Using predictability for segmentation

Çağrı Çöltekin

University of Groningen

Abstract

This paper presents a computational study of a well-known cue, predictability statistics, for learning and finding lexical units in continuous speech. Certain statistical properties of the naturally occurring speech stream allow predicting syllables or phonemes from their context in varying degrees. These properties have been shown to be used by humans for segmenting continuous speech as early as eight-months of age. However, the interest in direct use of this cue in computational models of segmentation is relatively scarce so that the cue is under explored. This study is an attempt to fill this gap by demonstrating the utility of predictability statistics in the segmentation task by corpus analysis and computational simulations. The computational simulations are carried out using an incremental model on child-directed speech corpora. The results show that one can achieve good segmentation performance by using only the predictability cue. The accuracy of the results rivals state-of-the-art systems even though the only cue exploited involves predictability and the system presented is both unsupervised and strictly incremental, unlikely most other systems that have been examined. Besides an in-depth analysis of predictability, the model presented in this study also offers a natural way to combine multiple cues for solving the segmentation problem.

Keywords: word segmentation, computational modeling, transitional probabilities, mutual information, entropy, successor variety

1. Introduction

Segmenting continuous speech into lexical units is one of the early tasks an infant needs to tackle during language acquisition. The segmentation problem is more difficult than may be appreciated at first sight. Children need to find words in a continuous stream of speech, with no knowledge of words to start with. Fortunately, experimental studies suggest that children are not helpless in this task. They are sensitive to, and make use of, some properties of naturally occurring speech very early in the acquisition process, which lead to relatively simple computational strategies for segmenting input utterances. Children are known to attend to a number of cues that are useful for discovering lexical units. These cues include, but are not limited to, *lexical stress* (Cutler and Butterfield, 1992; Jusczyk et al., 1999b), *phonotactics* (Jusczyk et al., 1993), *predictability statistics* (Saffran et al., 1996a), *allophonic differences* (Jusczyk et al., 1999a), and *coarticulation* (Johnson and Jusczyk, 2001). However, most of these cues are language specific and can be useful only after the learner has a lexicon populated with some of the words of the target language. This suggests that some cues, particularly *predictability*, are better candidates for bootstrapping the acquisition of lexical units.

The principle behind the predictability-based segmentation strategy is simple: *the basic units (e.g., syllables or phonemes) within a lexical unit predict one another in sequence, while units across boundaries do not*. The reason why predictability is useful for discovering boundaries and the lexical units has to do with the fact that the input to the learner is constructed from a set of reoccurring lexical units which tend to follow certain regularities. This process is invariant with respect to the language(s) children are exposed to. As a result, predictability is a language-general cue that can be put into operation without any language-specific information. Naturally, we expect children to use language-specific cues as well, but only after they tune into particular aspects of the language(s) they are exposed to. Predictability statistics is the best candidate we know for the first step into building a lexicon and enabling the use of other, language-specific, cues. Therefore, a better understanding of this cue and computational mechanisms exploiting it is important for understanding how children start extracting their first lexical units from continuous speech input.

Predictability statistics, as a way of discovering lexical units, has been a topic of interest in quantitative and computational linguistics for a long time, at least dating back to Harris (1955). Furthermore, this strategy is shown to be used by humans in many tasks including segmentation. The seminal work by Saffran et al. (1996a) showed that eight-month-old infants use the predictability of consecutive syllables to extract word-like units from an artificial sound sequence where all other cues were removed. Following this study, a large number of experimental studies have confirmed that predictability-based strategies are used by adults and children for learning different aspects of language (e.g., Aslin et al., 1998; Thiessen and Saffran, 2003; Newport and Aslin, 2004; Graf Estes et al., 2007; Thompson and Newport, 2007; Perruchet and Desaulty, 2008).

Besides direct experimentation, computational modeling and simulation has been a fruitful method for studying cognitive phenomena, among which segmentation is not an exception. There have been an increasing number of models of segmentation particularly within the last two decades. Earlier influen-

tial models of segmentation were based on connectionist models (e.g., Elman, 1990; Aslin, 1993; Christiansen et al., 1998). These models have been instrumental in understanding the segmentation process. Particularly, these models clearly demonstrate the usefulness of predictability statistics and combination of multiple cues. However, connectionist systems, in general, have been subject to the criticism that what a connectionist model learns is rather difficult to interpret. Furthermore, even though connectionist models perform better than random processes, the segmentation performance that can be achieved using connectionist models is lower than what we expect from adult segmentation performance. Models that use explicit representations in combination with statistical procedures (e.g., Brent and Cartwright, 1996; Brent, 1999; Venkataraman, 2001; Goldwater et al., 2009; Johnson and Goldwater, 2009; Monaghan and Christiansen, 2010) avoid both problems: these models typically perform well, and it is easier to reason about what they learn. However, these models lack at least two aspects of connectionist models that fit human processing better. First, even though we know that human segmentation is incremental and predictive, most of these models process their input either in a batch fashion, or they require the complete utterance to be presented before attempting to discover any boundaries or units. Second, even though the models can be augmented to use a set of cues, it is difficult to incorporate arbitrary cues into most of these models.

The present study will introduce an incremental framework of segmentation that follows human performance as closely as the former models do, while using an explicit statistical procedure and performing comparably to the state-of-the-art models. The focus of the current study is investigating the use of the predictability statistics in depth. As a result, we will not explore the use of other cues. However, the framework described here offers a natural way of combining cues from multiple sources. This paper, first, reports on an in-depth investigation of common predictability measures for segmentation and the relations between them. Second, it describes a novel predictability-based model that can easily be extended to include other cues for segmentation, and reports on the simulation results obtained with this model on a well-known child-directed speech corpus. The model described here follows what we know about human performance in the segmentation task closely while performing competitively with the state-of-the-art segmentation models.

The remainder of this paper is organized as follows. The next section will briefly discuss predictability statistics and its use in the segmentation task by humans and the computational models in the literature. The ways of quantifying predictability differ, so Section 3 takes a closer look at some of the ways predictability can be quantified and offers a few extensions to the way it is commonly used in the literature. Besides describing the data used in this study, Section 4 discusses a set of metrics to assess the success of a computational model of segmentation. Section 5 introduces two reference models, and Section 6 describes a predictability-based segmentation model, and the results of the simulations on a transcribed child-directed speech corpus. Section 7 provides a summary and concludes.

2. Predictability and distributional regularities

Predicting things to come is a natural activity of the human brain. Our cognitive machinery constantly predicts the next state of the environment at many levels. An interesting aspect of human cognition is not only how we set expectations about the next step on a task, but how we react when expectations fail and we are surprised. We remember and we learn most from surprising events. It seems that prediction is an important aspect of human cognition, and when it fails, it has further consequences on the cognitive system.

Besides this common-sense notion of predictability in human cognition, children and adults use it for language processing. An earlier use of the cue in linguistics is due to Harris (1955). Harris suggested using a measure of predictability to segment a given utterance into morphemes by positing boundaries after an initial utterance segment which may be followed by many different phoneme types—in other words, when it is difficult to predict which phoneme comes next. The idea has been used in natural language processing after Harris (e.g., Hafer and Weiss, 1974). However investigations of predictability as a cue for discovering lexical units was delayed until 1990's in the psycholinguistics and cognitive science literature, and gained popularity after an influential study by Saffran et al. (1996a).

Saffran et al. (1996a) familiarized eight-month-old infants with stimuli constructed from three-syllabic artificial words. In the stimuli used during the familiarization phase the transition of syllables within the words was deterministic, while the transition probability between the words was lower (1/3). Crucially, the stimuli did not contain any other cues to 'word' boundaries. After only two minutes of familiarization, infants were able to distinguish a novel sequence constructed from the artificial words in the familiarization phase from another sequence formed by part-words with the same frequency of occurrence of the syllables as in the training stimuli. The computational principle children apply to the task seems to be the same: the syllable segments that predict each other well are identified as (lexical) units, positing boundaries where it is difficult to predict the next syllable.

The reason why this simple principle works for segmentation has to do with the way natural language utterances are generated. Utterances are strings of words from the speaker's lexicon. Words tend to reoccur, and the formation of words follows certain regularities. On the other hand, the sequence of words is relatively unpredictable. As a result, the basic units (e.g., phonemes or syllables) within words predict each other, while it is more difficult to predict the next basic unit on a word boundary.[1]

This fact has led a large number of successful segmentation algorithms to focus on the generative side of the process.

[1] To enable comparison with earlier research, we use a phonemically-transcribed reference corpus in this paper. However, the assumption of a certain basic unit, such as the phoneme or the syllable, is not essential for a predictability-based segmentation strategy. The strategy works with any basic unit, even with non-linguistic collections of *acoustic events* (Räsänen, 2011).

These systems define a *generative model* that is conjectured to produce the input corpus, and find the segmentation with the highest probability under the model (e.g., Brent and Cartwright, 1996; Brent, 1999; Goldwater et al., 2009; Johnson and Goldwater, 2009). These models typically perform better than the models that make use of the cues that are known to be used by humans on the task directly, and they are instrumental in answering many questions about the segmentation process. However, the models that operate on cues available to the learner, and do not assume any knowledge of the system that generates the input take the learner's perspective with higher fidelity. Such models, at least in principle, can provide explanations at lower levels (e.g., Marr's (1982) algorithmic level in comparison to computational level).

Models that take the learner's perspective and use predictability as a segmentation strategy are rarely studied in depth. The most common use of predictability for modeling learning segmentation are connectionist models that implement the strategy implicitly. For example, the popular *simple recurrent networks* (SRN, Elman, 1990) are typically trained for predicting the next input unit. Non-connectionist examples of the use of predictability in psychologically motivated computational models of segmentation are rather scarce, and offer little detail. For example, Brent (1999) employs simple uses of *transitional probabilities* and *pointwise mutual information* as baselines for comparison with the proposed model in his study, and Swingley (2005) uses *pointwise mutual information* in his batch segmentation method. The only detailed treatment of a predictability measure, *entropy*, for segmentation is the study by Cohen et al. (2007). The segmentation algorithm described in Cohen et al. (2007) shares a number of similarities with the model presented in Section 6 of this study. Besides the differences in the measures used and the ways in which they are combined and weighted, the present study also differs from Cohen et al. (2007) in its its design, structured to allow the combination of other cues that are known to be used by humans in discovering lexical items from continuous speech.

The next section provides a corpus analysis for investigating the usefulness of a number of ways to quantify predictability and the relations between them.

3. Measures of predictability for segmentation

It is clear from the psycholinguistics literature that predictability is used by humans for the task of segmentation. Particularly, it seems, when consecutive units do not predict each other, even eight-month-olds tend to assume that there is a word boundary (Saffran et al., 1996a). This section will formally introduce four measures of (un)predictability, namely, *transitional probability* and *successor variety*, *pointwise mutual information* and *boundary entropy*, and present an analysis of child-directed speech that investigates usefulness of these measures as indications of word boundaries.

Before analyzing the measures listed above, another measure previously studied by Hockema (2006) will be presented. This measure is not suitable for unsupervised learning, and hence, to model how children learn to segment. Nevertheless, the study uncovers an interesting property of speech sequences relevant to the segmentation problem.

3.1. Boundary probability

Hockema (2006) analyzed a large corpus of child-directed speech according to a measure he called *conditional boundary probability*, which is defined as the probability of observing a word boundary between two phonemes l and r,

$$P_{wb}(l, r) = P(boundary|lr)$$

where lr is the phoneme bigram obtained by concatenating the phonemes l and r.

He transcribed all child-directed utterances in the American English section of the CHILDES that were available at the time using the Carnegie Mellon Pronouncing Dictionary (Carnegie Mellon University, 1998). For each possible phoneme pair lr, he estimated $P_{wb}(l, r)$, and plotted the histogram of these probability values. The result showed that the distribution is strongly bimodal. Phoneme pairs show a high tendency to occur either word-internally or at word boundaries.

Figure 1 presents graphs produced by the same procedure using the child-directed speech in the corpus collected by Bernstein Ratner (1987) (henceforth, BR corpus). The differences between data used for producing these graphs and Figure 2 in Hockema (2006) are in the size of the corpus and the number of phonemes used for transcribing the corpus. Hockema's data consisted of 8,078,540 phoneme pairs transcribed using a 39-phoneme alphabet. In contrast, the analysis used here is based on 86,019 phoneme pairs that are transcribed with a 50-phoneme alphabet. Despite the differences, the same trends hold: the distribution of phoneme pairs is strongly bimodal.

The presentation of the data is also different. All histograms presented in this section are like Hockema's *normalized* histograms: they represent token frequencies, counting of the relevant values as many times as the they occur in the corpus. As a result, compared to a histogram that is based on phoneme-pair types, these histograms are better representations of the distributions that a child would hear.

Figure 1a presents the histogram of $P_{wb}(l, r)$ for all phoneme pairs that were observed in the corpus. Large portions of the probability mass are lumped together either at the very first bin where the probability of a word boundary is zero or close to zero, or on the opposite end of the scale, where the probability of a word boundary is one or close to one. This clearly shows that there is a tendency for some phoneme pairs to appear only word internally, and some others to appear on word boundaries. Figure 1b–c presents the two separate histograms of the same quantity. In Figure 1b only the probabilities of phoneme pairs that straddle word boundaries are shown, while in Figure 1c only word-internal phoneme pairs are considered. These histograms clearly show that bimodality of the measure is indeed due to the differences between the phoneme pairs that occur at the word boundaries and the word internal positions. Figure 1d presents the segmentation performance of a simple segmentation algorithm that segments between phoneme pairs for which $P_{wb}(l, r)$ is greater than a threshold value. The graph

presents precision, recall and F-score for varying threshold values. The results indicate that a very high level of performance is attainable for a large range of threshold values. For example, for a threshold of 0.5, we get 91.2% precision, 87.3% recall which amounts to an F-score of 89.2%. These figures seem to be somewhat lower (86.5% precision, 76.0% recall) for Hockema's larger CDS data set.

