決定性文脈を用いた可変長 n-gram モデルによる日本語単語分割*

小田 裕樹　北 研二
徳島大学 工学部
{hiroki,kita}@is.tokushima-u.ac.jp

1 はじめに

日本語処理において，単語の同定，すなわち文の単語分割は，最も基本的かつ重要な処理である．日本語単語分割では未知語の存在が重大な問題となる．本稿では，日本語単語分割の方法として，文字モデルに基づくものを提案する．文字モデルのパラメータ数は単語モデルよりもはるかに少ないため，頑健な推定を行うことができる．文献 [1] において指摘されているように，文字モデルは，未知語モデルとしても機能するために，学習データに含まれていない単語に対してもある程度の対応が可能となる．

文字モデルとしては，決定性文脈 [2] を利用して実現した可変長 n-gram モデルを用いる．決定性文脈とは，無限長文脈を取り扱うことのできる PPM* [2] (高性能データ圧縮アルゴリズム PPM の一種) において導入されているものであり，その文脈から予測される文字が一意に決められる文脈のことを指している．可変長 n-gram モデルを用いることにより，確率推定の条件部 (文脈) に長い文字列が必要な場合や，逆に短い文字列で文字の生起が予測できる場合に対応できる．

以下，本稿では，まず，基本となる文字 n-gram モデルに基づく単語分割モデル [3] について簡単に説明する．次に，決定性文脈を利用した可変長 n-gram モデルの実現方法について説明し，それを用いた単語分割モデルを提案する．ATR 対話データベースを用いた評価実験において，各々の言語モデルを用いた場合の単語分割精度を比較し，提案した手法の評価を行う．

2 文字 n-gram モデルの適用

単語分割モデルの学習データは「⟨s⟩ はい ⟨d⟩ 、⟨d⟩ そう ⟨d⟩ です ⟨d⟩ 。 ⟨/s⟩」のように単語境界を付与したデータを用いる．ここで，特殊記号 ⟨d⟩, ⟨s⟩, ⟨/s⟩ は各々「単語境界」,「文頭」,「文末」を表している．

本節では，以上のような学習データから作成した文字 trigram (3-gram) モデルを用いて単語分割を行うことを考える．与えられた「ベタ書き」文を単語列に分割するためには，文中の各文字位置に対し，その文字の前で単語分割が起こる (1) か否 (0) かを求めればよい．このために，各文字位置に対し，直前の単語境界の有無に相当する 2 つの状態 1 と 0 を仮定する．文字位置 $i(\geq 2)$ の状態の推定は次式で与えられる．なお，$P_j(c_1^i)$ は文字列 $c_1^i = c_1 \cdots c_i$ を生成して状態 j に到達する確率を表す．

$$P_0(c_1^i) = \max(P_0(c_1^{i-1})A_i,\ P_1(c_1^{i-1})B_i) \quad (1)$$

$$P_1(c_1^i) = \max(P_0(c_1^{i-1})C_i,\ P_1(c_1^{i-1})D_i) \quad (2)$$

$$A_i = p(c_i|c_{i-2}c_{i-1}),\ B_i = p(c_i|\langle \mathrm{d}\rangle c_{i-1}),$$
$$C_i = p(\langle \mathrm{d}\rangle|c_{i-2}c_{i-1})p(c_i|c_{i-1}\langle \mathrm{d}\rangle),$$
$$D_i = p(\langle \mathrm{d}\rangle|\langle \mathrm{d}\rangle c_{i-1})p(c_i|c_{i-1}\langle \mathrm{d}\rangle)$$

ここで，文字位置 1 の状態 0 の確率は，$P_0(c_1) = p(c_1|\langle \mathrm{s}\rangle)$ により求めることができる．また，学習データ中の文字位置 1 の前には単語境界記号がないため，状態 1 の確率 $P_1(c_1)$ を 0 と定義する．

文 $s = c_1^m$ に対する最適な単語分割は，各文字位置に対する状態 1 と 0 の最適な状態遷移系列として与えられる．単語分割モデルの計算のため，実際の入力文には，文頭記号と文末記号を各々 0 番目と $m+1$ 番目の文字として加えて処理を行う．学習データ中の文末記号 ⟨/s⟩ の前には単語境界 ⟨d⟩ がないので，最適な状態遷移系列は $\max P_0(c_1^{m+1})$ となるような状態遷移系列である．これを求めるためには，動的計画法の一種であるビタビ・アルゴリズム (Viterbi algorithm) を用いることができる (図 1参照)．

