An Efficient Algorithm for Unsupervised Word Segmentation with Branching Entropy and MDL

Valentin Zhikov
Interdisciplinary Graduate School
of Science and Engineering
Tokyo Institute of Technology
zhikov@lr.pi.titech.ac.jp

Hiroya Takamura
Precision and Intelligence Laboratory
Tokyo Institute of Technology
takamura@pi.titech.ac.jp

Manabu Okumura
Precision and Intelligence Laboratory
Tokyo Institute of Technology
oku@pi.titech.ac.jp

Abstract

This paper proposes a fast and simple unsupervised word segmentation algorithm that utilizes the local predictability of adjacent character sequences, while searching for a least-effort representation of the data. The model uses branching entropy as a means of constraining the hypothesis space, in order to efficiently obtain a solution that minimizes the length of a two-part MDL code. An evaluation with corpora in Japanese, Thai, English, and the "CHILDES" corpus for research in language development reveals that the algorithm achieves an accuracy, comparable to that of the state-of-the-art methods in unsupervised word segmentation, in a significantly reduced computational time.

1 Introduction

As an inherent preprocessing step to nearly all NLP tasks for writing systems without orthographical marking of word boundaries, such as Japanese and Chinese, the importance of word segmentation has lead to the emergence of a micro-genre in NLP focused exclusively on this problem.

Supervised probabilistic models such as Conditional Random Fields (CRF) (Lafferty et al., 2001) have a wide application to the morphological analysis of these languages. However, the development of the annotated training corpora necessary for their functioning is a labor-intensive task, which involves multiple stages of manual tagging. Because of the scarcity of labeled data, the domain adaptation of morphological analyzers is also problematic, and semi-supervised algorithms that address this issue have also been proposed (e.g. Liang, 2005; Tsuboi et al., 2008).

Recent advances in unsupervised word segmentation have been promoted by human cognition research, where it is involved in the modeling of the mechanisms that underlie language acquisition. Another motivation to study unsupervised approaches is their potential to support the domain adaptation of morphological analyzers through the incorporation of unannotated training data, thus reducing the dependency on costly manual work. Apart from the considerable difficulties in discovering reliable criteria for word induction, the practical application of such approaches is impeded by their prohibitive computational cost.

In this paper, we address the issue of achieving high accuracy in a practical computational time through an efficient method that relies on a combination of evidences: the local predictability of character patterns, and the reduction of effort achieved by a given representation of the language data. Both of these criteria are assumed to play a key role in native language acquisition. The proposed model allows experimentation in a more realistic setting, where the learner is able to apply them simultaneously. The

Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 832–842,
MIT, Massachusetts, USA, 9-11 October 2010.

method shows a high performance in terms of accuracy and speed, can be applied to language samples of substantial length, and generalizes well to corpora in different languages.

2 Related Work

The principle of least effort (Zipf, 1949) postulates that the path of minimum resistance underlies all human behavior. Recent research has recognized its importance in the process of language acquisition (Kit, 2003). Compression-based word induction models comply to this principle, as they reorganize the data into a more compact representation while identifying the vocabulary of a text. The minimum description length framework (MDL) (Rissanen, 1978) is an appealing means of formalizing such models, as it provides a robust foundation for learning and inference, based solely on compression.

The major problem in MDL-based word segmentation is the lack of standardized search algorithms for the exponential hypothesis space (Goldwater, 2006). The representative MDL models compare favorably to the current state-of-the-art models in terms of accuracy. Brent and Cartwright (1996) carried out an exhaustive search through the possible segmentations of a limited subset of the data. Yu (2000) proposed an EM optimization routine, which achieved a high accuracy, in spite of a lower compression than the gold standard segmentation.

As a solution to the aforementioned issue, the proposed method incorporates the local predictability of character sequences into the inference process. Numerous studies have shown that local distributional cues can serve well the purpose of inducing word boundaries. Behavioral science has confirmed that infants are sensitive to the transitional probabilities found in speech (Saffran et al., 1996). The increase in uncertainty following a given word prefix is a well studied criterion for morpheme boundary prediction (Harris, 1955). A good deal of research has been conducted on methods through which such local statistics can be applied to the word induction problem (e.g. Kempe, 1999; Huang and Powers, 2003; Jin and Tanaka-Ishii, 2006). Hutchens and Adler (1998) noticed that entropic chunking has the effect of reducing the perplexity of a text.

Most methods for unsupervised word segmentation based solely on local statistics presume a certain – albeit minimum – level of acquaintance with the target language. For instance, the model of Huang and Powers (2003) involves some parameters (Markov chain order, numerous threshold values) that allow its adaptation to the individuality of written Chinese. In comparison, the method proposed in this paper generalizes easily to a variety of languages and domains, and is less dependent on annotated development data.