Considering that this segmentation performance can be achieved using only statistics over phoneme pairs, it is an impressive result. However, there are two major problems with this analysis. First, the learner has no access to the information needed (the knowledge of word boundaries) to build this distribution. As a result, even though it uncovers a nice regularity about the data, it is of little direct use for an unsupervised segmentation algorithm. Second, since the method is based only on phoneme pairs, there is no way of distinguishing occurrences of a phoneme pair that occurs both word-internally, and at word boundaries. This becomes particularly problematic for some of the frequent phoneme pairs. For example, /sI/,[2] occurs 153 times on a word boundary, such as in *what's **it***, and 163 times word internally, such as in ***sit***, in the BR corpus. The method suggests that either all occurrences of the word *sit* and the words including the phoneme pair /sI/ will be oversegmented, or phrases like *what's it* will be undersegmented.

The analysis provided above indicates that given a correctly segmented corpus, one can come up with relatively accurate segmentation based on the likelihood that a phoneme pair occurs at word boundaries. Even though this is not immediately useful to a learner without access to an already segmented corpus, similar results may be obtained based on various measures of predictability that do not require a segmented corpus. The rest of this section provides similar analyses for four such measures.

3.2. Transitional probability

As a measure of predictability, most studies in the psycholinguistic literature use *conditional probability*—or *transitional probability* (TP), as it is known in this field (e.g., Saffran et al., 1996a). Transitional probability of two phonemes l and r is defined as

$$\mathrm{TP}(l, r) = P(r|l) = \frac{P(lr)}{P(l)} \approx \frac{frequency(lr)}{frequency(l)} \qquad (1)$$

Note that this is the same measure used by Saffran et al. (1996a), except we use phonemes (and later sequences of phonemes) instead of syllables.

Intuitively, if the phoneme pair lr is highly probable, it is likely that l and r are part of a word for two reasons. First, words repeat, and that makes parts of the words repeat as well. Second, since words are not formed randomly, certain sequences are more likely to be within words. These observations indicate that the *joint probability*, $P(lr)$, is a useful measure. However, if l is very frequent, the reason that lr is also frequent may often be just by chance. For example, since the phoneme /i/ is rather frequent in English, the sequence /iI/ occurs frequently even though it rarely occurs within words. On the other hand, even though the phoneme sequence /WI/ occurs exclusively within words in the BR corpus, the probability estimate of P(/iI/) is 3.67 times P(/WI/). As Equation 1 suggests, transitional probability is high if joint probability is high. The division by $P(l)$ in the definition of TP, reduces this 'chance effect' to some extent. For the same example, even though it is still higher, TP(/i/, /I/) is only 1.71 times TP(/W/, /I/).

Figure 2a presents distribution of transitional probability values. Unfortunately, there is no clear indication of a bimodal distribution. If we plot histograms of the transitional probabilities at boundaries and word-internal positions separately (Figure 2b–c), we can see that the distributions are somewhat different. As expected, the probability mass for boundaries is found more towards the lower end of the distribution. However, even though the distribution of word-internal transitional probabilities tends more towards the higher values, there is still a large number of word-internal positions with low transitional probabilities. Figure 2d presents the performance scores for a strategy that segments at the locations where transitional probability is lower than a threshold. The gray line in this graph presents the performance of a segmentation strategy where boundaries are inserted randomly with the constraint that the number of boundaries inserted is the same as the number of boundaries in the gold-standard segmentation. The random segmentation strategy is explained in Section 5. Since precision and recall scores of the random segmentation are the same, the F-score is also the same. As a result they appear as a single line in Figure 2d. It should be noted that even though the boundaries are chosen at random, this particular segmentation strategy is a rather informed baseline: it knows the number of boundaries.

This analysis indicates that even though it is not as impressive as the measure suggested by Hockema (2006), a naive segmentation strategy based on TP performs consistently better than random. Crucially, this measure is more suitable for unsupervised methods, since calculation of transitional probabilities does not require the knowledge of word boundaries.

Using threshold values for unsupervised segmentation is problematic because it requires a non-trivial way to set a threshold value without knowing which value is a good option. This problem and possible solutions will be discussed further in Section 6 where an explicit unsupervised algorithm for segmentation will be described. The analysis provided in Figure 2d serves as an indication that this measure is useful, and allows us to compare it with the other measures.

Using the transitional probability measure as presented here has two other weaknesses. First, like the P_{wb} measure discussed above, TP calculated on only two consecutive phonemes cannot handle effects of larger sequences of phonemes or non-adjacent phonemes. This is not an intrinsic property of the measure, and use of larger phoneme context will be discussed in Section 3.6. Second, as is also noted in Brent (1999), the conditional probability is asymmetric, $P(l|r)$ is not the same as $P(r|l)$, and $P(l|r)$ can also provide useful information for segmentation. The util-

[2] The symbols used for phonemes in these examples and for the rest of this paper follow the conventions used by Brent and Cartwright (1996) in transcribing the BR corpus.

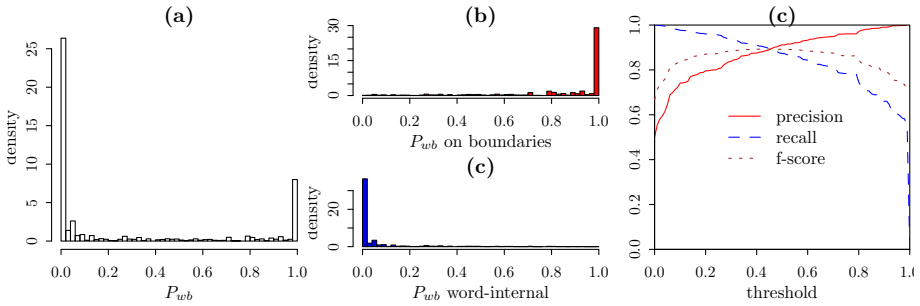


Figure 1: (a) The bi-modal histogram of $P_{wb}(l, r)$ values. (b–c) Histograms of P_{wb} for all pairs that occur at word boundaries (b), and word-internal positions (c). (d) Precision, recall and F-score values for against changing threshold.

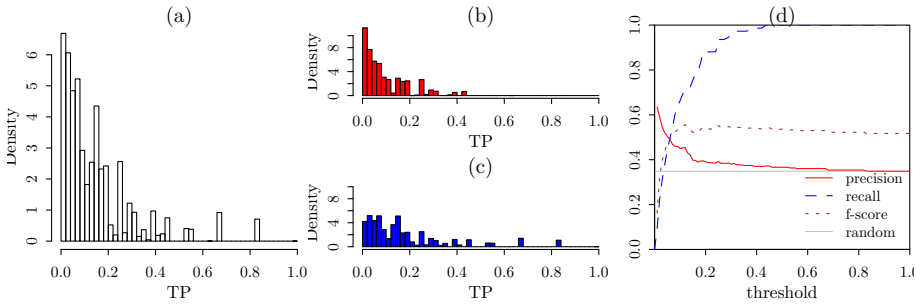


Figure 2: (a) Distribution of transitional probabilities. (b–c) Distribution of TP for boundaries and word-internal positions, respectively. (d) Performance of algorithms that segment at locations where $P(r|l)$ is lower than a threshold value. The solid gray line in (d) represents the precision, recall and F-score of a pseudo-random segmentation method that inserts as many boundaries as in the gold-standard segmentation.

ity of the backward version of the measure will be discussed in Section 3.7.

3.3. Pointwise mutual information

Pointwise (or specific) mutual information is an information theoretic measure of association between two random variables. It is used in many natural language processing tasks, and its use in segmentation, albeit rare, is not exceptional (e.g., Brent, 1999; Swingley, 2005). Pointwise mutual information (MI) is defined as,[3]

$$\mathrm{MI}(l,r) = log_2 \frac{P(l,r)}{P(l)P(r)}$$

Neglecting the logarithm for now, in this definition, the joint probability is divided by $P(l) \times P(r)$. As a result, the high association one would get by chance for highly frequent phonemes is reduced just as in the case of TP. Unlike TP, the MI score is affected by frequencies of both phonemes, and it is symmetrical. The logarithm defines the unit of the measure. The binary logarithm (base two) is commonly used in information theory, and the resulting unit is called *bit*.

There has been some work on computational modeling of segmentation which used MI (Brent, 1999; Swingley, 2005). However, it is virtually unmentioned in the psycholinguistic literature.

Figure 3 presents the same analysis for MI that Figure 2 presents for the TP. The first difference to note is that the shape of the graph is different from TP. This is because of the fact that the probability values are estimated from frequencies of phonemes and phoneme bigrams. Like many other frequency distributions on linguistic units, distribution of probability values, such as TP, follows an exponential trend. On the other hand, MI is the logarithm of a combination of probability values,[4] and the logarithm function transforms the exponential-like distribution into a roughly normal distribution. In addition, the difference between the distributions of MI values for boundary and non-boundary phoneme pairs seems to be slightly better separated. This is also evident from the differences of performance graphs in Figure 2d and Figure 3d. F-score for TP barely exceeds 50%, while F-score for MI is well over 60% for some threshold values. Before providing a more detailed comparison, two more measures will be introduced.

3.4. Successor variety

Among the measures we consider in this paper, the *successor variety* (SV) (Harris, 1955) is probably the earliest measure suggested for lexical segmentation. SV is defined as

$$\mathrm{SV}(l) = \sum_{r \in A} c(l,r)$$

where,

$$c(l,r) = \begin{cases} 1 & \text{if substring } lr \text{ occurs in the corpus} \\ 0 & \text{otherwise} \end{cases}$$

and A is the list of phonemes (the alphabet).

Unlike the measures discussed previously, SV is only a function of the initial sequence, l. In Harris (1955), this sequence is the sequence from the beginning of the utterance to the position to be evaluated. Figure 4 presents the successor values for the utterance /hizkwIkR/ 'he's quicker'. The SV value after the word *he's* is the highest, and a reasonable algorithm based on SV would segment this utterance correctly. However, Figure 4 also points to a problem. As the initial sequence gets longer, the likelihood that it has never occurred before in the input increases. As a result, even for child-directed speech, which is characteristically repetitive, the SV values drop to 0 and become useless after a short initial sequence. A segmentation algorithm based on the SV values calculated as in Figure 4 is likely to fail to find boundaries after a few initial boundaries. There are ways to solve this problem, but even in its simple form, SV has been popular in morphological segmentation literature (e.g., Hafer and Weiss, 1974; Déjean, 1998; Al-Shalabi et al., 2005; Bordag, 2005; Goldsmith, 2006; Bordag, 2007; Demberg, 2007; Stein and Potthast, 2008). Morphological segmentation is the task of segmenting words into morphemes, it is useful in many natural language processing tasks ranging from stemming to machine translation of agglutinative languages. Since words are more repetitive than utterances, the measure works better for morphological segmentation. However, the measure may benefit from some improvements in this task as well (Çöltekin, 2010).

To adapt the SV measure to the segmentation of utterances into lexical units, the discussion here is based on calculations made using a varying size phoneme context. It is not very useful to use SV as a segmentation measure calculated using a single-phoneme context. For example, in BR corpus, SV after the phoneme /W/ is 7 and SV after the phoneme /t/ is 46. A threshold value between these numbers will always segment after the phoneme /t/ and will never segment after /W/. However, to provide a comparison with the other measures, Figure 5 presents an analysis of SV values where boundaries are classified using the SV value of a single preceding phoneme. Nevertheless, Figure 5 indicates that, even in this form, the measure performs similarly to others.

Some improvements to make SV-like measures more effective will be discussed in Section 3.6 and 3.7. The next section finalizes the discussion of individual predictability measures with a similar but theoretically more attractive and better studied measure.

3.5. Boundary entropy

Entropy (also called *Shannon entropy* when there is a need to distinguish from entropy in thermodynamics) is the information-theoretic measure of average uncertainty.[5] Entropy is also known

[3] *Mutual information* is a related but different information theoretic measure. However, in this paper, following the related work in computational models of segmentation, the term mutual information and the abbreviation MI always refers to pointwise mutual information between two consecutive sequences.

[4] It should be noted that the quantity $\frac{P(l,r)}{P(l)P(r)}$ is not a probability. For positively correlated phonemes this value is greater than one, and MI score is positive.

[5] The inventor of the measure, Claude Shannon initially named the quantity 'uncertainty', but based on suggestion of John von Neumann, another pioneer

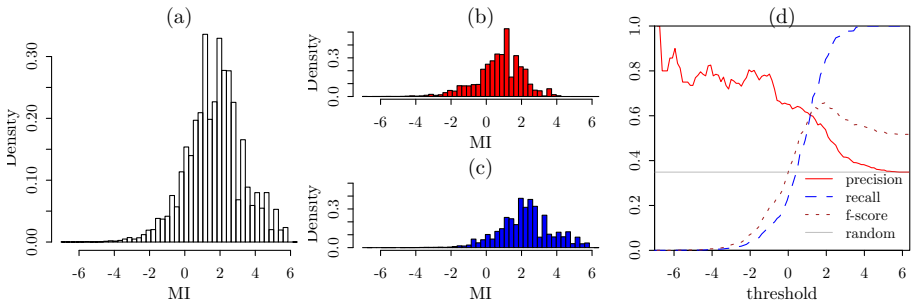


Figure 3: (a) Distribution of MI. (b–c) Distribution of MI for boundaries and word-internal positions, respectively. (d) Performance of algorithms that segment at locations where MI is lower than a threshold value. The solid gray line in (d) represents precision, recall and F-score of a pseudo-random segmentation method that inserts as many boundaries as in the gold-standard segmentation.

phoneme	h	i	z	k	w	I	k	R
SV_{BR}	16	13	22	2	0	0	0	0

Figure 4: Successor variety values calculated from BR corpus for the utterance /he's quicker/.

