擬似確率的単語分割コーパスによる言語モデルの改良

森　信介[†]　小田　裕樹

言語モデルの分野適応において、適応対象の分野の単語境界情報のない生コーパスの有効な利用方法として、確率的単語分割コーパスとしての利用が提案されている。この枠組では、生コーパス中の各文字間に単語境界が存在する確率を付与し、それを用いて単語 n-gram 確率などが計算される。本論文では、この単語境界確率を最大エントロピー法に基くモデルによって推定することを提案する。さらに、確率的単語分割コーパスを従来の決定的に単語に分割されたコーパスで模擬する方法を提案し、言語モデルの能力を下げることなく計算コストが大幅に削減できることを示す。

キーワード：　言語モデル　確率的単語分割　音声認識　仮名漢字変換

Language Model Improvement by a Pseudo-Stochastically Segmented Corpus

Shinsuke MORI[†] and Hiroki ODA

Language model (LM) building needs a corpus whose sentences are segmented into words. For languages in which the words are not delimited by whitespace, an automatic word segmenter built from a general domain corpus is used. Automatically segmented sentences, however, contain many segmentation errors especially around words and expressions belonging to the target domain. To cope with segmentation errors, the concept of stochastic segmentation has been proposed. In this framework, a corpus is annotated with word boundary probabilities that a word boundary exists between two characters. In this paper, first we propose a method to estimate word boundary probabilities based on an maximum entropy model. Next we propose a method for simulating a stochastically segmented corpus by a segmented corpus and show that the computational cost is drastically reduced without a performance degradation.

KeyWords: *Language Modeling, Stochastic Segmentation, Speech Recognition, Kana-kanji Convertor*

† 京都大学学術情報メディアセンター (本研究の一部は日本アイ・ビー・エム在籍中になされた), Academic Center for Computing and Media Studies, Kyoto University (Some parts of this research were done when the first author was at IBM Japan, Ltd.)

1 はじめに

一般的な分野において精度の高い単語分割済みコーパスが利用可能になってきた現在、言語モデルの課題は、言語モデルを利用する分野への適応、すなわち、適応対象分野に特有の単語や表現の統計的振る舞いを的確に捉えることに移ってきている。この際の標準的な方法では、適応対象のコーパスを自動的に単語分割し、単語 n-gram 頻度などが計数される。この際に用いられる自動単語分割器は、一般分野の単語分割済みコーパスから構築されており、分割誤りの混入が避けられない。特に、適切に単語分割される必要がある適応対象分野に特有の単語や表現やその近辺において誤る傾向があり、単語 n-gram 頻度などの信頼性を著しく損なう結果となる。

上述の単語分割誤りの問題に対処するため、確率的単語分割コーパスという概念が提案されている (森, 宅間, 倉田 2007)。この枠組では、適応対象の生コーパスは、各文字の間に単語境界が存在する確率が付与された確率的単語分割コーパスとみなされ、単語 n-gram 確率が計算される。従来の決定的に自動単語分割された結果を用いるより予測力の高い言語モデルが構築できることが確認されている。また、仮名漢字変換 (森 2007) や音声認識 (Kurata, Mori, and Nishimura 2006) においても、従来手法に対する優位性が示されている。

確率的単語分割コーパスの初期の論文では、単語境界確率は、自動分割により単語境界と推定された箇所で単語分割の精度 α (例えば 0.95) とし、そうでない箇所で $1-\alpha$ とする単純な方法により与えられている[1]。実際には、単語境界が存在すると推定される確率は、文脈に応じて幅広い値を取ると考えられる。例えば、学習コーパスからはどちらとも判断できない箇所では 1/2 に近い値となるべきであるが、既存手法では 1 に近い α か、0 に近い $1-\alpha$ とする他ない。

この問題に加えて、既存の決定的に単語分割する手法よりも計算コスト (計算時間、記憶領域) が高いことが挙げられる。その要因は 2 つある。1 つ目は、期待頻度の計算に要する演算の種類と回数である。通常の手法では、学習コーパスは単語に分割されており、これを先頭から単語毎に順に読み込んで単語辞書を検索して番号に変換し、対応する単語 n-gram 頻度をインクリメントする。単語辞書の検索は、辞書をオートマトンにしておくことで、コーパスの読み込みと比較して僅かなオーバーヘッドで行える (森 1997)。これに対して、確率的単語分割コーパスにおいては、全ての連続する n 個の部分文字列 (L 文字) に対して、$L+1$ 回の浮動小数点数の積を実行して期待頻度を計算し、さらに 1 回の加算を実行する必要がある (第 2.2 項参照)。2 つ目の要因は、学習コーパスのほとんど全ての部分文字列が単語候補になるため、語彙サイズが非常に大きくなることである。この結果、単語 n-gram の頻度や確率の記憶領域が膨大となり、個人向けの計算機では動作しなくなるなどの重大な制限が発生する。例えば、本論文で実験に用いた 44,915 文の学習コーパスに出現する句読点を含まない 16 文字以下の部分文字列

