [0 0 1]. Then, the first training iteration of the HMM will be [1 1 0], the second training iteration will be [0 0 1], and the third will be [0 0 0].

To estimate the values of β, Eq. (1) will be maximized with the constraint

$$\sum_x \sum_y \beta_{xy} = 1 \tag{6}$$

using

$$\hat{x}, \hat{y} = \underset{x,y}{argmax} \left(\sum_{h=1}^{2^n-2} \prod_{i=1}^{n} \alpha_{ij_h(i)} \left[\prod_{j_h(i)=0} P(v_{xi} \mid w_0) \right] \left[\prod_{j_h(i)=1} P(v_{yi} \mid w_1) \right] \right) \tag{7}$$

Since the learned α values are associated with the states of the syntax HMM and the sequence of the states is unknown, the values for α in Eq. (7) must be estimated using the state probabilities from the training of HMM described above, as:

$$\alpha_{ij} = \sum_{s=0}^{2} \alpha_{is} P(state(t) = s \mid word(t) = j) \tag{8}$$

where t is a time index, *state(t)* is the state of the syntax HMM at time t, and *word(t)* is the word recognized at time t. The time index t increments each time a word is heard and at the end of the two-word sentence.

The syntax model parameter β_{syn} will be trained by first computing β_{est} according to:

$$\beta_{est} = \begin{cases} 1, if\ \hat{x} = \hat{y} \\ 0, else \end{cases} \tag{9}$$

This estimated value will then be used to update the value of β_{syn} using the stochastic gradient:

$$\beta_{syn}^{k+1} = \beta_{syn}^{k} + \varepsilon(\beta_{est} - \beta_{syn}^{k}) \tag{10}$$

where β_{syn}^{k} is the estimated value of β_{syn} after the k^{th} iteration and ε is the learning rate.

5.6 *Word production*

In order to test the learned syntax model, it will be used to produce sentences that describe what the robot observes. The words will be generated by initializing the state of the syntax HMM to the most likely state to follow the delimiter state. Then, the word that best describes the perceptual modalities relevant to the current HMM state will be determined using:

$$\hat{k} = \underset{k}{argmax} \sum_{i=1}^{n} \alpha_s(i) P(v_{xi} \mid w_k) \tag{11}$$

where v_{xi} is perceptual modality i in feature vector x, w_k is word k of the word model, $\alpha_s(i)$ is the probability that the current HMM state describes perceptual modality i. The probabilities in this equation are calculated according to Eqs. (2) and (3). In the case that the robot is observing more than one object, the robot will decide which object will be described by the first word, reflected by the variable x in Eq. (11).

After this word is chosen, the next state is determined by choosing the most likely next state in the syntax HMM, and the word to output is chosen using:

$$\hat{k} = \underset{k}{argmax} \sum_{y=1}^{m} \beta_{xy} \sum_{i=1}^{n} \alpha_s(i) P(v_{yi} \mid w_k) \tag{12}$$

The compositional semantics information is included in Eq. (12) where x is same value as in Eq. (11). Here, β is determined using β_{syn} by the same method used for Eq. (5). This is repeated until the delimiter state is detected, which is determined when each of the $\alpha_s(i)$ values for a state are less than 0.5.

6. Robotic Experiments

6.1 *Training*

As stated earlier, the proposed syntax model will be trained by hearing noun-verb sentences that describe what the robot is observing.

The robotic environment used is an approximately 6' by 5' pen, with white walls and a white floor. The robot "Illy" is used in all of the experiments described here. The objects used in this experiment are a toy dog, a toy cat, and a soda can. To create the objects' actions that will be described by the verbs, the robot will perform one of three actions when an object is directly in front of it: move forward 6 inches, move forward 18 inches, or raise the front gripper. After performing this action, the robot will perform a counter-action: move backward 6 inches, move backward 18 inches, or lower the front gripper, respectively, and observe the effect on the object. Each of the robot's cognitive models will be trained using these objects and actions.

It is noted that the robotic environment described here constitutes a simplified version of the world. While ideally, language learning should occur in the real world, this is simply not feasible with LAR-UIUC's current robotic implementation. These simplifications are necessary in order to study higher language function. In the robotic environment used in these experiments, the learning tasks proposed are not trivial.