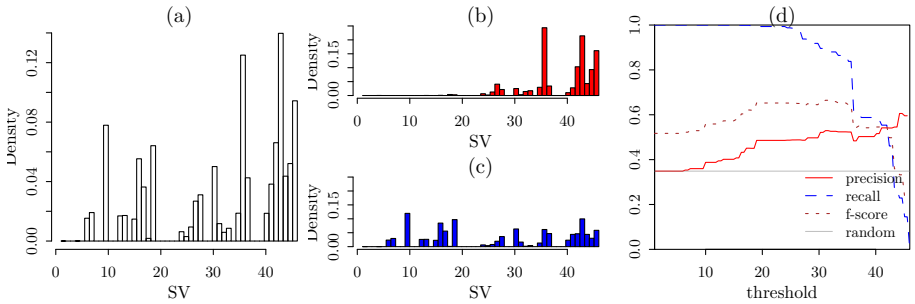


Figure 5: (a) Distribution of SV. (b–c) Distribution of SV for boundaries and word-internal positions, respectively. (d) Performance of algorithms that segment at locations where SV is higher than a threshold value. The solid gray line in (d) represents the precision, recall and F-score of a pseudo-random segmentation method that inserts as many boundaries as in the gold-standard segmentation.

as *average surprisal*, where surprisal ($-logP(l)$) is another information theoretic measure suggested by Shannon (1948). As a result, entropy is one of the natural choices for measuring (un)predictability. However, in psychologically motivated models of segmentation entropy is rarely mentioned. The use of entropy is common in segmentation of written text, particularly for languages like Chinese and Japanese, which are typical examples of languages that use writing systems without a word boundary marker (e.g., Kempe, 1999; Huang and Powers, 2003; Zhikov et al., 2010). As far as I can determine, Cohen et al. (2007) is the only study of entropy-based segmentation motivated by human (or human-like) performance.

The measure that will be used in this paper, *boundary entropy* (H) defined as,[6]

$$\mathrm{H}(l) = -\sum_{r \in A} P(r|l)\, log_2\left(P\left(r|l\right)\right) \qquad (2)$$

where the sum ranges over all phonemes in the alphabet, A. Given the sequence l, this formula gives a measure of how much uncertainty still exist. As in MI, the binary (base 2) logarithm makes the unit of the measure the *bit*. In more intuitive terms, this quantity measures how many yes/no questions are necessary on average to predict the next phoneme.

Even though it may not be clear at first sight, the entropy measure has strong similarities with the SV. Both measure the promiscuity of l. That is, if l combines with many different phonemes, then both SV and entropy are high. The difference is that entropy is sensitive to the token frequencies of the sequences, while SV only considers types. The difference may be easier to grasp with an example: Assume we have a corpus consisting of three words *xa*, *xb* and *xc*, and we are interested in unpredictability after *x*. Obviously SV is three, and calculating entropy using Equation 2 we find that entropy is 1.56 bits. If we had a corpus where *xa* occurred twice while the other two words in our previous corpus occurred once, that would not make any difference for the SV, it is still three. However, since the knowledge that *a* is a more probable phoneme after *x* reduces uncertainty, the new value for entropy (1.5 bits) reflects this.

Like SV, calculating entropy values conditioned on a single phoneme is not a good strategy. However, for the sake of completeness, Figure 6 presents the analysis presented for other measures for boundary entropy.

3.6. Effects of phoneme context

It is plausible to assume that humans use a predictability strategy based on a larger phoneme context. Many studies in psycholinguistics have shown that humans are sensitive to transitions between syllables, which are typically multi-phoneme units. Furthermore, at least at some level, adults seem to be sensitive to expectations about longer and even discontinuous sequences of syllables (Dilley and McAuley, 2008). On the other hand, almost all computational models of segmentation use predictability measures calculated only on consecutive phonemes. For example, although Brent (1999) notes that calculating TP and MI values on single-phoneme context does not reflect their full utility, he nevertheless calculates these values on the basis of single-phoneme context. Here, I will extend the analysis carried out in the previous subsections and discuss the effect of calculating predictability measures on larger sequences of initial phonemes.

Figure 7 presents a set of graphs that visualize the effect of increasing the length of preceding phoneme context, l to two and three. The figure also provides a direct comparison of the predictability measures discussed so far. In this figure, the first three columns display the distribution of the measures with changing phoneme context size between one and three. The last column compares the performance of segmentation algorithms using a single measure with varying phoneme context size. Performance comparison is presented using precision/recall graphs. The horizontal axes of these graphs are precision values, and the vertical axes are recall values. Perfect segmentation corresponds to upper right corner where both precision and recall are one. Otherwise, the closer the curve to the upper left corner, the better the performance is. In other words, a large *area under curve* is indication of a measure that performs well over a range of threshold values. The first four rows, separated by dotted lines, correspond to the measures: TP, MI, SV, H respectively. Each row contains two sub-rows of histograms, a top histogram depicts the distribution of the measure at boundary locations and a bottom histogram depicts the distribution of the measure at word-internal positions. The fifth row presents precision/recall graphs comparing measures that use the same context length.

Figure 7 demonstrates that increasing the context size increases the separation between the distributions of boundary and non-boundary locations. This is particularly visible for context size two, and measures TP, SV and H. The separation is not that clear for MI, and for context size of three. Although there is a general trend of increase with the context size, the increase of phoneme context from two to three does not seem to have a dramatic effect on the performance.. This trend is visible from the area under the precision/recall curves. Especially the precision/recall curves at the bottom row of Figure 7 demonstrate this clearly. The area under the curves increases in these graphs from left to right (by increasing phoneme context).

Figure 7 shows that increasing the phoneme context for all predictability measures affects how well they predict the word boundaries, making the measure more useful. However, an interesting question to ask is whether they give the same information or not. For example, does calculating TP conditioned on previous two phonemes give us all the information we get from calculating it by conditioning on a single previous phoneme? The question is important, because if different context sizes provide different information, than instead of using the higher context size, one can use both to achieve a better performance com-

of the field, he named it entropy (Tribus and McIrvine, 1971).

[6] Boundary entropy defined here is similar to but different from a well known entropy measure, conditional entropy, which is defined as $-\sum_{r \in A} P(r,l)\, log_2\left(P\left(r|l\right)\right)$. For example, Moberg et al. (2007) use conditional entropy to model phonetic recognition in semi-communication. In preliminary experiments conducted, the results obtained for both measures in segmentation task were similar. The boundary entropy is adopted here since it was used in previous research for segmentation (e.g., Hafer and Weiss, 1974).

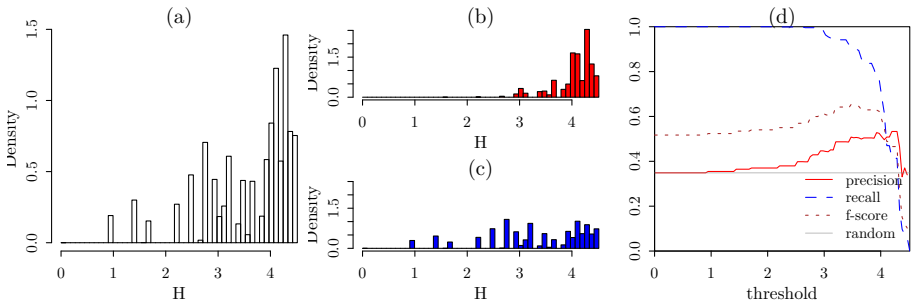


Figure 6: (a) Distribution of entropy. (b–c) Distribution of entropy for boundaries and word-internal positions, respectively. (d) Performance of algorithms that segment at locations where entropy is higher than a threshold value. The solid gray line in (d) represents the precision, recall and F-score of a pseudo-random segmentation method that inserts as many boundaries as in the gold-standard segmentation.

pared to the performance achieved by using the better of them. Using multiple context sizes is appealing, also because the unpredictability of word boundaries is due to their being dependent on different linguistic units, such as morphemes, syllables and phonemes. Changing the phoneme context size may capture regularities that exist because of different linguistic units. The relation between the phoneme context size and the linguistic units, of course, is not clear-cut. However, for example, it is likely that a context size of two or three captures more about relationships between syllables, while context size of one mostly captures the relationships between single pairs of phonemes. If we expect regularities at both levels, then we expect combination of different context sizes to be helpful.

Section 6 will investigate the effect of varying context size on an unsupervised segmentation algorithm. Here, I will provide some evidence that different context sizes provide different information. The evidence comes from the fact that if two sources of information contribute independently to evidence in favor of a certain conclusion, their correlation is expected to be lower when we know the conclusion is correct. They correlate in the first place because they measure the same quantity. However, given the conclusion, they should not be correlated if they make errors independently. If they are not completely independent, but still provide some independent information, we expect the correlation to be lower when the conclusion is known. Returning to the segmentation problem, if two context sizes, say one and two, used with the same measure provide independent information regarding word boundaries, we expect their correlation given we know there is a boundary to be lower than their correlation independent of the word boundaries.

Table 1 presents correlation coefficients for context sizes between one and three for all measures, for all possible boundary positions, and only for word boundaries. With some variability of the magnitude of the change depending on the measure, the correlations at word boundaries are lower than the correlations for the overall corpus. The results indeed indicate that the measures calculated using each phoneme-context size provide some information about the word boundaries that the other context-length options do not provide. This result (based on cases of genuine boundaries) gives some indication that using statistics with phoneme sequences of varying lengths may be useful for the segmentation task. The use of information from multiple measures calculated using varying context length will be investigated empirically in Section 6.

3.7. Predicting the past

Except MI, all three predictability measures discussed in this section are asymmetric. They take an initial sequence of phonemes, and measure the predictability of the next phoneme. Moreover, the SV and entropy measures do that without actually seeing the next phoneme. It is clear that the reverse quantities that measure the predictability of the previous phoneme given the current phoneme or phoneme sequence provide some additional information. Taking TP as an example, we know that $P(l|r) \neq P(r|l)$. If they are both useful for segmentation, using both measures is, in principle, better than using only one of them.

This section will show empirically that the reverse versions of the measures discussed so far are also good measures for segmentation. However, for a truly online-predictive system, predicting past events based on current information may seem odd. The justification of using reverse predictability measures for segmentation comes from two sources. First, intuitively, it seems that what we hear at a particular moment changes our interpretation of past input, especially if the previous interpretation is uncertain in some way. It is not unusual that when reading some text or listening to someone, things we read or heard start making sense only after we hear or read more. The second, more concrete evidence is from developmental psycholinguistics. Pelucchi et al. (2009) showed that eight-month-old infants (the same age as the infants in Saffran et al. (1996a) study) were able to track statistical regularities that are only possible to detect if they were sensitive to some reverse predictability measure between the successive syllables. Pelucchi et al. (2009) carefully selected words from a natural but unfamiliar language with sequences of syllables that differed only in their 'backward' transitional probabilities. Results were similar to Saffran et al. (1996a), confirming that infants do use backward predictability.

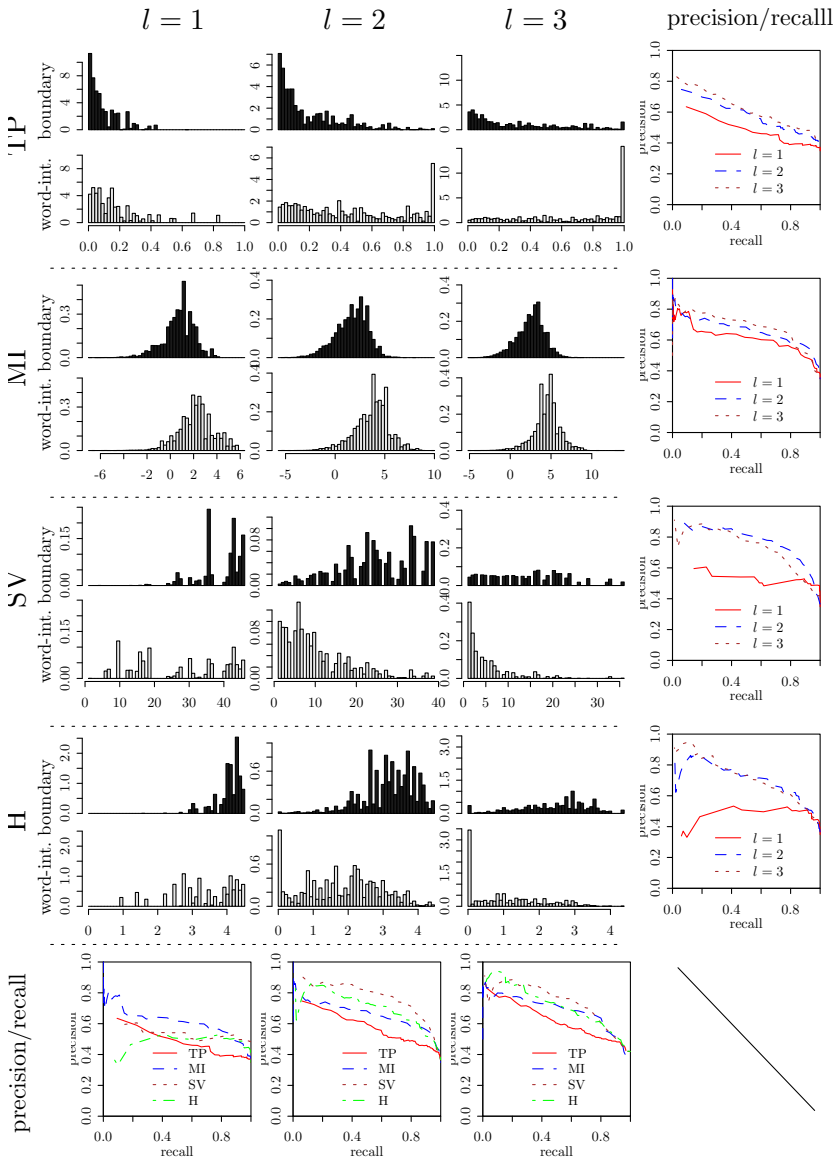


Figure 7: The effect of context on predictability measures. The first three columns in the first four rows (rows are separated by dotted line) present distributions of measure values for varying context size. The last column presents the precision/recall graphs for each context size. The last row presents the precision/recall values for each context for all measures.