The state-of-the-art in unsupervised word segmentation is represented by Bayesian models. Goldwater et al. (2006) justified the importance of context as a means of avoiding undersegmentation, through a method based on hierarchical Dirichlet processes. Mochihashi et al. (2009) proposed extensions to this method, which included a nested character model and an optimized inference procedure. Johnson and Goldwater (2009) have proposed a novel method based on adaptor grammars, whose accuracy surpasses the aforementioned methods by a large margin, when appropriate assumptions are made regarding the structural units of a language.

3 Proposed Method

3.1 Word segmentation with MDL

The proposed two-part code incorporates some extensions of models presented in related work, aimed at achieving a more precise estimation of the representation length. We first introduce the general two-part code, which consists of:

- the model, embodied by a codebook, i.e., a lexicon of unique word types $M = \{w_1, ..., w_{|M|}\}$,
- the source text D, obtained through encoding the corpus using the lexicon.

The total description length amounts to the number of bits necessary for simultaneous transmission of the codebook and the source text. Therefore, our objective is to minimize the combined description length of both terms:

$$L(D, M) = L(M) + L(D|M).$$

The description length of the data given M is calculated using the Shannon-Fano code:

$$L(D|M) = -\sum_{j=1}^{|M|} \#w_j \log_2 P(w_j),$$

where $\#w_j$ stands for the frequency of the word w_j in the text.

Different strategies have been proposed in the literature for the calculation of the codebook cost. A common technique in segmentation and morphology induction models is to calculate the product of the total length in characters of the lexicon and an estimate of the per-character entropy. In this way, both the probabilities and lengths of words are taken into consideration. The use of a constant value is an effective and easily computable approach, but it is far from precise. For instance, in Yu (2000) the average entropy per character is measured against the original corpus, but this model does not capture the effects of the word distributions on the observed character probabilities. For this reason, we propose a different method: the codebook is modeled as a separate Markov chain of characters.

A lexicon of characters M' is defined. The description length of the lexicon data D' given M' is then calculated as:

$$L(D'|M') = -\sum_{i=1}^{|C|} \#c_i \log_2 P(c_i),$$

where $\#c_i$ denotes the frequency of a character c_i in the lexicon of hypothesis M. The term $L(M')$ is constant for any choice of hypothesis, as is represents the character set of a corpus.

The total description length under the proposed model is thus calculated as:

$$L(M) + L(D|M) = L(M') + L(D'|M') + L(D|M) =$$
$$-\sum_{i=1}^{|C|} \#c_i \log_2 P(c_i) - \sum_{j=1}^{|M|} \#w_j \log_2 P(w_j) + O(1).$$

A rigorous definition should include two additional terms, $L(\theta|M)$ and $L(\theta'|M')$, which give the representation cost of the parameters of both models. The $L(\theta|M)$ can be calculated as:

$$L(\theta|M) = \frac{|M| - 1}{2} * \log_2 S,$$

where $|M| - 1$ gives the number of parameters (degrees of freedom), and S is the size of the dataset (the total length of the text in characters). The parametric complexity term is calculated in the same way for the lexicon. For a derivation of the above formula, refer to e.g. Li (1998).

MDL is closely related to Bayesian inference. Depending on the choice of a universal code, the two approaches can overlap, as is the case with the two-part code discussed in this paper. It can be shown that the model selection in our method is equivalent to a MAP inference, conducted under the assumption that the prior probability of a model decreases exponentially with its length (Goldwater, 2006). Thus, the task that we are trying to accomplish is to conduct a focused search through the hypothesis space that will allow us to obtain an approximation of the MAP solution in a reasonable time.

The MDL framework does not provide standard search algorithms for obtaining the hypotheses that minimize the description length. In the rest of this section, we will describe an efficient technique suitable for the word segmentation task.

3.2 Obtaining an initial hypothesis

First, a rough initial hypothesis is built by an algorithm that combines the branching entropy and MDL criteria.

Given a set $\mathcal{X}$, comprising all the characters found in a text, the entropy of branching at position k of the text is defined as:

$$H(X_k|x_{k-1}, ..., x_{k-n}) =$$
$$-\sum_{x \in \mathcal{X}} P(x|x_{k-1}, ..., x_{k-n}) \log_2 P(x|x_{k-1}, ..., x_{k-n}),$$

where x_k represents the character found at position k, and n is the order of the Markov model over characters. For brevity, hereafter we shall denote the observed sequence $\{x_{k-1}, ..., x_{k-n}\}$ as $\{x_{k-1:k-n}\}$.

