ELSEVIER

Cognition 83 (2002) 167–206

COGNITION

www.elsevier.com/locate/cognit

Bootstrapping the lexicon: A computational model of infant speech segmentation

Eleanor Olds Batchelder*

The Graduate Center of the City University of New York, New York, NY, USA

Received 9 May 2000; received in revised form 31 August 2001; accepted 5 January 2002

Abstract

Prelinguistic infants must find a way to isolate meaningful chunks from the continuous streams of speech that they hear. BootLex, a new model which uses distributional cues to build a lexicon, demonstrates how much can be accomplished using this single source of information. This conceptually simple probabilistic algorithm achieves significant segmentation results on various kinds of language corpora – English, Japanese, and Spanish; child- and adult-directed speech, and written texts; and several variations in coding structure – and reveals which statistical characteristics of the input have an influence on segmentation performance. BootLex is then compared, quantitatively and qualitatively, with three other groups of computational models of the same infant segmentation process, paying particular attention to functional characteristics of the models and their similarity to human cognition. Commonalities and contrasts among the models are discussed, as well as their implications both for theories of the cognitive problem of segmentation itself, and for the general enterprise of computational cognitive modeling.

Keywords: Speech segmentation; Infants; Computational algorithm; Probabilistic cues; Cognitive models

1. Introduction

One of the infant's early tasks is to break up continuous streams of speech into more manageable chunks that can be attached to meaning. The problem can be represented schematically:

`themaninthemoon`	(what the child hears)
`the man in the moon`	(a successful segmentation)
`them anin them oon`	(an unsuccessful segmentation)

A successful segmentation – one which locates "words" – is a logically necessary prepara-

* 70 West 95th Street, 9H, New York, NY 10025-6752, USA.
E-mail address: eob@post.harvard.edu (E.O. Batchelder).

0010-0277/02/$ - see front matter
PII: S0010-0277(02)00002-1

tion for the more complex language learning which follows. Since each language has different words, and different regularities for word formation, successful segmentation cannot be due to innate knowledge.[1]

That the child succeeds in discovering words early and often is clear. According to Mandel, Jusczyk, and Pisoni (1995), infants as young as 4.5 months can distinguish their own names, said in isolation, from other names which are similar in stress pattern (e.g. *Joshua* vs. *Agatha*, *Brandon* vs. *Kevin*) and prefer them, as shown by significantly longer looking times. At 6 months English-learning children understand "mommy" and "daddy" to refer to their own parents (Tincoff & Jusczyk, 1999). Although there is wide individual variation,[2] by 1 year 4 months of age most children have a comprehension vocabulary of at least 50 words (Harris & Chasin, 1999).

This first word comprehension, or "the child's dawning appreciation of some of the conventional meaning units of the adult language" (Vihman, 1996, p. 122), is one result of a successful chunking or segmentation process. Various sources of information that the infant might use for word segmentation have been proposed, and behavioral experiments with infants have tested the availability and effectiveness of prosodic information like pauses, stress, and intonational contours,[3] phonetic cues to word boundaries,[4] phonotactics,[5] and the distribution of sounds in the speech stream,[6] as well as tests of two or more of these strategies working in combination.[7] Research in this area has expanded lately to the point where space does not permit a proper review here; for comprehensive surveys, see Jusczyk (1997, 1999) and Aslin, Jusczyk, and Pisoni (1998).

In this paper, I will focus on just one of these sources of information – the distribution of

[1] The term "word segmentation" is also used in the literature to refer to processes used by adults in understanding spoken language, but it is important not to conflate the two contexts. The infant has no words at all to begin with, while the adult can take advantage of a rich mental lexicon.

[2] Illustrative ranges are found in Waxman (1999), where 36 infants aged 12.7–14.5 months (average 13.5) were surveyed, finding production vocabularies of 0–112 words (average 16) and comprehension of 5–327 (average 112).

[3] Prosody: research showing the sensitivity of infants to prosodic features of language and/or evidence that babies use such features for segmentation includes: Christophe, Dupoux, Bertoncini, & Mehler, 1994; DeCasper & Fifer, 1980; Echols, Crowhurst, & Childers, 1997; Gerken, Jusczyk, & Mandel, 1994; Hayashi, Tamekawa, Deguchi, & Kiritani, 1996; Hirsh-Pasek et al., 1987; Hohne & Jusczyk, 1994; Johnson & Jusczyk, 2001; Jusczyk, 1998b; Jusczyk, Cutler, & Redanz, 1993; Jusczyk et al., 1992; Mattys, Jusczyk, Luce, & Morgan, 1999; Mehler et al., 1988.

[4] Allophones, including coarticulation cues: Johnson & Jusczyk, 2001; Jusczyk, Hohne, & Bauman, 1999; Mattys & Jusczyk, 2001a.

[5] Phonotactics refers to "the specific sequences of sounds that occur in a language" (Crystal, 1987, p. 427). Permissible sound sequences in English vary by their position within syllables, words, and morphemes, and across their boundaries: /gd/ is okay word-finally (*begged*) and across a word, morpheme, and/or syllable boundary (*big deal*), but not word-initially (**gdum*). Research on the use of phonotactics in early word segmentation includes: Friederici & Wessels, 1993; Jusczyk, Friederici, Wessels, Svenkerud, & Jusczyk, 1993; Mattys & Jusczyk, 2001b; Mattys, Jusczyk, Luce, & Morgan, 1999.

[6] Distribution: Aslin, Saffran, & Newport, 1998; Goodsitt, Morgan, & Kuhl, 1993; Johnson & Jusczyk, 2001; Saffran, Aslin, & Newport, 1996a,b; Saffran, Johnson, Aslin, & Newport, 1999.

[7] Combination: Jusczyk, Houston, & Newsome, 1999; Mattys, Jusczyk, Luce, & Morgan, 1999; Morgan, 1996; Morgan & Saffran, 1995.

segmental information,[8] or the relative frequency of sounds and sound clusters, and their tendencies to co-occur with each other and with utterance boundaries. Distributional information comes from observing the frequency of events in the environment, a skill available to even the tiniest infant, and indeed to most non-human animals; for reviews of research on the cognitive effects of frequency, see Hasher and Zacks (1984), Alloy and Tabachnik (1984), and Kelly and Martin (1994). In experiments specific to language stimuli, 8-month-old infants successfully segmented an artificial speech stream based solely on distributional information – frequency and order (Saffran, Aslin, & Newport, 1996a,b) – and the same stimuli drew similar responses from tamarin monkeys (Hauser, Newport, & Aslin, 2001). The infant experiment has been replicated with naturally spoken syllables (Johnson & Jusczyk, 2001).

Here we will be concerned not with the behavioral data, but rather with computational models of the use of distributional cues to segment words. In particular, this paper describes BootLex, a model of early word segmentation which uses the distribution of segments and pauses to discover word boundaries in several language corpora from three different languages. Second, several previously reported computer models of the same cognitive process are reviewed and compared to BootLex, not only in terms of the usual quantitative measures of effectiveness, but also by contrasting their more global functional characteristics. I hope to show that comparison of models of this small but critical cognitive process can highlight aspects of the problem – both cognitive and computational – that might otherwise be overlooked.

Section 2 of the paper describes how speech segmentation is modeled by computers, and how the performance of such models has been evaluated quantitatively, and then previews the qualitative characteristics that we will contrast in the several models. Section 3 presents the BootLex algorithm in detail. Section 4 discusses three groups of other computer models, and compares them with BootLex and with each other. Section 5 compares the cognitive plausibility of these models, and considers some broader implications.

2. Distributional models of infant speech segmentation

A number of computational models of the use of statistical cues for infant speech segmentation have been presented recently. These computer models, including BootLex, are inductive, or self-organizing, algorithms. With the significant exception of the categories implicit in the coded input, they have no linguistic knowledge to begin with. That is, there is no lexicon of known words or knowledge of applicable rules or regularities, such as phonotactics. They can only try to discover any structure implicit in the linear association of their elementary codes. Such models thus presuppose that there is structure to be discovered: in particular, that each utterance can be resolved into a series of "words" – a non-overlapping sequence of small chunks, which chunks recur across utterances in somewhat varying orders. The problem, then, is to find those recurring chunks. All of the models discussed here cast the problem in similar ways:

[8] The term "segment" is potentially confusing in the context of speech segmentation. While "segmentation" refers to any partitioning or chunking process, whatever the degree of granularity, "segment(s), segmental" refers to a minimal unit in phonetic/phonological theory, a phone such as [t] or a phoneme such as /e/.

• *Model inputs and outputs.* The models take as input a computer-readable text, generally a transcription of spoken language. Word boundaries are removed from the text before input, but utterance boundaries are usually retained, as these are assumed to be salient in the speech stream as pauses, and thus available to infants. Since each utterance boundary is also a word boundary, a subset of the true word boundaries is thus represented. The input text is read by a computer program, which selects and records certain characteristics, and then uses these to decide where in the text to place "word" boundaries. All the models generate a word-spaced version of the input text, and some also produce a lexicon of words "learned" during the segmentation process.

• *Language representation.* While speech is acoustically a continuum of sound, language is composed of a series of abstract categorical units. Transforming actual speech sounds (analog wave forms) into such categorical units is still beyond the ability of machines unless considerable linguistic knowledge is supplied, such as a list of possible words. Since our models are intended to *discover* words, some non-lexical way of representing language – the analog of the speech that the child hears – is necessary. Thus, there seems no way to do this without using some set of codes. All of the models we discuss here used some phonemic (segmental) notation system, such as:

Phonemic symbols:	lUk	D*z	6	b7	wIT	hIz	h&t
Orthography:	look	there's	a	boy	with	his	hat

The three connectionist models to be discussed used binary features derived from phonemes instead of the phonemes themselves.[9]

In addition to segmental units and utterance boundaries, other information sources have been modeled with varying degrees of success, including word stress (Christiansen, Allen, & Seidenberg, 1998) and phonotactic regularities (Brent & Cartwright, 1996). In this paper, I compare models in terms of their performance using distributional information, specifically the interrelationships of speech segments and utterance boundaries, so I omit models which are not concerned with distributional cues.[10]

• *Corpus preparation.* In addition to encoding, input texts are standardized in other ways. Punctuation and numerals are removed, capital letters (in orthographic corpora) are made lower-case, and sometimes further efforts toward uniformity are made, such as

[9] Several reasons were cited for using component features instead of discrete phonemes. One, because their designers believed that subsegmental (featural) representations were more "cognitive" or "psychologically realistic" than phones or phonemes: "An account of speech segmentation based on an input composed of phonological features may be both more parsimonious and more psychologically realistic" (Cairns, Shillcock, Chater, & Levy, 1997, p. 130). For another, to facilitate generalization: "Our strategy was to code the input as a set of articulatory features… This was done to allow the model to learn not just which specific phoneme ended an utterance, but also to generalize this end-of-utterance information to other phonemes in English that shared one or more features with this phoneme." (Aslin, Woodward, LaMendola, & Bever, 1996, p. 126f).

[10] Using only phonemic or orthographic units ignores two other subphonemic aspects of the sound stream. Phonetic regularities, such as allophones and coarticulation effects, are omitted, introducing a conservative bias, since these would provide even more information and statistical structure if they were included. Secondly, idiosyncratic differences in pronunciation from speaker to speaker are not represented. These do not conform to any pattern, so that the listener must abstract away from such differences when identifying words, and the child also has to be able to ignore them in order to find "words" successfully.

removing non-words like *um* and *huh*. Since it is often claimed that speech is more "natural" than written texts, it is ironic that model preparation proceeds to remove or transform much of the naturalness, such as variation in pronunciation, prosody, and even vocabulary. However, modelers fear that such variation in language, which is otherwise desirable to capture, might overwhelm the computer's weak skills, so they try to strike a balance between unnaturally controlled transcripts and the chaos of lifelike language.

2.1. Comparing speech segmentation models

A major goal of this paper is a close comparison of various models, including BootLex, both qualitatively and quantitatively. In this section I outline some of the ways that the models can differ in function, and also describe the quantitative measures which will be used. Table 1 serves as a reference during this and the later discussions of functional differences.

- *Build lexicon?* All the models produce word-segmented utterances, but there are two types of knowledge, or competence, that underlie this production. Some models build a lexicon as they go along, and use it as a knowledge store to recognize previously learned words when segmenting utterances. A second group of models does not create a lexicon, but learns the characteristics typical of word boundaries and then uses that knowledge to segment utterances on the basis of phonotactic regularities. For these non-lexical models, words are a side-effect of the segmenting process rather than a primary output.
- *Cluster or divide?* Another difference between the models is their learning strategy. While all the models use co-occurrence frequencies as measures of cohesion, some focus on the points of lowest cohesion and place a word boundary there, while others look for points of high cohesion and aggregate at those points to form clusters. The "divide" strategy starts with a full utterance as the default "word" and gradually breaks it into smaller and smaller pieces by inserting more word boundaries, while the "cluster" strategy begins with minimal "words" which gradually get bigger and bigger.
- *Cumulative?* Most of the models "learn" by gradually accumulating knowledge in small increments, receiving inputs in small amounts one after another and adjusting the state of knowledge after each input. Shifts in the nature of the input cause corresponding changes in the state of learned knowledge. The size of the inputs and the granularity of learning increments can differ from model to model.
- *Feedback from outputs?* All these models "learn" from the inputs, but some also learn from their own outputs by a feedback process. The first prerequisite for such a system is that it learn incrementally, as discussed above. Secondly, the system must receive inputs

Table 1
Summary of functional characteristics (as discussed in the text)

Model	Build lexicon?	Cluster or divide?	Cumulate?	Feedback from outputs?	Constraints?
BootLex	Lexicon	Cluster	Cumulate	Feedback	Optlen value
Networks	No	Divide	Cumulate	No	Threshold value
MDL	Lexicon	Cluster	No	No	Compute-intensive
MBDP	Lexicon	Divide	Cumulate	Feedback	External parameters

and produce outputs in an overlapping and continuous series.[11] (The overlapped inputs and outputs do not necessarily relate to each other one-to-one, but rather may be two continuing streams with a diffuse relationship across a delay in time.) Some systems which meet these two criteria are arranged so that the output of the system feeds back to affect the state of the system's learned knowledge.