1 前後の文字種 (漢字、平仮名、片仮名、記号、アラビア数字、西洋文字) によって場合分けし、単語境界確率を学習コーパスから最尤推定しておく方法 (森 宅間 2004) も提案されているが、構築されるモデルの予測力は単語分割の精度を用いる場合よりも有意に低い。後述する実験条件では、文字種を用いる方法によって構築されたモデルと単語分割の精度を用いる方法によって構築されたモデルによるエントロピーはそれぞれ 4.723[bit] と 3.986[bit] であった。

は 9,379,799 種類あった。このうち、期待頻度が 0 より大きい部分文字列と既存の語彙を加えて重複を除いた結果を語彙とすると、そのサイズは 9,383,985 語となり、この語彙に対する単語 2-gram 頻度のハッシュによる記憶容量は 10.0GB となった。このような時間的あるいは空間的な計算コストにより、確率的単語分割コーパスからの言語モデル構築は実用性が高いとは言えない。このことに加えて、単語クラスタリング (Brown, Pietra, deSouza, Lai, and Mercer 1992) や文脈に応じた参照履歴の伸長 (Ron, Singer, and Tishby 1996) などのすでに提案されている様々な言語モデルの改良を試みることが困難になっている。

本論文では、まず、確率的単語分割コーパスにおける新しい単語境界確率の推定方法を提案する。さらに、確率的単語分割コーパスを通常の決定的に単語に分割されたコーパスにより模擬する方法を提案する。最後に、実験の結果、言語モデルの能力を下げることなく、確率的単語分割コーパスの利用において必要となる計算コストが大幅に削減可能であることを示す。これにより、高い性能の言語モデルを基礎として、既存の言語モデルの改良法を試みることが容易になる。

2 確率的単語分割コーパスからの言語モデルの推定

確率的言語モデルを新たな分野に適応する一般的な方法は、適応分野のコーパスを用意し、それを自動的に単語分割し、単語の頻度統計を計算することである。この方法では、単語分割誤りにより適応分野のコーパスにのみ出現する単語が適切に扱えないという問題が起こる。この解決方法として、適応分野のコーパスを確率的単語分割コーパスとして用いることが提案されている (森他 2007)。この節では、確率的単語分割コーパスからの確率的言語モデルの推定方法について概説する。

2.1 確率的単語分割コーパス

確率的単語分割コーパスは、生コーパス C_r(以下、文字列 $\boldsymbol{x}_1^{n_r}$ として参照) とその連続する各 2 文字 x_i, x_{i+1} の間に単語境界が存在する確率 P_i の組として定義される。最初の文字の前と最後の文字の後には単語境界が存在するとみなせるので、$i = 0,\ i = n_r$ の時は便宜的に $P_i = 1$ とされる。確率変数 X_i を

$$X_i = \begin{cases} 1 & x_i, x_{i+1} \text{ の間に単語境界が存在する場合} \\ 0 & x_i, x_{i+1} \text{ が同じ単語に属する場合} \end{cases}$$

とし $(P(X_i = 1) = P_i,\ P(X_i = 0) = 1 - P_i)$、各 $X_0, X_1, \ldots, X_{n_r}$ は独立であることが仮定される。

文献 (森他 2007) の実験で用いられている単語境界確率の推定方法は次の通りである。まず、単語に分割されたコーパスに対して自動単語分割システムの境界推定精度 α を計算しておく。

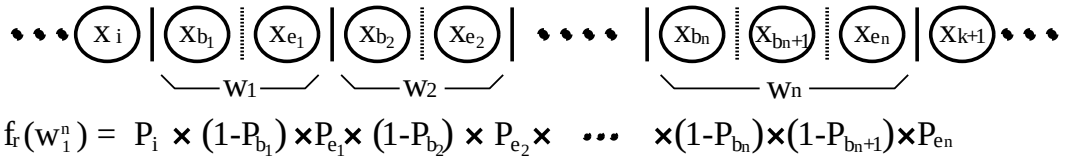

図 1 確率的単語分割コーパスにおける単語 n-gram 頻度

次に、適応分野のコーパスを自動単語分割し、その出力において単語境界であると判定された点では $P_i = \alpha$ とし、単語境界でないと判定された点では $P_i = 1 - \alpha$ とする。後述する実験の従来手法としてこの方法を採用した。

2.2 単語 n-gram 頻度

確率的単語分割コーパスに対して単語 n-gram 頻度が以下のように定義される。

単語 0-gram **頻度** 確率的単語分割コーパスの期待単語数として以下のように定義される。

$$f(\cdot) = 1 + \sum_{i=1}^{n_r - 1} P_i \tag{1}$$

単語 1-gram **頻度** 確率的単語分割コーパスに出現する文字列 $\boldsymbol{x}_{i+1}^{k}$ が $l = k - i$ 文字からなる単語 $w = \boldsymbol{x'}_1^l$ である必要十分条件は以下の 4 つである。