The above definition is extended to combine the entropy estimates in the left-to-right and right-to-left directions, as this factor has reportedly improved performance figures for models based on branching entropy (Jin and Tanaka-Ishii, 2006). The estimates in both directions are summed up, yielding a single value per position:

$$H'(X_{k;k-1}|x_{k-1:k-n}; x_{k:k+n-1}) =$$
$$-\sum_{x\in\mathcal{X}} P(x|x_{k-1:k-n}) \log_2 P(x|x_{k-1:k-n})$$
$$-\sum_{x\in\mathcal{X}} P(x|x_{k:k+n-1}) \log_2 P(x|x_{k:k+n-1}).$$

Suffix arrays are employed during the collection of frequency statistics. For a character model of order n over a testing corpus of size t and a training corpus of size m, suffix arrays allow these to be acquired in $O(tn \log m)$ time. Faster implementations reduce the complexity to $O(t(n+\log m))$. For further discussion, see Manber and Myers (1991). During the experiments, we did not use the caching functionality provided by the suffix array library, but instead kept the statistics for the current iterative pass (n-gram order and direction) in a local table.

The chunking technique we adopt is to insert a boundary when the branching entropy measured in sequences of length n exceeds a certain threshold value ($H(X|x_{k-1:k-n}) > \beta$). Both n and β are fixed.

Within the described framework, the increase in context length n promotes precision and recall at first, but causes a performance degradation when the entropy estimates become unreliable due to the reduced frequencies of long strings. High threshold values produce a combination of high precision and low recall, while low values result in low precision and high recall.

Since the F-score curve obtained as decreasing values are assigned to the threshold is typically unimodal as in many applications of MDL, we employ a bisection search routine for the estimation of the threshold (Algorithm 1).

All positions of the dataset are sorted by their entropy values. At each iteration, at most two new hypotheses are built, and their description lengths are calculated in time linear to the data size. The computational complexity of the described routine is $O(t \log t)$, where t is the corpus length in characters.

The order of the Markov chain n used during the entropy calculation is the only input variable of the proposed model. Since different values perform the best across the various languages, the most appropriate settings can be obtained with the help of a small annotated corpus. However, the MDL objective also enables unsupervised optimization against a sufficiently large unlabeled dataset. The order that minimizes the description length of the data can be discovered in a few iterations of Algorithm 1 with increasing values of n, and it typically matches the optimal value of the parameter (Table 1).

Algorithm 1 Generates an initial hypothesis.

```
thresholds[] := sorted H(X_k) values;
threshold := median of thresholds[];
step := length of thresholds[]/4;
direction := ascending;
minimum := +∞;
while step > 0 do
    nextThreshold := thresholds[] value one step in last direction;
    DL = calculateDL(nextThreshold);
    if DL < minimum then
        minimum:= DL; threshold := nextThreshold;
        step := step/2; continue;
    end if
    reverse direction;
    nextThreshold := thresholds[] value one step in last direction;
    if DL < minimum then
        minimum:= DL; threshold := nextThreshold;
        step := step/2; continue;
    end if
    reverse direction;
    step := step/2;
end while
```

Corpus	[1]	[2]	[3]	[4]
CHILDES	394655.52	**367711.66**	368056.10	405264.53
Kyoto	1.291E+07	**1.289E+07**	1.398E+07	1.837E+07

Table 1: Length in bits of the solutions proposed by Algorithm 1 with respect to the character n-gram order.

Although an acceptable initial segmentation can be built using the described approach, it is possible to obtain higher accuracy with an extended model that takes into account the statistics of Markov chains from several orders during the entropy calculation. This can be done by summing up the entropy estimates, in the way introduced earlier for combining the values in both directions:

$$H''(X_{k;k-1}|x_{k-1:k-n}; x_{k:k+n-1}) =$$
$$-\sum_{n=1}^{n_{max}} (\sum_{x\in\mathcal{X}} P(x|x_{k-1:k-n}) \log_2 P(x|x_{k-1:k-n})$$
$$+\sum_{x\in\mathcal{X}} P(x|x_{k:k+n-1}) \log_2 P(x|x_{k:k+n-1})),$$

where n_{max} is the index of the highest order to be taken into consideration.

3.3 Refining the initial hypothesis

In the second phase of the proposed method, we will refine the initial hypothesis through the reorganization of local co-occurrences which produce redundant description length. We opt for greedy optimization, as our primary interest is to further explore the impact that description length minimization has on accuracy. Of course, such an approach is unlikely to obtain global minima, but it is a feasible means of conducting the optimization process, and guarantees a certain increase in compression.

Since a preliminary segmentation is available, it is convenient to proceed by inserting or removing boundaries in the text, thus splitting or merging the already discovered tokens. The ranked positions involved in the previous step can be reused here, as this is a way to bias the search towards areas of the text where boundaries are more likely to occur. Boundary insertion should start in regions where the branching entropy is high, and removal should first occur in regions where the entropy is close to zero. A drawback of this approach is that it omits locations where the gains are not immediately obvious, as it cannot assess the cumulative gains arising from the merging or splitting of all occurrences of a certain pair (Algorithm 2).