• *Constraints?* While these computer models are self-organizing algorithms, each of them is externally controlled to some extent. It is important to note what external information sources are used (aside from the utterance inputs, and knowledge of the representational codes) to constrain the incremental process or limit the possibility space.

• *Quantitative success.* The foregoing characteristics are qualitative, which are important criteria for a cognitive computer model. However, it is also true that one criterion of a successful cognitive strategy is its effectiveness as reflected in quantitative measures. For quantitative measures of success, the common practice is to compare the word boundaries created by the model with the word boundaries in the original text (the "standard").[12] In particular, two measures from information retrieval are widely used: **recall** (the proportion of correct items which were identified) and **precision** (the proportion of identified items which were correct). Usually, running words (tokens) are the items evaluated as a measure of segmentation effectiveness, but sometimes "cuts" (word divisions inserted) are used. For those models which create a lexicon, this list of word types hypothesized can be evaluated relative to the list of word types in the original text. Such measures, called "lexical recall and precision", give equal weight to all words regardless of their frequency.

3. The BootLex algorithm

Olivier (1968) was the first to create a working probabilistic segmentation routine. His algorithm was a deceptively simple exercise in self-organization, using only letter co-occurrence frequencies to segment utterances into words, and the BootLex model is a new implementation based on his idea.[13] Because Olivier's algorithm had an unfortunate tendency to create longer and longer "words" as it proceeded, BootLex incorporates a mechanism to constrain word length, as well as other modifications (Batchelder, 1997).

BootLex can be best understood as two complementary and concurrent processes: language input is used to create a lexicon, and each input utterance is parsed using the lexicon in its current state. Technically, BootLex is a **word grammar**: "a stochastic grammar whose language is finite; it is thus equivalent to a finite dictionary of strings, each with an associated probability" (Olivier, 1968, p. 30). This dictionary, called the **lexicon**, contains a list of pairs – a character string ("word") and its current frequency

[11] In other contexts, the terms "continuous" and "incremental" have been construed as opposites, where "increment" means an abrupt, non-continuous change. Here we use these terms in a synonymous sense, where increments are successive changes that are small enough to produce an effect of continuity.

[12] It has often been suggested (e.g. Plunkett, 1993) that young children consider some clumps of words as single lexical items (*thank you*, *happy birthday*), so an evaluation metric which credits such segmented items might be closer to the child's reality. Batchelder (1997) introduces a novel scheme that incorporates these elements, but since the present paper focuses on comparison of several models, we discuss only widely used metrics.

[13] After describing BootLex, I will discuss in more detail the differences between it and Olivier's routine, as well as some other probabilistic algorithms, in Section 3.5.

value. **Parsing** is the process of exhaustively segmenting an utterance into non-overlapping words. A possible segmentation, or parse, of a given utterance is created by selecting a set of "words" from the current lexicon and arranging them end to end in such a way that the utterance is completely replicated with no gaps and no overlapping. There may be many such possible sets and arrangements for that utterance, given the current lexicon's word candidates. Each possible segmentation is assigned a score according to the current word frequencies, and the highest-scoring one is selected as the final version.

The following two sections explain these procedures in more detail; the third section gives quantitative results; and the fourth section gives a qualitative assessment using the five characteristics described above.

3.1. Building the lexicon

As the input is seen and parsed, an iterative process records information about it in the lexicon:

(a) Initialization: The starting lexicon contains the set of basic symbols (the "alphabet" of letters or other graphemes), each one as its own lexical entry with a frequency of 1. The input is presented as a series of "utterances" (lines) of varying lengths.[14]

(b) Cycle 1: Using this initial lexicon, the first utterance is parsed into "word" tokens of one symbol each.

(c) For each word token in the utterance just parsed, the matching word type in the lexicon has its frequency incremented by one.

(d) Before beginning the next utterance, the lexicon is augmented by adding to it potential new words, consisting of contiguous pairs of words in the utterance just parsed. Each such pair that is not already in the lexicon is added, with an initial frequency of 1 (if the pair is already in the lexicon, its frequency there is not altered). The lexicon as updated (frequencies incremented and new entries added) is now ready for the next utterance.

(e) Cycle 2: The second utterance is parsed into words, using only those words which are found in the current lexicon, and a score for each possible parse is calculated based on its likelihood in light of experience to date, using the frequency counts recorded in the lexicon. (The parsing procedure is discussed in more detail below.) The word tokens which make up the highest-scoring parse are used to update the frequency counts in the lexicon (step (c) above) and to make new lexical entries (step (d)).

(f) Cycles 3 to N: Repeat step (e) to the end of the text, each time using the lexicon as just modified.

The effect of this procedure is described by Olivier:

> …if a section included the string 'abcdefgh' parsed as four words 'a bc de fgh' the three new words 'abc', 'bcde', and 'defgh' would be added to the dictionary… A principal advantage of this system for adding words to the dictionary is that it swells the dictionary with large numbers of words rather quickly. (Olivier, 1968, p. 67)

[14] To simplify the computer programming, some finite number of characters per line is built into the software. In the experiments described here, a limit of 150 characters allowed all lines in the corpora being used.

Table 2
BootLex builds the lexicon while parsing a toy corpus line by line[a]

Cycle	Utterance as parsed	New lexical items added to lexicon this cycle
0	–	a b c d e f g h i j k l m n o p q r s t u v w x y z
1	a b i g p e t a c u t e d o g	ab bi ig gp pe et ta ac cu ut te ed do og
2	t h e pe t t h ed og	th he epe pet tt th hed edog
3	a d og a pet	ad dog oga apet
4	t he bi g dog t he cu te pet	the hebi big bdog dogt the hecu cute tepet
5	a big pet a cute dog	abig bigpet peta acute cutedog
6	the pet the dog	thepet petthe thedog
7	a dog apet	adog dogapet
8	the big dog the cute pet	thebig bigdog dogthe thecute cutepet
9	abig peta cutedog	abigpeta petacutedog
10	thepet thedog	thepetthedog
11	a dogapet	adogapet
12	thebig dogthe cutepet	thebigdogthe dogthecutepet
13	abig petacutedog	abigpetacutedog
14	thepetthedog	–
15	adogapet	–
16	thebig dogthecutepet	thebigdogthecutepet

[a] Lexical items which later appear as words in at least one parse are underscored.

Consequently, a large part of the lexicon at any time will consist of such pairs which have been added as potential words, but have not (yet) actually been used in a parse.[15]

The operation of BootLex is briefly illustrated in Table 2, using a miniature corpus consisting of just four "utterances" repeated four times, each with its parsed version and lexicon output. This tiny corpus has been constructed with short words and lots of repetition so that the effect of the algorithm will be quickly apparent. On the left is shown the utterance as it is parsed based on the lexicon from the preceding cycle. On the right are the new word candidates added to the lexicon as the result of this parse, by combining contiguous pairs of parsed words. Those word candidates which subsequently are used as words in at least one parse are underscored in the table; those not underscored are those "potential words" which are never used in this example.

The parsing process begins in cycle 1 by producing only single letters as "words". Then (cycle 2) it finds some letter pairs that have been entered in the lexicon, and those are preferred over single letters in the parse of the second utterance. As will be explained below, the basic algorithm will tend to create longer and longer units, a process which is constrained only by the length of an utterance, since word pairs cannot occur across an utterance boundary. In this miniature corpus, where every utterance is repeated, eventually each utterance will become a single lexical item. In natural language the length of words is

[15] BootLex tallies the occurrences of a subset of the sound patterns that it observes, using the "word pair" scheme to select just which patterns it records. While we do not imagine that the child's processing would be so mechanical, we feel that children must similarly observe and record the frequency of sound patterns in their environment.

not so unlimited, so we modified the basic algorithm to constrain its tendency to produce unnaturally long words, as described in the following section.

3.2. The parsing procedure

In BootLex, each possible parse is scored by taking the probability for each word in the parse – that is, for each word token in the parse, the probability of its word type in the lexicon, which in turn is its frequency count divided by the sum of all such counts in the lexicon (**maximum-likelihood estimate**, or **MLE**). Then all these probabilities are multiplied together to produce a combined probability which represents the probability of that parse, and the probabilities of the various possible parses can be directly compared with one another.

As described in the previous section, the lexicon is modified after each utterance is parsed, so that each succeeding utterance is parsed with respect to a slightly different lexicon. This iterative process – parse decisions leading to changes in the lexicon, which in turn lead to new parse decisions, etc. – is a "training" or learning process. In BootLex, parsed words are used to create word pairs which are entered in the lexicon as candidates for future words, with the result that the candidate word types tend to increase in length. In addition, MLE scoring always favors a longer unit over two shorter ones, simply because of the mathematics of fractional products. Since probabilities are fractional (less than unity), two multiplied together will generally be less than the probability of the combined string, so "the estimated frequency of a word is much larger than the product of the estimated frequencies of its parts" (Olivier, 1968, p. 59). This relationship is illustrated in Table 3, where *ofthe* as one word scores higher than *of the* as two words.

Each occurrence of a word in a successful parse increases its frequency count by one. Thus, as longer and longer words become eligible for parsing, they tend to be selected over shorter words, which raises their frequency and further increases their likelihood of future selection. As the algorithm proceeds through a text, unless it is constrained, the parsed words tend to get longer. A typical unconstrained run tends to underdivide the text, creating words which are on average longer than those in the standard.

To correct this, BootLex uses an external constraint – the **optimum-length parameter** – which serves as a target for the average word token length. By referring to this value, each parse is evaluated not solely by maximum likelihood, but also by how closely the average word length of the parsed utterance approaches the **optlen**. Parses with overly long words have their "goodness" rating lowered slightly. Thus, the final score of each

Table 3
Example of parse detail calculations by maximum-likelihood estimation (MLE), taken from a corpus with 74,951 total tokens

Word	Number of occurrences	Parse likelihood (MLE)
of	1839 times	$1839 \div 74951 = 0.0245$
the	3184 times	$3184 \div 74951 = 0.0424$
of + *the* (2 words)	(product)	$0.0245 \times 0.0424 = 0.0010$
ofthe (1 word)	393 times	$393 \div 74951 = 0.0052$

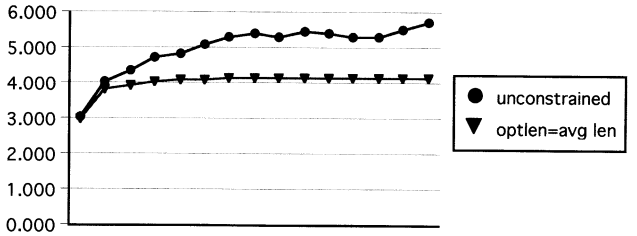

Fig. 1. Comparison of the average word length over successive 5000-word sections of a text as parsed by BootLex with and without the optimum-length parameter.

parse will be a combination of its raw probability and its relative nearness to the optlen target. Though optlen is not a rigid threshold, but rather the focus of a graded scale of adjustments, its cumulative effect over a large number of parsing decisions is to control fairly well the number of words produced by the algorithm and their average length (Fig. 1).[16] The question of cognitive parallels for an external constraint like optlen will be discussed below. For the moment, consider it as a way of exacting the maximum effectiveness from the algorithm on a given corpus. In general, setting optlen equal to the average word length (in characters) of the standard version of the text will produce a number of parsed words roughly equal to the number of word tokens in the standard, which gives the maximally accurate results when scored relative to that standard.

The effect of the optlen parameter is illustrated in Table 4, which shows the effect on the same miniature corpus from Table 2 of an optlen equal to its average standard word token length of 2.7 characters. Since the optimum-length criterion penalizes those parses which have an average word length greater than the optlen value, parses which use shorter words tend to win the competition and word length does not grow in an unconstrained fashion over the learning process. Fig. 2 charts the effect of optlen on a more realistic corpus, comparing segmentation results using various optlen values on a corpus of transcribed English speech. It can be seen here that, while the average-length value (1.0) produced the best performance, any value for optlen gave better results than not using it at all.

3.3. Quantitative results of BootLex

BootLex was exercised on a wide variety of text corpora, both to evaluate the effectiveness of the algorithm itself under various conditions, and as part of a larger project to explore differences among various language corpora (Batchelder, 1997). Six corpora were used, two each in English, Spanish, and Japanese. One of each language pair was from the CHILDES collection of corpora of transcribed child-directed adult speech (MacWhinney, 1995), and the second was a text which had been originally composed in writing (a science book for young children, and a novel and some short stories for an adult audience). All the

Table 4
BootLex parses of toy corpus without optlen (same as Table 2) and with optlen

Cycle	Parse without optlen (from Table 2)	Parse with optlen (= avglen of 2.7 chars.)
1	a b i g p e t a c u t e d o g	a b i g p e t a c u t e d o g
2	t h e pe t t h ed og	t h e pe t t h ed og
3	a d og a pet	a d og a pet
4	t he bi g dog t he cu te pet	t he bi g dog t he cu te pet
5	a big pet a cute dog	a big pet a cute dog
6	the pet the dog	the pet the dog
7	a dog apet	a dog apet
8	the big dog the cute pet	the big dog t he cute pet
9	abig peta cutedog	abig pet a cute dog
10	thepet thedog	the pet the dog
11	a dogapet	a dog apet
12	thebig dogthe cutepet	the big dog t he cute pet
13	abig petacutedog	abig pet a cute dog
14	thepetthedog	the pet the dog
15	adogapet	a dog apet
16	thebig dogthecutepet	the bi g dog the cute pet

results reported in this section were achieved with the parameter optlen set to the average length of a standard word in the corpus being tested.