最尤状態遷移系列において，状態 1 である文字位置の前で単語分割を行う (図 1中の点線)．これにより，入力文に対して最適な単語分割を得ることができる．

*Japanese Word Segmentation by a Variable-Length n-gram Model Using Deterministic Contexts
Hiroki Oda and Kenji Kita
Faculty of Engineering, Tokushima University

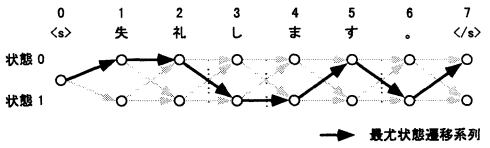

図 1: ビタビ・アルゴリズムを用いた文の分割

3 可変長モデルによる単語分割

文字 n-gram モデルは，文字の生起を $(n-1)$ 重マルコフ過程により近似したモデルである．ここで，単純に考えると，n の値を大きくすればより精度の高いモデルが得られると考えられる．しかし，高次のモデルではパラメータ数が指数的に増大するため，学習データから統計的に信頼性の高いパラメータ値を推定することがますます難しくなる．そこで，本節では，n-gram モデルの信頼性を損なうことなく，同時に精度を向上させるために，状況に応じて動的に n の値を変化させる可変長 n-gram モデルを実現し，それを用いて単語分割を行うことを考える．

3.1 可変長 n-gram モデルの実現方法

可変長 n-gram モデルを実現するためには，その状況に応じた最適な長さの文脈を動的に決定する方法が必要となる．データ圧縮アルゴリズム PPM* [2] では，決定性文脈という考えを導入することで，状況に応じた文脈長を選択することを実現している．決定性文脈とは，その文脈から予測される文字が一意に決められる文脈のことを指す．したがって，学習データにおいて，決定性文脈の次には必ずある一文字が出現していることとなる．本稿では，PPM* モデルのスムージング (と文脈選択規則) を変更した以下の 2 つの可変長 n-gram モデルについて試みる．

3.1.1 決定性文脈を用いた可変長モデル I

決定性文脈は，一意の文字予測を行う文脈であるので，それ以上次数を大きくしても学習データ中で次に観測される文字が一種類であることに変化はない．したがって，決定性であるという情報から，それより次数を増やす必要はないと考えることとする．PPM* モデルでは，文脈が学習データ中に出現しているかどうかを調べながら，次の規則に基づき文字予測の文脈を選択する (詳細に関しては文献 [2] で説明されている)．

規則 1: もし決定性の文脈があれば，その中から最短の文脈を選択する．

規則 2: もし決定性文脈がない場合は，非決定性文脈の中から最長のものを選択する．たとえば，直前の文字が未知ならば文脈長 0 (unigram) となる．

以上の規則で選択された文脈から出現回数に基づいた確率を最尤推定する．文献 [2] では，ウィトン・ベル・スムージング付きの確率値を計算しているが，本稿では文字 n-gram モデルと条件を統一するために，バックオフ・スムージング付きの確率値を計算する．

3.1.2 決定性文脈を用いた可変長モデル II

本稿では，3.1.1 節 (PPM*モデル) の文脈選択規則を次のように変更した場合の可変長モデルについても試みることとする．

規則 1: 決定性文脈が複数ある場合は，その中から最短の決定性文脈を選択する．

規則 2: 規則 1 の条件を満たさない場合は，事前に決めておいた次数の文脈を選択する (n-gram)．

3.2 単語分割の解探索

文字 n-gram モデルを用いる場合と異なり，可変長 n-gram モデルを用いる場合は，文字位置 i での生起に影響を与える文字列の長さが動的に変化する．したがって，可変長 n-gram モデルを用いて単語分割を行う場合は，解探索における単語分割候補の指数的増加が大きな問題となる．

そこで，本稿では，ビームサーチ法により，各文字位置において確率の高い候補のみを後続する文字位置での探索に用いるようにして，確率の低い候補の枝刈りを行うこととした．このとき，文字位置 i の直前が単語境界である (1) 場合の単語分割候補と，単語境界でない (0) 場合の単語分割候補を各々別枠で比較し (別々にビーム幅を用意する)，必ず両方の可能性を残した状態で後続する文字位置 $i+1$ での探索を行うようにする．もし文字 trigram モデルによる単語分割モデルと同様に，文字位置 i の直前が単語境界である (1) か否 (0) かの 2 つの仮定に対する各々の最尤解のみに関して解探索を行うならば (図2参照)，その探索空間は，図 1 に示す探索空間と同じものとなる [3]．