(1) 文字列 $\boldsymbol{x}_{i+1}^{k}$ が単語 w に等しい。

(2) 文字 x_{i+1} の直前に単語境界がある。

(3) 単語境界が文字列中にない。

(4) 文字 x_k の直後に単語境界がある。

したがって、確率的単語分割コーパスの単語 1-gram 頻度 f_r は、単語 w の表記の全ての出現 $O_1 = \{(i,k) \mid \boldsymbol{x}_{i+1}^{k} = w\}$ に対する期待頻度の和として以下のように定義される。

$$f_r(w) = \sum_{(i,k) \in O_1} P_i \left[\prod_{j=i+1}^{k-1} (1 - P_j) \right] P_k \tag{2}$$

単語 n-gram **頻度** ($\boldsymbol{n \geq 2}$) L 文字からなる単語列 $\boldsymbol{w}_1^n = \boldsymbol{x'}_1^L$ の確率的単語分割コーパス $\boldsymbol{x}_1^{n_r}$ における頻度、すなわち単語 n-gram 頻度について考える。このような単語列に相当する文字列が確率的単語分割コーパスの $(i+1)$ 文字目から始まり $k = i + L$ 文字目で終る文字列と等しく ($\boldsymbol{x}_{i+1}^{k} = \boldsymbol{x'}_1^L$)、単語列に含まれる各単語 w_m に相当する文字列が確率的単語分割コーパスの b_m 文字目から始まり e_m 文字目で終る文字列と等しい ($\boldsymbol{x}_{b_m}^{e_m} = w_m,\ 1 \leq \forall m \leq n;\ e_m + 1 = b_{m+1},\ 1 \leq \forall m \leq n-1;\ b_1 = i + 1;\ e_n = k$) 状況を考える (図 1 参照)。確率的単語分割コーパスに出現する文字列 $\boldsymbol{x}_{i+1}^{k}$ が単語列 $\boldsymbol{w}_1^n = \boldsymbol{x'}_1^L$

である必要十分条件は以下の 4 つである。

(1)　文字列 $\boldsymbol{x}_{i+1}^{k}$ が単語列 $\boldsymbol{w}_1^n$ に等しい。

(2)　文字 x_{i+1} の直前に単語境界がある。

(3)　単語境界が各単語に対応する文字列中にない。

(4)　単語境界が各単語に対応する文字列の後にある。

確率的単語分割コーパスにおける単語 n-gram 頻度は以下のように定義される。

$$f_r(\boldsymbol{w}_1^n) = \sum_{(i,e_1^n)\in O_n} P_i \left[\prod_{m=1}^{n} \left\{ \prod_{j=b_m}^{e_m-1} (1-P_j) \right\} P_{e_m} \right] \tag{3}$$

ここで

$$\begin{aligned} e_1^n &= (e_1, e_2, \cdots, e_n) \\ O_n &= \{(i, e_1^n) | \boldsymbol{x}_{b_m}^{e_m} = w_m, 1 \le m \le n\} \end{aligned}$$

である。

2.3　単語 n-gram 確率

確率的単語分割コーパスにおける単語 n-gram 確率は、単語 n-gram 頻度の相対値として計算される。

単語 1-gram 確率　以下のように単語 1-gram 頻度を単語 0-gram 頻度で除することで計算される。

$$P_r(w) = \frac{f_r(w)}{f_r(\cdot)} \tag{4}$$

単語 n-gram 確率 ($n \ge 2$)　以下のように単語 n-gram 頻度を単語 $(n-1)$-gram 頻度で除することで計算される。

$$P_r(w_n | \boldsymbol{w}_1^{n-1}) = \frac{f_r(\boldsymbol{w}_1^n)}{f_r(\boldsymbol{w}_1^{n-1})} \tag{5}$$

2.4　単語 n-gram 頻度の計算コスト

ある単語列 $\boldsymbol{w}_1^n = \boldsymbol{x'}_1^L$ $(n \ge 1)$ のある 1 箇所の出現位置に対する期待頻度の計算に必要な演算は、式 (2) や式 (3) から明かなように、$L-n$ 回の浮動小数点に対する減算と $L+1$ 回の浮動小数点に対する乗算である。動的に単語 n-gram 確率を計算する方法では、この演算が文字列 $\boldsymbol{x'}_1^L$ の出現回数だけ繰り返される。通常の決定的単語分割コーパスの場合には、単語列の出現回数がそのまま頻度となるので、上述の浮動小数点に対する演算が全て付加的な計算コストであり、言語モデルの応用の実行速度を大きく損ねる。あらかじめ単語 n-gram 確率を計算しておく場合は、ある文 (文字数 h) に出現する全ての n 個の連続する部分文字列に対して行う必要がある。上述の減算や乗算が重複して行われるのを避けるために、まず文の両端を除く全ての位