6.1.1 *Perceptual modalities*

The perceptual modalities that the robot will use are color, shape, the robot's action, and the object's change in position. The color, shape, and change in position features are computed using the visual processing program developed by Lin,[17] as described in section 3.2.2. Nine total perceptual features are used in this experiment, divided amongst the perceptual modalities as follows. The color modality used for the syntax experiment uses five color histogram bins. The shape modality contains two features: the ratio and moment features computed by the visual processing program. It is noted here that the shape parameters do not constitute a three-dimensional representation of the objects, the effects of which will be discussed in section 6.2.2. The robot's action modality is a

single feature that is a ternary number, representing one of the three possible actions described in section 6.1. The object's change in position is also a single feature, computed as the difference in the object's position in the robot's visual field before the chosen action is taken and after the counter-action is taken. In the case that the object leaves the robot's field of vision, a large number is chosen for the change in position measurement.

6.1.2 *Word set*

The set of words used for the experiment are "kitten", "puppy", "can", "stay", "move", and "gone". The word "kitten" is used to name the toy cat, "puppy" is used to name the toy dog, and "can" is used to name the soda can. The word "stay" is used to describe no change in position of the object, "move" describes the case when the object's position changes and remains in the robot's field of vision, and "gone" describes a large change in position or the case where the object leaves the robot's field of vision.

6.2 ***Cognitive model implementation***

Sections 6.2.1 through 6.2.4 describe the implementation of the models in Figure 3.

6.2.1 *Word model*

The word model used in this experiment uses log area ratios (LARs) to represent the spectral information of each segment of the speech signal. Programs written by Kleffner are used,[18] which provide feature extraction and speech synthesis functions. The speech signal is recorded at 22 kHz, and eight LARs are computed on 300 ms windows of speech every 100 ms. Single word recognition is performed by computing the distance between a test and example word using a method of dynamic time warping, originally used by Vintsyuk,[26] where the distance between two segments of speech is the L2 norm of the LARs of the two segments. Two word sentences are recognized using a similar method except that the test utterance is compared against all concatenated combinations of pairs of single words. This model was trained by recording examples of each of the words in the word set.

6.2.2 *Perceptual model*

A hidden Markov Model with six states is used for the robot's perceptual model. The states of the HMM are representations of each of the words that the robot learns. The observable outputs of these states are the nine perceptual features described in section 6.1.1, each modeled with discrete distributions. The state transition matrix of this HMM is set to be uniform, since the sequence information is irrelevant.

The perceptual model was created using recorded samples of the robot performing actions on the objects. It is important to note here how each of the perceptual modalities correlate with each of the perceptual model states. For example, the verbs' perceptual representations have very low variance with regards to the change in position feature, because there is a high correlation between the change in position observed and the verb used to describe it. However, the nouns' perceptual representations have higher variance with regards to the shape features, because the shape features used do not constitute three-dimensional representations of the objects (see section 6.1.1). Thus, it is expected

that more errors will be made when inferring the lexical categorization parameters for the shape modality than the change in position modality. Specifically, if low probability shape features for a noun state are encountered, the syntax model may infer that the second word uttered in the two word sentence describes the shape modality. Since the models proposed here are meant to be robust, the syntax model should be able to learn the two-word syntax as long as the number of these errors is small.

6.2.3 *Name model*

The name model is implemented using an HMM with six states (one for each object/action and the corresponding words that name them). This model uses seven observable outputs, which are modelled using discrete distributions. One of these outputs is the word recognized by the word model. The remaining six outcomes are the probabilities of each of the states of the perceptual model. This is necessary, as opposed to using a single outcome that represents the perceptual state that is recognized, because each perceptual scene has two correct single word labels.

The name model's learned distributions over the perceptual states will be used to compute the probability of each of the perceptual states given a name state, using:

$$P(p_x \mid n_y) = \sum_{i=0}^{5} b_y(x,i) * q(x,i) \qquad (13)$$

where p_x is perceptual model state x, n_y is name model state y, $b_y(x,i)$ is the value in the i^{th} bin of name state y's distribution for perceptual state x, and $q(x,i)$ is the quantization lower bound of the i^{th} bin for perceptual state x's distributions in the name model.

The name model was created using state probabilities from the perceptual model and a word recognized by the word model that applies to the perceptual information.

6.2.4 *Syntax model*

The syntax model's HMM uses four observable outputs which are the lexical categorization parameters. These represent the probabilities that each state describes each of the four perceptual feature modalities: shape, color, robot action, and change in position. Binary distributions will be used for these features.