A clean-up routine, which compensates for this shortage, is also implemented (Algorithm 3). It operates directly on the types found in the lexicon produced by Algorithm 2, and is capable of modifying a large number of occurrences of a given pair in a single step. The lexicon types are sorted by their contribution to the total description length of the corpus. For each word type, splitting or merging is attempted at every letter, beginning from the center. The algorithm eliminates unlikely types with low contribution, which represent mostly noise, and redistributes their cost among more likely ones. The design of the merging routine makes it impossible to produce types longer than the ones already found in the lexicon, as an exhaustive search would be prohibitive.

The evaluation of each hypothetical change in the segmentation requires that the description length of the two-part code is recalculated. In order to

Algorithm 2 Compresses local token co-occurrences.

```
path[][]:= positions sorted by H(X_k) values;
minimum := DL of model produced at initialization;
repeat
  for i = max H(X_k) to min H(X_k) do
    pos:= path[i][k];
    if no boundary exists at pos then
      leftToken := token to the left;
      rightToken := token to the right;
      longToken := leftToken + rightToken;
      calculate DL after splitting;
      if DL < minimum then
        accept split, update model, update DP variables;
      end if
    end if
  end for
  for i = min H(X_k) to max H(X_k) do
    merge leftToken and rightToken into longToken
    if DL will decrease (analogous to splitting)
  end for
until no change is evident in model
```

Algorithm 3 A lexicon clean-up procedure.

```
types[] := lexicon types sorted by cost;
minimum := DL of model produced by Algorithm 2;
repeat
  for i = min cost to max cost do
    for pos = middle to both ends of types[i] do
      longType := types[i];
      leftType := sequence from first character to pos;
      rightType:= sequence from pos to last character;
      calculate DL after splitting longType into leftType and rightType;
      if DL < minimum then
        accept split, update model, update DP variables;
        break out of inner loop;
      end if
    end for
  end for
  types[] := lexicon types sorted by cost;
  for i = max cost to min cost do
    for pos = middle to both ends of types[i] do
      merge leftType and rightType into longType if
      DL will decrease (analogous to splitting)
      break out of inner loop;
    end for
  end for
until no change is evident in model
```

make this optimization phase computationally feasible, dynamic programming is employed in Algorithms 2 and 3. The approach adopted for the recalculation of the source text term $L(D|M)$ is explained below. The estimation of the lexicon cost is analogous. The term $L(D|M)$ can be rewritten as:

$$L(D|M) = -\sum_{j=1}^{|M|} \#w_j \log_2 \frac{\#w_j}{N} =$$

$$-\sum_{j=1}^{|M|} \#w_j \log_2 \#w_j + N \log_2 N = T_1 + T_2,$$

where $\#w_j$ is the frequency of w_j in the segmented corpus, and $N = \sum_{j=1}^{|M|} \#w_j$ is the cumulative token count. In order to calculate the new length, we keep the values of the terms T_1 and T_2 obtained at the last change of the model. Their new values are computed for each hypothetical split or merge on the basis of the last values, and the expected description length is calculated as their sum. If the produced estimate is lower, the model is modified and the new values of T_1 and T_2 are stored for future use.

In order to maintain precise token counts, Algorithms 2 and 3 recognize the fact that recurring sequences ("byebye" etc.) appear in the corpora, and handle them accordingly. Known boundaries, such as the sentence boundaries in the CHILDES corpus, are also taken into consideration.

4 Experimental Settings

We evaluated the proposed model against four datasets. The first one is *the Bernstein-Ratner corpus* for language acquisition based on transcripts from the CHILDES database (Bernstein-Ratner, 1987). It comprises phonetically transcribed utterances of adult speech directed to 13 through 21-month-old children. We evaluated the performance of our learner in the cases when the few boundaries among the individual sentences are available to it (B), and when it starts from a blank state (N). The *Kyoto University Corpus* (Kurohashi and Nagao, 1998) is a standard dataset for Japanese morphological and dependency structure analysis, which comprises newspaper articles and editorials from the Mainichi Shimbun. The *BEST* corpus for word segmentation and named entity recognition in Thai language combines text from a variety of sources including encyclopedias (E), newspaper articles (N), scientific articles (A), and novels (F). The *WSJ* subset of the *Penn Treebank II Corpus* incorporates selected stories from the Wall Street Journal, year 1989 (Marcus et al., 1994). Both the original text (O), and a version in which all characters were converted to lower case (L) were used.

Corpus	Language	Size (MB)	Chars (K)	Tokens (K)	Types (K)
CHILDES-B/N	English	0.1	95.8	33.3	1.3
Kyoto	Japanese	5.02	1674.9	972.9	39.5
WSJ	English	5.22	5220.0	1174.2	49.1
BEST-E	Thai	12.64	4360.2	1163.2	26.2
BEST-N	Thai	18.37	6422.7	1659.4	36.3
BEST-A	Thai	4.59	1619.9	438.7	13.9
BEST-F	Thai	16.18	5568.0	1670.8	22.6
Wikipedia	Japanese	425.0	169069.3	/	/
Asahi	Japanese	337.2	112401.1	/	/
BEST-All	Thai	51.2	17424.0	4371.8	73.4

Table 2: Corpora used during the evaluation. Precise token and type counts have been omitted for Wikipedia and Asahi, as no gold standard segmentations are available.