置に対して $1-P_i$ を計算 ($h-1$ 回の減算) し、さらにこれら $h-1$ 個の $1-P_i$ のうちの任意個の連続する位置に対する積 $\prod_{j=b}^{e}(1-P_j)$ $(b<e)$ を計算 ($\sum_{i=1}^{h-2}=(h-1)(h-2)/2$ 回の乗算) しておく。ある単語 n-gram の出現位置は、文に $n+1$ 個の単語境界を置くことで決るので、h 文字の文には重複を含め ${}_{h-1}C_{n+1}$ 個の単語 n-gram が含まれる。このそれぞれの期待頻度は、左端の P_i に n 個の $\prod_{j=e_{k-1}+1}^{e_k}(1-P_j)$ と n 個の P_{e_k} の積 ($2n$ 回の乗算) として得られる。この場合に必要な計算コストも、決定的単語分割コーパスの場合の単語数と同じ回数のインクリメントに比べて非常に大きい[2]。

このように、確率的単語分割コーパスに対する単語 n-gram 頻度の計算のコストは、従来の決定的単語分割コーパスに対する計算コストに比べて非常に大きくなる。文の長さの分布を無視すれば、計算回数はコーパスの文数に対しては比例する。文毎に独立なので複数の計算機による分散計算も可能であるが、ある程度の大きさのコーパスからモデルを作成する場合にはこの計算コストは問題になる。また、単語クラスタリング (Brown et al. 1992) や文脈に応じた参照履歴の伸長 (Ron et al. 1996) などの様々な言語モデルの改良においては、最適化の過程において言語モデルを何度も構築する。確率的単語分割コーパスにおける単語 n-gram 頻度の計算のコストによって、これらの改良を試みることが困難になっている。

3 最大エントロピー法による単語境界確率の推定

この節では、最大エントロピー法による単語分割器を単語境界確率の推定に用いる方法について述べる。

3.1 単語境界確率の推定

日本語の単語分割の問題は、入力文の各文字間に単語境界が発生するか否かを予測する問題とみなせる (風間, 宮尾, 辻井 2004; Tsuboi, Kashima, Mori, Oda, and Matsumoto 2008)。つまり、文 $\boldsymbol{x}=x_1x_2\cdots x_m$ に対して、x_ix_{i+1} の間が単語境界であるか否かを表すタグ t_i を付与する問題とみなす。付与するタグは、単語境界であることを表すタグ E と、非単語境界であることを表すタグ N の 2 つのタグからなる。各文字間のタグがこのいずれかであるかは、単語境界が明示されたコーパスから学習された点推定の最大エントロピーモデル (ME model; maximum entropy model) により推定する[3]。その結果、より高い確率を与えられたタグをその文字間のタ

2 後述の実験での条件では、自動分割結果 (決定的単語分割) からの頻度計算におけるインクリメントは 1,377,062 回で、確率的単語分割に対する $n=2$ の乗算回数の理論値は 20,181,679,570 となる。浮動小数点数に対する乗算とインクリメントでは、計算のコストが異なるが、回数を単純に比較しても実に 14,656 倍となる。実際の計算時間には、さらに、入力文の読み込みや文字列 (単語表記) から語彙番号への変換が含まれるので、この比率にはならない。

3 文献 (Tsuboi et al. 2008) のように CRF(conditional random fields) により推定することもできるが、計算コストと記憶領域が大きくなる。これらの差は、スパースな部分的アノテーションコーパスからの学習において顕著となる。つまり、CRF のように系列としてモデル化する方法では、アノテーションのない部分も考慮する必要があるのに対して、点推定の最大エントロピーモデルでは、アノテーションのある部分のみを考慮すればよい。このような考察から、本論文では計算コストの少ない最大エントロピーモデルを用いる。

グとし、単語境界を決定する。すなわち、以下の式が示すように、最大エントロピーモデルにより、単語境界と推定される確率が非単語境界と推定される確率より高い文字間を単語境界とする。

$$t_i = \begin{cases} \mathrm{E} & \text{if } P_{ME}(t_i = E|\boldsymbol{x}) > P_{ME}(t_i = N|\boldsymbol{x}) \\ \mathrm{N} & \text{otherwise} \end{cases}$$

これにより、入力文を単語に分割することができる。

本論文では、以下のように、タグ t_i の出現確率を確率的単語分割コーパスにおける単語境界確率 P_i として用いることを提案する。

$$P_i = P_{ME}(t_i = E|\boldsymbol{x})$$

これにより、注目する文字の周辺のさまざまな素性を参照し、単語境界確率を適切に推定することが可能になる。

3.2 参照する素性

後述する実験においては、$x_i x_{i+1}$ の間に注目する際の最大エントロピーモデルの素性としては、x_{i-1}^{i+2} の範囲の文字 n-gram および字種 n-gram$(n = 1, 2, 3)$ をすべて用いた[4]。ただし、以下の点を考慮している。

- 素性として利用する n-gram は、先頭文字の字種がその前の文字の字種と同じか否か、および、末尾文字の字種がその次の文字の字種と同じか否かの情報を付加して参照する[5]。
- 素性には注目する文字間の位置情報を付加する。