	1	2	3
1	1.00	0.58	0.40
2		1.00	0.74
3			1.00

(a) TP all

	1	2	3
1	1.00	0.67	0.52
2		1.00	0.80
3			1.00

(b) MI all

	1	2	3
1	1.00	0.63	0.44
2		1.00	0.74
3			1.00

(c) SV all

	1	2	3
1	1.00	0.61	0.43
2		1.00	0.74
3			1.00

(d) H all

	1	2	3
1	1.00	0.55	0.33
2		1.00	0.64
3			1.00

(e) TP boundaries

	1	2	3
1	1.00	0.64	0.46
2		1.00	0.74
3			1.00

(f) MI Boundaries

	1	2	3
1	1.00	0.35	0.27
2		1.00	0.58
3			1.00

(g) SV boundaries

	1	2	3
1	1.00	0.38	0.20
2		1.00	0.56
3			1.00

(h) H boundaries

Table 1: Correlation coefficients for different phoneme context sizes for each measures. The top row gives the correlation coefficients over all boundary locations. The bottom row presents the correlation coefficients calculated only at boundary positions. The correlation coefficients are calculated after a log-transforming TP, SV and H, since log-transforming makes these distributions roughly normal, and reduces the effect of extreme values.

Since the direction does not make sense for MI,[7] only the reverse versions of TP, SV and H will be analyzed in this section. Reverse measures will be indicated by a subscript 'r' here. The reverse of TP and SV are sometimes abbreviated as BTP (backwards TP) and PV (for predecessor variety) in the literature. It is easy to deduce the definitions of reverse measures from the forward counterparts. The definitions are provided here for the sake of completeness.

$$\mathrm{TP}_r(l, r) = P(l|r) = \frac{P(lr)}{P(r)} \approx \frac{frequency(lr)}{frequency(r)} \tag{3}$$

$$\mathrm{SV}_r(r) = \sum_{l \in A} c(l, r) \tag{4}$$

where,

$$c(l, r) = \begin{cases} 1 & \text{if substring } lr \text{ occurs in the corpus} \\ 0 & \text{otherwise} \end{cases}$$

and A is the set of phonemes (the alphabet).

$$\mathrm{H}_r(r) = -\sum_{l \in A} P(l|r)\, log_2 P(l|r) \tag{5}$$

As can be seen in Figure 8, the reverse measures seem to achieve similar segmentation performances as their forward counterparts. From their mathematical formulation, it is clear that the forward and reverse versions of the measures are not equal to each other. $P(l|r) \neq P(r|l)$, and SV_r and H_r calculations do not even share the strings that they are calculated on with their forward counterparts. Like the analysis for varying phoneme-context length in Section 3.6, we can also check if correlation between forward and reverse version of these measures provide independent information. Since both are useful for detecting boundaries, they will naturally be correlated. However, if they provide some independent information, we would expect the correlation of the measures for the boundary locations to be lower than the correlation for the complete corpus. Indeed, the correlation coefficients for TP, SV and H and corresponding reverse measures on the BR corpus are 0.62, 0.15 and 0.21 respectively. When calculated only on boundary locations, the correlation coefficients, respectively, are 0.52, -0.01 and -0.06. The question as how to combine the forward and backward information efficiently still remains, to which we will return in Section 6.

3.8. Predictability measures: summary and discussion

So far, this paper has discussed four predictability measures: transitional probability, mutual information, successor variety and entropy. The reason all these measures work is that in an unsegmented speech stream, predictability inside the lexical units is high and predictability at the lexical unit boundaries is low. Our analysis is based on two consecutive sequences of phonemes l and r. In informal terms, TP measures how likely it is to observe r after l is observed. If TP is high, we expect to be within a unit, if TP is low it indicates a possible boundary. MI measures whether l and r are highly associated or not. Again if MI is high, we expect lr to be a word-internal sequence, otherwise a boundary position. The other two measures, SV and H, are measures of unpredictability (surprise). Hence, high values of SV and H indicate word boundaries. Another difference of these measures is that they are functions of only l. Informally, they try to answer the question 'how much do I (not) know about r after observing l?'. The difference between these two measures is in their sensitivity to the distribution of the sequences that follow l. Entropy is affected by the frequency of these sequences, while SV is oblivious to it.

All measures discussed in this section so far have some overlap in what they measure, but they are not the same. Most psycholinguistic studies consider TP as the measure of predictability, but the results from these experimental studies are

[7] This is not strictly true if phoneme sequences of unequal length are used for l and r. However, for ease of comparison this section only considers measures calculated on single phonemes.

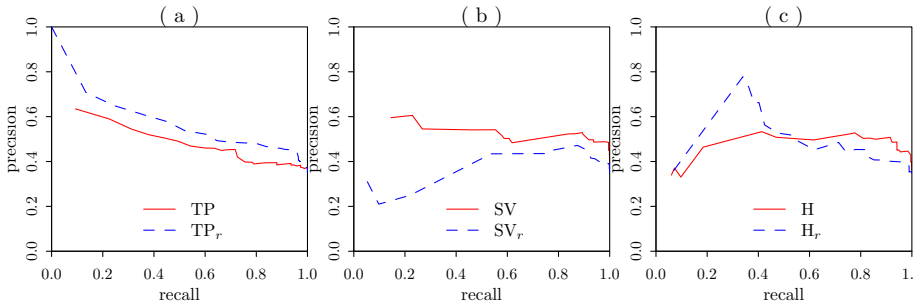


Figure 8: The precision/recall curves comparing the forward and reverse predictability measures: (a) TP and TP$_r$, (b) SV and SV$_r$, (c) H and H$_r$.

	TP	MI	SV	H	TP$_r$	SV$_r$	H$_r$
/bi-da/	1.0	3.4	1.0	0.0	1.0	1.0	0.0
/ku-pa/	0.5	2.4	2.0	1.0	0.5	2.0	1.0

Table 2: The predictability scores for syllable sequences /bi-da/ and /ku-pa/, given the sequence /bidakupadotigolabubidakugolabupadoti/ is observed. Note that for TP and MI lower values indicate word boundaries, while for SV and H higher values indicate word boundaries.

compatible with all four. For example, given a sequence similar to the stimuli presented to the infants in Saffran et al. (1996b) and subsequent studies, Table 2 presents the values of all measures discussed so far for two syllable pairs. One of the syllable pairs /bi-da/ is part of one of the artificial words /bidaku/ that form this sequence, while the other /ku-pa/ is not. Table 2 shows that, as expected, all measures indicate a higher chance for a word boundary between /bi-da/ compared to /ku-pa/. It would be interesting to see experimental results that would be compatible with only one of the measures but not the others. However, it is a difficult task to design such an experiment.

The analysis in this section showed that all the measures discussed here do something relevant to segmentation, all scoring consistently better than a random (but non-trivial) baseline. The performance analysis done by plotting precision/recall curves or by plotting precision, recall and F-scores gives an indication of the potential of a particular measure. The way they are used in an actual learning algorithm in combination with other information may result in different performance. Here, I will provide another way of looking at the similarities and differences of these measures before switching to explicit models of segmentation with concrete algorithms. Table 3 presents the correlation coefficients for all (forward) measures calculated on the BR corpus.

Table 3a confirms that all four measures are correlated. However, TP and MI are more strongly correlated with each other compared to their correlations with SV and H. Similarly SV and H are more strongly correlated with each other. Hence, the four measures fall into two groups: TP and MI in one, and SV and H in another. The correlations between the former and the latter

	TP	MI	SV	H
TP	1.00	0.77	-0.45	-0.40
MI		1.00	-0.51	-0.43
SV			1.00	0.76
H				1.00

(a) All phoneme pairs.

	TP	MI	SV	H
TP	1.00	0.77	-0.10	-0.13
MI		1.00	-0.13	-0.09
SV			1.00	0.82
H				1.00

(b) Boundaries.

Table 3: Correlation coefficients of predictability measures for all phonemes in BR corpus (a) and for the phoneme pairs that straddle a word boundary. The coefficients are calculated after log-transforming the TP, SV and H values.

group of measures is negative, since the former two measure predictability and the latter two measure unpredictability. Table 3b gives the correlation coefficients of the measures where a boundary is observed. This also reveals an interesting relationship between these groups of measures. Given boundaries, the correlations between the groups drop substantially, while correlations within the groups do not change much. This is an indication that the measures within the same group are highly dependent, while being relatively (conditionally) independent of the measures in the other group. Similar to the analysis provided for varying phoneme context size in Section 3.6, this is an indication that a learning algorithm that combines measures from different groups will gain additional information, while an algorithm that uses measures of the same sort will not.

This section provided an analysis of four measures of predictability (or unpredictability) for their use in lexical segmentation. All of them measure something relevant to segmentation as they all perform better than a random segmentation baseline. The analysis also showed that the use of additional context improves their performance, and it is useful to consider the reverse

of the asymmetric measures. Further analysis showed the similarities and differences between these measures. Before laying out an unsupervised model in Section 6, we will first introduce the data and evaluation method used in study in Section 4, and two baseline models in Section 5.

4. Data and Evaluation

As with other models of the acquisition of natural languages, we know rather little about our target, the human lexicon. However, everything else being equal, we would prefer the models that perform well against a theoretical *gold standard*. Furthermore, we need a quantitative measure of evaluation for comparing performances of different models. We will introduce the data used in this study followed by the definition of the evaluation metrics.

4.1. Data

The corpus used for testing the models in this paper is the corpus used by many recent studies. This corpus was collected by Bernstein Ratner (1987) and processed by Brent and Cartwright (1996). Following the convention in the literature the corpus will be called the *BR corpus*.

The original orthographic transcription of the corpus was converted to a phonemic transcription by Brent and Cartwright (1996). All words are transcribed the same at every occurrence, and onomatopoeia and interjections are removed. The BR corpus consists of 9790 utterances, 33,387 words, and 95,809 phonemes. A complete description of the corpus can be found in Brent (1999).

The BR corpus has been used by many other computational studies of segmentation. The corpus is also distributed with the implementation of the models presented by Venkataraman (2001) and Goldwater et al. (2009). The copies of the corpus in these sources are identical, and the same copy was used in this study without any modifications except for 12 boundary mismatches between segmentation of two words in the text version and phonemic transcriptions. The phonemic transcriptions of 10 instances of the word /ebisi/ 'ABC' and two instances of the word /Enim%/ 'anymore' have been modified to match the text version. In all cases, this resulted in removing boundaries in the instances of /e bi si/ and /Eni m%/.

4.2. Evaluation metrics

Two quantitative measures, *precision* and *recall*, originate in the information retrieval literature and have become the standards measures of evaluation of computational simulations. Precision can be seen as a measure of exactness, and it is sometimes called *accuracy* in the cognitive science literature.[8] Recall is a measure of *completeness*, and sometimes called so in cognitive science literature. In informal terms, high precision means that the model has found only correct items, but many relevant items might have been missed. High recall, on the other hand, means that the model has not missed anything, but it may have suggested many irrelevant items. To have a balanced indication, a derived measure, F_1*-score*,[9] is used, which is the harmonic mean of precision and recall.

$$\text{F}_1\text{-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

As in recent studies of computational segmentation, in this paper three different types of precision and recall values are distinguished.

- *Boundary* precision (BP) and boundary recall (BR) calculations consider the boundaries that match the gold standard segmentation as a *true positive* (TP), where the mistakenly proposed boundaries that do not exist in the gold standard are considered *false positives* (FP) and the boundaries that are in gold standard, but not spotted by the model, are considered *false negatives* (FN). Since utterance boundaries are clearly marked, not to give credit to the segmentation models for stating the obvious, the utterance boundaries are not included in calculation of the boundary scores. The F-score calculated using BP and BR will be denoted BF.

- *Token*, or word, precision (WP) and token recall (WR) scores require both boundaries of a word to be found to count positively in TP. Likewise, the words that are suggested by the model, but not in the gold standard, are FPs. The words that the model could not segment correctly are FNs. The token scores are naturally lower than the boundary scores. Similarly the F-score calculated from WP and WR will be denoted WF.

- *Type*, or lexicon, precision (LP), type recall (LR) and type F-score (LF) are similar to token scores, however, the comparisons are done over the word types the model proposed and word types in the gold standard. These scores are typically lower than the token scores. If a model does a good job only at segmenting high-frequency words (e.g., function words), type scores will be much lower than the token scores, but if the model is good at segmenting low frequency words as well, lexical scores will be closer to the token scores. In case the model is particularly bad at segmenting high-frequency words, but good at segmenting low-frequency words, the type scores can be higher than the token scores.

All segmentation models we are interested use unsupervised learning methods in the sense that the algorithms do not have access to information regarding real boundary locations. As a result, it is common practice to present the results on a single data set without training–test data separation.

[8] Unfortunately, accuracy is ambiguous in the cognitive science literature. Accuracy, as it is commonly used in many branches of science is different than precision.

[9] The subscript '1' indicates that the measure gives equal weights for precision and recall. In its more generic original formulation, F_α-score gives higher weight to recall for higher values of α, and lower values give higher weight to precision (van Rijsbergen, 1979).