The syntax model was trained using the RMLE algorithm developed by Squire.[19] The HMM state transition matrix was initialized to be ergodic. The lexical categorization parameters were initialized with a small bias to prevent the model from converging to a uniform solution. The compositional semantics parameter was initialized to have no bias ($\beta_{syn}^{0} = 0.5$). For training, 60 samples were recorded of the robot performing an action on an object and hearing a noun-verb description by the experimenter. These 60 samples were comprised of 30 trials with the toy kitten and 30 with the can, within which each action was performed 10 times. The toy dog was intentionally left out of the syntax training so that it could be used during testing to demonstrate the ability to produce novel sentences. The model was then trained with blocks of 1000 iterations, each using a randomly chosen sample from the set of 60. In each iteration, a second feature vector had a 50% chance of being present, representing a second object in the robot's field of vision. This second feature vector is randomly chosen from the set of samples involving the object not present in first feature vector.

For the RMLE algorithm, an exponentially decreasing learning rate was used with an initial value of 0.005, and an exponent of 0.25.

To determine convergence, the difference in all of the syntax parameter values from before and after each block of 1000 iterations was computed. If the sum of these differences was below a threshold, the model was determined to have converged. The results of the syntax model training are shown in section 6.3.4.

Figures 5 through 8 show the results of the lexical categorization parameter training. The final values of these parameters show that the syntax model has correctly learned a semantic definition of the two lexical categories used in this experiment: one state of the syntax model (state 0) is most likely to describe the color and shape modalities and

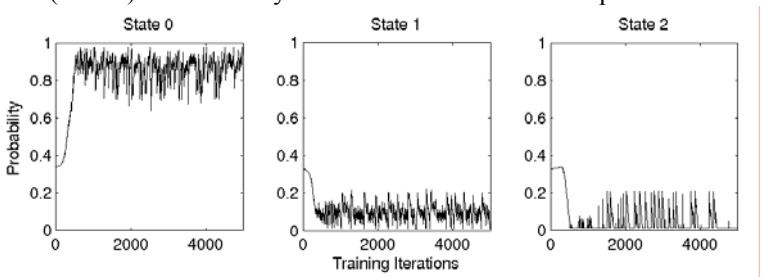


Figure 5: Probability that each syntax HMM state describes the color modality

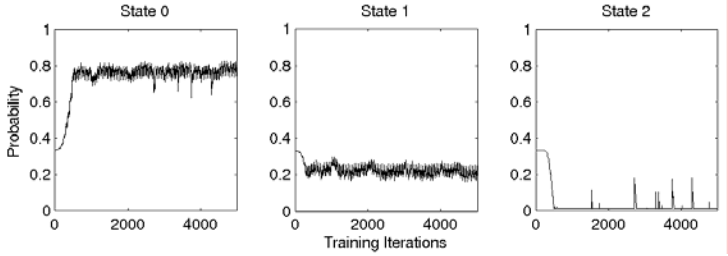


Figure 6: Probability that each syntax HMM state describes the shape modality

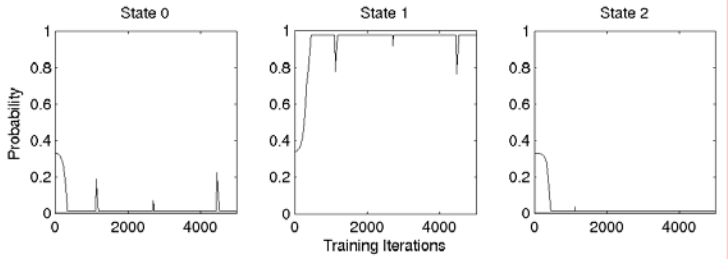


Figure 7: Probability that each syntax HMM state describes the change in position modality

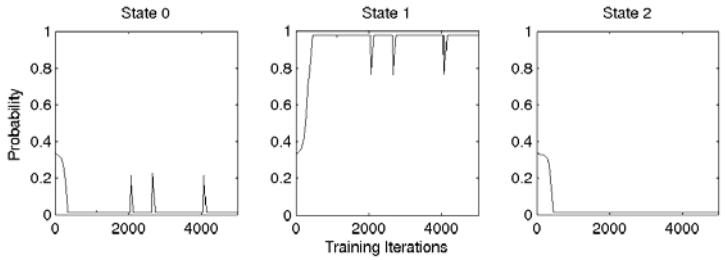


Figure 8: Probability that each syntax HMM state describes the robot action modality

another (state 1) is most likely to describe the change in position and action modalities. These two states can be interpreted as noun and verb lexical categories, respectively. As well, the remaining state can be interpreted as a delimiter state.