たとえば、文字列「文字列を単語に分割する」の「語」「に」の文字間の素性は、{ − 単 +|, + 語 |−, −| に −, |− 分 +, − 単語 |−, + 語 | に −, −| に分 +, − 単語 | に −, + 語 | に分 +, −K+|, +K|−, −|H−, |−K+, −KK|−, +K|H−, −|HK+, −KK|H−, +K|HK+, } となる。「|」は注目する文字間を表す補助記号であり、「+」と「−」は前後の文字が同じ字種である (+) か否 (−) かを表す補助記号である。「H」と「K」は字種の平仮名と漢字を表している。

なお、実験においては、パラメータ数を減らすために、学習データで 2 回以上出現する素性のみを用いた。また、最大エントロピーモデルのパラメータ推定には、GIS アルゴリズム (Darroch and Ratcliff 1972) を使用した。

4 疑似確率的単語分割コーパス

確率的単語分割コーパスに対する単語 n-gram 頻度は、高いコストの計算を要する。また、確率的単語分割コーパスは、頻度計算の対象となる単語や単語断片 (候補) を多数含む。ある単

4 字種は、漢字、ひらがな、カタカナ、アルファベット、数字、記号の 6 つとした。

5 パラメータ数の急激な増加を抑えつつ素性の情報量を増加させる。これにより、参照範囲を前後 1 文字拡張して x_{i-2}^{i+3} の範囲の n-gram$(n = 3, 4, 5)$ を参照する。

語 n-gram の頻度の計算に際しては、その単語の文字列としてのすべての出現に対して、頻度のインクリメントではなく、複数回の浮動小数点演算を実行しなければならない。この計算コストにより、より長い履歴を参照する単語 n-gram モデルや単語クラスタリングなどの言語モデルの改良が困難になっている。

上述の困難を回避する方法として、単語分割済みコーパスで確率的単語分割コーパスを近似する方法を提案する。具体的には、確率的単語分割コーパスに対して以下の処理を最初の文字から最後の文字まで $(1 \leq i \leq n_r)$ 行なう。

(1) 文字 x_i を出力する。

(2) 0 以上 1 未満の乱数 r_i を発生させ P_i と比較する。$r_i < P_i$ の場合には単語境界記号を出力し、そうでない場合には何も出力しない。

これにより、確率的単語分割コーパスに近い単語分割済みコーパスを得ることができる。これを疑似確率的単語分割コーパスと呼ぶ。

上記の方法では、文字列としての出現頻度が低い単語 n-gram の頻度が確率的単語分割コーパスと疑似確率的単語分割コーパスにおいて大きく異なる可能性がある。そもそも、出現頻度が低い単語 n-gram の場合、単語分割が正しいとしても、その統計的振る舞いを適切に捉えるのは困難であるが、近似によって誤差が増大することは好ましくない。従って、この影響を軽減するために、上記の手続きを N 回行ない、その結果得られる N 倍の単語分割済みコーパスを単語 n-gram 頻度の計数の対象とすることとする。このときの N を本論文では倍率と呼ぶこととする。

疑似確率的単語分割コーパスは、一種のモンテカルロ法となっている。モンテカルロ法による d 次元の単位立方体上 $[0,d]^d$ 上の定積分 $I = \int_{[0,1]^d} f(x)dx$ の数値計算法では、単位立方体 $[0,d]^d$ 上の一様乱数 $x_1,\ x_2,\ \ldots,\ x_N$ を発生させて $I_N = \sum_{i=1}^{N} f(x_i)$ とする。このとき、誤差 $|I_N - I|$ は次元 d によらずに $1/\sqrt{N}$ に比例する程度の速さで減少することが知られている。疑似確率的単語分割コーパスにおける単語 n-gram 頻度の計算はこの特殊な場合であり、n の値や文字数によらずに $1/\sqrt{FN}$ に比例する程度の速さで減少する。ここで F は単語 n-gram の文字列としての頻度である。

5 評価

単語境界確率の推定方法の評価として、言語モデルの適応の実験を行なった。まず、適応対象文野の大きな生コーパスに既存手法と提案手法のそれぞれで単語境界確率を付与した。次に、その結果得られる確率的単語分割コーパスから単語 2-gram モデルを推定し、これを一般分野の単語分割済みコーパスから推定された単語 2-gram モデルと補間した。最後に、適応分野のテストコーパスに対して、予測力と仮名漢字変換 (森 2007) の精度の評価を行なった。後者は、理想的な音響モデルを用いた場合の音声認識と考えることも可能である。この節では、実験の結

表 1　一般コーパス (単語分割済み)

用 途	文 数	単語数	文字数
学 習	13,181	436,785	623,653
テスト	1,464	48,819	69,503

表 2　適応対象コーパス (単語境界情報なし)