Precision, recall and F-score are the standard measures that are well understood and widely used in the literature. However, it is often more insightful to study where the system fails. For this reason, I will describe two error measures relevant to segmentation, and report these measures along with the precision, recall and F-score values for the models developed in this study.

A segmentation error can be due to one of two reasons. First, the model may fail to detect a boundary, causing *undersegmentation*. Second, the model may insert a boundary where there is none, causing *oversegmentation*. The simple counts of oversegmentation and undersegmentation errors change depending on the size of the corpus. Hence, they are not comparable across the simulations that run on different corpora. Furthermore, in a typical corpora, there are more word-internal positions than boundaries. As a result, there are more chances to make an oversegmentation error compared to an undersegmentation error. To overcome these difficulties we will use the following error measures for oversegmentation and undersegmentation respectively:

$$E_o = \frac{\text{FP}}{\text{FP} + \text{TN}}$$
$$E_u = \frac{\text{FN}}{\text{FN} + \text{TP}}$$

where TP, FP, TN and FN are true positives, false positives, true negatives, and false negatives respectively.

In plain words, E_o is the ratio of the false boundaries inserted by the model divided by the total number of word internal positions in the corpus. Similarly, E_u is the ratio of boundaries missed to the total number of boundaries. Similar to the definition of F-score, one could also define a combined, single error measure, e.g., harmonic mean of E_o and E_u. Since reporting both measures is more informative, and the combined measures can be calculated from the two measures trivially, the combined measure will not be reported in this paper.

The two error measures described above are related to precision and recall, but the quantities cannot be derived from each other directly. Undersegmentation will reduce true positives which, in turn, reduce both precision and recall. Oversegmentation, on the other hand, will cause false positives to increase, which will affect precision adversely, but will not have an effect on recall. As a result, good recall and bad precision are a typical sign of oversegmentation, and bad precision and bad recall are likely to be due to undersegmentation. So, the error measures are related precision and recall to some extent, but it will be useful to examine them directly as well.

A last note about all the performance scores discussed in this section is that they take values between zero and one. However, to use the space available for significant digits more efficiently, it is common to present values in percentages. In this paper, all values in the tables are percentages, and the values in the graphs are absolute scores (between zero and one).

5. Two reference models

Ideally, the performance of a model of the human cognitive capacity should be evaluated based on its match with the human performance. From this perspective we should prefer models that segment as children do—including the incorrect segmentations of children. However, we currently lack the theoretical understanding, the data, and the tools to do this in a realistic way. In any case, everything else being equal, we prefer models that perform better at the task in question. This is reasonable, since language learners eventually segment quite well. To be able to evaluate our models, we need references that we can compare our model's performance to. A trivial way to show that a model does something relevant to the task at hand is to compare it with the model that makes random choices. A second method is to compare the model with a state of the art alternative. This section defines two such models that will serve as references for the models that are developed in this study.

5.1. A random segmentation model

A trivial random model can be defined as one which makes a random boundary decision for each possible boundary location. For a boundary guessing algorithm, performing consistently better than this model would already indicate that the algorithm is finding something relevant for the solution of the segmentation problem. However, it is customary (since Brent and Cartwright, 1996) in segmentation literature to set the bar a little bit higher. The typical random baseline used in computational segmentation literature inserts boundaries with the probability of boundaries in the actual corpus. In other words, it inserts as many boundaries as in the gold-standard segmentation, however, at random locations. Throughout this paper, performance scores obtained by this particular random model (RM) will be presented as a baseline reference. Note that the RM knows an important fact about the language that no other unsupervised models of segmentation know: the average length of words (estimated from the corpus studied). Although expected error rates E_o and E_u and boundary scores are easy to calculate for the RM, the direct calculation of the word and lexicon scores is not trivial. Table 4 presents all performance scores discussed in Section 4 for both random procedures.

Since the RM inserts boundaries at random, its performance is varied. This variation is expected to be small for a large enough corpus. However, for additional reassurance, the results reported for RM baseline are obtained by averaging of 50 runs over the relevant corpus.

5.2. A state-of-the-art reference model

Differing theoretical and practical motivations aside, most successful computational models use a strategy based on *language models* in computational linguistics. Albeit simple, a typical example of this strategy is described by Equations 6 and 7. The model described here, which we will call LM (for language-modeling based model), assign probabilities to possible segmentations as described in Equations 6 and 7.

	boundary			word			lexicon			error	
model	P	R	F	P	R	F	P	R	F	E_o	E_u
random	27.4	50.0	35.4	8.6	13.6	10.5	7.4	38.1	12.4	50.0	50.0
RM	27.4	27.0	27.2	12.6	12.5	12.5	6.0	43.6	10.5	27.1	73.0

Table 4: Performance scores and error rates of two random segmentation strategies. The scores in the first row are obtained by a random algorithm that decides for boundaries with probability 0.5. The RM algorithm, as described, inserts boundaries with the probability of observing boundaries in the reference BR corpus. The scores presented are average of 50 runs, standard deviations for all scores were less than 0.01.

$$P(s) = \prod_{i=1}^{n} P(w_i) \tag{6}$$

$$P(w) = \begin{cases} (1-\alpha)f(w) & \text{if } w \text{ is known} \\ \alpha \prod_{i=1}^{m} P(a_i) & \text{if } w \text{ is unknown} \end{cases} \tag{7}$$

where w_i is the i^{th} word in the sequence (utterance or corpus), a_i is the i^{th} sound in the word, and α is the only parameter of the model. We turn to a description of how α functions.

For the incremental model defined here, a word is 'known', if it was used in a previous segmentation. The model accepts whole utterances as single words if the utterance does not contain any known words.

Major improvements over the LM in the segmentation literature include use of larger word context, e.g., bigrams and trigrams, to calculate the known probabilities (e.g. Goldwater et al., 2009) or using more elaborate models of phonotactics (Blanchard et al., 2010). However, these improvements bring rather small increase in the performance (Table 5 compares performances of some of the models in the literature with the LM). The performance differences, when observed, are also likely to be due to processing and search strategies as well as the way the scores are calculated.

In this modeling setup, α can be interpreted as the weight given to novel words. If α is large, the novel words get higher probability. If α is small, known words are more preferable. This probability can be estimated from type/token ratio (i.e., ratio of the number of novel words seen so far to the number all words seen so far). Some models in the literature (e.g., Venkataraman, 2001) use this intuition to remove the free parameter α. Even though a parameter-free model is indeed more desirable, the relationship between the value of α and segmentation performance is not trivial. Nevertheless, for most values of α, the performance of the LM is competitive with the recent models in the literature, performing better at some scores (see Çöltekin, 2011, chapter 5 for effects of varying α on segmentation performance). In this paper, all results reported for the LM is with α set to 0.5 (although, one can achieve slightly better performance by fine-tuning α).

The LM, as defined here, shares the basic structure of state-of-the-art segmentation models, and it achieves competitive results with other segmentation models on the known benchmark corpus. As a result, it serves as a good reference model.

As an added benefit of reimplementing the reference model, Table 5 also reports the error scores described in Section 4. Furthermore, it also enables us to investigate an incremental model's performance over time. Notice that the best performing model in Table 5 is the batch Bayesian model presented by Goldwater et al. (2009). Besides the modeling practice used, there are two more reasons why this model can perform better than an incremental model. First, since it has access to complete data, in principle, it can arrive at generalizations that are consistent with the complete corpus. Second, the performance of the incremental models in the same table includes the initial output of the learning process where errors are expected. A batch model, on the other hand, outputs its results after the learning process is completed. To demonstrate the performance of the LM with the increasing input, Figure 9 presents the performance scores plotted for each 500-utterance block during the learning process. The first value of each score in this graph is calculated using first 500 utterances, the second value is calculated on 501^{th} utterance to 1000^{th}, and each successive score is calculated on the next 500-utterance block. Since the corpus contains 9790 utterances, the last scores in this graphs are calculated using the last 290 utterances. As expected, the performance scores increase and errors drop as the learning progresses. It also seems that the learning is fast, since, after the third or fourth block, the scores stabilize. At the end of the last phase of the learning from this corpus, the performance scores of the LM are substantially better than the performance scores calculated on the output of the model during the complete learning process (BF=89.0%, WF=80.6%, LF=74.0%, E_o=4.4%, E_u=11.1%). And in fact, these performance scores are also higher than the performance scores reported in Goldwater et al. (2009).

A possible objection to reporting the performance scores for last 290 utterances is that the scores can be a result of idiosyncrasies of this particular small sample. Figure 9 shows that despite slight fluctuations, the scores obtained for earlier blocks of 500 utterances are also similar, and analysis provided in Çöltekin (2011) provides further assurances that the results are not due to chance effects.

The LM and the related models set a high standard of performance to achieve. However, this modeling paractice has a few shortcomings as a model of of human performance. The main shortcoming, as argued in Section 1, is that this strategy does not follow what we know about how humans go about

	boundary			word			lexicon			error	
model	P	R	F	P	R	F	P	R	F	E_o	E_u
Brent (1999)	80.3	84.3	82.3	67.0	69.4	68.2	53.6	51.3	52.4	–	–
Venkataraman (2001)	81.7	82.5	82.1	68.1	68.6	68.3	54.5	57.0	55.7	–	–
Goldwater et al. (2009)	90.3	80.8	85.2	75.2	69.6	72.3	63.5	55.2	59.1	–	–
Blanchard et al. (2010)	81.4	82.5	81.9	65.8	66.4	66.1	57.2	55.4	56.3	–	–
LM	84.1	82.7	83.4	72.0	71.2	71.6	50.6	61.0	55.3	5.9	17.3

Table 5: Performance scores and error rates of the baseline model LM (with $\alpha = 0.5$) in comparison to the other models using the similar strategy. The performance scores for other models are listed as reported in the related research. If there were multiple models reported in a study, the model with the highest lexicon F-score is presented. All scores are obtained on the BR corpus.

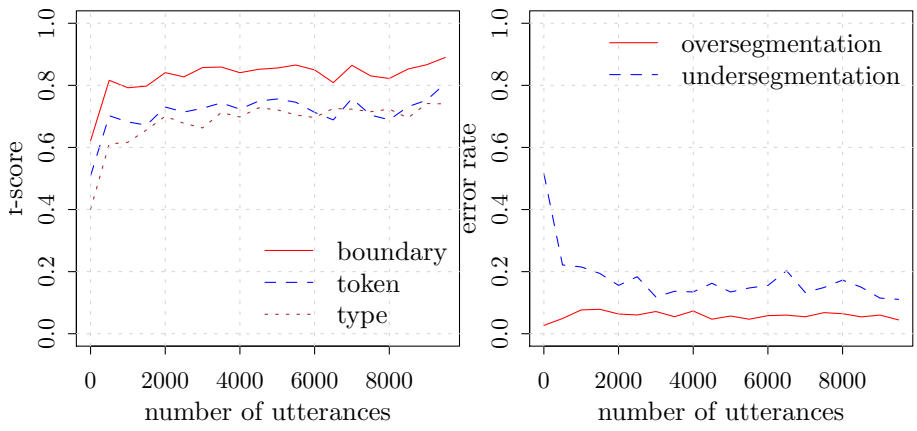


Figure 9: (a) Boundary, word token and word type F-scores and (b) oversegmentation and undersegmentation rates of the LM on the BR corpus for successive blocks of 500 utterances each.

solving the segmentation problem. The model that we will develop next provides a better fit to humna performance by (1) directly using a cue, predictability, known to be explited by infants, (2) following a strictly incremental processing model, and (3) providing an easy way to integrate other arbitrary cues.

6. A predictability based segmentation model

Existing predictability-based computational models of segmentation typically use a single measure of predictability calculated on single phoneme (and rarely syllable) contexts. However, the analysis of child-directed utterances in Section 3 indicates that the four measures discussed (transitional probability, mutual information, successor value and boundary entropy) are useful indicators of word boundaries. This analysis has also shown that even though these measures are similar in many ways, they measure different aspects of the input. As a result, the combination of these measures should help finding boundaries more than each measure alone. Another aspect discussed during this analysis is the effect of the phoneme context, which is also shown to affect the performance of the measures. According to the analysis, increasing the number of phonemes that the measures are calculated on, and combining measures calculated on varying context size is expected to increase the performance. Section 3 presented the effectiveness of each measure using a simple threshold based algorithm, leaving the development of an unsupervised algorithm that combines information from multiple sources for later. This section aims to fulfill this promise by developing an unsupervised algorithm for learning lexical units from continuous speech.

Before describing the segmentation method developed in this study, it should be stressed that the aim of this modeling practice is twofold. First, we would like achieve good performance using only predictability. Second, we would like to propose a model that is useful for understanding human-like segmentation starting from no initial knowledge of the lexical units of the target language. The use of predictability, a cue known to be used by adults and children, takes a step in this direction. However, there are other aspects that motivate the modeling practice here. First, although we focus on predictability in this study, we keep in mind that humans use multiple cues, and a plausible model of segmentation should be able to make use of the cues available to children. Second, an unsupervised and strictly incremental model should follow what we know about human segmentation well. The remainder of this section will describe a model motivated by these concerns in a number of incremental steps.

6.1. Peaks in unpredictability

Besides the non-trivial problem of choosing a threshold, the segmentation algorithms based on thresholds do not exploit the relation between predictability and lexical units fully. Deciding for a boundary when an unpredictability measure exceeds a threshold (or equivalently a predictability measure is less than a threshold) is in line with the idea that predictability is low between the lexical units. However, the thresholds do not directly utilize the fact that predictability is high within the units. In the following pages a completely unsupervised strategy that explicitly attends to high predictability within the units and low predictability between the units will be discussed. That is, this strategy posits a boundary if an unpredictability measure at the position is greater than the measure before and after the position. Following the previous research (e.g., Harris, 1955; Hafer and Weiss, 1974), I will call the strategy *peak-based* predictability strategy. However, it should be stressed that the term peak is valid for only unpredictability measures, such as SV and H. For predictability measures such as TP and MI, we look for 'troughs' rather than peaks. As well as reflecting the intuition 'high predictability within the words, low predictability between words', the peak-based segmentation strategy is also completely unsupervised: we do not need to tune any parameters, or use any labeled data where word boundaries are segmented.