These figures also reflect the expected errors that occurred in training. As noted in section 6.3.3, the shape feature distributions in the noun states have higher variance than the change in position distributions in the verb states. This is due to the nature of the shape features extracted by the robot's visual processing, in that they do not give a three dimensional representation of the objects. This resulted in errors in training that are reflected in the final values of the parameters for state 0 and state 1in Figure 6. In contrast, Figures 7 and 8 show that the parameters for state 1 reach a higher final value.

During training, the lexical categorization parameter estimates for each iteration were recorded. The estimates that were chosen at least once, along with the number of times that they were chosen, are shown in Table 1. As can be seen, the correct hypothesis was chosen 71% of the time. Thus, despite the fact that many errors were made, the model proposed here is robust enough to correctly learn the lexical categorization.

Table 1: Lexical categorization hypothesis estimates

Word Hypothesized to Describe the Modality				
Color	Shape	Change in Position	Action	Number of times chosen
1	0	0	0	60
1	1	0	0	20
1	1	1	0	4
1	1	0	1	2
0	0	1	1	3548
1	0	1	1	318
0	1	1	1	1048

The state transition parameters, along with the lexical categorization information, give the ordering of the lexical categories. Figure 9 shows the trends of the state transition parameters. From this, it can be seen that the syntax HMM converges to a left-to-right model in which the state sequence is [2, 0, 1, 2, …]. With the learned lexical categorization information shown in Figures 5 through 8, the syntax HMM can be interpreted as representing the production rule S→NV.

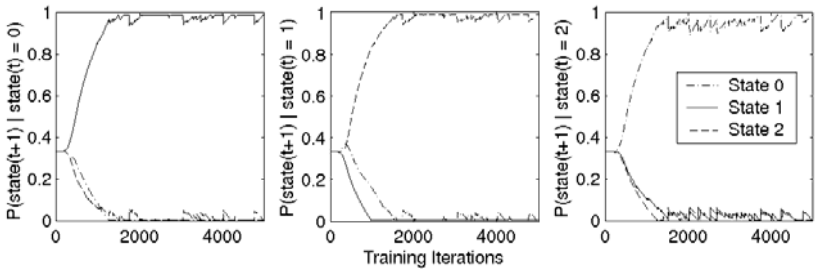


Figure 9: State transition parameters

The compositional semantics parameter was defined to be the probability that the each word in the two-word sentence describes perceptual features regarding the same object. The training of this parameter is shown in Figure 10. Since the final value of this parameter is close to 1, this parameter was learned correctly.

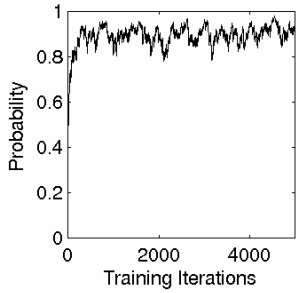


Figure 10: Probability that each word describes modalities from the same perceptual feature vector

6.4 *Testing*

The syntax model learned by the robot was tested by using the model to produce two-word sentences that describe what the robot observes. In this test, the robot chooses an action to perform on an object (placed in front of it by the experimenter) and outputs a two-word sentence using the method described in section 5.6. The testing involved 45 trials: fifteen trials with each of the objects and five each of the robot actions with each object. The robot was able to produce the correct sentence 41 out of the 45 trials (91% accuracy). In each case that the robot produced an incorrect sentence, the error occurred with the first word only. Thus, the accuracy of producing correct words was 96% (86 out of 90). As was required by the model, the robot was able to produce two-word sentences describing the toy dog, which means that it was able to produce sentences that it never heard before. A video showing an example of three test iterations can be found at http://www.ifp.uiuc.edu/speech/acquisition/acquisition.html, in the "Demonstration Videos" section, under "Semantic Based Learning of Syntax."

In each of the errors in sentence production, the robot outputted a verb when the correct word was "puppy". An inspection of the perceptual features in these trials showed that in each of these cases, a low probability shape or color feature was encountered for the perceptual state corresponding to the word "puppy". Thus, these

errors occurred because of an inaccurate perceptual model of the "puppy" object and not because the syntax training did not include the toy dog object.