用 途	文 数	単語数	文字数
学 習	44,915	—	1,890,041
テスト	7,000	—	293,318

主に医療文書からなる。

果を提示し、評価を行なう。

5.1 実験の条件

実験に用いたコーパスは、「現代日本語書き言葉均衡コーパス」モニター公開データ（2008年度版）中の人手による単語分割の修正がなされている文 (一般コーパス) と医療文書からなる適応対象のコーパスである。一般コーパスの各文は正しく単語に分割され、各単語に入力記号列 (読み) が付与されている。これを 10 個に分割し、この内の 9 個を学習コーパスとし、残りの 1 個をテストコーパスとした (表 1 参照)。自動単語分割器や単語境界確率の推定のための最大エントロピーモデルはこの学習コーパスから構築される。一方、適応対象のコーパスは大量にあるが、単語境界情報を持たない。この内の 7,000 文に入力記号列 (読み) を付与しテストコーパスとし、残りを確率的単語分割コーパスとして言語モデルの学習に用いた (表 2 参照)。テストコーパスの内の 1,000 文には、単語境界情報も付与し、言語モデルの予測力の評価に用いた。

5.2 評価基準

確率的言語モデルの予測力の評価に用いた基準は、テストコーパスにおける単語あたりのパープレキシティである。まず、テストコーパス C_t に対して未知語の予測も含む文字単位のエントロピー H を以下の式で計算する (森 山地 1997)。

$$H = -\frac{1}{|C_t|}\log_2 \prod_{\boldsymbol{w}\in C_t} M_{w,n}(\boldsymbol{w})$$

ここで、$M_{w,n}(\boldsymbol{w})$ は単語 n-gram モデルによる単語列 $\boldsymbol{w}$ の生成確率を、$|C_t|$ はテストコーパス C_t の文字数を表す。次に、単語単位のパープレキシティを以下の式で計算する。

$$PP = 2^{H\times\overline{|\boldsymbol{w}|}}$$

ここで $\overline{|\boldsymbol{w}|}$ は平均単語長 (文字数) である。これらの計算に際しては、単語境界情報が付与された 1,000 文を用いた[6]。

仮名漢字変換の評価基準は、文字誤り率である。文字誤り率は $\mathrm{CER} = 1 - N_{LCS}/N_{COR}$ と

6 本論文での言語モデルの予測力の評価は、文字列の予測のみならず、人手で付与された単語境界の予測も含まれている。これは、言語モデルの応用を考慮してのことである。純粋に予測力が高いモデルが必要な場合は、既存の単語単位を用いず、文字単位でモデル化する方がよいと考えられる (森, 山地, 長尾 1997; 持橋大地, 山田武士, 上田修功 2009)。

表 3 単語境界確率の推定方法と言語モデルの能力の関係

	単語境界確率の推定方法	エントロピー [bit]	パープレキシティー	文字正解率 [%]
BL	単語自動分割器の精度	3.986	49.79	97.51
ME	最大エントロピーモデル	3.872	44.53	97.58

定義される。ここで、N_{COR} は正解に含まれる文字数であり、N_{LCS} は各文を一括変換することで得られる最尤解と正解との最長共通部分列 (LCS; longest common subsequence)(Aho 1990) の文字数である。

5.3 単語境界確率の推定方法の評価

単語境界確率の推定方法の差異を調べるために、以下の 2 つの確率的単語分割コーパスを作成しそれらから推定された単語 2-gram モデルの能力を調べた。

BL: 従来手法
各単語境界確率は、単語 2-gram モデルに基づく自動単語分割器の判断に応じて α 又は $1-\alpha$ とする。ここで、$\alpha = 67372/68039$ は一般分野のテストコーパスにおける単語境界推定精度である (第 2.1 項参照)。

ME: 提案手法
各単語境界確率は、最大エントロピーモデルを用いて文脈に応じて推定される (第 3.1 項参照)。

適応対象分野のテストコーパスにおける予測力と文字誤り率を表 3 に示す。この結果から、本論文で提案する最大エントロピー法による単語境界確率の推定方法により約 11%のパープレキシティの削減が実現されている。この結果から、最大エントロピー法により推定された単語境界確率を持つ確率的単語分割コーパスを用いることで適応対象分野における単語 2-gram 確率がより正確に推定されていることがわかる。応用の仮名漢字変換においても、文字正解率の比較から、提案手法により、従来手法の文字誤りの約 3.1%が削減さた。検定の結果、有意水準 5%で有意差があるとの結果であった。この点からも言語モデルが改善されていることが確認される。従来手法の文字正解率は 97.51%と高いので、提案手法により実現された誤りの削減は十分有意義であろう。

5.4 疑似確率的単語分割コーパスの評価

本論文のもう一つの論点は、単語分割済みコーパスによる確率的単語分割コーパスの近似である。この評価として、3 種類の大きさ (1/1, 1/2, 1/4) の適応分野の疑似確率的単語分割コーパスから推定した言語モデルのテストコーパスに対するパープレキシティと文字正解率を複数の倍率 ($N = 1,\ 2,\ 4,\ \cdots, 256$) に対して計算した。表 4〜表 6 はその結果である。まず、自動