Figure 10a presents values for all the measures discussed in this paper for each possible boundary position in the utterance /IzD&t6kIti/ 'is that a kitty'. The measures calculated for the beginning and the end of the utterance are useful for discovering peaks at neighboring positions, but, for a segmentation algorithm, there is no point in trying to discover boundaries at these locations. The values where the peak strategy suggests a boundary for each measure are indicated with boldface. Figure 10b represents the values for MI and H graphically.

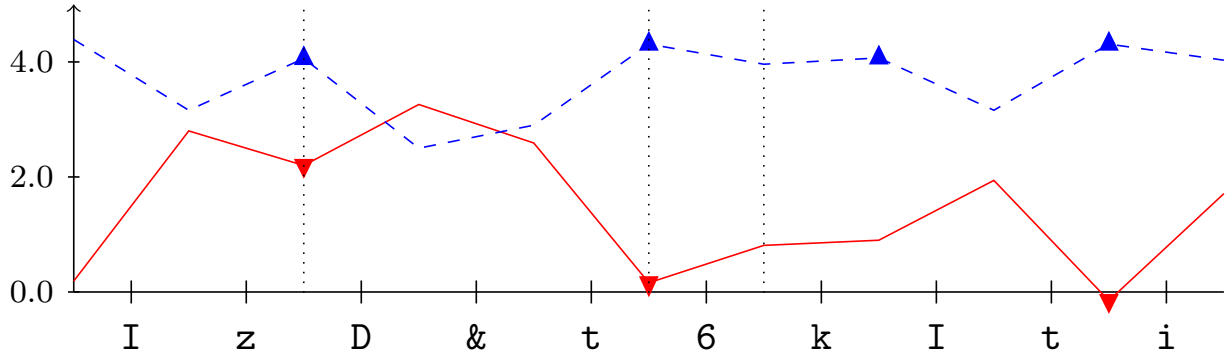
The measures presented in Figure 10 are calculated using single phoneme contexts. That is, the sequence l and when required the sequence r are taken to be single phonemes. As a result, the performance of a peak based-segmentation algorithm is bound to be adversely affected by the short context length. Since SV, SV_r, H, and H_r are functions of only l or only r, their performance is particularly low. However, unlike the threshold strategy which gives the same decision before or after a certain phoneme, the peak strategy considers the surrounding values as well. As a result, even with short context used for calculating the measure, the segmentation decision is affected by a larger surrounding context.

Even though the benefits of peak strategy for discovering boundaries are clear, there are a few weaknesses to note here. First, the peak-based boundary decision is rather conservative. It requires both sides of the boundary candidate to have the right kind of slope. Even a very sharp increase on one side will be discarded unless it is followed by a fall. Considering that most of the measures we discussed here are asymmetric, and their indications are stronger in one direction than the other, the problem certainly deserves some attention. This problem becomes more serious for single-phoneme words. Since the peak-based algorithm never makes two boundary decisions in a row, it never detects single-phoneme words. This problem will be revisited in Section 6.4. A second problem that I will leave relatively unexplored in this study is the fact that peaks do not take into account how steep the slopes are. Intuitively, the sharper the slope the higher the expected boundary indication. However, the peak-based boundary strategy used here ignores this fact.

The peak-based segmentation method that is demonstrated informally in Figure 10 can easily be implemented as an un-

	I	z	D	&	t	6	k	I	t	i	
TP	0.05	0.17	**0.15**	0.29	0.39	**0.03**	0.06	0.08	0.25	**0.03**	0.26
MI	0.19	2.80	**2.20**	3.26	2.59	**0.16**	0.81	0.90	1.94	**-0.15**	1.71
SV	38	19	**43**	10	30	**46**	36	**43**	19	**46**	42
H	4.39	3.16	**4.06**	2.50	2.90	**4.31**	3.96	**4.07**	3.16	**4.31**	4.03
TP_r	0.09	0.29	**0.11**	0.31	0.18	0.07	**0.05**	0.06	0.16	**0.06**	0.09
SV_r	42	34	38	40	**41**	40	40	**42**	41	33	39
H_r	4.47	3.57	**3.80**	3.37	**3.87**	3.50	4.06	**4.47**	3.87	4.12	4.41

(a) Example predictability scores



(b) Graphical representation of MI and H values.

Figure 10: Predictability measures for example utterance /IzD&t6kIti/ 'is that a kitty'. (a) presents all predictability measures discussed in this paper calculated on the BR corpus using single-phoneme context. The values where unpredictability peaks are marked with boldface. (b) represents a graphical representation of the MI (solid line) and the H (dashed line) values for the example utterance. Dotted vertical lines mark expected boundary locations, and the triangles mark the positions where the measures indicate a boundary according to peak criterion. Note that 'troughs' rather than peaks are indications boundaries for MI.

supervised segmentation algorithm for each measure alone. A possible realization of the peak-based segmentation is described in Algorithm 1. For all measures, the algorithm essentially follows the same steps. The predictability measures for each phoneme position in the utterance are calculated using the definitions given in Section 3. Unlike the values presented in Figure 10, the calculation of measures is not done using the complete corpus. The frequencies of phonemes and phoneme pairs are updated in an incremental fashion, using only the corpus seen so far. The beginnings and ends of the utterances are treated as special phonemes for the calculation of the measures, and otherwise the utterance boundaries are not used as separate cues.

Table 6 presents the results obtained on the BR corpus for each predictability measure by Algorithm 1 in comparison to the random baseline (RM) and the reference recognition algorithm (LM) described in Section 5.

Results in Table 6 indicate clearly that the performance of the peak-based prediction strategy as used here is far behind the LM. However, the results also show that the algorithm performs consistently better than random for all measures. As it will be discussed next, this is all we need to know about these measures for now.

Using peaks in unpredictability, Algorithm 1 exemplifies a completely unsupervised method of segmentation. However, two other problems raised in Section 3, combination of measures and making use of larger phoneme context, are still left

Algorithm 1: A peak-based segmentation algorithm.

Input: A sequence of utterances without word boundaries
Output: The sequence of utterances with boundaries

```
foreach utterance u in the input do
    foreach phoneme position i in u do
        Update frequencies of phoneme_i, phoneme_{i+1}
        and phoneme-pair_{i,i+1};
        P_i ← predictability value between i and i + 1;
        if P_{i-2} > P_{i-1} and P_{i-1} < P_i then
            insert a boundary between phoneme_{i-1} and
            phoneme_i;
        end
    end
    output the segmented utterance ;
end
```

	boundary			word			lexicon			error	
measure	P	R	F	P	R	F	P	R	F	E_o	E_u
TP	57.6	68.9	62.7	42.8	48.7	45.6	15.0	37.2	21.3	19.2	31.1
MI	66.3	74.1	70.0	52.2	56.6	54.3	18.5	42.5	25.8	14.3	25.9
SV	49.3	53.4	51.3	34.3	36.3	35.3	12.3	38.1	18.5	20.7	46.6
H	51.3	56.5	53.8	38.1	40.8	39.4	13.8	38.8	20.4	20.3	43.5
TP_r	53.3	67.5	59.6	36.3	43.1	39.4	14.4	35.5	20.5	22.4	32.5
SV_r	36.7	40.0	38.3	22.7	24.1	23.4	8.4	32.3	13.3	26.0	60.0
H_r	43.5	49.6	46.3	28.9	31.7	30.2	10.1	33.7	15.6	24.4	50.4
RM	27.4	27.0	27.2	12.6	12.5	12.5	6.0	43.6	10.5	27.1	73.0
LM	84.1	82.7	83.4	72.0	71.2	71.6	50.6	61.0	55.3	5.9	17.3

Table 6: Boundary/word/lexicon precision/recall/F-score values and oversegmentation and undersegmentation errors for the peak-based segmentation algorithm on the BR corpus. RM represents a pseudo-random segmentation that inserts a word boundary with the probability of word boundaries in the gold-standard segmentation. The LM is the recognition-based reference model. Both models are described in Section 5. The performance and error scores are described in Section 4.

unanswered. The next subsection will offer solutions to these problems, starting with the former.

6.2. *Combining multiple measures and varying phoneme context*

The discussion so far supports the expectation that using multiple measures and varying context size may be beneficial for segmentation performance. Using multiple measures is expected to be better than a single one, since, even though they have a lot in common, each measure seems to be measuring some aspects of the input that the others do not. It was also shown in Section 3 that the phoneme context size makes a difference in the performance of all measures. Furthermore, combining the measures calculated on varying phoneme context size was also conjectured to be useful. Here, Algorithm 1 will be extended to handle multiple sources of information coming from multiple measures calculated on varying phoneme-context length.

In its essence, the peak-based segmentation method presented in Algorithm 1 is a binary classifier. It classifies each possible boundary position in an utterance as boundary or non-boundary. Using different measures results in multiple classifiers that do the same task. Viewing the problem as combining a number of classifiers for achieving a better performance than each individual classifier is a relatively well studied problem in the machine learning literature, where the sets of classifiers are known as *ensembles* or *committees* (e.g., Bishop, 2006, chapter 14). For an effective combination, the classifiers should be accurate and diverse (Hansen and Salamon, 1990). Accuracy refers to the requirement that the individual classifiers perform better than random. Diversity is taken as the requirement that, to some extent, the classifiers are independent. Most combination methods in machine learning, such as *bagging* and *boosting*, are typically suitable for supervised classifiers. However, the field offers a set of practical and theoretical tools for the problem at hand. Here a simple and well-known method, *majority voting*, will be used for combining the multiple measures for segmentation in an unsupervised fashion.

As well as machine learning applications, majority voting is also a common (and arguably effective) method in everyday social and political life. As a result it has been well studied, and known to work well especially if the accuracy and the diversity requirements are met. A theoretical justification of majority voting is given by well-known 'Condorcet's jury theorem' which dates back to late 18$^{\text{th}}$ century (Boland, 1989). Provided that each member's decision is better than random, and the votes are cast independently, the Condorcet's jury theorem states that the probability that a jury arrives at the correct decision by majority vote monotonically approaches to one as the number of members is increased. Informally, this states that in the long run the decision of a large number of less competent individuals is better than the decision of a single individual with the greatest competence. In practice, even though the votes are almost never independent (especially in the social scene) majority voting is still an effective way of combining outcomes of multiple classifiers (see Narasimhamurthy, 2005, for a recent review and a discussion of the effectiveness of the method).

Majority voting provides a simple way to incorporate information from multiple and (somewhat) independent measures and the information provided by calculating these measures on varying context size. Instead of calculating a single value for a measure and for a given context size, we can calculate multiple values for multiple measures with multiple context sizes. Each *measure–context size* pair forms a voter. If there is a peak in unpredictability according to this pair, we get a boundary vote. If the majority of the voters vote for a boundary for a possible segmentation position, we insert a boundary at that position. Algorithm 2 describes this version of the segmentation method using majority voting. For the forward measures, context size defines the length of the sequence l, while for the reverse measures context size defines the sequence r. For all boundary positions, the number of votes Algorithm 2 considers is equal to 'the maximum context size' times 'the number of measures'. For example, assuming that we run the algorithm only for H and H_r with the maximum context size of two, and the algo-

Algorithm 2: The majority voting algorithm for multiple measures and multiple context size. The function $m()$ at line 9 calculates the predictability score (hence, unpredictability measures are multiplied by -1) according to measure m on given sequences of phonemes. If the required n-gram is not available, the algorithm backs off to the n-gram with the highest available rank.

Input: A sequence of utterances without word boundaries and the maximum context size M
Output: The sequence of utterances with boundaries

```
1  foreach utterance u in the input do
2      for n = 1 ... M + 1 do
3          update n-gram frequencies for the n-grams in u;
4      end
5      foreach phoneme position i in u do
6          votecount ← 0;
7          foreach measure m do
8              foreach context size n = 1 ... M do
9                  P_i ← m(n-gram ending at i-1, phoneme_i)
                   ;
10                 if P_{i-2} > P_{i-1} and P_{i-1} < P_i then
11                     votecount ← votecount + 1 ;
12                 else
13                     votecount ← votecount − 1 ;
14                 end
15             end
16         end
17         if votecount > 0 then
18             insert a boundary between phoneme_{i-1} and
               phoneme_i;
19         end
20     end
21     output the segmented utterance ;
22 end
```

rithm is about to decide if there is a boundary after *ki* in *akitty*, it checks each condition

1. $H(i) > H(k)$ *and* $H(i) > H(t)$
2. $H_r(i) > H_r(k)$ *and* $H_r(i) > H_r(t)$
3. $H(ki) > H(ak)$ *and* $H(ki) > H(it)$
4. $H_r(ki) > H_r(ak)$ *and* $H_r(ki) > H_r(it)$

Then, the algorithm increases the vote count by one for each condition met. If the vote count is greater than half of the votes (two in this case) it inserts a boundary.

The results of combining all measures with varying context size using majority voting on the BR corpus are presented in Table 7. Each row in the table lists the common segmentation scores we use in this paper for context size between one and eight. Maximum context size one means that the measures are calculated with single phoneme context. As a result, the scores in the first line of Table 7 are obtained by the majority decision of seven voters (TP, MI, SV, H, TP_r, SV_r and H_r, all calculated on single phoneme context), while the scores in line two are obtained by the majority decision of 14 voters, each representing context sizes one or two for all seven measures.