The best results were achieved on an English corpus of child-directed speech derived from the Bernstein-Ratner corpus (Bernstein Ratner, 1996) in the CHILDES collection. The original Bernstein-Ratner corpus was collected from recordings taken over a period of a year from nine mothers, each talking to her infant daughter aged 1;1 to 1;11, and transcribed in standard English orthography. For use with a computer model it was transcribed into 50 phonemic symbols – "an ASCII-based phonetic representation [which] paralleled the IPA alphabet… diphthongs, r-colored vowels and syllabic consonants were each represented as one character" (Cartwright & Brent, 1994, p. 150). For instance, 'b7' was *boy*, 'lebL' was *label*, and 'bRd' was *bird*. Parallel samples of both transcriptions are shown in Table 5 to illustrate the representational system, and also to show BootLex's

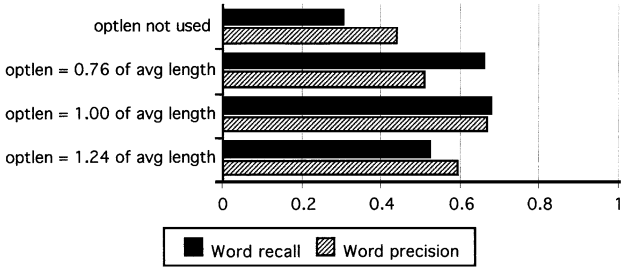

Fig. 2. BootLex segmentation results using various optimum-length parameters on one corpus.

segmentation results as it progresses through successive utterances in both orthographic and phonemic versions of the same corpus.

Results of BootLex runs on several variations of this corpus[17] are shown in the rightmost two columns of Table 6: *word recall*, *word precision*. The dimensions along which the corpora were varied are described in the columns to the left: child- vs. adult-directed speech (the latter corpus created from separately recorded interviews with each mother), size (*number of characters*, *words*), and coding structure (*number of codes* and *code type*: phonemic symbols vs. orthography). The results demonstrate the greater segmentation effectiveness of a larger corpus, of phonemic coding, and of child-directed speech. The best results (highest recall and precision) were obtained on the *Large* phoneme-coded child-directed corpus, and smaller amounts of data in the same series (*Medium* and *Small*) resulted in corpora with poorer performance, probably due to lower repetition levels (*TTR*). The adult-directed corpus, with a much longer utterance length (*Avg # chars/utt*), and consequently a lower percentage of word boundaries given as utterance boundaries (*%UB*), performs the worst of the five, even though it is the second biggest; it also has low repetition.[18] The medium-size phonemic and the orthographic corpora had the same number of words and the same content, simply coded differently, and the phonemic coding did a little better. However, we show below that this tendency was not borne out with other corpora.

• *Multilingual tests.* To test whether these results were limited to English, we also ran BootLex on child-directed corpora in Japanese and in Spanish (Table 7), finding that the algorithm did not do as well on these. Not only were the languages different, but also the statistical characteristics of the various corpora, so it is difficult to draw conclusions about the effect of language alone without testing a greater variety of corpora in each language. In an extensive series of controlled tests, Batchelder (1997) found that certain characteristics of the input affected segmentation performance. When other factors were held constant, short utterance length, short words, and a high rate of lexical repetition correlated with a faster rate of "learning" – a more accurate segmentation with less text input. In turn, since these text characteristics were more typical of spoken than written texts, and of child-directed than of adult-directed speech, child-directed speech tended to be better segmented than other types of texts. However, significant amounts of learning and

[17] So that interested readers can tell which chart references are to the same corpus versions, the variations of each corpus are listed here by our internal file names, in the order of their appearance in each chart. Table 6: brechall, brecar, bresh, breconew, bread. Table 7: brechall, akirom, akih, spano; maughamt, eucnew, eucnewr, doledt. Table 8 and Fig. 3: brecar, breconew; akirom, akih, akihr; spano, spanall, spanph; maughamph, mauword, maughamt; eucnew, eucnewr, eucnewrx. Table 11 and Figs. 2, 4, 5 and 6: brechall.

[18] The metrics reported here are for the most part those that have been reported for other similar models, the better to compare them, and they are each the results of a "training" run on a full corpus. For comparison, I also report one "held-out" test (Batchelder, 1997), which showed that performance was somewhat better when using the algorithm with a previously "trained" lexicon on a novel portion of text – one test of whether the "learning" is effective on new material:

	Recall	Precision
Training on full English written corpus	44.9	43.0
Training on 90% of the corpus	44.7	42.5
Test on remaining 10% of the corpus	48.1	46.7

Table 5
Samples as segmented by BootLex from two versions of the Bernstein-Ratner corpus

Orthographic version		Phonemic version	
Line no.	Utterances as segmented	Line no.	Utterances as segmented
100	is it da d d y onthe p hon e	100	IzIt d& d i anD6 fo n
101	p r es s the but t on	101	p r Es D 6bAt ~
102	there youg o	102	D*yug o
103	say	103	se
104	hello dada	104	hElo d &d&
520	t ur nthep age	1125	t 3n D6peG
521	okay	1126	oke
522	whats here	1127	WAts h(
523	flow ers	1128	fl QRz
524	you see the flowers	1129	yu si D6fl QRz
1000	whats that	2164	WAts D&t
1001	a cat right	2165	6k&t r9t
1002	there s another cat	2166	D*z 6 nADR k&t
1003	thatsa no ther cat	2167	D&ts 6 nADR k&t
1004	whats this cat doin	2168	WAts DIs k&t du IN
1005	what shedo in	2169	WAts hi duIN
1006	whats the cat doin	2170	WAts D6 k&t duIN
1007	ishe awa ke	2171	Izhi 6we k
4287	open that	9641	op ~ D&t
4288	now em ust nt openthat one	9642	no wi mAs ~t op ~ D&t wAn
4289	because this canget us alld irt y	9643	bI kAz DIs k&ngEt As Old 3t i
4290	keep this one closed	9644	kip DIs wAn kloz d
4291	dont open that one either	9645	dont op ~ D&t wAn iDR
4292	open that	9646	op ~ D&t
4293	you can open that	9647	yu k&n op ~ D&t
4294	canya close it	9648	k&n yu kloz It
4295	very good	9649	v*i gUd
4296	you cut your finger a little bit	9650	yu kAt y) fINgR 6 lItLbI t
4297	y uh	9651	y&
4298	that s right	9655	D&ts r9t
4299	put them away	9656	pUt DEm 6we
4300	good	9657	gUd
4301	you want the doll	9658	yu want D6 dal

segmentation took place even on corpora with long utterances and low repetition. From a cognitive point of view this is a favorable finding, since a segmentation model which yields results on a wide range of language input is a more realistic analog to the way that children can learn language even under very adverse conditions. The unexplained differences in performance among the three language groups deserve further exploration.

• *Code variations.* Our tests, however, failed to detect any systematic differences in performance depending on representational code. For each of five corpora, two or three variations in code representation were submitted to segmentation by BootLex (samples of

Table 6
Corpus characteristics and results of BootLex segmentation for five variations of the Bernstein-Ratner corpus[a]

	Corpus characteristics								Performance %	
	Number of			Code type	Avg # chars		%UB	Lex Rep (TTR)	Word recall	Word precision
	chars	words	codes		/utt	/word				
Child-directed speech										
Large	95,809	33,399	50	phon	9.8	2.9	30	25.3	68.2	67.2
Medium	40,792	14,184	50	phon	9.4	2.9	31	16.0	65.0	62.3
Small	13,690	4,746	50	phon	8.9	2.9	33	9.7	57.8	51.8
Other variations										
Orthographic	53,439	14,188	26	alph	12.4	3.8	31	15.5	61.9	58.4
Adult-directed	61,784	20,242	50	phon	19.6	3.1	16	10.7	53.7	50.5

[a] Code = primitive symbols (character set): phonemic or Roman alphabet; %UB = percent of standard word boundaries that are also utterance boundaries = chars/word ÷ chars/line; TTR = (# word tokens in standard) ÷ (# of word types in standard); word recall = (# word tokens same in model and standard) ÷ (# word tokens in standard) × 100; word precision = (# word tokens same in model and standard) ÷ (# word tokens in model) × 100.

Table 7
Corpus characteristics and results of BootLex segmentation for eight corpora in three languages[a]

	Corpus characteristics								Performance %	
	Number of			Code type	Avg # chars		%UB	Lex Rep (TTR)	Word recall	Word precision
	chars	words	codes		/utt	/word				
Transcribed speech										
English (large)	95,809	33,399	50	phon	9.8	2.9	30	25.3	68.2	67.2
Japanese (A)	312,818	77,219	24	alph	11.8	4.1	35	21.4	56.0	53.8
Japanese (H cvt)	184,022	77,219	77	hira	7.0	2.4	34	21.4	54.5	53.3
Spanish	103,892	26,552	32	alph	15.6	3.9	25	10.3	50.5	46.8
Written prose										
English	313,321	74,951	26	alph	42.1	4.2	10	10.6	44.9	43.0
Japanese (H)	37,060	19,255	78	hira	17.7	1.9	11	11.0	38.0	37.4
Japanese (A cvt)	74,120	19,255	30	alph	35.5	3.8	11	11.0	35.3	34.4
Spanish	126,946	25,926	26	alph	47.6	4.9	10	4.2	32.0	26.1

[a] (A) = as originally transcribed, in Roman alphabet; (H cvt) = mechanically converted from alphabet to hiragana; (H) = as originally published, in hiragana characters; (A cvt) = mechanically converted from hiragana to alphabet; Code = primitive symbols (character set): phonemic; alphabet (for Spanish, with no diacritics); Japanese hiragana syllabary; %UB = percent of standard word boundaries that are also utterance boundaries = chars/word ÷ chars/line; TTR = (# word tokens in standard) ÷ (# of word types in standard); word recall = (# word tokens same in model and standard) ÷ (# word tokens in standard) × 100; word precision = (# word tokens same in model and standard) ÷ (# word tokens in model) × 100.

Table 8
Comparison of different encodings for the same corpus, shown for five corpora

Corpus description	No. of codes	Percentage Recall	Percentage Precision	Sample encodings (content same across one corpus)
English spoken (ENGV)				
Phonemic	50	65.0	62.3	lUk D*z 6 b7 wlT hlz h&t
Orthographic	26	61.9	58.4	look theres a boy with his hat
Japanese spoken (JAPV)[a]				
Roman original	24	56.0	53.8	aki chan hashigosha ga dekiru yo
Hiragana transliteration	77	54.5	53.3	あきちゃんはしごしゃができるよ
Roman transliteration	29	52.4	50.5	A-KI TIyaN- HASIGOSIya GA DEKIRU YO
Spanish spoken (SPAV)				
Without diacritics	26	50.5	46.8	ver no lo veo bien desde aqui ensenamelo
With diacritics	32	50.4	46.3	ver no lo veo bien desde aqul ensENamelo
Phonemic	32	48.1	44.6	ber no lo beo bien dezde akl ensENamelo
English written (ENGW)[b]				
Phonetic	35	47.8	45.7	AY kAAnfEHs DHAEt WHEHn fERst AY mEYd AEkkwEYntAEns wlHTH CHAArAXlz
Expanded	26	45.4	43.3	ij cownfehss thazt whehn fijrst ij mazdeh azcqurazijntazncneh wijth chazrlehs
Orthographic	26	44.9	43.0	i confess that when first i made acquaintance with charles
Japanese written (JAPW)[c]				
Hiragana original	78	38.0	37.4	このきゃんぷのためにかんがえられたのが
Roman transliteration	30	35.3	34.4	KONO KIyaN-PU NO TAME NI KAN-GAE- RARE TA NO GA
Roman adjusted	29	32.5	31.8	KONO KyaNPU NO TAME NI KAngAE RARE TA NO GA

[a] The Japanese spoken corpus was first transcribed in Roman letters, then transliterated (by machine) into hiragana syllabary, then retransliterated (by machine) into alphabet.

[b] For the "expanded" version of the English written corpus, an extra character was added to every vowel (*az*, *eh*, *ij*, *ow*, *ur*) in order to make the overall number of characters roughly equal to the phonetic version.

[c] The Japanese written corpus was first written in hiragana, then transliterated by machine into Roman letters, then the Roman version was adjusted (by machine) to bring it closer to the standard Roman orthography.

the codings are shown in Table 8). The English spoken corpus was run in standard orthography and in the 50-code phonemic scheme which was shown above. The Japanese spoken corpus used the original roman alphabetic coding, plus a mechanical transliteration from alphabet to syllabic kana characters, and then mechanically converted back to alphabet. A Spanish spoken corpus used orthography with and without diacritics, and orthography adjusted toward a more phonemic representation. Segmentation results (recall and precision) for all the corpora are shown graphically in Fig. 3, in the same order as listed in Table 8. It can be seen that the results for different versions of the same corpus were more similar than those across corpora ($R = 0.96$, $P = 0.0001$), even within the same language. Corpus characteristics which were correlated with performance in these tests were the relative number of standard word boundaries which were provided as utterance boundaries (*%UB*, $R = 0.80$, $P = 0.0006$) and the rate of lexical repetition (*TTR*; recall, $R = 0.574$, $P = 0.032$; with precision, $R = 0.632$, $P = 0.015$).