The datasets listed above were built by removing the tags and blank spaces found in the corpora, and concatenating the remaining text. We added two more training datasets for Japanese, which were used in a separate experiment solely for the acquisition of frequency statistics. One of them was created from 200,000 randomly chosen Wikipedia articles, stripped from structural elements. The other one contains text from the year 2005 issues of Asahi Newspaper. Statistics regarding all described datasets are presented in Table 2.

One whole corpus is segmented in each experiment, in order to avoid the statement of an extended model that would allow the separation of training and test data. This setting is also necessary for the direct comparison between the proposed model and other recent methods evaluated against the entire CHILDES corpus.

We report the obtained precision, recall and F-score values calculated using boundary, token and type counts. Precision (P) and recall (R) are defined as:

$$P = \frac{\#correct\ units}{\#\ output\ units}, \quad R = \frac{\#correct\ units}{\#gold\ standard\ units}.$$

Boundary, token and lexicon F-scores, denoted as B-F and T-F and L-F, are calculated as the

Model	Corpus & Settings	B-Prec	B-Rec	B-F	T-Prec	T-Rec	T-F	DL (bits)	Ref.DL (bits)	Time (ms)
1	CHILDES, α = 1.2, n = [1-6]	0.8667	0.8898	**0.8781**	**0.6808**	**0.6990**	**0.6898**	**344781.74**	300490.52	1060.2
2a (H')	CHILDES, n = 2	0.7636	**0.9109**	0.8308	0.5352	0.6384	0.5823	367711.66		**753.1**
2b (H'')	CHILDES, n_{max} = 3	**0.8692**	0.8865	0.8777	0.6792	0.6927	0.6859	347633.07		885.3
1	Kyoto, α = 0, n = [1-6]	**0.8208**	0.8208	0.8208	0.5784	0.5784	0.5784	1.325E+07	1.120E+07	54958.8
2a (H')	Kyoto, n = 2	0.8100	0.8621	0.8353	0.5934	0.6316	0.6119	1.289E+07		**22909.7**
2b (H'')	Kyoto, n_{max} = 2	0.8024	**0.9177**	**0.8562**	**0.6093**	**0.6969**	**0.6501**	**1.248+E07**		23212.8

Table 3: Comparison of the proposed method (2a, 2b) with the model of Jin and Tanaka-Ishii (2006) (1). Execution times include the obtaining of frequency statistics, and are represented by averages over 10 runs.

harmonic averages of the corresponding precision and recall values ($F = 2PR/(P + R)$). As a rule, boundary-based evaluation produces the highest scores among the three evaluation modes, as it only considers the correspondence between the proposed and the gold standard boundaries at the individual positions of the corpora. Token-based evaluation is more strict – it accepts a word as correct only if its beginning and end are identified accurately, and no additional boundaries lie in between. Lexicon-based evaluation reflects the extent to which the vocabulary of the original text has been recovered. It provides another useful perspective for the error analysis, which in combination with token scores can give a better idea of the relationship between the accuracy of induction and item frequency.

The system was implemented in Java, however it handled the suffix arrays through an external C library called Sary.[1] All experiments were conducted on a 2 GHz Core2Duo T7200 machine with 2 GB RAM.

5 Results and Discussion

The scores we obtained using the described instantiations of the branching entropy criterion at the initialization phase are presented in Table 3, along with those generated by our own implementation of the method presented in Jin and Tanaka-Ishii (2006), where the threshold parameter α was adjusted manually for optimal performance.

The heuristic of Jin and Tanaka-Ishii takes advantage of the trend that branching entropy decreases as the observed character sequences become longer; sudden rises can thus be regarded as an indication of locations where a boundary is likely to exist. Their method uses a common value for thresholding the entropy change throughout all n-gram orders, and combines the boundaries discovered in both directions in a separate step. These properties of the method would lead to complications if we tried to employ it in the first phase of our method (i.e. a step parameter for iterative adjustment of the threshold value, rules for combining the boundaries, etc.).

The proposed criterion with an automatically determined threshold value produced slightly worse results than that of Jin and Tanaka-Ishii at the CHILDES corpus. However, we found out that our approach achieves approximately 1% higher score when the best performing threshold value is selected from the candidate list. There are two observations that account for the suboptimal threshold choice by our algorithm. On one hand, the correspondence between description length and F-score is not absolutely perfect, and this may pose an obstacle to the optimization process for relatively small language samples. Another issue lies in the bisection search routine, which suggests approximations of the description length minima. The edge that our method has on the Kyoto corpus can be attributed to a better estimation of the optimal treshold value due to the larger amount of data.