6.5 *Demonstration*

In order for the ideas developed in this paper to be useful to LAR-UIUC, the syntax model was incorporated into the robot's autonomous exploration program. In the demonstration, the robot looks for objects in its environment based on the visual (color and shape) modalities and chooses a target object once one is found. The experimenter can also direct the robot to a specific object. This is done by first calling to the robot, which uses its sound source localization abilities to turn toward the speaker. Then, the robot listens for a word and then attempts to find that word's referent in its environment. Once the target is in the robot's visual field, the robot navigates toward it.

When the object is directly in front of the robot, an action (from the possible actions described in Section 6.1) is performed with the purpose of affecting the object. This is followed by the appropriate counter-action (again, described in Section 6.1). After this, the experimenter prompts the robot to either produce a sentence that describes what the robot observed or to listen for a two-word sentence for training. Then, the robot backs up to prepare to search for another object and remembers the last target object so that the same object is not used as the next target.

A video showing the syntax demonstration can be found at http://www.ifp.uiuc.edu/speech/acquisition/acquisition.html, in the "Demonstration Videos" section, under "Semantic Based Learning of Syntax."

7. Discussion and Future Work

This section will discuss the results of the experiments and provide some detail of work that can be done to improve or extend the syntax acquisition framework.

7.1 *Discussion*

The syntax model presented in this paper is shown to correctly acquire the sentence production rule S→NV using semantics. That is, the model learned the lexical categories N (noun) and V (verb) as defined by the perceptual modalities that the categories describe, the correct ordering of the lexical categories in the production rule, and the compositional semantic information as defined for this work. The syntax model's knowledge of lexical categories can be viewed as learning the production rules of the form X→x, where X is either N or V and x is a word is the robot's lexicon. Thus, the syntax model has demonstrated the ability to learn syntax production rules in Chomsky Normal Form. If the model presented here can be extended to learn additional sentence production rules, in particular recursive rules, the robot will have the ability to create and understand sentences of English generated by context-free grammars.

The learned syntax model was used to produce novel two-word sentences that describe the robot's environment. The errors that were made in sentence production were due to deficiencies in the learned perceptual model and not errors in the learned syntax model.

The implementation of the syntax model presented in this paper is shown to be robust. The model was able to converge to the correct values despite errors made during the training.

7.1.1 *Issues with the proposed syntax model*

The computation of the likelihood function in Eq. (1) becomes intractable if the number of perceptual modalities is large. This is because the number of lexical categorization hypotheses grows exponentially with the number of perceptual modalities. Thus, the syntactic parameter estimation proposed in this work may be inappropriate if the number of perceptual modalities used increases significantly. Perhaps there is also additional information available to this learning situation which could reduce the number of lexical categorization hypotheses that need to be computed.

The syntax model used in this paper requires that only two-word sentences of the form S→NV are heard by the robot to describe its world. This is obviously inconsistent with the natural speech that children are exposed to while learning language. Thus, the formulation of the robotic syntax acquisition here is meant to introduce a possible framework for inferring syntactic information using semantics.

7.2 *Compositional semantics representation*

In order to create a syntax model that is capable of learning additional rules of sentence production, better representations of compositional semantics information must be developed. A shortcoming of the syntax model presented in this paper is that the information about the compositional semantics to be learned was defined by the designer of the model. In general, any such knowledge about syntax should be inferred from interpretations of how the sentences heard describe the world. A problem that must eventually be solved is how design a syntax model that can determine what information is relevant to lexical semantic and what is relevant to the compositional semantics. Such a solution will require more general representations of the semantic interpretation of a sentence than provided by Eq. (1).

7.3 *Learning additional production rules*

The model presented in this paper provides a means of representing one sentence production rule in Chomsky Normal Form (CNF) with an HMM. The extension of this work to multiple production rules is not trivial. The two types of production rules that will be discussed here are other production rules of the form S→XY (production rules that start sentences) and Z→VW (production rules that do not start sentences).

In order to extend the current work to represent additional production rules of the form S→XY, a possible start to the solution would be to add two more states per additional production rule to the syntax HMM. The S state will serve as the start state for each of the production rules. After learning the rules of syntax, the state that follows the S state will no longer be deterministic, because the first non-terminal symbol of each production rule will be possible. Thus, the state following the S state will need to be determined using compositional semantics information. For example, if the syntax model has learned the two production rules S→NV and S→VN (for example, “move ball”), the robot would need to understand the difference in the compositional semantics

of each: the S→NV rule signifies that the speaker intends to describe something and the S→VN rule signifies a command. To accommodate this, a more complex representation of the compositional semantics would be necessary, especially because the model would need to account for which compositional semantics parameters apply to which non-terminal symbols represented by the HMM states.