We expected that representations which were closer to the speech signal (more phone-

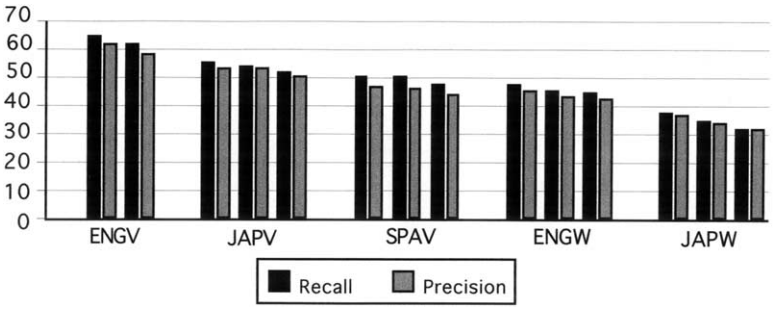

Fig. 3. Comparison of recall and precision (%) of BootLex performance using different encodings on five corpora (corpora shown here in the same order as listed in Table 8).

mic or phonetic) would perform better than more traditional – and less transparent – orthography, but we could not find a consistent effect either way. The BootLex algorithm seems to segment more or less effectively regardless of the coding scheme.

One explanation for this finding is that any transcription system which is devised by humans to "make sense" – that is, to be consistent with our knowledge of language – is sufficiently abstract and grounded in the (human) transcriber's knowledge of the language, that the difference between two or more such systems will be comparatively small and will not give a significant advantage in segmentation. In other words, BootLex can detect word-form-based regularities using any idiosyncratic representation which is both categorical and consistent. By analogy, then, we can say that early word segmentation which is based on probabilistic parsing like the BootLex algorithm would allow infants to have initially idiosyncratic mental representations during this early stage and still be successful word-finders.

• *Lexical measures.* Lexical recall and precision are similar to ordinary recall and precision, but computed on lexical types rather than on running-word tokens. In computing this measure for BootLex (Fig. 4), all word types which were used in at least one parse were considered part of the lexicon which had been "learned". The recall measure is hits (those "correctly" parsed by the model) divided by the standard, while precision is hits divided by the total number of found words. Since the number of word types created by BootLex was significantly greater than the number of word types in the standard, lexical

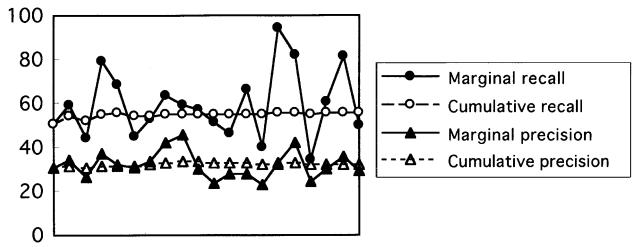

Fig. 4. Lexical recall and precision (%), both marginal and cumulative, for BootLex on the large phonemic child-directed version of the Bernstein-Ratner corpus, by 500-utterance sections.

recall was higher than lexical precision: hits accounted for more than 50% of the types in the standard (recall varied from 51.0 to 56.0%), but for only about 30% of the total types found by the model (precision from 30.4 to 33.7%). Marginal rates are also shown, calculated for just the new types in each 500-utterance section, and these rates can be seen to fluctuate considerably. The apparent stability of the cumulative figures is due to the fact that there are fewer and fewer new types in each successive section, and they have a correspondingly smaller and smaller influence on the cumulative percentage.

The quantitative measures we have discussed here show that BootLex is an effective segmenter, able to use statistical measures of co-occurrence tendency to divide the text stream into chunks that closely resemble traditional and intuitive "words". Further, its segmenting effectiveness works across a broad range of texts which vary in language, encoding, and content.

3.4. Functional characteristics of BootLex

Here, BootLex is positioned with respect to the five functional characteristics discussed in Section 2.1.

- *Build lexicon?* As it receives the input and produces segmented output, BootLex is simultaneously building a lexicon which serves as the store of its cumulative knowledge. The lexicon contains particular words (word types) which have been "learned" and are reused (as word tokens) in segmentations of new utterances as they are encountered.
- *Cluster or divide?* BootLex is a clustering algorithm, beginning with minimal "words" consisting of the individual codes which were provided at the outset. These are combined into clusters based on which groupings are seen most frequently in the input, so the "words" get longer and longer with experience.
- *Cumulative?* In BootLex, learning is incremental and cumulative, continuing as long as text is received. Possible words are added to the lexicon, and some of these candidates gradually become more and more likely as evidence for their existence accumulates – while others fall further and further from consideration if not much additional evidence is encountered, and if they consistently fail to win in competition with other candidates. Should the nature of the input change midway in some significant respect, the BootLex learning process gradually adapts itself to this change, with segmentation decisions increasingly reflecting the newer input as more of it is seen. Words could even be "unlearned", or abandoned, if a better segmentation comes into use (though there is no provision in the computer model to actually remove such words; they would just cease to appear in newly parsed utterances).
- *Feedback from output?* During BootLex processing, both inputs and outputs occur continuously and overlap with each other, each input utterance immediately followed by a segmented output. However, this is not just a mechanical alternation of input and output, but a more diffuse learning process based on feedback from output to input. After an input utterance is segmented based on the information in the current lexicon, the results of that segmentation are recorded in the lexicon as new information. Thus, the lexicon represents knowledge accumulated not from the input utterances directly, but only after their transformation into system outputs. Further, because each processed utterance becomes part of the knowledge store, the effect of a particular input/output sequence is not completely

exhausted in the same cycle, but it continues to influence decisions made many cycles later.

• *Constraints?* The "optimum-length" parameter in BootLex sets an upper limit on the algorithm's inherent tendency to produce longer and longer words as more input is processed. This constraint penalizes the creation and use of words which are longer than the "optlen" when there are suitable shorter candidates, thus giving very long words a probabilistic disadvantage without absolutely preventing their use. The actual value of the optlen parameter used in the experiments was calculated to produce about the same number of word tokens as exist in the standard, in order to realize the most optimal match with the standard and thus compare the algorithm's peak performance on various texts. While there are cognitive analogs for a general constraint on the word-lengthening tendency, which we will discuss at greater length in Section 5, we do not claim that the child actually calculates or uses such a parameter. It is an arbitrary external constraint, such as is used by many computer models.

3.5. BootLex vs. Olivier and other probabilistic models

Olivier (1968) was the first to create a working probabilistic segmentation routine. In a dissertation entitled *Stochastic grammars and language acquisition mechanisms*, he was concerned both with psychological reality and empirical effectiveness. Although Olivier (1968) has been widely cited in subsequent research, it is usually as a curiosity or exception. Batchelder (1997) was the first attempt to resurrect his ideas in a viable cognitive model. In detail, the differences between Olivier's algorithm as originally implemented and BootLex are:

Olivier	BootLex
Parse sections contain exactly 480 characters.	Parse sections contain varying numbers of characters,[19] representing utterances or sentences.
Parse sections may end mid-word.	Parse sections never break a word.
Dictionary is purged of unused candidates when it gets too full.	Dictionary is allowed to expand (now we have larger computers).
Lexicon tallies are updated for each word in the parse, and also for each word-pair candidate	Lexicon tallies are updated for each word in the parse, but word-pair candidates are not tallied until used in a parse.
No constraint; words are allowed to get longer and longer.	The parsing algorithm is constrained by an ideal average word length (optlen) and parses are penalized if they exceed it.

[19] The effect of supplied boundaries is significant. Trials with BootLex on utterances of various number of words indicate that as the ratio %UB (# utterance boundaries/# word boundaries) increases, the recall rate tends to increase, generally by an amount about one-half the amount of the ratio increase. Thus, for instance, the written English (maughamt) corpus was tested with %UB ratios of 5, 10, and 19%, and the recall rates were 43, 45, and 49%, respectively. Accuracy rates also increased comparably.

Wolff (1975, 1977) used a very similar approach. He knew Olivier's work only by a report in Brown (1973), and apparently developed the idea independently, inspired by the artificial-language-learning experiments of Hayes and Clark (1970). Wolff also began with single letters and entered in the dictionary pairs of parsed "words", with the difference that, while Olivier entered all possible pairs, Wolff entered only those pairs that passed some frequency threshold. Wolff's results, both lexicon and parsed text, highlighted the various levels of structure in the language. His lexicon also gave a count of the number of text letters which had been traversed when the item was added, which can be seen as an indicator of structural level, or strength of association. In fact, Wolff was less concerned with the segmented results of parsing or the "words" found in the final dictionary than he was with the relative strengths of association. He showed that the general trend of his routine was not only to discover boundaries, but to rank them in terms of strength, with intra-morpheme connections stronger than morpheme boundaries, which in turn were stronger than word boundaries, words stronger than phrases, and so on.

A rather offhand experiment in segmentation was carried out by Redlich (1993) as a trial and demonstration of techniques to be applied later in visual image processing. It used entropy as the evaluation criterion – specifically, the redundancy of the word distribution relative to the character distribution. Like Olivier and Wolff, he started by assuming that each letter was a word and gradually combined them to make larger and larger words. But, like the minimum description length (MDL) techniques to be discussed below, his algorithm created a lexicon by optimizing the representation of a particular and finite corpus, a strategy which we will show has serious limitations as a cognitive model.

More recently, work by Perruchet and Vinter (1998) is also relevant and interesting. Their model combined observed characteristics of perception and memory, such as attention, forgetting, and interference. The operation of the model was startlingly similar to that of BootLex, despite its different theoretical origins:

> Our account assumes that the material is mandatorily perceived as a succession of small and disjunctive chunks composed of a few primitives. This characteristic is thought to be inherent in the attentional processing of ongoing information. When a chunk is repeatedly perceived, its components are associated and form a new representational unit as an automatic by-product of the joint attentional processing of the components. The units of the language initially emerge thanks to a sort of natural selection process: among all the units that are created, only those matching the words (or parts of words) are sufficiently repeated to resist forgetting and inference. These initial representational units in turn become able to guide perception and to enter as components of other percepts, and this process continues recursively. (p. 258)

This model was able to duplicate the results of the several human experiments by Saffran et al. (Saffran et al., 1996a,b; Saffran, Newport, Aslin, Tunick, & Barrueco, 1997), which used a small number of nonsense words and syllables, but the authors correctly recognize that tests on larger and more natural sample of language are necessary before much can be said about its overall effectiveness as a model of early language learning.

4. Other model strategies

A number of computational models of segmentation using other paradigms have been reported recently, falling into three main groups:

(i) Three connectionist networks
(ii) Two algorithms using the minimum description length principle
(iii) Two algorithms based on a formal statistical model called "Model-based dynamic programming" (MBDP)

All these models interpret the cognitive problem of word segmentation similarly, as discussed above, but there are significant differences among them in goals and methods. Each group will be described and discussed, particularly with respect to the five functional characteristics introduced earlier, and then compared with each other and with BootLex.

4.1. Connectionist models

Three groups of researchers have created connectionist networks to study segmentation from a developmental point of view.[20] These models do not create a lexicon or directly refer to words as objects; they learn phonotactic regularities rather than particular words. Table 9 gives more detailed information about the architecture and training parameters for each network.

4.1.1. University of Rochester

Aslin, Woodward, LaMendola, and Bever (1996) of the University of Rochester used a simple feed-forward network which was trained to predict utterance boundaries from phoneme and pause information. Each of 44 phonemes was coded as 18 articulatory binary features of a type now traditional in linguistics (sonorant, coronal, round, etc.), and the input stream was presented as moving triples – phonemes 1 2 3, then phonemes 2 3 4, etc. There were thus 55 input units – three phonemes ($3 \times 18 = 54$), plus a unit indicating the presence of an utterance boundary. The output of the network was a single unit that indicated the presence or absence of an utterance boundary. A very small corpus of child-directed speech was input, and the state of this output unit was noted after each input sequence (once for each phoneme). As expected, the highest activation of the output unit occurred at utterance boundaries, but there was also greater activation on average at lexical (word) boundaries than between word-internal phonemes. This meant that the network was "learning" something about word boundaries by observing the beginnings and ends of words at utterance boundaries.

4.1.2. University of Southern California (USC)

At USC, Christiansen, Allen and Seidenberg (Christiansen et al., 1998) tested the effect of

[20] Two additional connectionist models have been reported, both aimed at replicating the results of the Saffran et al. experiments with nonsense syllables: Gasser and Colunga (1999) and Dominey and Ramus (2000). They will not be treated here because, like Perruchet and Vinter (1998), discussed in Section 3.5, they do not report tests on natural language inputs.