The experimental results obtained at the completion of Algorithm 3 are summarized in Tables 4 and 5. Presented durations include the obtaining of frequency statistics. The n_{max} parameter is set to the value which maximizes the compression during the initial phase, in order to make the results representative of the case in which no annotated development corpora are accessible to the algorithm.

It is evident that after the optimization carried out in the second phase, the description length is reduced to levels significantly lower than the ground truth. In this aspect, the algorithm outperforms the EM-based method of Yu (2000).

[1] http://sary.sourceforge.net

Corpus & Settings	B-F	T-F	L-F	Time (ms)
CHILDES-B, n_{max}=3	0.9092	0.7542	0.5890	2597.2
CHILDES-N, n_{max}=3	0.9070	0.7499	0.5578	2949.3
Kyoto, n_{max}=2	0.8855	0.7131	0.3725	70164.6
BEST-E, n_{max}=5	0.9081	0.7793	0.3549	738055.0
BEST-N, n_{max}=5	0.8811	0.7339	0.2807	505327.0
BEST-A, n_{max}=5	0.9045	0.7632	0.4246	250863.0
BEST-F, n_{max}=5	0.9343	0.8216	0.4820	305522.0
WSJ-O, n_{max}=6	0.8405	0.6059	0.3338	658214.0
WSJ-L, n_{max}=6	0.8515	0.6373	0.3233	582382.0

Table 4: Results obtained after the termination of Algorithm 3.

Corpus & Settings	Description Length (Proposed)	Description Length (Total)
CHILDES-B, n_{max}=3	**290592.30**	300490.52
CHILDES-N, n_{max}=3	**290666.12**	300490.52
Kyoto, n_{max}=2	**1.078E+07**	1.120E+07
BEST-E, n_{max}=5	**1.180E+07**	1.252E+07
BEST-N, n_{max}=5	**1.670E+07**	1.809E+07
BEST-A, n_{max}=5	**4438600.32**	4711363.62
BEST-F, n_{max}=5	**1.562E+07**	1.634E+07
WSJ-O, n_{max}=6	**1.358E+07**	1.460E+07
WSJ-L, n_{max}=6	**1.317E+07**	1.399E+07

Table 5: Description length - proposed versus reference segmentation.

We conducted experiments involving various initialization strategies: scattering boundaries at random throughout the text, starting from entirely unsegmented state, or considering each symbol of the text to be a separate token. The results obtained with random initialization confirm the strong relationship between compression and segmentation accuracy, evident in the increase of token F-score between the random initialization and the termination of the algorithm, where description length is lower (Table 6). They also reveal the importance of the branching entropy criterion to the generation of hypotheses that maximize the evaluation scores and compression, as well as the role it plays in the reduction of computational time.

T-F-Score		Description Length	Time (ms)
Random Init	Refinement		
0.0441 (0.25)	0.3833	387603.02	6660.4
0.0713 (0.50)	0.3721	383279.86	4975.1
0.0596 (0.75)	0.2777	412743.67	3753.3

Table 6: Experimental results for CHILDES-N with randomized initialization and search path. The numbers in brackets represent the seed boundaries/character ratios.

The greedy algorithms fail to suggest any optimizations that improve the compression in the extreme cases when the boundaries/character ratio is either 0 or 1. When no boundaries are given, splitting operations produce unique types with a low frequency that increase the cost of both parts of the MDL code, and are rejected. The algorithm runs slowly, as each evaluation operates on candidate strings of enormous length. Similarly, when the corpus is broken down into single-character tokens, merging individual pairs does not produce any increase in compression. This could be achieved by an algorithm that estimates the total effect from merging all instances of a given pair, but such an algorithm would be computationally infeasible for large corpora.

Finally, we tried randomizing the search path for Algorithm 2 after an entropy-guided initialization, to observe a small deterioration in accuracy in the final segmentation (less than 1% on average).

Figure 1a illustrates the effect that training data size has on the accuracy of segmentation for the Kyoto corpus. The learning curves are similar throughout the different corpora. For the CHILDES corpus, which has a rather limited vocabulary, token F-score above 70% can be achieved for datasets as small as 5000 characters of training data, provided that reasonable values are set for the n_{max} parameter (we used the values presented in Table 4 throughout these experiments).

Figure 1b shows the evolution of token F-score by stage for all corpora. The initialization phase seems to have the highest contribution to the formation of the final segmentation, and the refinement phase is highly dependent on the output it produces. As a consequence, results improve when a more adequate language sample is provided during the learning of local dependencies at initialization. This is evident in the experiments with the larger unlabeled Thai and Japanese corpora.

