An Improved MDL-Based Compression Algorithm for Unsupervised Word Segmentation

Ruey-Cheng Chen
National Taiwan University
1 Roosevelt Rd. Sec. 4
Taipei 106, Taiwan
rueycheng@turing.csie.ntu.edu.tw

Abstract

We study the mathematical properties of a recently proposed MDL-based unsupervised word segmentation algorithm, called regularized compression. Our analysis shows that its objective function can be efficiently approximated using the negative empirical pointwise mutual information. The proposed extension improves the baseline performance in both efficiency and accuracy on a standard benchmark.

1 Introduction

Hierarchical Bayes methods have been mainstream in unsupervised word segmentation since the dawn of hierarchical Dirichlet process (Goldwater et al., 2009) and adaptors grammar (Johnson and Goldwater, 2009). Despite this wide recognition, they are also notoriously computational prohibitive and have limited adoption on larger corpora. While much effort has been directed to mitigating this issue within the Bayes framework (Borschinger and Johnson, 2011), many have found minimum description length (MDL) based methods more promising in addressing the scalability problem.

MDL-based methods (Rissanen, 1978) rely on underlying search algorithms to segment the text in as many possible ways and use description length to decide which to output. As different algorithms explore different trajectories in the search space, segmentation accuracy depends largely on the search coverage. Early work in this line focused more on existing segmentation algorithm, such as branching entropy (Tanaka-Ishii, 2005; Zhikov et al., 2010) and bootstrap voting experts (Hewlett and Cohen, 2009; Hewlett and Cohen, 2011). A recent study (Chen et al., 2012) on a compression-based algorithm, *regularized compression*, has achieved comparable performance result to hierarchical Bayes methods.

Along this line, in this paper we present a novel extension to the regularized compressor algorithm. We propose a lower-bound approximate to the original objective and show that, through analysis and experimentation, this amendment improves segmentation performance and runtime efficiency.

2 Regularized Compression

The dynamics behind regularized compression is similar to digram coding (Witten et al., 1999). One first breaks the text down to a sequence of characters (W_0) and then works from that representation up in an agglomerative fashion, iteratively removing word boundaries between the two selected word types. Hence, a new sequence W_i is created in the i-th iteration by merging all the occurrences of some selected bigram (x, y) in the original sequence W_{i-1}. Unlike in digram coding, where the most frequent pair of word types is always selected, in regularized compression a specialized decision criterion is used to balance compression rate and vocabulary complexity:

$$\begin{array}{ll} \text{min.} & -\alpha f(x,y) + |W_{i-1}| \Delta\tilde{H}(W_{i-1}, W_i) \\ \text{s.t.} & \text{either } x \text{ or } y \text{ is a character} \\ & f(x,y) > n_{\text{ms}}. \end{array}$$

Here, the criterion is written slightly differently. Note that $f(x, y)$ is the bigram frequency, $|W_{i-1}|$ the sequence length of W_{i-1}, and $\Delta\tilde{H}(W_{i-1}, W_i) = \tilde{H}(W_i) - \tilde{H}(W_{i-1})$ is the difference between the empirical Shannon entropy measured on W_i and W_{i-1}, using maximum likelihood estimates. Specifically, this empirical estimate $\tilde{H}(W)$ for a sequence W corresponds to:

$$\log |W| - \frac{1}{|W|} \sum_{x:types} f(x) \log f(x).$$

For this equation to work, one needs to estimate other model parameters. See Chen et al. (2012) for a comprehensive treatment.

	$f(x)$	$f(y)$	$f(z)$	$\lvert W\rvert$
W_{i-1}	k	l	0	N
W_i	$k-m$	$l-m$	m	$N-m$

Table 1: The change between iterations in word frequency and sequence length in regularized compression. In the new sequence W_i, each occurrence of the x-y bigram is replaced with a new (conceptually unseen) word z. This has an effect of reducing the number of words in the sequence.

3 Change in Description Length

The second term of the aforementioned objective is in fact an approximate to the change in description length. This is made obvious by coding up a sequence W using the Shannon code, with which the description length of W is equal to $|W|\tilde{H}(W)$. Here, the change in description length between sequences W_{i-1} and W_i is written as:

$$\Delta L = |W_i|\tilde{H}(W) - |W_{i-1}|\tilde{H}(W_{i-1}). \quad (1)$$

Let us focus on this equation. Suppose that the original sequence W_{i-1} is N-word long, the selected word type pair x and y each occurs k and l times, respectively, and altogether x-y bigram occurs m times in W_{i-1}. In the new sequence W_i, each of the m bigrams is replaced with an unseen word $z = xy$. These altogether have reduced the sequence length by m. The end result is that compression moves probability masses from one place to the other, causing a change in description length. See Table 1 for a summary to this exchange.