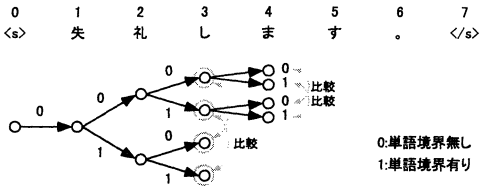

図 2: ビームサーチ法による解探索

4 評価実験

以上で提案した単語分割の方法を評価するために，ADD(ATR Dialogue Database) コーパスを用いた評価実験を行った．コーパスのサイズを表 1に示す．

表 1: 学習データと評価データのサイズ

	学習データ	評価データ
文数	10,000	2,697
単語数	141,658	31,724
文字数	251,699	58,522

4.1 単語分割実験

表 2に，可変長 n-gram モデルまたは文字 n-gram モデルによる単語分割モデルの単語分割精度を示す[1]．表中の可変長モデルの括弧内は文脈選択規則 2 でどの次数の文脈を選択するかを示している．本実験では，すべてのモデルでバックオフ・スムージング付きの確率値を計算した．また，必要となる記憶量との関係から，可変長 n-gram モデルにおける文脈同定処理には上限 (20-gram) を設けた．ここで，文字 n-gram モデル により単語分割を行う場合は，各文字位置で 2^{n-1} の分割候補を計算することで，ビタビ・アルゴリズムにより最適解を求めることができる．しかし，本実験では，高次 n-gram モデル ($n > 3$) および可変長 n-gram モデルはすべて trigram モデルを用いた場合と同じ探索空間として，高速な解探索を行った (図 2参照)．

実験結果より，文字 n-gram モデルを用いる場合，文字 4-gram モデルの精度が最も高く，それ以上次数を大きくすることはかえって精度が低下していることが分かる．また，可変長モデル I (3.1.1節) による精度は，文字 4-gram モデルによる精度にわずかながら劣る結果となった．それに対し，3.1.2節の文脈選択規則 2 で次数 3 (4-gram) または次数 4 (5-gram) を選択する可変長モデル II を用いた場合，文字 4-gram モデルでの精度を上回り，可変長モデル II (4-gram) が本実験で最も高い精度を達成した．文字 trigram モデル以外は枝刈りによる近似解であったが，良い解が得られている．また，学習データと評価データの変更による再評価も行ったが，本実験のように少ない計算量で高速な分割をする場合，やはり 3.1.2節の規則 2 において 4-gram または 5-gram を用いたモデルによる本手法が n-gram の精度を上回り最も高精度であった．

[1]再現率 recall = M/Std と適合率 precision = M/Sys により性能を評価する [4]．ここで，Std はコーパス中の単語数，Sys は本手法で分割された単語数，M は照合した単語数である．

表 2: 単語分割モデルの精度 (オープンテスト)

言語モデル	再現率	適合率
3-gram	96.78%	97.47%
4-gram	97.55%	98.09%
5-gram	97.02%	97.76%
6-gram	96.30%	97.42%
可変長 I (最長)	97.31%	97.99%
可変長 II (4-gram)	97.75%	98.27%
可変長 II (5-gram)	97.66%	98.21%

表2に示すように，文字モデルに基づく本手法はオープンテストであるにもかかわらず，全体としてかなり高精度であった．これは，パラメータ数の少ない文字モデルの頑健性を示しており，学習データに含まれていない単語に対してもある程度の対応ができていることが分かる．今回用いたような小規模の学習データからでも，信頼性の高い確率値を推定することができることは，文字モデルを用いる利点の一つである．

4.2 言語モデルのエントロピー評価

単語分割に用いた各言語モデルの性能を直接評価することを試みる．言語モデルの性能尺度であるクロス・エントロピー H は以下の式で定義される．

$$H(M,T) = -\frac{\sum_{i=1}^{n} \log p_M(s_i)}{\sum_{i=1}^{n} |s_i|} \qquad (3)$$

ここで，M は言語モデル，s_i は評価データ T 中の i 番目の文である．また，文の生起確率 $p_M(s_i)$ は，確

率の条件部に文頭記号を仮定した文字 c_1 から文末記号までの生起確率の積とする．ゆえに，$|s_i|$ は文区切りとして文末記号までを含めた文 s_i の文字数となる．