The results certainly improve compared to single-measure segmentation results presented in Table 6. Some of the scores also exceed the performance of the LM, the state-of-the-art reference model. The performance of the majority voting algorithm is good at spotting boundaries and words. The boundary and word precision scores are consistently better than the corresponding recall scores. When increasing maximum context parameter, both precision and recall increase at first. This is expected since we incorporate information from higher level n-gram frequencies that are good predictors of the boundaries. After context length three, the recall starts to go down, while precision still gets better with the increased parameter value. Since the increased number of voters requires a higher consensus, it is natural that the precision is high. However, the higher number of voters also means that the disagreement on real boundaries will also increase. As a result recall drops. With the decreased boundary recall, the word and lexicon precision start going down as well. One of the reasons for this may be because higher level n-grams suffer from data sparseness, so that the voters that use higher level n-grams start to become less competent. As a result, increasing the number of voters that calculate the results on higher level n-grams violates the requirement of the successful combination that the individual voters need to perform better than random.

Despite being precise at spotting boundaries (and as a result words) the majority voting algorithm is still bad at lexical precision. The low lexical scores mostly stem from two causes. The first reason has to with the fact that this algorithm does not build and use an explicit lexicon. As a result it does not get any reward for reusing the previously discovered lexical items. Second, the algorithm starts with no prior knowledge at all, and it takes time to build useful n-gram statistics. Until a reasonable amount of statistics is collected, many wrong word-types are inserted into the lexicon, and this affects the lexical precision adversely.

	boundary			word			lexicon			error	
max. context	P	R	F	P	R	F	P	R	F	E_o	E_u
1	73.6	68.2	70.8	57.3	54.3	55.8	16.7	49.5	24.9	9.2	31.8
2	86.6	72.9	79.2	70.7	62.8	66.6	22.0	58.6	32.0	4.3	27.1
3	89.7	77.5	83.1	75.6	68.3	71.8	27.7	63.3	38.6	3.4	22.5
4	93.4	73.6	82.3	76.4	65.0	70.2	26.1	63.1	36.9	2.0	26.4
5	94.1	72.2	81.7	76.3	63.7	69.4	26.2	64.0	37.2	1.7	27.8
6	94.9	66.1	77.9	73.4	57.7	64.6	22.8	61.7	33.3	1.4	33.9
7	95.1	63.5	76.2	72.4	55.5	62.8	21.4	60.1	31.5	1.2	36.5
8	95.4	58.7	72.7	70.3	51.2	59.2	19.6	58.3	29.3	1.1	41.3
RM	27.4	27.0	27.2	12.6	12.5	12.5	6.0	43.6	10.5	27.1	73.0
LM	84.1	82.7	83.4	72.0	71.2	71.6	50.6	61.0	55.3	5.9	17.3

Table 7: Performance and error scores for peak-based majority voting algorithm with varying context. Two reference models, the RM and LM are defined in Section 5. The performance and error scores are described in Section 4.

6.3. *Weighing the competence of the voters*

The majority voting algorithm presented in Section 6.2 treats all the voters equally. Even though this may be a virtue in the social and political context, it is a shortcoming for a learner. A better learner is expected to identify the value of the information provided by each source, and increase the weight of the sources that perform well consistently. Weighted majority voting is an extension of the majority voting algorithm which weighs the vote of each source according to their competence (Littlestone and Warmuth, 1994).

For the particular instantiation of the weighted majority voting algorithm used here, we will first assign a weight, w_i, in range $[0, 1]$ to each voter. Second, instead of increasing or decreasing a vote count by one, we will increase or decrease the vote count by w_i. To do that we replace line 11 in Algorithm 2 with '$votecount \leftarrow votecount + w_i$' and replace line 13 with '$votecount \leftarrow votecount - w_i$'. The rest of the segmentation algorithm is essentially the same. Note that if all weights are set to one, the algorithms are equivalent.

So far we have described how to adjust the majority voting algorithm to be able to weigh its sources of information. However, we also need a way of setting the weights, so that they reflect the usefulness of the particular voter's decision. As with many examples in the literature, we will set all the weights to one at the beginning. After each decision, we will update the weights. In supervised models, where exact error is known, one can adjust weights in a way to reduce the error. Here we do not know boundary locations, and we cannot be certain about which decisions are correct. However, we will take the (weighted) majority decision as the correct decision. That is, if the voter agrees with the majority decision, we count this as a correct decision, and if it disagrees we will assume that it is an error. To finalize our adjustments to Algorithm 2, we keep count of errors made by each voter i, e_i, which is incremented when the voter does not agree with the majority decision. After every boundary decision, first the error counts are updated for each voter. Then, the weights w_i, of all voters are updated using,

$$w_i \leftarrow 2\left(0.5 - \frac{e_i}{N}\right)$$

where N is the number of boundary decisions so far, including the current one.

This update rule sets the weight of a voter that is half the time wrong (a voter that votes at random) to zero, eliminating the incompetent voters. If the votes of a voter are in accordance with the most of the voters almost all the time, the weight stays close to one.

The performance scores of the weighted majority algorithm for the maximum phoneme context parameter between one to eight on the BR corpus are presented in Table 8. In general weighted majority voting algorithm performs slightly better than majority voting algorithm. The performance of the algorithm can be improved by further extensions, for example, by using a better method for setting weights, or using modified versions of peak-based boundary detection. However, the purpose of the current work is not only to find a well-performing performing segmentation algorithm, but also proposing an explicit model of segmentation that can combine information from multiple sources. The weighted version of the algorithm is more attractive in this regard. First, it makes it easy to include possibly irrelevant sources of information. If they are irrelevant, they will be left out by the weight update procedure reducing their weights to zero. Second, it may explain certain shifts during learning. For example, a weak cue that is not very useful before enough input is seen may become stronger in time, as its predictions become more effective with the additional information. In other words, a weak source of information may be bootstrapped by other sources if the information it collects is relevant in the long run.

6.4. *Two sides of a peak*

Some of the performance scores of both the weighted and not-weighted version of the models described in this section exceeds the performance of the LM, but in general the model described so far is slightly behind the LM in performance comparison. This is related to a problem mentioned earlier. Section 6.1 pointed out a particular weakness of the peak criterion defined here. It is too conservative, and in some cases this is a

	boundary			word			lexicon			error	
max. context	P	R	F	P	R	F	P	R	F	E_o	E_u
1	72.1	71.7	71.9	57.0	56.8	56.9	17.4	48.3	25.6	10.5	28.3
2	83.7	77.6	80.5	70.3	66.6	68.4	25.1	59.4	35.3	5.7	22.4
3	89.3	78.2	83.4	75.6	68.9	72.1	28.0	62.8	38.8	3.5	21.8
4	92.7	76.0	83.5	77.2	67.4	72.0	28.4	65.1	39.6	2.3	24.0
5	94.1	71.4	81.2	75.8	62.8	68.7	26.3	64.8	37.4	1.7	28.6
6	94.7	66.8	78.3	73.9	58.5	65.3	23.5	63.2	34.3	1.4	33.2
7	95.1	62.1	75.2	71.9	54.3	61.8	21.1	60.6	31.3	1.2	37.9
8	95.1	58.5	72.4	70.2	51.1	59.1	19.6	58.5	29.4	1.1	41.5
RM	27.4	27.0	27.2	12.6	12.5	12.5	6.0	43.6	10.5	27.1	73.0
LM	84.1	82.7	83.4	72.0	71.2	71.6	50.6	61.0	55.3	5.9	17.3

Table 8: Performance and error scores for the peak-based weighted majority voting algorithm with varying context. Two reference models, the RM and LM are defined in Section 5.

	boundary			word			lexicon			error	
max. context	P	R	F	P	R	F	P	R	F	E_o	E_u
1	52.5	89.2	66.1	34.2	51.2	41.0	24.9	30.3	27.3	30.5	10.8
2	63.7	92.5	75.4	49.6	65.4	56.4	34.3	39.7	36.8	19.9	7.5
3	72.4	92.7	81.3	60.5	72.5	66.0	36.8	50.8	42.7	13.3	7.3
4	79.8	90.3	84.7	68.5	74.9	71.5	38.1	60.6	46.8	8.7	9.7
5	84.0	85.6	84.8	71.8	72.8	72.3	34.8	65.9	45.5	6.2	14.4
6	86.2	80.2	83.1	72.5	69.0	70.7	30.2	66.0	41.4	4.8	19.8
7	87.5	75.1	80.9	72.2	64.9	68.4	26.2	63.6	37.1	4.0	24.9
8	88.1	70.8	78.5	71.2	61.3	65.9	23.8	61.7	34.3	3.6	29.2
RM	27.4	27.0	27.2	12.6	12.5	12.5	6.0	43.6	10.5	27.1	73.0
LM	84.1	82.7	83.4	72.0	71.2	71.6	50.6	61.0	55.3	5.9	17.3

Table 9: Performance and error scores for the peak-based weighted majority voting algorithm that incorporates the information from local changes at the both sides of the boundary candidate. Two reference models, the RM and LM are defined in Section 5. The performance and error scores are defined in Section 4.

serious problem. For example, since there cannot be two peaks in a row, it can never find single-phoneme words. This is also evident in the performance scores presented so far, all combinations presented have high precision (and low oversegmentation error), but low recall (and high undersegmentation error).

The solution to this problem has been delayed up to this point since the combination methods described in previous sections provide a natural approach to solve it. We can interpret the increase or decrease of uncertainty on either side of a boundary separately. The majority voting algorithm can easily incorporate these additional voters' decisions. The weighted majority voting provides an additional reassurance by eliminating useless votes. Furthermore, since most of the measures discussed here are asymmetric, their indication in one direction is stronger. For example, one expects TP to give better indications while processing the stream left-to-right, so on the left side of a boundary candidate. On the contrary, TP_r should provide a better indication on the right side. Weighted combination will automatically discover the value of these decisions.

As a result, the last improvement to the boundary discovery algorithm discussed here is to incorporate the local changes on the two different sides of a boundary candidate as separate voters in the weighted majority voting algorithm. Table 9 presents these results for varying maximum context size parameter. As expected, undersegmentation errors get lower, but this comes with the cost of higher oversegmentation errors.

In comparison to previously presented results the benefit of this approach may not be immediately clear. It seems the approach trades the oversegmentation for reduced undersegmentation. However, as well as in the increased lexical performance seen in Table 9, the benefit of this more eager segmentation approach is particularly useful when more varied cues are added.

Following the evaluation strategy described in Section 4, Figure 11 presents change of F-score and error values for the final model with maximum context size set to 3 for each 500 utterance block of the BR corpus. For the last 290 utterances, the performance scores are significantly better than the scores calculated for the complete corpus (BF=83.8%, WF=69.8%, LF=60.2%, E_o=13.7%, E_u=3.4%).

7. Summary and discussion

This paper investigated use of predictability for learning lexical units from unsegmented speech input. Consecutive phonemes or syllables in natural speech tend to relate to each other in a way that is useful for predicting which sequences are part of lexical units (e.g., words) and which sequences straddle lexical unit boundaries. This tendency leads to a computational strategy that is useful for the segmentation of any natural language. We also know that this strategy is used by children early on in this task. Even though we know that adults and children use multiple cues to segment speech into lexical units, the fact that predictability is a language-general cue makes it a good candidate for one of the cues to bootstrap the lexical acquisition process. This study contributes to understanding of this particular cue for segmentation by corpus analysis and computational simulations.

First, we examined a set of quantitative measures used for characterizing predictability: *transitional probabilities*, *successor variety*, *mutual information* and *entropy*. The findings indicate that these measures are all relevant measures of predictability for segmentation task. Furthermore, they are similar in what they measure, but they are not equivalent. In other words, there is no single best measure that can replace all others, and their combination, in principle, leads to better segmentation. Second, the analysis also indicates that a more careful, but also more realistic, use of predictability can improve the segmentation performance as well. Previous uses of the predictability cue in the literature tend to use segmentation in a rather simplified settings which do not exploit its full utility. Particularly, using multiple phoneme context sizes for the calculation of segmentation is useful. This also corresponds to the fact that different phoneme lengths should (roughly) correspond to different linguistic units. Hence, using multiple context sizes at once allows one to arrive at generalizations at multiple levels.

After a careful analysis of these measures and their combination, we described a completely unsupervised method for combining multiple measures calculated on varying phoneme-context size. Arguably, we could do with a single measure of predictability. The reason for the effort spent for combination of these measures here is twofold. First, as the analysis in Section 3 showed, none of the measures alone performs as well as the combination of multiple measures. With the interest of getting the most out of the predictability cue, it makes sense to combine them. In this way, we can also take a step towards finding the full potential of the cue in segmentation task. We do not know to what extent the human cognitive system utilizes predictability. However, the findings in this study suggests that predictability-based segmentation strategy has a higher potential than is typically assumed in the literature, for example by characterizing it by transitional probabilities. Second, the method developed here for combining multiple measures provides a framework for combining more diverse cues, such as phonotactics, existing lexical knowledge or stress patterns, which go beyond predictability.

The segmentation algorithm developed here is completely unsupervised: it takes a set of unsegmented utterances, and returns a segmentation for each utterance based only on predictability statistics. However, the model has a free parameter, *maximum context size*. In this paper, we did not attempt to fix a certain value for this parameter, but reported results using a range of parameter values. As we increase the parameter value, we typically get an improvement at first, followed by a decrease in performance. The computational reason behind this decrease in performance with increased context size has to do with sparseness of the data when we calculate relevant statistics on longer sequences of strings. On the other hand, the value of the parameter can also be linked to working memory and processing limitations. As we increase the value of this parameter, the number of 'chunks' to remember increases, and as a result one expects realistic settings of this parameter to be related to what we know about limitations of human processing (See Miller, 1956, for a relevant discussion).