Table 9
Comparison of three connectionist networks designed for word segmentation

	Rochester	USC	Edinburgh
Training corpus			
Transcribed speech directed to	infant	infant	adult
Size in words	1300	25,000	300,000
Size in segments (characters)	<5000	73,947	1,000,000
Average segments per word	~3.75[a]	3.00	3.30
Average segments per utterance	10–15	9	n.a.[a]
Token/type ratio (TTR) for words	≤10	30	≤25
Coding			
# segments	~44[a]	33[b]	45
# binary features (bits)	18	11	9
Training process			
Iterations	2–3	1	2
Total segments input	≤15,000	73,947	2,000,000
Total bits input	≤270,000	813,417	18,000,000
Next-segment training?	yes	yes	yes
Utterance boundary training?	yes	yes	no
Net architecture[c]	FF (window of 3)	SRN	BPTT
Input units	54 + 1	11 + 1[b]	9
Hidden units	30	80	60
Context units	–	80	60
Output units	1	36 + 1[d]	27

[a] ~ indicates an approximate estimate, where precise figures are not reported; n.a., information not reported.
[b] Although 36 phonemes were intended, in fact only 33 phonemes were input. In three cases, identical feature codings were used for two different phonemes: /I/ and /V/, /e/ and /i/, and /k/ and /g/, the latter as erroneously coded (Christiansen et al., 1998, p. 266).
[c] FF, feed-forward; SRN, simple recurrent network; BPTT, back-propagation through time.
[d] Input was by features, output was local coding (one unit for each phoneme).

combined cues to word segmentation – sequential phonotactic structure, utterance boundaries, and word stress information. They used a simple recurrent network (SRN) design (Elman, 1990), which combines new input with a "memory" of its own state in the preceding cycle, recorded in "context units". A fairly large corpus was used, from the CHILDES database (MacWhinney, 1995), of speech directed to infants of 6–16 weeks of age. The corpus was edited to avoid unusual or idiosyncratic words, deleting about one-quarter of the text for this reason, and then transcribed using 33 phonemes, which were represented to the network as 11 binary features. Utterance boundaries were represented as a single binary unit. After supervised training, for both the next phoneme and boundary unit status, outputs at test showed a much higher mean activation level of the boundary unit at lexical boundaries than word-internally. The results were very similar to those of the Rochester group; see the comparison in Table 10. The difference between the low and high activation is about the same in both studies, but in one the "lexical boundary" level is closer to "utterance boundary", while in the other it is closer to "word-internal". Since there was no lexical boundary

Table 10
Comparison of two SRN networks: activation of the output unit at three types of position

Mean boundary unit activation at:	USC[a]	University of Rochester[b]
Word-internal (not a boundary)	0.05	0.08
Lexical (word) boundary	0.25	0.18
Utterance boundary	0.34	0.35

[a] Including the stress component, estimated from Fig. 4 in Christiansen et al. (1998, p. 242).
[b] Estimated from Fig. 8.5 in Aslin et al. (1996, p. 129).

information given to the nets, the activation of the boundary unit at those points can only be a generalization from utterance boundary information. Christiansen et al. (1998) thus claim that their net has a greater ability to generalize than was found by Aslin et al. (1996).

4.1.3. University of Edinburgh

A group at the University of Edinburgh (Cairns, Shillcock, Chater, & Levy, 1994, 1997; Shillcock, Cairns, Chater, & Levy, 2000) used the "back-propagation through time" (BPTT) network architecture, a computationally more intensive method than the SRN, and one which "allows the error signal to be back-propagated through longer stretches of time than in the SRN… The task is to echo the current slice of input [phoneme], to remember the previous, and to predict the next… [B]oundaries are proposed at peaks in the error score on the prediction output units" (Cairns et al., 1997, p. 133f). This group used a very large corpus, the London-Lund Corpus of adult speech, converted to a phonemic feature-based representation using 45 phonemes and nine features (Shillcock, Lindsey, Levy, & Chater, 1992), but even the best results were poor. The authors concluded that "…the network does not segment more whole words from the test stretch than it would by chance" (Cairns et al., 1997, p. 140). Although other differences, such as adult-directed rather than child-directed speech and an unusual feature scheme, may also have contributed to the lackluster results, it seems likely that the lack of boundary training was primarily responsible. While utterance or pause information was represented in the input to the other two networks, no boundary information at all was supplied here.

4.1.4. Functional characteristics of connectionist models

• *Build lexicon?* None of the networks produces a lexicon.[21] For networks, the store of knowledge learned is in the hidden units of the network, which mediate between input and output, not a lexicon. Thus, we can say that what the networks are "learning" is different in principle from what BootLex and the other models to be described here are "learning": the networks' goal is not to discover and learn words, but to acquire word-finding skills. They model the learning of phonotactic characteristics of words by observing those phoneme (or featural) clusters which tend to occur at utterance boundaries and then generalizing these

[21] While this statement is restricted to the networks described here, I do not know of any connectionist model which creates "lexical entries" from subword units in the course of processing and provides for them to be the objects of activation, though some, like TRACE (McClelland & Elman, 1986), can recognize words which are "built-in" to the architecture of the model as objects from the outset.

to discover the most likely utterance-internal word boundaries. The end result of the process is thus not a store of learned words, but knowledge of phonotactic regularities of the language.

• *Cluster or divide?* While BootLex clusters, the networks divide. While segmentation is a side-effect of the clustering process, with boundaries arising between clusters, the converse occurs in the networks – points of unusually low cohesion are treated as word boundaries and words arise as a side-effect of this dividing process.

• *Cumulative?* This is the only one of these five functional characteristics for which the connectionist models and BootLex are similar: both proceed by small increments toward the learning goal. If the input should change, both will gradually reflect the new input in the output.

• *Feedback from output?* While the connectionist models learn incrementally, input and output are not interleaved, which is a precondition for feedback of output to input, so the nets cannot make use of such feedback. In these models, input events as a group are separated in time from output events. The network is instructed either to "learn" (training phase) or to "perform" (testing phase), but never both concurrently. During the training phase, a data stream is presented and each increment of data causes a slight modification of the internal weights. The mechanism of learning is the network's prediction about the next item of data (based on its observations so far), which is "corrected" when that item of data arrives, and the gap between the prediction and the correction becomes part of the training process. In this sense, temporally "local" outputs feed back to affect learning, but a distinction can be made between these local outputs, which are not recorded or reported, and those "global" outputs which are the goal of the learning process. After training is deemed complete, the network is instructed not to further modify its representations, but to preserve its internal state and continue to make predictions about the incoming data stream, using its accumulated knowledge. It is these predictions that are captured by the researchers and interpreted as global outputs, the end results of the learning process being modeled. Thus, the global inputs and outputs of the model's learning process are not overlapped, and the reported outputs do not affect the learning process, since it is already complete.

• *Constraints?* Networks have a number of external constraints, beginning with the various technical decisions involved in designing and running the network (as detailed in Table 9). In all three networks reported here, a crucial external constraint was the boundary output threshold which was needed to convert the outputs from graded values (a range of activation levels between 0 and 1) to discrete data (word boundary: yes or no). Those outputs which had activation levels exceeding this criterion threshold were reported as word boundaries. The Rochester and USC nets both trained a special output unit by matching it to utterance boundary information, and used as a threshold the mean activation level of all these output units. The Edinburgh group used a cross-entropy error measure (0–1) of the output of a phoneme prediction task, and a threshold that "maximized the mutual information" (Cairns et al., 1997, p. 134). Using that cutoff value, the network greatly underdivided the corpus, with the number of inserted boundaries only 35% of the standard number. It is not clear from their report whether a different cutoff value would have yielded better results.

4.2. Minimum Description Length

Minimum description length refers to an evaluation criterion (Ristad & Thomas, 1995) that looks for the shortest combined length of both the data and its analysis – that is, the encoded text and the lexicon derived from it. The MDL technique was originally developed for data compression, but it is also effective as a segmentation technique, since it finds the best representation of a text as a series of recurring small units.

Carl de Marcken (1995, 1996a,b) used the MDL metric to segment text, though he made no claim to be modeling cognitively realistic processes. He used an optimized approach to avoid testing every possible lexicon, and his results on a variety of corpora were quite remarkable as a testimony to both the amount of structure in language and the ability of the MDL algorithm to extract it. However, his algorithm discovered structure at various levels rather than creating a single series of words. For example (de Marcken, undated):

```
[s[h[or]]t] [c[ut]]
[[[ju][st]]_] [a[s_]]
[m[er]] [e[[ly]_]]
```

This kind of graded segmentation can be an advantage for some purposes, but it makes a quantitative evaluation of segmentation performance more difficult.

Brent and Cartwright (1996) used the MDL algorithm for segmentation in a model which exhaustively tested every possible segmentation of a text in order to select the best one – the one with the shortest combined length of the lexicon and the coded representation of the text. This process is equivalent to choosing as "words" those combinations which occur together most frequently and in the greatest variety of contexts. Brent and Cartwright (1996) ran experiments on the same phonemic corpus that we have described above (Cartwright & Brent, 1994). The results reported by Brent and Cartwright (1996) were based on a tiny part of this corpus, since the algorithm was so computationally intensive that it could not handle a larger amount of data.

To characterize the MDL algorithm reported by Brent and Cartwright (1996) in terms of our functional characteristics, it produced a lexicon and used a clustering process. Further, like other MDL algorithms, it was constrained by the tension between the length of the word type and its frequency. Minimizing the combined length of the coded text and the lexicon results in the avoidance of two extremes: long words which occur rarely, and very short words which occur too frequently.

However, in the MDL-based algorithm the timing of input and the consequent learning were not incremental processes. All input was accepted as a single event and then processed repeatedly, at the end producing a lexicon and a segmented version of the input text.

> The search algorithm… operates in batch mode, reading in the entire input before segmenting any part of it. Clearly, children do not work this way. Rather, they add to their lexicons incrementally as new input becomes available. (Brent & Cartwright, 1996, p. 117)

Since the input and output did not overlap at any point, no feedback from output to input

was possible. Further, the MDL principle assumes a bounded set of data to work on, with no possibility of the input changing or being extended during the learning process. This restriction is so unlike the ongoing process of human word-learning that, despite the fact that Brent and Cartwright (1996) is the most widely cited simulation of infant word segmentation, it is hard to consider MDL a successful cognitive model.

4.3. Model-based dynamic programming (MBDP)

Brent (1999a) presented another, and truly incremental, algorithm called MBDP-1, and a very similar one is reported in Venkataraman (2001).[22] These models are similar to BootLex in many respects. Like BootLex, they build a lexicon, and select words from the lexicon to parse utterances, keeping a tally of the number of times each word appears in a successful parse. This "parse frequency" is used to evaluate succeeding parses, thus providing for output to feed back into the learning process. However, while BootLex is a "clusterer", MBDP is a "divider". The MBDP models begin with an empty lexicon and create new "words" in two ways:

(i) When an utterance cannot be segmented, by default the whole utterance is added to the lexicon as a single "word". The first utterance in the corpus, therefore, always becomes the first item in the lexicon – the first "word".

(ii) During the process of segmenting an utterance by recognizing one or more lexical items in it, any unrecognized portion is considered to be one or more new words. For instance, suppose the utterance *hello* is encountered and entered in the lexicon (because it could not be segmented). Later, the utterance *hellojoe* occurs, and the listed word *hello* is recognized within it. The residue of this second utterance – *joe* – is then added to the lexicon as a word.

As discussed above for BootLex, it is often the case that a given utterance can yield a number of different parses by using different items from the lexicon and/or by segmenting residue strings differently. The method of determining the best parse in MBDP is similar to BootLex: compute the estimated probability ("preference value") of each word in each possible parse and select the parse with the largest product of these preference values. In the MBDP algorithms, there are two formulas for calculating preference values: one for lexicalized words (those previously seen) and one for novel words.

- The preference value for a lexicalized word is approximately its "parse frequency" – the number of parsed occurrences for its type in the lexicon, divided by the total parsed word tokens so far.
- The preference value for a novel word is its likelihood purely as a string of phonemes, estimated as the product of individual phoneme probabilities. Each phoneme's probability is estimated as its frequency of occurrence in the list of word types in the lexicon.[23] The algorithm reduces the preference value by a constant factor for each novel word, so a given

[22] Because the algorithm in Venkataraman (2001) is much simpler than, and the results are virtually identical to, Brent (1999a), my description of the algorithm leans more heavily on the former paper.

[23] Venkataraman (2001) tested three bases for phoneme frequencies: types (frequency of phonemes in the lexicon), as used by Brent (1999a); tokens (frequency of phonemes in the full text seen so far); and a flat (equal) distribution. There was a significant performance disadvantage to the flat distribution, but only a very small advantage to the type-based over the token-based frequencies.

phoneme string will always score better as a single word than as several. However, short novel words will generally receive a higher score than longer ones as the product of phoneme frequencies, since products of fractions tend to decrease with the number of terms (cf. Table 3). This means that parses which leave shorter residue strings will be preferred in the overall calculation.

Thus, the MBDP algorithms and BootLex both model an incremental and cumulative learning process which creates a lexicon. Input and output are overlapped, with feedback from the output to the lexicon. The major difference is that in MBDP "words" start out long and get smaller, whereas in BootLex they start small and get longer. The occurrence and co-occurrence frequency patterns of phonemes and pauses in the text, which are central to the BootLex learning strategy, are less important in MBDP. The MBDP algorithms learn words as whole utterances with no internal structure, and then use a "divide" strategy to recognize previously learned chunks in order to segment new utterances.

• *Principled vs. intuitive.* Constraints used in the MBDP algorithms include several mathematically complex and computationally intensive parameters. The implementation in Brent (1999a) incorporates several heuristic approximations which use externally derived parameters, such as particular distributions on the positive integers. Venkataraman (2001, p. 361) says of the model in Brent (1999a) that it "requires the explicit calculation of the probability of the [current] lexicon in order to calculate the probability of any single segmentation", rather than simply using dynamically recorded frequencies. The Venkataraman (2001) model requires only the calculation of phoneme frequencies in the current lexicon.

MBDP is based on a formal statistical model as defined by a set of complex mathematical formulas. Such mathematical foundations are typical of many computer models, and they represent an issue which affects scholarly exchange among members of the modeling community. Computer models of language acquisition and processing like those discussed here are potentially the offspring of three quite different research traditions – linguistics, psychology, and computational linguistics. Computational linguistics has close ties to computer science and artificial intelligence, in which a "formal model" consists of a system of mathematical relationships which are intended to express hypothesized relationships among the processes being modeled (Charniak, 1993). Such models have certain predictable properties which have been well documented, and are thus good research vehicles that can be developed and compared in an orderly fashion. The MBDP algorithms are statistical models in this tradition, and the connectionist networks also have a mathematical basis.

On the other hand, BootLex does not have such a mathematical justification. It would thus be termed "ad hoc" or "heuristic" by those who work closer to algorithmic theory than to psychological ones. However, mathematical notation, which is indeed the most precise description of a model's operation, is frequently incomprehensible to all but specialists in that field. The necessary "leap" from a cognitive hypothesis to its mathematical expression may thus not be easily available for review. A simpler and more intuitive algorithm can be easier to comprehend, and thus easier to evaluate as plausible or not, even though it is not guaranteed to be mathematically consistent.