単語境界記号を含む学習データから構築された言語モデルのクロス・エントロピーを表3に示す．表中の H_1 は単語境界記号を含む (分かち書き) 評価データにおける一文字当たりのエントロピーである．したがって，H_1 は単語間のスペースまで考慮した値となる．実験結果より，言語モデルとして本稿の可変長モデルを評価した場合，文字 n-gram モデル ($n = 4,5$) よりも可変長モデル II (n-gram; $n = 4,5$) のほうが H_1 の値が小さくなっている．したがって，決定性文脈を利用して文脈長を可変とすることによって，予測力という点で優れたモデルを得ることができている．

表 3: 一文字当たりのクロス・エントロピー

言語モデル	H_1	H_2
3-gram	2.3978	9.5884
4-gram	2.1205	9.5074
5-gram	2.0585	9.5027
6-gram	2.0774	9.5074
可変長 I (最長)	2.1083	9.6351
可変長 II (4-gram)	2.0886	9.6382
可変長 II (5-gram)	2.0531	9.6359

表3の H_1 の値から分かるように，単語分割実験の結果とは異なり，可変長モデル II (5-gram) が正解分かち書き文の予測という点で最も優れたモデルであった．前述したように，表2の単語分割精度は近似解によるものであるので，解探索におけるビーム幅を大きく (探索空間を広く) すれば，可変長モデル II (5-gram) の精度はより高くなることが期待できる．ただし，その場合は解探索の計算時間が増加する．

また，ここで単語分割精度に影響を与える要因は，必ずしも H_1 のみではないことにも注意しておく必要がある．単語分割モデルでは，誤り単語分割候補も含めた文の生起確率を計算・比較する．このとき問題となるのは単語境界の予測力であるので，正解分かち書き文を高い確率で予測する (H_1) だけでなく，相対的に，誤った単語分割候補の確率を低くできているかどうかが精度の鍵となる．各言語モデルでは，分かち書きされた学習データの生成確率を最大とするようなパラメータ値を推定しているが，そこに誤った分割候補の確率を低くするという視点はない．たとえば，表3中の H_2 は，単語境界記号を含まない評価データにおける一文字当たりのエントロピーの値である．各言語モデルは分かち書きされた学習データから確率値を最尤推定しているので，ほとんどのベタ書き文は例外的な文字列 (誤り候補) となる．表2の単語分割精度を達成しているのであるから，当然 H_2 は大きな値となるが，n-gram モデルは全体的に可変長モデルよりも H_2 を小さな値としているといった現象が生じている．

極端な例での確率値は別としても，局所的に正解候補の確率よりも誤り候補の確率を高くしやすいモデルほど，正解よりも確率の高い誤った解が多く存在してしまっていることが考えられる．したがって，単語分割に用いる文字モデルとしては，正解分かち書き文を高い確率で予測でき，かつ，誤った分割候補の確率を低くおさえることのできるモデルがより有効であると思われる．以上のことから，さらに単語分割に適した言語モデルを獲得するために，誤り単語分割候補の確率を低くして，正解単語分割候補の確率を高くするような識別学習を試みることを予定している．

5 おわりに

本稿では，文字モデルに基づく単語分割モデルを提案した．ATR 対話データベースを用いた評価実験で，文字 n-gram モデルを用いた場合と，可変長 n-gram モデルを用いた場合の単語分割精度の比較を行い，決定性文脈の考えを利用して実現した可変長 n-gram モデルによる単語分割モデルがかなりの高精度であるという結果を得た．本手法は未知語を含むデータに対してかなりの精度を達成したが，さらに高速かつ頑健な手法とするために識別学習の導入を検討している．

参考文献

[1] 山本幹雄, 増山正和: “品詞・区切り情報を含む拡張文字の連鎖確率を用いた日本語形態素解析”, 言語処理学会第 3 回年次大会, pp. 421–424, 1997.

[2] Cleary, J. G. and Teahan, W. J.: “Unbounded length contexts for PPM”, *Computer Journal*, Vol. 40, No. 2, pp. 67–75, 1997.

[3] 小田裕樹, 北研二: “PPM* モデルによる日本語単語分割”, 情処研報, 98-NL-128, pp. 9–16, 1998.

[4] Nagata, M.: “A Stochastic Japanese Morphological Analyzer Using a Forward-DP Backward-A^* N-Best Search Algorithm”, COLING-94, pp. 201–207, 1994.