For Japanese language with the setting for the n_{max} parameter that maximized compression, we observed an almost 4% increase in the token F-score produced at the end of the first phase with the Asahi corpus as training data. Only a small (less than 1%) rise was observed in the overall performance. The quite larger dataset of randomly chosen Wikipedia articles achieved no improvement. We attributed this

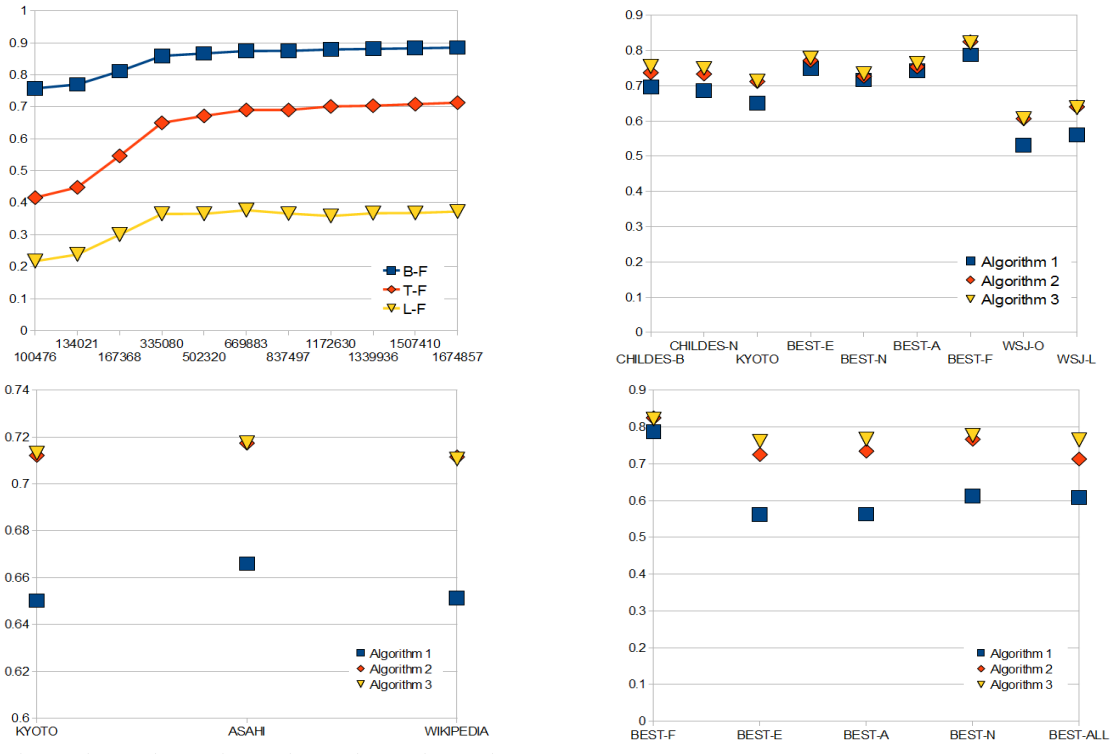

Figure 1: a) corpus size / accuracy relationship (Kyoto); b) accuracy levels by phase; c) accuracy levels by phase with various corpora for frequency statistics (Kyoto); d) accuracy levels by phase with different corpora for frequency statistics (BEST).

to the higher degree of correspondence between the domains of the Asahi and Kyoto corpora (Figure 1c).

Experiments with the BEST corpus reveal better the influence of domain-specific data on the accuracy of segmentation. Performance deteriorates significantly when out-of-domain training data is used. In spite of its size, the assorted composite corpus, in which in-domain and out-of-domain training data are mixed, produces worse results than the corpora which include only domain-specific data (Figure 1d).

Finally, a comparison of the proposed method with Bayesian n-gram models is presented in Table 7. Through the increase of compression in the refinement phase of the algorithm, accuracy is improved by around 3%, and the scores approach those of the explicit probabilistic models of Goldwater et al. (2009) and Mochihashi et al. (2009). The proposed learner surpasses the other unsupervised word induction models in terms of processing speed. It should be noticed that a direct comparison of accuracy is not possible with Mochihashi et al. (2009), as they evaluated their system with separate datasets for training and testing. Furthermore, different segmentation standards exist for Japanese, and therefore the "ground truth" provided by the Kyoto corpus cannot be considered an ideal measure of accuracy.

6 Conclusions and Future Work

This paper has presented an efficient algorithm for unsupervised word induction, which relies on a combination of evidences. New instantiations of the branching entropy and MDL criteria have been proposed and evaluated against corpora in different languages. The MDL-based optimization eliminates the discretion in the choice of the context length and threshold parameters, common in segmentation models based on local statistics. At the same time, the branching entropy criterion enables a constrained search through the hypothesis space, allowing the proposed method to demonstrate a very high