We compare the results of the model with a non-trivial ran-

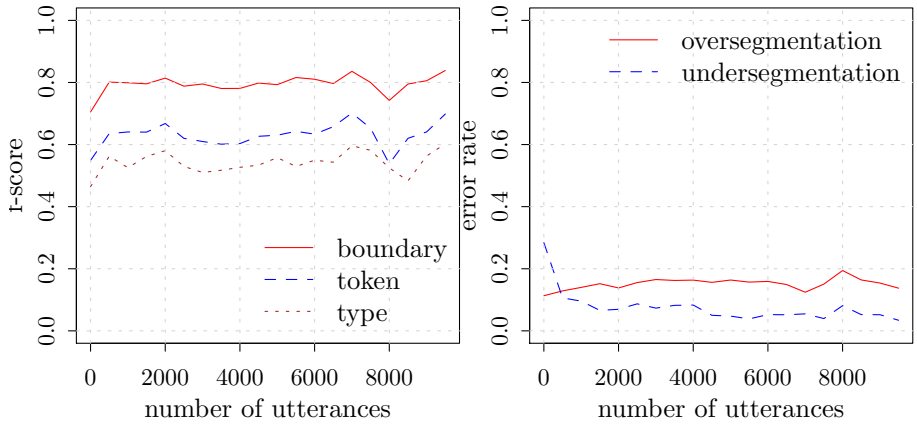


Figure 11: (a) Boundary, word token and word type F-scores and (b) oversegmentation and undersegmentation rates of the predictability-based segmentation model with maximum context size of 3 on the BR corpus for successive blocks of 500 utterances each.

dom baseline, and a model similar to many successful state-of-the-art models of segmentation. It is clear that the method developed here is relevant to segmentation: it performs substantially better than the baseline segmentation method. When compared with the performance of the state-of-the-art LM strategy, on the other hand, the performance of the predictability-based segmentation model is not that impressive. However, it is not too far behind either. It performs comparably for a range of parameter values, and even outperforms the LM in some performance scores, except lexical precision. The low lexical-precision scores are due to the fact that this model does not make use of a lexicon. As a result, it does not give any preference to the boundary decisions that reuse lexical units.

The performance degradation with respect to LM and similar models is also likely to be related to the greedy search strategy used in the proposed model. The model does not search all possible segmentations of an utterance as most successful segmentation algorithms do. This aspect of the model is in line with what we expect from human processing. It is hardly plausible that humans consider all possible segmentation of an utterance before finding the lexical units in the utterance. Human processing is known to be incremental and predictive. In this respect, the model presented in this paper fits human processing better. We expect that the possible performance deficiency caused by the greedy nature of the algorithm can be compensated for when other cues are used.

The comparison of the predictability model with the LM provides an indirect comparison with the state-of-the art models presented in Table 5. In summary, the predictability-based segmentation model described here performs comparably to the other successful models in the literature. Because of the use of different corpora and different sets of evaluation methods, it is difficult to compare the performance scores with other related models that utilize predictability. Nevertheless, the performance scores reported in three earlier studies are presented here to aid a rough comparison. Graphs presented in in Brent (1999) indicates about 50%–60% WP and WR and 20%–30% LP for his baseline model utilizing mutual information on the BR corpus. Cohen et al. (2007) report 76% BP, and 75% BR on George Orwell's 1984. Christiansen et al. (1998) report 37% WP and 40% WR with an SRN using phonotactics and utterance boundary cues on another child-directed speech corpus Korman (1984). Although these results are not directly comparable, it is clear that the performance scores presented in Table 9 are the best scores presented to date for models using only the predictability cue.

We want our models to achieve better segmentation scores, as humans eventually segment well. However, for a model of human cognition, high performance scores are not the only desirable properties. We would also like our models to match what we know about human cognition well, and provide further insight into the process being modeled. In this respect, the model proposed here has at least four desirable properties. First, the model is completely unsupervised, as it does not depend on any prior knowledge of boundaries. This aspect is shared by almost all other models discussed in this paper. Second, the model presented in this paper is strictly incremental. Most successful models in the literature are either batch (they process a large set of utterances at once, possibly multiple times), or require complete utterances to be processed before deciding any boundaries. Third, the model presented here takes strategies suggested by psycholinguistic research, rather than mathematically or computationally attractive strategies. We use a cue, predictability, known to be used both by adults and children. Furthermore, following the findings that humans use multiple cues for segmentation, the framework presented here is designed to deal with combination of arbitrary cues in mind. Finally, the model here is built on explicit representations, as opposed to a black-box model of input and output. As a result, model's behavior can easily be traced back to modeling assumptions, and various modeling assumptions can be modified systematically and tested.[10]

[10]This not to say that contrary modeling practices are completely irrelevant

This paper presented a cognitively-motivated model of segmentation. It demonstrates that we can get better segmentation performance by only using predictability statistics compared to the uses of the method in previous literature. Even though our focus in this study has been predictability, the modeling framework described here use an incremental and predictive segmentation strategy, and provides a simple mechanism of combining multiple sources of information. The results of this study also raise a number of questions for future research, two of which are particularly interesting. First, integration of other, more varied cues to the framework described here can shed a light on the combination and interaction of different cues. As well as an expected gain in segmentation performance, we expect such a study to also show the relative importance of various cues in certain settings of stages of acquisition. Second, alternative combination methods may provide better results and more principled modeling practices. The weighted majority voting method used here is a simple method which has been proven to be useful. However, other combination techniques such as Bayesian cue combination methods used in other areas of cognition may allow us to give better answer to the questions regarding both acquisition and processing. As a result, studying other combination methods points to another direction for future research.

Al-Shalabi, R., Kannan, G., Hilat, I., Ababneh, A., and Al-Zubi, A. (2005). Experiments with the successor variety algorithm using the cutoff and entropy methods. *Information Technology Journal*, 4(1).

Aslin, R. N. (1993). Segmentation of fluent speech into words: Learning models and the role of maternal input. In Boysson-Bardies, B. D., de Schonen, S., Jusczyk, P., MacNeilage, P., and Morton, J., editors, *Developmental Neurocognition: Speech and Face Processing in the First Year of Life*, pages 305–315. Kluwer Academic Publishers.

Aslin, R. N., Saffran, J. R., and Newport, E. L. (1998). Computation of conditional probability statistics by 8-month-old infants. *Psychological Science*, 9(4):321–324.

Bernstein Ratner, N. (1987). The phonology of parent-child speech. In Nelson, K. and van Kleeck, A., editors, *Children's language*, volume 6, pages 159–174. Erlbaum, Hillsdale, NJ.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Blanchard, D., Heinz, J., and Golinkoff, R. (2010). Modeling the contribution of phonotactic cues to the problem of word segmentation. *Journal of Child Language*, 37(Special Issue 03):487–511.

Boland, P. J. (1989). Majority systems and the Condorcet jury theorem. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 38(3):181–189.

Bordag, S. (2005). Unsupervised knowledge-free morpheme boundary detection. In *The Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP)*.

Bordag, S. (2007). Unsupervised and knowledge-free morpheme segmentation and analysis. In *The Working Notes for the CLEF Workshop 2007*.

Brent, M. R. (1999). An efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learning*, 34(1-3):71–105.

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

Carnegie Mellon University (1998). The carnegie mellon university pronouncing dictionary. `http://http://www.speech.cs.cmu.edu/cgi-bin/cmudict`. version 0.6d.

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

for studying cognition. Various models that violate one or more of these properties have been useful in providing insight into the segmentation process. However, the modeling practice followed in this study allows a model to be used in investigation of finer details regarding human cognition.

Cohen, P., Adams, N., and Heeringa, B. (2007). Voting experts: An unsupervised algorithm for segmenting sequences. *Intelligent Data Analysis*, 11(6):607–625.

Çöltekin, Ç. (2010). Improving successor variety for morphological segmentation. In Westerhout, E., Markus, T., and Monachesi, P., editors, *Proceedings of the 20th Meeting of Computational Linguistics in the Netherlands*, pages 13–28. LOT.

Çöltekin, Ç. (2011). *Catching Words in a Stream of Speech: Computational simulations of segmenting transcribed child-directed speech*. PhD thesis, University of Groningen.

Cutler, A. and Butterfield, S. (1992). Rhythmic cues to speech segmentation: Evidence from juncture misperception. *Journal of Memory and Language*, 31(2):218–236.

Déjean, H. (1998). Morphemes as necessary concept for structures discovery from untagged corpora. In *Workshop on Paradigms and Grounding in Natural Language Learning*, pages 295–299.

Demberg, V. (2007). A language-independent unsupervised model for morphological segmentation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 920–927, Prague, Czech Republic. Association for Computational Linguistics.

Dilley, L. C. and McAuley, J. D. (2008). Distal prosodic context affects word segmentation and lexical processing. *Journal of Memory and Language*, 59(3):294–311.

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

Goldsmith, J. (2006). An algorithm for the unsupervised learning of morphology. *Natural Language Engineering*, 12(04):353–371.

Goldwater, S., Griffiths, T. L., and Johnson, M. (2009). A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112:21–54.

Graf Estes, K., Evans, J. L., Alibali, M. W., and Saffran, J. R. (2007). Can infants map meaning to newly segmented words? statistical segmentation and word learning. *Psychological Science*, 18(3):254–260.

Hafer, M. A. and Weiss, S. F. (1974). Word segmentation by letter successor varieties. *Information Storage and Retrieval*, 10(11-12):371–385.

Hansen, L. K. and Salamon, P. (1990). Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12:993–1001.

Harris, Z. S. (1955). From phoneme to morpheme. *Language*, 31(2):190–222.

Hockema, S. A. (2006). Finding words in speech: An investigation of American English. *Language Learning and Development*, 2(2):119–146.

Huang, J. H. and Powers, D. (2003). Chinese word segmentation based on contextual entropy. In *Proceedings of Pacific Asia Conference on Language, Information and Computation*, pages 121–127.

Johnson, E. K. and 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.

Johnson, M. and Goldwater, S. (2009). Improving nonparameteric Bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 317–325.

Jusczyk, P. W., Cutler, A., and Redanz, N. J. (1993). Infants' preference for the predominant stress patterns of English words. *Child Development*, 64(3):675–687.

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

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

Kempe, A. (1999). Experiments in unsupervised entropy-based corpus segmentation. In *Proc. Workshop on Computational Natural Language Learning (CoNLL'99)*, pages 7–13, Bergen, Norway.

Korman, M. (1984). Adaptive aspects of maternal vocalizations in differing contexts at ten weeks. *First Language*, 5:44–45.

Littlestone, N. and Warmuth, M. K. (1994). The weighted majority algorithm. *Information and Computation*, 108(2):212–261.

Marr, D. (1982). *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Freeman, New York.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63(2):81–97.

Moberg, J., Gooskens, C., Nerbonne, J., and Vaillette, N. (2007). Conditional entropy measures intelligibility among related languages. In Dirix, P., Schuurman, I., Vandeghinste, V., and Eynde, F. V., editors, *Computational Linguistics in the Netherlands 2006: Selected papers from the 17th CLIN Meeting*, pages 51–66. LOT.

Monaghan, P. and Christiansen, M. H. (2010). Words in puddles of sound: modelling psycholinguistic effects in speech segmentation. *Journal of Child Language*, 37(Special Issue 03):545–564.

Narasimhamurthy, A. (2005). Theoretical bounds of majority voting performance for a binary classification problem. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27:1988–1995.

Newport, E. L. and Aslin, R. N. (2004). Learning at a distance: I. statistical learning of non-adjacent dependencies. *Cognitive Psychology*, 48(2):127–162.

Pelucchi, B., Hay, J. F., and Saffran, J. R. (2009). Learning in reverse: Eight-month-old infants track backward transitional probabilities. *Cognition*, 113(2):244–247.

Perruchet, P. and Desaulty, S. (2008). A role for backward transitional probabilities in word segmentation? *Memory and Cognition*, 36(7):1299–1305.

Räsänen, O. (2011). A computational model of word segmentation from continuous speech using transitional probabilities of atomic acoustic events. *Cognition*, 120(2):149–176.

Saffran, J. R., Aslin, R. N., and Newport, E. L. (1996a). Statistical learning by 8-month old infants. *Science*, 274(5294):1926–1928.

Saffran, J. R., Newport, E. L., and Aslin, R. N. (1996b). Word segmentation: The role of distributional cues. *Journal of Memory and Language*, 35(4):606–621.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656.

Stein, B. and Potthast, M. (2008). Putting successor variety stemming to work. In Decker, R. and Lenz, H., editors, *Advances in Data Analysis*, pages 367–374. Springer.

Swingley, D. (2005). Statistical clustering and the contents of the infant vocabulary. *Cognitive Psychology*, 50(1):86–132.

Thiessen, E. D. and Saffran, J. R. (2003). When cues collide: Use of stress and statistical cues to word boundaries by 7- to 9-month-old infants,. *Developmental Psychology*, 39(4):706–716.

Thompson, S. P. and Newport, E. L. (2007). Statistical learning of syntax: The role of transitional probability. *Language Learning and Development*, 3(1):1–42.

Tribus, M. and McIrvine, E. C. (1971). Energy and information. *Scientific American*, 224:178–184.

van Rijsbergen, C. J. (1979). *Information Retrieval*. Butterworth-Heinemann, 2nd edition.

Venkataraman, A. (2001). A statistical model for word discovery in transcribed speech. *Computational Linguistics*, 27(3):351–372.

Zhikov, V., Takamura, H., and Manabu, O. (2010). An efficient algorithm for unsupervised word segmentation with branching entropy and mdl. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 832–842, Stroudsburg, PA, USA. Association for Computational Linguistics.