The ideal solution might be an approach like that of Perruchet and Vinter (1998), discussed above, in which psychological principles form the basis for a computational

model, which would then be more intuitive than mathematically-based models but more principled than a completely heuristic one.

4.4. Quantitative comparison of models

Representative quantitative results reported for the models discussed above are summarized in Table 11. Both cut and word measures are shown wherever available, in order to provide as many points of comparison as possible despite different reporting practices. Evaluating the correctness of cuts is less stringent than evaluating words, since a correct word requires that at least two cuts be correctly placed. Table 11 shows that BootLex does as well as or better than the others by both measures.

MBDP are the most recently reported models, and their performance is much better than earlier models, competitive with BootLex. Since both the MBDP models and BootLex report results on the Bernstein-Ratner(–Cartwright) corpus, they can be directly compared. Fig. 5 shows both precision and recall plotted for successive sections of 500 utterances in the corpus, plus a smoothed version for easier interpretation. On this corpus of almost 10,000 utterances, the two algorithms are performing at almost identical levels by the end of the learning process.

Neither Brent (1999a) nor Venkataraman (2001) report lexical precision and recall information like that shown for BootLex in Fig. 4. Brent (1999a) does give a statistic which he calls "lexical precision", but which is computed differently. The word types parsed from a given 500-utterance section of the corpus are compared with all word types in the entire standard corpus up to and including that section. This statistic will generally be greater than the marginal and cumulative precision we showed in Fig. 4, because it uses two different text spans to define correct word types.[24] Applying Brent's formula also to BootLex (Fig. 6), we see that BootLex consistently outperforms MBDP on this novel measure.

The performances of MBDP and BootLex are thus very similar. However, it should not be surprising to find that quite different techniques can be equally successful in harnessing the inherent statistics of language:

> When two computationally powerful systems are given the same set of input data, they both extract every bit of data regularity from that input. (MacWhinney, 1993, p. 295)

Taken together, these results demonstrate that there is a good deal of information in the statistical structure of the speech signal that could help the infant locate initial word boundaries, and that quite different techniques succeed in extracting this information.

• *Effect of kind of language input.* BootLex and MBDP performed very similarly on the

[24] More exactly: let "precision" be defined as the number of correct words found in some area x divided by the count of all word types found in the same area, and let "correct words" be defined as those word types found which also occur in the list of standard word types for some area y. Then, for marginal precision as in Fig. 4, both areas x and y are the current section, and for cumulative precision in Fig. 4, both areas x and y are the entire corpus up through the current section; but for the Brent (1999a) measure, area x is the current section while area y is the entire corpus up through the current section. Since the corpus so far has more word types than the current section alone, relatively more of the words found will be likely to match and be deemed "correct" in Brent's measure.

Table 11
Comparison of reported quantitative performance of segmentation models, measured as cuts or words matching the respective standard

Model type and project	Training set (phonemes)	Evaluating cuts		Evaluating words	
		Recall	Precision	Recall	Precision
BootLex	96,000	83%	81%	68%	67%
Connectionist					
Rochester	≤ 15,000	62	74	–	–
USC[a]	74,000	71	66	40	37
Edinburgh	2,000,000	21	60	–	–
MDL					
Brent and Cartwright (DR)	1,520	–	–	47	41
MBDP					
Brent (1999a)[b]	96,000	–	–	69	67
Venkataraman (2001, 1-gram) [b]	96,000	–	–	70	68

[a] The network trained on stress information in addition to phonemic and utterance boundary information achieved slightly higher levels: 74% recall and 70% precision on cuts, 45% recall and 43% precision on words.
[b] From Table 2 in Venkataraman (2001); no comparable figures were given in Brent (1999a).

same inputs. However, only BootLex was tested on a number of other texts, using three languages and varying the encoding and other characteristics of the input as well. These tests showed that the BootLex algorithm achieved significant segmentation results on a variety of corpora, but it performed better on the spoken corpora tested. The probable reasons for this are not hard to guess. Spoken corpora have shorter utterances, which give more information about word boundaries by providing relatively more utterance boundaries, and shorter words, which tend to involve less of a confound between the phonotactics of syllables and words. When longer and more complex words are encountered, phonotactic strategies which (in English) are largely syllable-based will be likely to misdivide words like *seg ment* and *mis read*.

The MBDP algorithms reported results only on a spoken child-directed corpus, but it would also seem likely to do less well with more complex texts, perhaps tending to incorrectly segment out embedded words when faced with multimorphemic words like *cat er pillar* or *house keeper*. Although it has been shown that short utterances and high repetition rates are typical of child-directed speech, there is also ample evidence that "the abused and neglected children of the world, regardless of their other difficulties, adequately acquire the language of their communities" (Gleitman & Bloom, 1999, p. 435), so an algorithm which requires input to have particular characteristics would not be a satisfactory model of this learning process. MBDP, as well as the connectionist nets, should be tested on a wider range of language input.

• *Distribution of what?* Before proceeding to more general cognitive issues, we should observe that, although all of the models used distributional information to make segmentation decisions, they did not all attend to the distribution of the same elements. As Redington and Chater (1998, p. 145) point out:

> To state that a particular aspect of language is acquired from distributional informa-

tion has, by itself, no more explanatory power than to say that a particular aspect of language is known innately.

Two kinds of distributional information have been discussed. One is the tendency of individual codes (segments) and code clusters to occur at utterance boundaries, and the second is their tendency to form within-utterance clusters.

Both of these were recorded by two of the connectionist networks, but the Edinburgh net used only segmental information, and no utterance boundaries. Since all the networks coded phonemes as constellations of features, it is possible that the models recognized both the actual code clusters and also clusters with similar features, but the reports do not quantify the extent to which such generalization of features actually occurred, and other code representations were not tested, neither alternate feature schemes nor local representations of the phonemes.[25]

BootLex was influenced by segment/boundary co-occurrences, but only by that portion of segment/segment clusters which happened to occur in the parsed output and thus flowed into the lexicon. To illustrate, consider the parsed utterances in Table 2. Although there are 16 occurrences of the cluster 'dog' in the text, only five will be directly counted in the lexicon (plus one for the first "trial entry", which was created from the parsed pair 'd' and 'og' in line 3). All other instances of 'dog' were divided, or became part of a longer word. Thus, while the networks continued to collect information about all trigrams throughout their processing, the BootLex algorithm selected some combinations, entered them in the lexicon, and then used this stored knowledge to shape its processing of succeeding inputs. Utterance boundaries were important because they represent fixed word boundaries and, as stated earlier[19], the proportion of utterance boundaries relative to word boundaries has a significant effect on BootLex's performance.

The MBDP algorithms also made use of those clusters which found their way into the lexicon after parsing, though these were a still smaller subset than in BootLex. MBDP depended more heavily on the full character string enclosed between two utterance boundaries, which became the starting lexical entries. Parsing a residue string into new "words" relied on the frequencies of individual codes in the lexicon (a list of word types) rather than on code clusters previously collected from the text. Thus, although both BootLex and MBDP hypothesized lexical entries and then let future parses determine which ones survived, their hypotheses were based on quite different information sources.

5. From computer model to infant cognition

The previous two sections have presented the BootLex algorithm and compared it in some detail with two other groups of models, both in terms of quantitative performance and more global characteristics of design and function. In this final section, we examine

[25] Aslin et al. (1996) tested a local coding scheme (one unit for each different phoneme), and reported that learning of word boundaries failed to occur. However, this group used a very small corpus. It is possible that phonemes might have performed as well as the lower-level features if comparably more data were available; that is, the net itself might perform the featural analysis if sufficient training were given.

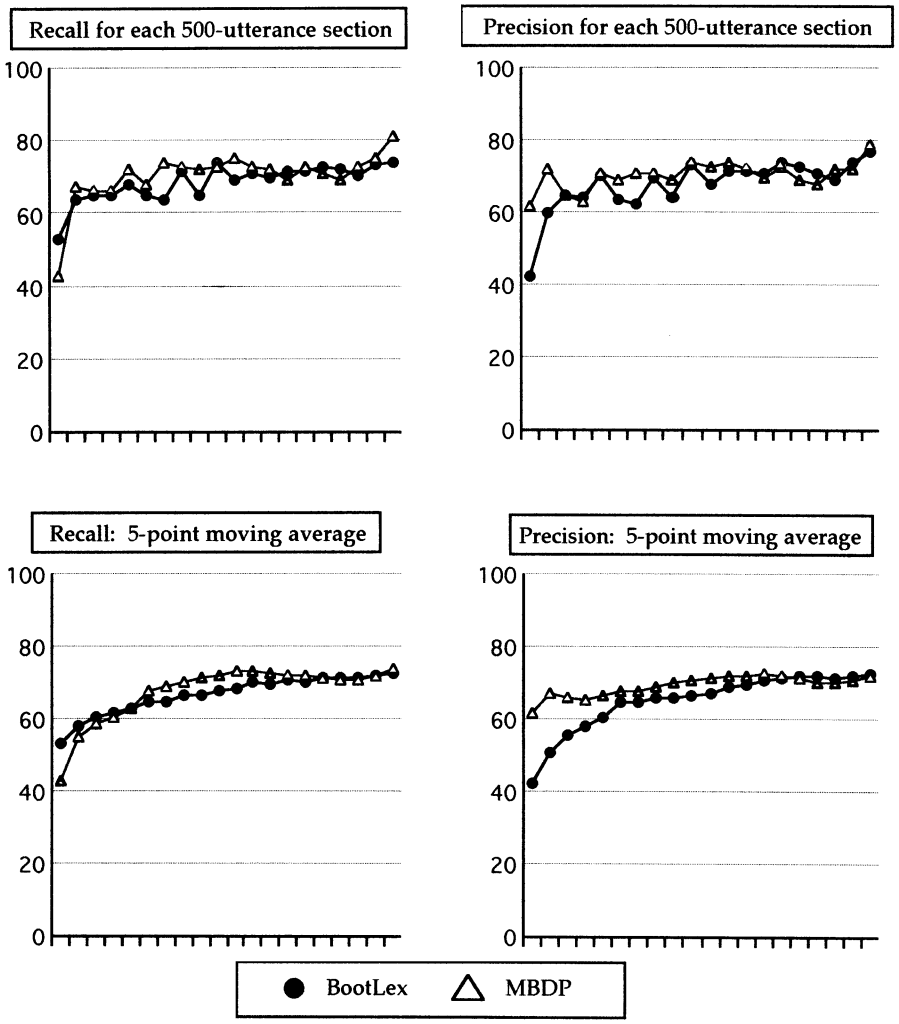

Fig. 5. Quantitative comparison of BootLex and MBDP algorithms by 500-utterance sections [MBDP data points estimated from graphs published in Brent, 1999a; Venkataraman, 2001].

the claims of these computational models to be cognitive models – to go beyond the purely engineering goal of an end product that is comparable with that realized by human infants, and also demonstrate similarities in process.

The relation between any model and its original is at best a loose analogy, and these single-purpose computer programs are very far from being true models of human cognition, even for this limited task. However, it may be instructive to look at the extent to which, in each case, human attributes can be seen in the machine's workings, not in the detailed mechanisms that are used, but in their processing characteristics. The following

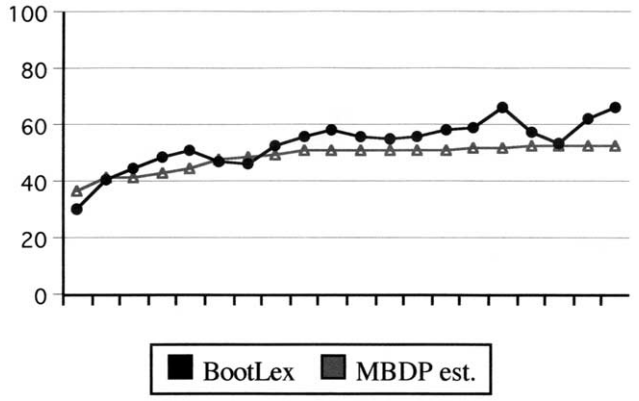

Fig. 6. Lexical precision (%) by the Brent (1999a) measure for BootLex and MBDP algorithms by 500-utterance sections [MBDP data points estimated from Fig. 5 in Brent (1999a)].

subsections discuss some cognitive implications of the functional characteristics which were introduced in Section 2.1.

• *Issues of timing and feedback.* The infant's learning process appears to be a continuous and ongoing one, with "input" being received from birth and "output", or learning, also taking place incrementally from the beginning. Thus, we have considered that the minimum requirement of a useful learning model is that it also receive language and learn from it in a continuous and cumulative fashion. One group of models discussed here, those based on the MDL algorithm, was unable to demonstrate such a continuous and incremental learning process, and was omitted from further discussion.

Babies also appear to interleave inputs and outputs. This capability was lacking in the connectionist networks, which partitioned inputs and outputs into separate processing phases ("training" and "testing"). Although local outputs which represented learning proceeded concurrently with input, no results issued from the model during the input phase. Thus, when networks model cognitive processes which would normally alternate input and output, we must interpret them loosely, as more of an abstraction from the human original.

Because the networks did not provide for overlapping of inputs and outputs, they had no possibility of learning from the feedback of output to input. Both the BootLex and MBDP models used this additional source of information – one which is also likely to be used by language-learning infants as they engage in an ongoing dialog with their environment, testing hypotheses and modifying their knowledge based on the response.