Now, as we expand Equation (1) and reorganize the remaining, we find that:

$$\begin{aligned}\Delta L = {} & (N-m)\log(N-m) - N\log N \\ & + k\log k - (k-m)\log(k-m) \\ & + l\log l - (l-m)\log(l-m) \\ & + 0\log 0 - m\log m\end{aligned} \quad (2)$$

Note that each line in Equation (2) is of the form $x_1\log x_1 - x_2\log x_2$ for some $x_1, x_2 \geq 0$. We exploit this pattern and derive a bound for ΔL through analysis. Consider $g(x) = x\log x$. Since $g''(x) > 0$ for $x \geq 0$, by the Taylor series we have the following relations for any $x_1, x_2 \geq 0$:

$$\begin{aligned} g(x_1) - g(x_2) &\leq (x_1 - x_2)g'(x_1), \\ g(x_1) - g(x_2) &\geq (x_1 - x_2)g'(x_2). \end{aligned}$$

Plugging these into Equation (2), we have:

$$m\log\frac{(k-m)(l-m)}{Nm} \leq \Delta L \leq \infty. \quad (3)$$

The lower bound[1] at the left-hand side is a best-case estimate. As our aim is to minimize ΔL, we use this quantity to serve as an approximate.

4 Proposed Method

Based on this finding, we propose the following two variations (see Figure 1) for the regularized compression framework:

- G_1: Replacing the second term in the original objective with the lower bound in Equation (3). The new objective function is written out as Equation (4).
- G_2: Same as G_1 except that the lower bound is divided by $f(x, y)$ beforehand. The normalized lower bound approximates the per-word change in description length, as shown in Equation (5). With this variation, the function remains in a scalarized form as the original does.

We use the following procedure to compute description length. Given a word sequence W, we write out all the induced word types (say, M types in total) entry by entry as a character sequence, denoted as C. Then the overall description length is:

$$|W|\tilde{H}(W) + |C|\tilde{H}(C) + \frac{M-1}{2}\log|W|. \quad (6)$$

Three free parameters, α, ρ, and n_{ms} remain to be estimated. A detailed treatment on parameter estimation is given in the following paragraphs.

Trade-off α This parameter controls the balance between compression rate and vocabulary complexity. Throughout this experiment, we estimated this parameter using MDL-based grid search. Multiple search runs at different granularity levels were employed as necessary.

Compression rate ρ This is the minimum threshold value for compression rate. The compressor algorithm would go on as many iteration as possible until the overall compression rate (i.e.,

[1] Sharp-eyed readers may have noticed the similarity between the lower bound and the negative (empirical) pointwise mutual information. In fact, when $f(z) > 0$ in W_{i-1}, it can be shown that $\lim_{m\to 0}\Delta L/m$ converges to the empirical pointwise mutual information (proof omitted here).

$$G_1 \equiv f(x,y)\left(\log\frac{(f(x)-f(x,y))(f(y)-f(x,y))}{|W_{i-1}|f(x,y)}-\alpha\right) \quad (4)$$

$$G_2 \equiv -\alpha f(x,y)+\log\frac{(f(x)-f(x,y))(f(y)-f(x,y))}{|W_{i-1}|f(x,y)} \quad (5)$$

Figure 1: The two newly-proposed objective functions.

word/character ratio) is lower than ρ. Setting this value to 0 forces the compressor to go on until no more can be done. In this paper, we experimented with predetermined ρ values as well as those learned from MDL-based grid search.

Minimum support n_{ms} We simply followed the suggested setting $n_{\text{ms}} = 3$ (Chen et al., 2012).

5 Evaluation

5.1 Setup

In the experiment, we tested our methods on Brent's derivation of the Bernstein-Ratner corpus (Brent and Cartwright, 1996; Bernstein-Ratner, 1987). This dataset is distributed via the CHILDES project (MacWhinney and Snow, 1990) and has been commonly used as a standard benchmark for phonetic segmentation. Our baseline method is the original regularized compressor algorithm (Chen et al., 2012). In our experiment, we considered the following three search settings for finding the model parameters:

(a) Fix ρ to 0 and vary α to find the best value (in the sense of description length);

(b) Fix α to the best value found in setting (a) and vary ρ;

(c) Set ρ to a heuristic value 0.37 (Chen et al., 2012) and vary α.