表 4　1/1 のサイズの疑似確率的単語分割コーパスから推定された言語モデルの能力

方法	学習コーパス	倍率	エントロピー [bit]	パープレキシティー	文字正解率 [%]
ME	決定的単語分割	–	4.149	58.41	97.36
ME	確率的単語分割	–	3.872	44.53	97.58
ME	疑似確率的単語分割	×1	4.036	52.27	97.40
		×2	4.010	50.95	97.45
		×4	3.983	49.66	97.48
		×8	3.955	48.29	97.51
		×16	3.944	47.77	97.53
		×32	3.925	46.92	97.54
		×64	3.917	46.54	97.55
		×128	3.905	46.00	97.56
		×256	3.896	45.58	97.56

表 5　1/2 のサイズの疑似確率的単語分割コーパスから推定された言語モデルの能力

方法	学習コーパス	倍率	エントロピー [bit]	パープレキシティー	文字正解率 [%]
ME	決定的単語分割	–	4.238	63.76	96.93
ME	確率的単語分割	–	3.971	49.08	97.13
ME	疑似確率的単語分割	×1	4.157	58.89	96.96
		×2	4.123	56.95	97.01
		×4	4.082	54.69	97.04
		×8	4.057	53.40	97.06
		×16	4.046	52.80	97.09
		×32	4.029	51.95	97.10
		×64	4.019	51.42	97.11
		×128	4.001	50.51	97.11
		×256	4.006	50.80	97.12

表 6　1/4 のサイズの疑似確率的単語分割コーパスから推定された言語モデルの能力

方法	学習コーパス	倍率	エントロピー [bit]	パープレキシティー	文字正解率 [%]
ME	決定的単語分割	–	4.363	72.09	96.44
ME	確率的単語分割	–	4.116	56.59	96.65
ME	疑似確率的単語分割	×1	4.316	68.82	96.45
		×2	4.279	66.35	96.53
		×4	4.239	63.82	96.57
		×8	4.211	62.11	96.58
		×16	4.193	61.01	96.60
		×32	4.179	60.17	96.62
		×64	4.165	59.33	96.64
		×128	4.154	58.68	96.64
		×256	4.145	58.17	96.65

分割の結果を決定的単語分割コーパスとして用いる場合についてである。これと、確率的単語分割コーパスとして用いる場合との比較では、文献 (森他 2007) の報告と同じように確率的単語分割により予測力が向上し、文献 (森 2007) の報告と同じように仮名漢字変換の文字正解率も向上している。さらに、本論文で提案する倍率が 1 の疑似確率的単語分割は、決定的単語分

割に対して、予測力と文字正解率の双方において優れていることが分る。倍率が 1 の疑似確率的単語分割と決定的単語分割の唯一の違いは、自動単語分割の際に単語境界確率を 0.5 と比較するか、0 から 1 の乱数と比較するかであり、モデル構築の計算コストはほとんど同じである。にもかかわらず、予測力と文字正解率の双方が向上している点は注目に値するであろう。

次に、確率的単語分割と疑似確率的単語分割の比較について述べる。倍率が 1 の場合は、予測力や文字正解率は、確率的単語分割コーパスから推定された言語モデルに対して少し低く、倍率を上げることによりこれらは確率的単語分割コーパスによる結果に近づいていくことがわかる。これは、疑似確率的単語分割がモンテカルロ法による数値演算の一種になっていることを考えれば当然の結果である。このことから、ある程度の倍率の疑似確率的単語分割コーパスは、確率的単語分割コーパスのよい近似となっているといえる。適応分野のコーパスの大きさに係わらず、倍率が 256 の場合の疑似確率的単語分割による結果は、確率的単語分割の結果とほぼ同じといえる。

最後に、確率的単語分割と疑似確率的単語分割の計算コストの比較について述べる。確率的単語分割の語彙サイズは、適応対象の学習コーパスにおける期待頻度が 0 より大きい 16 文字以下の部分文字列と一般コーパスの語彙の合計 9,383,985 語であった。この語彙に対する単語 2-gram 頻度をハッシュ (Berkeley DB 4.6.21) を用いてファイルに出力すると 10.0GB となった。これを RAM ディスク上で計算するのに 61147.45 秒 (約 17 時間) を要した[7]。同じ計算機で、16 倍の疑似確率的単語分割コーパスから単語 2-gram 頻度を RAM Disk 上で計算すると、語彙サイズが 46,777 語であり、単語 2-gram 頻度のファイルサイズは 9.98MB であり、計算時間は 1009.95 秒 (約 17 分) と約 61 分の 1 となった。疑似確率的単語分割コーパスを用いた場合には、倍率が 256 の場合でも 20.2MB と、ファイルサイズが大きくないので、現在の多くの計算機で主記憶上で計算が可能である (主記憶上での計算時間は 303.29 秒)。これに対して、確率的単語分割コーパスからの推定では、一部の計算機においてのみ主記憶上での計算が可能である。さらに、実験で用いた適応対象の分野のコーパスは 44,915 文と決して大きくはなく、適応分野によっては 1 桁か 2 桁ほど大きい学習コーパスが利用できることも十分考えられる。この場合には、確率的単語分割では 2 次記憶 (RAM ディスクかハードディスク) 上での計算が避けられず、モデル作成にかかる計算時間の違いは非常に大きくなる。したがって、本論文で提案する疑似確率的単語分割は、この点から有用であると考えられる。