• *The modeling goal: words or rules?* BootLex and MBDP both built data structures in memory (lexicons), while the network models extracted co-occurrence regularities for later application. This is the difference between "learning how" and "learning what" – the networks were learning how to segment, while BootLex and MBDP were discovering particular words and remembering them for later use. In the lexical approach, a segmentation of *thebunny* into *the* | *bunny* means that *the* and *bunny* have been identified as words. In the regularity-oriented approach, it implies that the occurrence of a boundary between *e*

and *b* is a more likely linguistic event than one at any other point in the string. For instance, *theb* and *thebu* are determined to be less likely linguistic units than *bun* or *bunny*. The two approaches model different cognitive assumptions – either focused on learning how to recognize patterns of sounds typical of words and word boundaries, or on remembering particular words.

However, when we turn from models to babies, it is difficult, perhaps impossible, to sharply distinguish between the object of learning and the path by which it is reached. In the absence of any guidelines, learners may make random guesses and then extract patterns from those guesses which are successful. Memorization of successful guesses (words) proceeds hand in hand with observation of which cues most reliably lead to success (regularities). As more and more words are learned, more and more regularities that led to their hypothesis are confirmed. "Rules", in the sense of probabilistic regularities, are thus both the cause and the result of successful word-finding. The words themselves are consciously "known", while the "rules" or cues that led to successful learning are implicit knowledge, beneath conscious awareness (Cleeremans, 1993; Reber, 1993).

In this way, infants are undoubtedly engaged in both "learning what" and "learning how" simultaneously. Ongoing research in several laboratories continues to try to tease apart the relative contributions of different cues at various stages in the child's development (Hauser et al., 2001; Johnson & Jusczyk, 2001; Mattys & Jusczyk, 2001a,b; Saffran, 2001).

• *Cluster vs. divide?* Both clustering and dividing are plausible cognitive strategies, given different starting assumptions. Indeed, the child undoubtedly uses both at various points, and different children may use different strategies, depending on their learning styles (Peters, 1983) and the kind of input they receive. However, there are open questions about their respective uses as first segmentation strategies.

The networks' divide strategy consisted of locating points of lowest cohesion between phonemes in the speech stream. This strategy requires that a stream of speech longer than a word have a mental representation which can then be divided into smaller parts. To aid such a process, the child might use prosodic features like stress, intonation contours, and pauses to subdivide long utterances into pieces small enough for computation. Since the network models did not attempt to use prosody for a preliminary subdividing, it is likely that their success depended on the use of a corpus of child-directed speech with short utterances. In fact, the Edinburgh group used adult-directed speech and did not have good success.

In the case of the MBDP models, their use of a very different divide strategy also may have presumed sufficient short (one-word) utterances to bootstrap the rest of the lexicon. A recent report by Brent and Siskind (2000) points out that not just "utterances", but also mid-utterance words which are bounded by pauses in the speech stream will serve as well. That paper reported that when child-directed speech was transcribed with attention to the actual length of pauses, it contained more of such instances, and in more variety, than had previously been documented. However, both the networks and the MBDP models need to demonstrate that they are not dependent on particular kinds of input for successful segmentation.

Secondly, neither of these divide strategies can explain the experimental results of Saffran et al. (1996a,b), which showed that "when confronted with long stretches of speech containing no familiar words and no utterance boundaries, infants can still discover novel

words" (Brent, 1999b, p. 299).[26] By contrast, BootLex, which also requires utterance boundaries at intervals, can succeed in discovering words in this task by breaking the stream arbitrarily at various points to create "utterances". Randomly placed boundaries will incorrectly divide a few words, but not sufficiently frequently to interfere with the learning process, which also includes observation of within-utterance segment clusters. The experimental babies, too, must have done something like this – observe the relationships among small chunks of sound – in order to infer boundaries.

Of the models presented here, BootLex is the only viable example of a clustering algorithm. Although the mechanical nature of a computer algorithm is evident in the implausibility of some of its early one-unit "words" (like 't' and 'h'), the general approach of starting with small bits and building up is a cognitively plausible strategy for even very young infants, doing what Perruchet and Vinter (1998, p. 249) call "chunking… a ubiquitous phenomenon, due to the intrinsic constraints of attentional processing". One further objection that may be raised is that, as Redington and Chater (1998) point out, a clustering strategy taken literally predicts that children will learn small words before longer ones, presumably including the function words, which tend to be among the shortest words in a language. This may be answered by saying that, though infants do not produce function words, it is possible that they do comprehend them (Gerken, Landau, & Remez, 1990).

• *Constraints?* Any computer model requires that the target problem be highly structured, defining in fair detail both data and processes and, for an incremental model, determining the beginning and end points. All of these design decisions serve to constrain the problem (the "search space") so that it is finite and soluble. When there is no motivated or principled way to make these decisions, they must be made arbitrarily. Since one goal of all of these projects was to demonstrate just how useful their respective sources of statistical information might be for segmentation, it is likely that optimal values were selected for all parameters so as to achieve maximal results in the experimental situation.

For human learning and growth also, there must be something to give that growth direction and structure. So we must ask, with Locke (1996), "Why do infants begin to talk?" What forces drive their learning? We do not know much about the cognitive structures which are necessary to shape the infant's learning of language, neither that portion which must be taken in completely from the environment, nor that which may be an actualization of innate capacities. In both cases, those constraints which succeed in directing similar learning in models can perhaps give us hints about what kinds of constraints to look for in human babies.

One candidate for a constraint on babies' learning is an innate tendency to attend to certain stimuli in the environment, such as human voices (Aslin, Jusczyk, & Pisoni, 1998, p. 158, and references therein). Another possibility is restricted attentional and computational resources, or what Jusczyk (1998a, p. 212) calls "the size of the learner's processing window". Such limitations are widely assumed to be developmentally determined, and the

[26] One anonymous reviewer has pointed out that several issues remain unresolved from this study. Critics have charged that the task is an artificial one and, though work by Johnson and Jusczyk (2001) replicated the results, the same paper found that infants ignored statistical cues when these conflicted with other information. Second, infants may be recognizing not "words" but merely recurring sound patterns (Mattys & Jusczyk, 2001b); this issue was reconsidered in Saffran (2001).

"less is more" hypothesis (Goldowsky & Newport, 1996; Kareev, 1995; Newport, 1990) suggests that they may also be cognitively advantageous. We will not attempt to match one for one such plausible constraints with each model's constraints, but we will ask what in particular acted to constrain the incremental learning process in each model.

For the networks, we know that learning was incremental because that is the way connectionist models function, but the direction of the incremental process is not clear. For instance, would more training have tended to produce more boundaries, or just to mark the same ones more strongly? The end of training is the ultimate limit of the learning process, but the reports do not state how the end of training was determined – how the number of iterations of the input corpus was decided – and they describe very few tests of variations in the training or testing parameters, or the input corpus.

For the MBDP algorithms, we can logically assume that words tend to become shorter and shorter as larger chunks are divided into smaller ones. However, none of the external pieces of data provided – such as the number of words currently in the lexicon and the frequency of individual phonemes in the lexicon needed by Venkataraman (2001), or the more elaborate derivations of Brent (1999a) – seemed to be acting as a constraint on such a tendency. Perhaps this logical tendency was controlled implicitly by tensions among the various processes within the algorithm itself, though the authors do not make such a claim. If so, this would be perhaps the most cognitively plausible kind of constraint. On the other hand, the tendency to find words within words may, as we mentioned above, have been minimized by using input with a simple vocabulary, and some further constraint might become necessary with more complex vocabulary.

The BootLex algorithm's tendency to produce longer and longer words as its process continued was curbed by an "optimal length" parameter. As we discussed in Section 3.4 above, this is an arbitrary external constraint which was optimized separately for each corpus tested, in order to compare their performances. Since BootLex was the only model that reported results on more than one corpus, we do not know whether the other models would also have had to alter their operation in some way to achieve good results on widely differing corpora.

However, how a particular value is assigned to this parameter can be separated from the fact that such a constraint is necessary for BootLex's learning process. In the case of BootLex, the optlen constraint serves to curb the model's tendency to produce longer and longer "words" as more language is experienced. We claim that there is a cognitive analog for such a constraining influence in the child's segmentation process – the short time period over which the bootstrapping process is hypothesized to operate, as described in the following section.

• *A developmental stage or a lifelong process?* A final difference among the models is their relation to the developmental process. BootLex was designed to model the early bootstrapping process of the infant who builds a small first lexicon without much linguistic knowledge. We hypothesize that this temporary nonlinguistic process – a purely probabilistic strategy – comes to a natural end when the lexicon contains sufficient linguistic information to enable the child to forge more sophisticated tools. This hypothesis is supported by at least one estimate of an early "comprehension spurt" between the ages of 0;11 and 1;3 (Harris & Chasin, 1999).

The other models do not share with BootLex this view of their task as a temporary or

nonlinguistic process, but see it as the learning of skills that become a permanent part of the linguistic repertoire. The networks model a phonotactics-based segmentation process which is seen as continuing to assist speech processing in adulthood (Cairns et al., 1997; Christiansen et al., 1998; Shillcock et al., 2000). The MBDP models are based on locating previously defined words in an utterance and then also treating as words any residual chunks. Brent believes that this strategy is learned early and persists throughout the lifespan, and Dahan and Brent (1999) demonstrated experimentally its use by adults. Thus, both the networks and MBDP are modeling the learning of permanent skills, though different ones.

This issue of continuity arises in all developmental research. It requires that, in addition to estimating the contribution of various information sources to cognitive performance at a certain age, we must also be alert to the possibility that the same information may be used in substantially different ways at different stages of development. Further research with both infants and adults is needed to ascertain just how the information sources modeled here actually aid segmentation at various life stages.

6. Conclusion

A new model, BootLex, was shown to be a conceptually simple and effective segmentation procedure. Based on observation of frequently appearing phoneme clusters and their relationship to utterance boundaries, a lexicon was built incrementally and used to recognize words and parse incoming utterances, with the results fed back to further modify the lexicon. The algorithm was tested on a number of corpora with a variety of characteristics. Then, two other groups of models which have been applied to similar segmentation problems – connectionist networks and MBDP algorithms – were closely compared to BootLex, and the suitability of all three groups as cognitive analogs was examined. This case study demonstrated that word segmentation can be accomplished to a significant degree by purely probabilistic techniques. Using different techniques and starting assumptions, all three groups were able to draw upon the statistical structure inherent in language, suggesting that children might do the same.

In one respect, however, the contrastive approach used here does a disservice to cognitive science. What have been presented as disjunctive choices may in fact be cooperating influences. It is probable that infants use not just one approach, but a number of sources of information and a variety of strategies, in their struggle to make sense of the speech stream surrounding them.

Our efforts so far to model segmentation, as well as other cognitive problems, have with few exceptions been limited to "one-trick" programs – not necessarily because we believe these are the truth, but because our modeling technologies have been unable to do justice to the observed complexity of organic cognitive systems. However, as Markman and Dietrich (2000, p. 162f) say:

> The diversity of representational schemes is to be embraced rather than avoided… [C]ognitive science must find ways to integrate processes involving different kinds of representations.

If we could create more complex and layered models, we could explore the effects of using both connectionist-style activation and object-oriented lexicons in cooperation. We could combine several distributional cues in varying proportions. We could test various timing assumptions, with different abilities "kicking in" at different stages of development. In the future, we must challenge ourselves to create more cognitively plausible models and to find ways to more accurately reflect the probable reality of multiple representations and cooperative strategies.

Acknowledgements

The research reported here was conducted as partial fulfillment of the requirements for the degree of Doctor of Philosophy. I thank my thesis supervisor, Virginia Teller, and the members of my committee, Virginia Valian and Martin Chodorow. Portions of this manuscript were written while I was a Foreign Research Fellow of the Japanese Society for the Promotion of Science, appointed on the recommendation of the National Science Foundation, and hosted by Nobuo Ohta at the University of Tsukuba. Thanks for help on various drafts is due to Virginia Valian, Virginia Teller and Hartvig Dahl, Anne Christophe, Terry Joyce, and Marshall R. Childs, and to two anonymous reviewers.

References

Alloy, L. B., & Tabachnik, N. (1984). Assessment of covariation by humans and animals: the joint influence of prior expectations and current situational information. *Psychological Review*, *91*, 112–149.

Aslin, R. N., Jusczyk, P. W., & Pisoni, D. B. (1998). Speech and auditory processing during infancy. In D. Kuhn & R. S. Siegler (Eds.), *Handbook of child psychology,* 5th ed. Cognition, perception, & language (pp. 147–198). Vol.2. New York: Wiley.

Aslin, R. N., Saffran, J. R., & Newport, E. L. (1998). Computation of conditional probability statistics by 8-month-old infants. *Psychological Science*, *9*, 321–324.

Aslin, R. N., Woodward, J. Z., LaMendola, N. P., & Bever, T. G. (1996). Models of word segmentation in fluent maternal speech to infants. In J. L. Morgan & K. Demuth (Eds.), *Signal to syntax* (pp. 117–134). Mahwah, NJ: Lawrence Erlbaum Associates.

Batchelder, E. O. (1997). *Computational evidence for the use of frequency information in discovery of the infant's first lexicon.* PhD dissertation, City University of New York.

Bernstein Ratner, N. (1996). From 'signal to syntax': but what is the nature of the signal? In J. L. Morgan & K. Demuth (Eds.), *Signal to syntax* (pp. 135–150). Mahwah, NJ: Lawrence Erlbaum Associates.

Brent, M. R. (1999a). An efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learning*, *34*, 71–105.

Brent, M. R. (1999b). Speech segmentation and word discovery: a computational perspective. *Trends in Cognitive Sciences*, *3*, 294–301.

Brent, M. R., & Cartwright, T. A. (1996). Distributional regularity and phonotactic constraints are useful for segmentation. *Cognition*, *61*, 93–125.

Brent, M. R., & Siskind, J. M. (2000). The role of exposure to isolated words in early vocabulary development. NECI TR 2000-067R.