Settings (a) and (b) can be seen as running a stochastic grid searcher one round for each parameter[2]. Note that we tested (c) here only to compare with the best baseline setting.

5.2 Result

Table 2 summarizes the result for each objective and each search setting. The best (α, ρ) pair for G_1 is (0.03, 0.38) and the best for G_2 is (0.002, 0.36). On one hand, the performance of G_1 is consistently inferior to the baseline across all settings. Although approximation error was one possible cause, we noticed that the compression process was no longer properly regularized, since $f(x,y)$ and the ΔL estimate in the objective are intermingled. In this case, adjusting α has little effect in balancing compression rate and complexity.

Run		P	R	F
Baseline		76.9	81.6	79.2
G_1 (a)	$\alpha : 0.030$	76.4	79.9	78.1
G_1 (b)	$\rho : 0.38$	73.4	80.2	76.8
G_1 (c)	$\alpha : 0.010$	75.7	80.4	78.0
G_2 (a)	$\alpha : 0.002$	**82.1**	80.0	81.0
G_2 (b)	$\rho : 0.36$	79.1	81.7	80.4
G_2 (c)	$\alpha : 0.004$	79.3	**84.2**	**81.7**

Table 2: The performance result on the Bernstein-Ratner corpus. Segmentation performance is measured using word-level precision (P), recall (R), and F-measure (F).

The second objective G_2, on the other hand, did not suffer as much from the aforementioned lack of regularization. We found that, in all three settings, G_2 outperforms the baseline by 1 to 2 percentage points in F-measure. The best performance result achieved by G_2 in our experiment is 81.7 in word-level F-measure, although this was obtained from search setting (c), using a heuristic ρ value 0.37. It is interesting to note that G_1 (b) and G_2 (b) also gave very close estimates to this heuristic value. Nevertheless, it remains an open issue whether there is a connection between the optimal ρ value and the true word/token ratio ($\approx$ 0.35 for Bernstein-Ratner corpus).

The result has led us to conclude that MDL-based grid search is efficient in optimizing segmentation accuracy. Minimization of description length is in general aligned with performance improvement, although under finer granularity MDL-based search may not be as effec-

[2] A more formal way to estimate both α and ρ is to run a stochastic searcher that varies between settings (a) and (b), fixing the best value found in the previous run. Here, for simplicity, we leave this to future work.

Method		P	R	F
Adaptors grammar, colloc3-syllable	Johnson and Goldwater (2009)	86.1	88.4	87.2
Regularized compression + MDL, G_2 (b)	—	79.1	81.7	80.4
Regularized compression + MDL	Chen et al. (2012)	76.9	81.6	79.2
Adaptors grammar, colloc	Johnson and Goldwater (2009)	78.4	75.7	77.1
Particle filter, unigram	Börschinger and Johnson (2012)	–	–	77.1
Regularized compression + MDL, G_1 (b)	—	73.4	80.2	76.8
Bootstrap voting experts + MDL	Hewlett and Cohen (2011)	79.3	73.4	76.2
Nested Pitman-Yor process, bigram	Mochihashi et al. (2009)	74.8	76.7	75.7
Branching entropy + MDL	Zhikov et al. (2010)	76.3	74.5	75.4
Particle filter, bigram	Börschinger and Johnson (2012)	–	–	74.5
Hierarchical Dirichlet process	Goldwater et al. (2009)	75.2	69.6	72.3

Table 3: The performance chart on the Bernstein-Ratner corpus, in descending order of word-level F-measure. We deliberately reproduced the results for adaptors grammar and regularized compression. The other measurements came directly from the literature.

tive. In our experiment, search setting (b) won out on description length for both objectives, while the best performance was in fact achieved by the others. It would be interesting to confirm this by studying the correlation between description length and word-level F-measure.

In Table 3, we summarize many published results for segmentation methods ever tested on the Bernstein-Ratner corpus. Of the proposed methods, we include only setting (b) since it is more general than the others. From Table 3, we find that the performance of G_2 (b) is competitive to other state-of-the-art hierarchical Bayesian models and MDL methods, though it still lags 7 percentage points behind the best result achieved by adaptors grammar with colloc3-syllable. We also compare adaptors grammar to regularized compressor on average running time, which is shown in Table 4. On our test machine, it took roughly 15 hours for one instance of adaptors grammar with colloc3-syllable to run to the finish. Yet an improved regularized compressor could deliver the result in merely 1.25 second. In other words, even in an 100×100 grid search, the regularized compressor algorithm can still finish 4 to 5 times earlier than one single adaptors grammar instance.