The ideas behind learning production rules of the form S→XY can be generalized to learning any CNF sentence production rule. In this work, the learning of a rule of syntax is treated as the learning of how the information encapsulated in the non-terminal symbol on the left-hand side of the production rule is divided between the non-terminal symbol on the right-hand side and the compositional semantics. Viewed in this way, the learning of additional production rules of the form Z→VW is possible.

In order to acquire a new sentence production rule of the type Z→VW, the syntax model would need to have already learned a rule of the form S→XY, where Z in the first rule is the same symbol as either Y or Z in the second. Then, when a three word sentence is heard, the lexical categorization information will be inferred about each word, and a rule of the form Z→VW will need to be inferred that combines the lexical categorization information about the two possible pairs of the words into one of the lexical categories in the right hand side of the production rule S→XY. To illustrate this, the "ball move" example form Figures 1 and 2 will be extended to "red ball move". Here the rule of syntax already learned would be S→XY, where the lexical category X describes shape and color information and Y describes change in position. Then, when the sentence "red ball move" is heard, a new lexical categorization will need to be inferred for each word: A→"red", B→"ball", and C→"move", where A describes color information, B describes shape, and C describes change in position. A new rule of syntax will need to be inferred to transform the string ABC into XY. The intuition here is that the new rule Z→VW should mean that the lexical category Z describes all of the information described by V and W, with some additional compositional information. So, the inference algorithm should find that X→AB is the relevant rule. As well, it will need to be inferred that Y and C are equivalent lexical categories.

7.4 *Perceptual model improvements*

The training results in section 6.2.4 show that the variance of the perceptual model in the syntax acquisition experiment had a direct affect on the accuracy of the syntactic parameter estimation. Therefore, improvements to the perceptual model will help the accuracy of the syntax acquisition and the sentence production. Some possible improvements include using shape features that are rotation-invariant, using linear discriminant analysis for modalities with multiple perceptual features, and using Gaussian distributions in the HMM instead of independent discrete distributions.

8. Conclusions

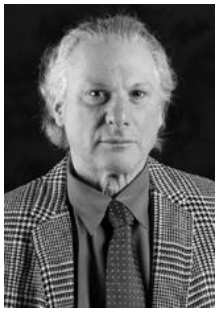
The current work provides a method for acquiring information about syntax using semantics. The models and algorithms developed in this paper are meant to be intuitive and straightforward. It is hoped that the results here give weight to idea that the human use of language is driven by semantics and that a wealth of innate knowledge of language is not necessary.