Brown, R. (1973). *A first language*. Cambridge, MA: Harvard University Press.

Cairns, P., Shillcock, R., Chater, N., & Levy, J. (1994). Lexical segmentation: the role of sequential statistics in supervised and un-supervised models. In A. Ram & K. Eiselt (Eds.), *Proceedings of the 16th Annual Conference of the Cognitive Science Society* (pp. 136–141). Hillsdale, NJ: Lawrence Erlbaum Associates.

Cairns, P., Shillcock, R., Chater, N., & Levy, J. (1997). Bootstrapping word boundaries: a bottom-up corpus-based approach to speech segmentation. *Cognitive Psychology*, *33*, 111–153.

Cartwright, T. A., & Brent, M. R. (1994). Segmenting speech without a lexicon: evidence for a bootstrapping model of lexical acquisition. In A. Ram & K. Eiselt (Eds.), *Proceedings of the 16th Annual Conference of the Cognitive Science Society* (pp. 148–152). Hillsdale, NJ: Lawrence Erlbaum Associates.

Charniak, E. (1993). *Statistical language learning*. Cambridge, MA: MIT Press.

Christiansen, M. H., Allen, J., & Seidenberg, M. S. (1998). Learning to segment speech using multiple cues: a connectionist model. *Language and Cognitive Processes*, *13* (2/3), 221–268.

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.

Cleeremans, A. (1993). *Mechanisms of implicit learning: connectionist models of sequence processing*. Cambridge, MA: MIT Press.

Crystal, D. (1987). *The Cambridge encyclopedia of language*. Cambridge: Cambridge University Press.

Dahan, D., & Brent, M. R. (1999). On the discovery of novel wordlike units from utterances: an artificial-language study with implications for native-language acquisition. *Journal of Experimental Psychology: General*, *128* (2), 165–185.

DeCasper, A. J., & Fifer, W. P. (1980). Of human bonding: newborns prefer their mothers' voices. *Science*, *208*, 1174–1176.

de Marcken, C. (1995). The unsupervised acquisition of a lexicon from continuous speech. A.I. Memo No. 1558, MIT Artificial Intelligence Lab. Available: http://xxx.lanl.gov/abs/cmp-lg/9512002

de Marcken, C. (1996a). Linguistic structure as composition and perturbation. In *Proceedings of the 34th Meeting of the Association for Computational Linguistics* (pp. 335–341). Available: http://xxx.lanl.gov/abs/cmp-lg/9606027

de Marcken, C. G. (1996b). *Unsupervised language acquisition*. PhD dissertation, MIT, Cambridge, MA. Available: http://xxx.lanl.gov/abs/cmp-lg/9611002

de Marcken, C. (undated). Language acquisition by recursive text compression, ms., MIT Artificial Intelligence Laboratory.

Dominey, P. F., & Ramus, F. (2000). Neural network processing of natural language: I. Sensitivity to serial, temporal and abstract structure of language in the infant. *Language and Cognitive Processes*, *15* (1), 87–127.

Echols, C. H., Crowhurst, M. J., & Childers, J. B. (1997). The perception of rhythmic units in speech by infants and adults. *Journal of Memory and Language*, *36*, 202–225.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*, 179–211.

Friederici, A. D., & Wessels, J. M. I. (1993). Phonotactic knowledge of word boundaries and its use in infant speech perception. *Perception & Psychophysics*, *54* (3), 287–295.

Gasser, M., & Colunga, E. (1999). How babies learn to find words. *Proceedings of the International Conference on Cognitive Science*, *2*, 277–281.

Gerken, L., Jusczyk, P. W., & Mandel, D. R. (1994). When prosody fails to cue syntactic structure: nine-month-olds' sensitivity to phonological vs. syntactic phrases. *Cognition*, *51*, 237–265.

Gerken, L., Landau, B., & Remez, R. E. (1990). Function morphemes in young children's speech perception and production. *Developmental Psychology*, *26*, 204–216.

Gleitman, L., & Bloom, P. (1999). Language acquisition. In R. A. Wilson & F. C. Keil (Eds.), *MIT encyclopedia of the cognitive sciences* (pp. 434–438). Cambridge, MA: MIT Press.

Goldowsky, B., & Newport, E. (1996). Modeling the effects of processing limitations on the acquisition of morphology: the less is more hypothesis. In J. Mead (Ed.), *Proceedings of the 11th West Coast Conference on Formal Linguistics* (pp. 234–247). Stanford, CA: CSLI.

Goodsitt, J. V., Morgan, J. L., & Kuhl, P. K. (1993). Perceptual strategies in prelingual speech segmentation. *Journal of Child Language*, *20* (2), 229–252.

Harris, M., & Chasin, J. (1999). Developments in early lexical comprehension: a comparison of parental report and controlled testing. *Journal of Child Language*, *26*, 453–460.

Hasher, L., & Zacks, R. T. (1984). Automatic processing of fundamental information: the case of frequency of occurrence. *American Psychologist*, *39*, 1372–1388.

Hauser, M. D., Newport, E. L., & Aslin, R. N. (2001). Segmentation of the speech stream in a non-human primate: statistical learning in cotton-top tamarins. *Cognition*, *78* (3), B53–B64.

Hayashi, A., Tamekawa, Y., Deguchi, T., & Kiritani, S. (1996). *Developmental change in perception of clause boundaries by 6- and 10-month-old Japanese infants*. Paper presented at the Fourth International Conference on Spoken Language Processing (ICSLP), Philadelphia, PA.

Hayes, J. R., & Clark, H. H. (1970). Experiments on the segmentation of an artificial speech analogue. In J. R. Hayes (Ed.), *Cognition and the development of language* (pp. 221–234). New York: Wiley.

Hirsh-Pasek, K., Kemler Nelson, D. G., Jusczyk, P. W., Cassidy, K. W., Druss, B., & Kennedy, L. (1987). Clauses are perceptual units for young infants. *Cognition*, *26*, 269–286.

Hohne, E. A., & Jusczyk, P. W. (1994). Two-month-old infants' sensitivity to allophonic differences. *Perception & Psychophysics*, *56*, 613–623.

Johnson, E. K., & Jusczyk, P. W. (2001). Word segmentation by 8-month-olds: when speech cues count more than statistics. *Journal of Memory and Language*, *44* (4), 548–567.

Jusczyk, P. W. (1997). *The discovery of spoken language*. Cambridge, MA: MIT Press.

Jusczyk, P. W. (1998a). Constraining the search for structure in the input. *Lingua*, *106*, 197–218.

Jusczyk, P. W. (1998b). Dividing and conquering linguistic input. In M. C. Gruber, D. Higgins, K. Olson & T. Wysocki (Eds.), *CLS 34, The panels* (pp. 293–310). Vol. II. Chicago, IL: University of Chicago Press.

Jusczyk, P. W. (1999). How infants begin to extract words from speech. *Trends in Cognitive Sciences*, *3*, 323–328.

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

Jusczyk, P. W., Friederici, A. D., Wessels, J., Svenkerud, V., & Jusczyk, A. M. (1993). Infants' sensitivity to the sound structure of native language words. *Journal of Memory and Language*, *32*, 402–420.

Jusczyk, P. W., Hirsh-Pasek, K., Kemler Nelson, D. G., Kennedy, L. J., Woodward, A., & Piwoz, J. (1992). Perception of acoustic correlates of major phrasal units by young infants. *Cognitive Psychology*, *24*, 252–293.

Jusczyk, P. W., Hohne, E. A., & Bauman, A. (1999). Infants' sensitivity to allophonic cues for word segmentation. *Perception & Psychophysics*, *61* (8), 1465–1476.

Jusczyk, P. W., Houston, D. M., & Newsome, M. (1999). The beginnings of word segmentation in English-learning infants. *Cognitive Psychology*, *39*, 159–207.

Kareev, Y. (1995). Through a narrow window: working memory capacity and the detection of covariation. *Cognition*, *56*, 263–269.

Kelly, M. H., & Martin, S. (1994). Domain-general abilities applied to domain-specific tasks: sensitivity to probabilities in perception, cognition, and language. *Lingua*, *92*, 105–140.

Locke, J. L. (1996). Why do infants begin to talk? Language as an unintended consequence. *Journal of Child Language*, *23*, 251–268.

MacWhinney, B. (1993). Discussion: connections and symbols: closing the gap. *Cognition*, *49*, 291–296.

MacWhinney, B. (1995). *The CHILDES project: tools for analyzing talk*, (2nd ed.) Hillsdale, NJ: Lawrence Erlbaum Associates.

Mandel, D. R., Jusczyk, P. W., & Pisoni, D. B. (1995). Infants' recognition of the sound patterns of their own names. *Psychological Science*, *6* (5), 314–317.

Markman, A. B., & Dietrich, E. (2000). In defense of representation. *Cognitive Psychology*, *40*, 138–171.

Mattys, S. L., & Jusczyk, P. W. (2001a). Do infants segment words or recurring contiguous patterns? *Journal of Experimental Psychology: Human Perception & Performance*, *27* (3), 644–655.

Mattys, S. L., & Jusczyk, P. W. (2001b). Phonotactic cues for segmentation of fluent speech by infants. *Cognition*, *78*, 91–121.

Mattys, S. L., Jusczyk, P. W., Luce, P. A., & Morgan, J. L. (1999). Phonotactic and prosodic effects on word segmentation in infants. *Cognitive Psychology*, *38* (4), 465–494.

McClelland, J. L., & Elman, J. L. (1986). The TRACE model of speech perception. *Cognitive Psychology*, *18*, 1–86.

Mehler, J., Jusczyk, P., Lambertz, G., Halsted, N., Bertoncini, J., & Amiel-Tison, C. (1988). A precursor of language acquisition in young infants. *Cognition*, *29*, 143–178.

Morgan, J. L. (1996). Prosody and the roots of parsing. *Language and Cognitive Processes*, *11* (1/2), 69–106.

Morgan, J. L., & Saffran, J. R. (1995). Emerging integration of sequential and suprasegmental information in preverbal speech segmentation. *Child Development*, *66* (4), 911–936.

Newport, E. L. (1990). Maturational constraints on language learning. *Cognitive Science*, *14*, 11–28.

Olivier, D. C. (1968). *Stochastic grammars and language acquisition mechanisms*. PhD dissertation, Harvard University, Cambridge, MA.

Perruchet, P., & Vinter, A. (1998). PARSER: a model for word segmentation. *Journal of Memory and Language*, *39* (2), 246–263.

Peters, A. M. (1983). *The units of language acquisition*. Cambridge: Cambridge University Press.

Plunkett, K. (1993). Lexical segmentation and vocabulary growth in early language acquisition. *Journal of Child Language*, *20*, 43–60.

Reber, A. S. (1993). *Implicit learning and tacit knowledge: an essay on the cognitive unconscious*, New York: Oxford University Press.

Redington, M., & Chater, N. (1998). Connectionist and statistical approaches to language acquisition: a distributional perspective. *Language and Cognitive Processes*, *13* (2/3), 129–191.

Redlich, A. N. (1993). Redundancy reduction as a strategy for unsupervised learning. *Neural Computation*, *5*, 289–304.

Ristad, E. S., & Thomas, R. G. (1995). New techniques for context modeling, ms. Available: http://xxx.lanl.gov/abs/cmp-lg/9505002

Saffran, J. R. (2001). Words in a sea of sounds: the output of infant statistical learning. *Cognition*, *81*, 149–169.

Saffran, J. R., Aslin, R. N., & Newport, E. L. (1996a). Statistical cues in language acquisition: word segmentation by infants. In G.W. Cottrell (Ed.), *Proceedings of the 18th Annual Conference of the Cognitive Science Society*. (pp. 376–380). Hillsdale, NJ: Lawrence Erlbaum Associates.

Saffran, J. R., Aslin, R. N., & Newport, E. L. (1996b). Statistical learning by 8-month-old infants. *Science*, *274*, 1926–1928.

Saffran, J. R., Johnson, E. K., Aslin, R. N., & Newport, E. L. (1999). Statistical learning of tone sequences by human infants and adults. *Cognition*, *70*, 27–52.

Saffran, J. R., Newport, E. L., Aslin, R. N., Tunick, R. A., & Barrueco, S. (1997). Incidental language learning: listening (and learning) out of the corner of your ear. *Psychological Science*, *8* (2), 101–105.

Shillcock, R., Cairns, P., Chater, N., & Levy, J. (2000). Statistical and connectionist modelling of the development of speech segmentation. In P. Broeder & J. M. J. Murre (Eds.), *Models of language acquisition: inductive and deductive approaches* (pp. 103–120). Oxford: Oxford University Press.

Shillcock, R., Lindsey, G., Levy, J., & Chater, N. (1992). A phonologically motivated input representation for the modelling of auditory word perception in continuous speech. *In Proceedings of the 14th Annual Conference of the Cognitive Science Society* (pp. 408–413). Hillsdale, NJ: Lawrence Erlbaum Associates.

Tincoff, R., & Jusczyk, P. W. (1999). Some beginnings of word comprehension in 6-month-olds. *Psychological Science*, *10* (2), 172–175.

Venkataraman, A. (2001). A statistical model for word discovery in transcribed speech. *Computational Linguistics*, *27* (3), 352–372.

Vihman, M. M. (1996). *Phonological development: the origins of language in the child*. Cambridge, MA: Blackwell.

Waxman, S. R. (1999). Specifying the scope of 13-month-olds' expectations for novel words. *Cognition*, *70*, B35–B50.

Wolff, J. G. (1975). An algorithm for the segmentation of an artificial language analogue. *British Journal of Psychology*, *66*, 79–90.

Wolff, J. G. (1977). The discovery of segments in natural language. *British Journal of Psychology*, *68*, 97–106.