Method	Time (s)
Adaptors grammar, colloc3-syllable	53826
Adaptors grammar, colloc	10498
Regularized compressor	1.51
Regularized compressor, G_1 (b)	0.60
Regularized compressor, G_2 (b)	1.25

Table 4: The average running time in seconds on the Bernstein-Ratner corpus for adaptors grammar (per fold, based on trace output) and regularized compressors, tested on an Intel Xeon 2.5GHz 8-core machine with 8GB RAM.

6 Concluding Remarks

In this paper, we derive a new lower-bound approximate to the objective function used in the regularized compression algorithm. As computing the approximate no longer relies on the change in lexicon entropy, the new compressor algorithm is made more efficient than the original. Besides runtime efficiency, our experiment result also shows improved performance. Using MDL alone, one proposed method outperforms the original regularized compressor (Chen et al., 2012) in precision by 2 percentage points and in F-measure by 1. Its performance is only second to the state of the art, achieved by adaptors grammar with colloc3-syllable (Johnson and Goldwater, 2009).

A natural extension of this work is to reproduce this result on some other word segmentation benchmarks, specifically those in other Asian languages (Emerson, 2005; Zhikov et al., 2010). Furthermore, it would be interesting to investigate stochastic optimization techniques for regularized compression that simultaneously fit both α and ρ. We believe this would be the key to adapt the algorithm to larger datasets.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback.

References

Nan Bernstein-Ratner. 1987. The phonology of parent child speech. *Children's language*, 6:159–174.

Benjamin Borschinger and Mark Johnson. 2011. A particle filter algorithm for bayesian word segmentation. In *Proceedings of the Australasian Language Technology Association Workshop 2011*, pages 10–18, Canberra, Australia, December.

Benjamin Börschinger and Mark Johnson. 2012. Using rejuvenation to improve particle filtering for bayesian word segmentation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 85–89, Jeju Island, Korea, July. Association for Computational Linguistics.

Michael R. Brent and Timothy A. Cartwright. 1996. Distributional regularity and phonotactic constraints are useful for segmentation. In *Cognition*, pages 93–125.

Ruey-Cheng Chen, Chiung-Min Tsai, and Jieh Hsiang. 2012. A regularized compression method to unsupervised word segmentation. In *Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology*, SIGMORPHON '12, pages 26–34, Montreal, Canada. Association for Computational Linguistics.

Thomas Emerson. 2005. The second international chinese word segmentation bakeoff. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, volume 133. Jeju Island, Korea.

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

Daniel Hewlett and Paul Cohen. 2009. Bootstrap voting experts. In *Proceedings of the 21st international jont conference on Artifical intelligence*, IJCAI'09, pages 1071–1076, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Daniel Hewlett and Paul Cohen. 2011. Fully unsupervised word segmentation with BVE and MDL. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2*, HLT '11, pages 540–545, Portland, Oregon. Association for Computational Linguistics.

Mark Johnson and Sharon Goldwater. 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*, NAACL '09, pages 317–325, Boulder, Colorado. Association for Computational Linguistics.

Brian MacWhinney and Catherine Snow. 1990. The child language data exchange system: an update. *Journal of child language*, 17(2):457–472, June.

Daichi Mochihashi, Takeshi Yamada, and Naonori Ueda. 2009. Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling. In *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 AFNLP: Volume 1 - Volume 1*, ACL '09, pages 100–108, Suntec, Singapore. Association for Computational Linguistics.

Jorma Rissanen. 1978. Modeling by shortest data description. *Automatica*, 14(5):465–471, September.

Kumiko Tanaka-Ishii. 2005. Entropy as an indicator of context boundaries: An experiment using a web search engine. In Robert Dale, Kam-Fai Wong, Jian Su, and Oi Kwong, editors, *Natural Language Processing IJCNLP 2005*, volume 3651 of *Lecture Notes in Computer Science*, chapter 9, pages 93–105. Springer Berlin / Heidelberg, Berlin, Heidelberg.

Ian H. Witten, Alistair Moffat, and Timothy C. Bell. 1999. *Managing gigabytes (2nd ed.): compressing and indexing documents and images*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Valentin Zhikov, Hiroya Takamura, and Manabu Okumura. 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, Cambridge, Massachusetts. Association for Computational Linguistics.