疑似確率的単語分割において、どの程度の倍率がよいかは要求する精度と利用可能な計算機資源との兼ね合いである。例えば倍率が 16 の場合は、単語に分割された 718,640 文から言語モデルを推定することになる。モデル構築に要する計算時間は、決定的単語分割の場合の 16 倍程度であり、現在の計算機はこの大きさのコーパスを処理する能力が十分ある。したがって、疑似確率的単語分割により、単語 3-gram モデルや可変長記憶マルコフモデル、あるいは言語モデル

7 この計算に用いた計算機の中央演算装置は Intel Core 2 Duo 3.91GHz であり、主記憶は 4GB である。

のための単語クラスタリングなどさらなる言語モデルの改善を容易に試みることが可能となる。

6 おわりに

本論文では、確率的単語分割コーパスにおける新しい単語境界確率の推定方法を提案した。実験の結果、提案手法により約 11%のパープレキシティの減少と約 3.1%の文字誤りの削減が確認された。さらに、確率的単語分割コーパスを通常の決定的単語分割コーパスにより模擬する方法を提案した。実験の結果、言語モデルの能力を下げることなく、確率的単語分割コーパスの利用において必要となる計算コストが削減可能であることを示した。

謝辞

査読者から有意義なコメントを頂きました。心より感謝致します。

参考文献

Aho, A. V. (1990). “文字列中のパターン照合のためのアルゴリズム.” コンピュータ基礎理論ハンドブック, I: 形式的モデルと意味論巻, pp. 263–304. Elseveir Science Publishers.

Brown, P. F., Pietra, V. J. D., deSouza, P. V., Lai, J. C., and Mercer, R. L. (1992). “Class-Based *n*-gram Models of Natural Language.” *Computational Linguistics*, 18 (4), 467–479.

Darroch, J. and Ratcliff, D. (1972). “Generalized Iterative Scaling For Log-Linear Models.” *The annuals of Mathematical Statistics*, 43 (5), 1479–1480.

Kurata, G., Mori, S., and Nishimura, M. (2006). “Unsupervised Adaptation of a Stochastic Language Model Using a Japanese Raw Corpus.” In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*.

Ron, D., Singer, Y., and Tishby, N. (1996). “The Power of Amnesia: Learning Probabilistic Automata with Variable Memory Length.” *Machine Learning*, 25, 117–149.

Tsuboi, Y., Kashima, H., Mori, S., Oda, H., and Matsumoto, Y. (2008). “Training Conditional Random Fields Using Incomplete Annotations.” In *Proceedings of the 22th International Conference on Computational Linguistics*.

持橋大地, 山田武士, 上田修功 (2009). “ベイズ階層言語モデルによる教師なし形態素解析.” 情報処理学会研究報告, NL190 巻.

風間淳一, 宮尾祐介, 辻井潤一 (2004). “教師なし隠れマルコフモデルを利用した最大エントロピータグ付けモデル.” 自然言語処理, 11 (4), 3–24.

森信介 (1997). “DFA による形態素解析の高速辞書検索.” EDR 電子化辞書利用シンポジウム.
森信介 (2007). “無限語彙の仮名漢字変換.” 情報処理学会論文誌, 48, 3532–3540.
森信介 山地治 (1997). “**日本語の情報量の上限の推定**.” **情報処理学会論文誌**, 38 (11), 2191–2199.
森信介, **山地治**, **長尾真** (1997). “**予測単位の変更による** n-gram **モデルの改善**.” **情報処理学会研究報告**, SLP19 **巻**, pp. 87–94.
森信介 宅間大介 (2004). “**生コーパスからの単語** N-gram **確率の推定**.” **情報処理学会研究報告**, NL162 **巻**.
森信介, **宅間大介**, **倉田岳人** (2007). “**確率的単語分割コーパスからの単語** N-gram **確率の計算**.” **情報処理学会論文誌**, 48, 892–899.

略歴

森 信介: 1998 年京都大学大学院工学研究科電子通信工学専攻博士後期課程修了. 同年日本アイ・ビー・エム (株) 入社. 2007 年 5 月より京都大学学術情報メディアセンター准教授. 工学博士. 1997 年情報処理学会山下記念研究賞受賞. 情報処理学会会員.

小田 裕樹: 1999 年徳島大学大学院工学研究科博士前期課程知能情報工学専攻修了. 同年 NTT ソフトウェア (株) 入社. 言語処理・情報検索システム等の開発, コンサルティング業務に従事. 確率・統計的自然言語処理およびその応用に興味を持つ. 工学博士. 情報処理学会会員.