References

1. Y. Sugita and J. Tani, "A holistic approach to compositional semantics: a connectionist model and robot experiments," Advances in Neural Information Processing Systems 16 (NIPS2003), Vancouver and Whistler, Canada, (Eds) S. Thrun, L. K. Saul and B. Scholkopf, The MIT Press, pp.969-976, 2004.
2. D. Roy, "Learning visually-grounded words and syntax for a scene description task," Computer Speech and Language, 16(3), pp353-385, 2002.
3. D. Roy, "Learning visually grounded words and syntax of natural spoken language," Evolution of Communication, 4(1), pp33-56, 2001.
4. R. Brown, A First Language: The Early Stages, Cambridge, Massachusetts: Harvard University Press, 1973.
5. L. Bloom, Language Development: Form and Function in Emerging Grammars, Cambridge, Massachusetts: The MIT Press, 1970.
6. F. Chang, G. Dell, and K. Bock, "Becoming Syntactic," Psychological Review, under review.
7. H. Kamp and U. Reyle, From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory. Dordrecht, Holland: Kluwer Academic, 1993.
8. N. Chomsky, Syntactic Structures, The Hague: Mouton, 1957.
9. N. Chomsky, "On certain formal properties of grammars," Information and Control, 2, pp 137-167, 1959.
10. R. L. Cave and L. P. Neuwirth, "Hidden Markov models for English," Hidden Markov Models for Speech, vol. IDA-CRD, Princeton, NJ, 1980.
11. R. Brooks, "Achieving artificial intelligence through building robots," Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Tech. Rep. 899, 1986.
12. J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur and E. Thelen, "Autonomous Mental Development by Robots and Animals," *Science*, vol. 291, no. 5504, pp. 599 - 600, Jan. 26, 2000.
13. S. Zeng and J. Weng, "Online-learning and attention-based approach to obstacle avoidance using a range finder," *Journal of Intelligent and Robotic Systems*, vol. 43, no. 2, June 2005.
14. Y. Zhang and J. Weng, "Grounded Auditory Development by a Developmental Robot," in *Proc. INNS/IEEE International Joint Conference of Neural Networks 2001 (IJCNN 2001)*, Washington DC, pp. 1059-1064, July 14-19, 2001.
15. O. Sporns, and W.H. Alexander, "Neuromodulation and plasticity in an autonomous robot," *Neural Networks* 15, 761-774, 2002.
16. O. Sporns and W.H. Alexander, "Neuromodulation in a learning robot: interactions between neural plasticity and behavior," Proceedings of IJCNN 2003, 2789-2794, 2003.
17. R. Lin, "Unsupervised learning of nonlinear manifolds for map building on an autonomous robot," Ph.D. dissertation, University of Illinois at Urbana-Champaign, 2005.
18. M. Kleffner, "A method of automatic speech imitation via warped linear prediction," M.S. Thesis, University of Illinois at Urbana-Champaign, 2003.
19. K. Squire, "HMM-based semantic learning for a mobile robot," Ph.D. dissertation, University of Illinois at Urbana-Champaign, 2004.
20. M. McClain, "The role of exploration in language acquisition for an autonomous robot," M.S. Thesis, University of Illinois at Urbana-Champaign, 2003.
21. L. Bloom, Language Development From Two to Three, New York: Cambridge University Press, 1991.
22. R. Brown, Psycholinguistics, New York: The Free Press, 1970.

23. M. Tomasello, “Pragmatic Contexts for Early Verb Learning,” in Beyond Names for Things, ed. by M. Tomasello and W.E. Merriman, pp 115-146, 1995.
24. D. Jurafsky and J. H. Martin, Speech and Language Processing. Englewood, New Jersey: Prentice-Hall, 2000.
25. S. Pinker, Language Learnability and Language Development, Cambridge, Massachusetts: Harvard University Press, 1984.
26. T.K. Vintsyuk, “Recognition of words of oral speech by dynamic programming methods,” *Kibernetika*, 81(8), 1968.

Matt McClain received his M.S and Ph.D. degrees from the University of Illinois at Urbana-Champaign, in 2003 and 2006, respectively. His research interests are in language and cognitive modelling. He is currently employed as a Senior Research Engineer at 21st Century Technologies in Austin, Texas.



Stephen E. Levinson was born in New York City on September 27, 1944. He received the B. A. degree in Engineering Sciences from Harvard in 1966, and the M. S. and Ph.D. degrees in Electrical Engineering from the University of Rhode Island, Kingston, Rhode Island in 1972 and 1974, respectively. From 1966-1969 he was a design engineer at Electric Boat Division of General Dynamics in Groton, Connecticut. From 1974-1976 he held a J. Willard Gibbs Instructorship in Computer Science at Yale University. In 1976, he joined the technical staff of Bell Laboratories in Murray Hill, NJ where he conducted research in the areas of speech recognition and understanding. In 1979 he was a visiting researcher at the NTT Musashino Electrical Communication Laboratory in Tokyo, Japan. In 1984, he held a visiting fellowship in the Engineering Department at Cambridge University. In 1990, Dr. Levinson became head of the Linguistics Research Department at AT&T Bell Laboratories where he directed research in Speech Synthesis, Speech Recognition and Spoken Language Translation. In 1997, he joined the Department of Electrical and Computer Engineering of the University of Illinois at Urbana-Champaign where he teaches courses in Speech and Language Processing and leads research projects in speech synthesis and automatic language acquisition. Dr. Levinson is a member of the Association for Computing Machinery, a fellow of the Institute of Electrical and Electronic Engineers and a fellow of the Acoustical Society of America. He is a founding editor of the journal Computer Speech and Language and a former member and chair of the Industrial Advisory Board of the CAIP Center at Rutgers University. He is the author of more than 80 technical papers and holds seven patents. His new book is entitled "Mathematical Models for Speech Technology"