Model	Corpus	T-Prec	T-Rec	T-F	L-Prec	L-Rec	L-F	Time
NPY(3)	CHILDES	0.7480	0.7520	0.7500	0.4780	**0.5970**	0.5310	17 min
NPY(2)	CHILDES	0.7480	**0.7670**	**0.7570**	0.5730	0.5660	0.5700	17 min
HDP(2)	CHILDES	0.7520	0.6960	0.7230	0.6350	0.5520	**0.5910**	-
Ent-MDL	CHILDES	**0.7634**	0.7453	0.7542	**0.6844**	0.5170	0.5890	**2.60 sec**
NPY(2)	Kyoto	-	-	0.6210	-	-	-	-
NPY(3)	Kyoto	-	-	0.6660	-	-	-	-
Ent-MDL	Kyoto	0.6912	0.7365	**0.7131**	0.5908	0.2720	0.3725	70.16 sec

Table 7: Comparison of the proposed method (Ent-MDL) with the methods of Mochihashi et al., 2009 (NPY) and Goldwater et al., 2009 (HDP).

performance in terms of both accuracy and speed.

Possible improvements of the proposed method include modeling the dependencies among neighboring tokens, which would allow the evaluation of the context to be reflected in the cost function. Mechanisms for stochastic optimization implemented in the place of the greedy algorithms could provide an additional flexibility of search for such more complex models. As the proposed approach provides significant performance improvements, it could be utilized in the development of more sophisticated novel word induction schemes, e.g. ensemble models trained independently with different data. Of course, we are also going to explore the model's potential in the setting of semi-supervised morphological analysis.

References

Bernstein-Ratner, Nan 1987. The phonology of parent – child speech. *Childrens Language*, 6:159–174

Brent, Michael R and Timothy A. Cartwright. 1996. Distributional Regularity and Phonotactic Constraints are Useful for Segmentation. *Cognition* 61: 93–125

Goldwater, Sharon. 2006. Nonparametric Bayesian Models of Lexical Acquisition. *Brown University*, Ph.D. Thesis

Goldwater, Sharon, Thomas L. Griffiths and Mark Johnson. 2006. Contextual dependencies in unsupervised word segmentation. *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, Sydney*, 673–680

Goldwater, Sharon, Thomas L. Griffiths and Mark Johnson. 2009. A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112:1, 21–54.

Harris, Zellig. 1955. From Phoneme to Morpheme. *Language*, 31(2):190-222.

Huang, Jin H. and David Powers. 2003. Chinese Word Segmentation Based on Contextual Entropy. *Proceedings of 17th Pacific Asia Conference*, 152–158

Hutchens, Jason L. and Michael D. Alder. 1998. Finding structure via compression. *Proceedings of the International Conference on Computational Natural Language Learning*, 79–82

Jin, Zhihui and Kumiko Tanaka-Ishii. 2006. Unsupervised Segmentation of Chinese Text by Use of Branching Entropy. *Proceedings of the COLING/ACL on Main conference poster sessions*, 428–435

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

Kempe, Andre. 1999. Experiments in Unsupervised Entropy Based Corpus Segmentation. *Proceedings of CoNLL'99*, pp. 371–385

Kit, Chunyu. 2003. How does lexical acquisition begin? A cognitive perspective. *Cognitive Science* 1(1): 1–50.

Kurohashi, Sadao and Makoto Nagao. 1998. Building a Japanese Parsed Corpus while Improving the Parsing System. *Proceedings of the First International Conference on Language Resources and Evaluation, Granada, Spain*, 719–724

Lafferty, John, Andrew McCallum and Fernando Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *Proceedings of the International Conference on Machine Learning*.

Li, Hang. 1998. A Probabilistic Approach to Lexical Semantic Knowledge Acquisition and Structural Disambiguation. *University of Tokyo*, Ph.D. Thesis

Liang, Percy. 2005. Semi-Supervised Learning for Natural Language. *Massachusets Institute of Technology*, Master's Thesis.

Manber, Udi and Gene Myers. 1991. Suffix arrays: a new method for on-line string searches. *SIAM Journal on Computing* 22:935–948

Marcus, Mitchell, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz and Britta Schasberger. 1994. The Penn Treebank: Annotating Predicate Argument Structure. *Human Language Technology*, 114–119

Mochihashi, Daiichi, Takeshi Yamada and Naonori Ueda. 2009. Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, 1: 100–108

Rissanen, Jorma. 1978. Modeling by Shortest Data Description. *Aulomatica*, 14:465–471.

Saffran, Jenny R., Richard N. Aslin and Elissa L. Newport. 1996. Statistical learning in 8-month-old infants *Science*; 274:1926-1928

Tsuboi, Yuta, Hisashi Kashima., Hiroki Oda, Shinsuke Mori and Yuji Matsumoto. 2008. Training Conditional Random Fields Using Incomplete Annotations. *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*,897–904.

Yu, Hua. 2000. Unsupervised word induction using MDL criterion. *Proceedings of tne International Symposium of Chinese Spoken Language Processing, Beijing*.

Zipf, George K. 1949. Human Behavior and the Principle of Least Effort. *Addison-Wesley*.