Vol.2012-MPS-88 No.11
2012/5/17

言語モデルの違いによる HMM を用いたテキストセグメンテーションの性能比較

但馬 康宏[†1]

HMM によるテキストセグメンテーションの問題について，HMM の状態が表す言語モデルを変化させることによる，性能の変化を示す．一般に HMM でテキストをモデリングする場合，各状態は単語ユニグラムを言語モデルとして，対応する段落を表現する．これに対して本論文では，複数の単語を取りまとめて 1 つの出力記号とする手法を複数提案し，その性能の変化を考察する．評価実験の結果，1 文を出力記号単位とし，単語がその文章に含まれるか否かを確率として持つナイーブな言語モデルが高い性能であることが明らかとなった．また，提案手法は，本論文における設定よりも利用できる情報が多い，教師あり学習の枠組みによるアルゴリズムの性能には及ばないが，従来法である単語ユニグラムモデルを利用する HMM の性能を上回ることが確認された．

Performance comparision between different language models on a text segmentation problem via HMM

Yasuhiro Tajima[†1]

We investigate the performance of some text segmentation algorihtms which uses HMM. In general, HMM applied to a text segmentation problem uses the word unigram language model to express the text segments. In this paper, we propose some multi-gram language models for the states of HMM, and evaluate them by experiments. From the evaluation, the highest performace model among our proposals is the naive probabilistic vector model for a sentence. In addition, the performances of all our proposed models are higher than that of HMM which uses the word unigram language model.

†1 岡山県立大学 情報システム工学科
Department of Systems Engineering, Okayama Prefectural University

1. はじめに

テキストデータを段落や章，話題など意味のある分割位置で区切ることをテキストセグメンテーションもしくは，段落分割と呼ぶ．この問題に関して，以下のようないくつかの手法が知られている．まずはじめに，TextTiling[1] として知られている変化点を抽出する手法である．テキストデータに対し一定の範囲のテキスト窓を切り取り，その窓内のテキストを特徴付ける特徴量を算出する．テキスト窓をテキストの先頭から末尾まで動かしてゆき，特徴量の変化が大きい位置が分割位置であるとする手法である．例えば特徴量として，あるテキスト窓内に現れる単語の種類とその出現数をベクトルにしたものを考えると，ひとつの窓と隣接する窓との間には，2 つのベクトル間のなす角を類似度とみなすことができる．窓を動かしてゆき，類似度が大きく変動する位置が，大きく話題の転換する位置だとみなせ，分割位置の候補となる．この手法では，どの程度の変動を分割位置とするかという閾値問題など設定すべきパラメータが性能に大きな影響を与える．事前の学習にあたる部分がない点が特徴である．

次に HMM を用いた分割手法である[5]．一般的には，単語を 1 つの出力記号とし，HMM の各状態が 1 つの段落や話題を表すものとする．音声認識の分野では音素の抽出などに広く使われており，時系列データの処理での性能の高さがよく知られている．事前に学習データを用いてパラメータを設定することが多く，Baum-Welch などのアルゴリズムが知られている．この手法は，いくつかの発展形があり，状態に到着した時点で出力する記号を確率変数の長さをもった記号列とし，テキストセグメンテーションに適した改良を行う研究[4] や，出力記号と前状態から現在状態を決定する HMM (MEMM) への改良[3] などがある．いずれの研究においても，HMM の各状態は段落を表し，出力記号が一単語であるので，段落に対する単語ユニグラムによる言語モデルを構築している．

本論文では，HMM の 1 つの状態が表現する言語モデルについて複数の単語の列を表現するモデルを新たに提案する．一般に，複数の単語を扱う言語モデルは n-gram が知られているが，HMM において n-gram モデルを扱おうとすると，出力記号数の指数的増大を招き，実用的でない．本論文では，この点を考慮した手法を提案する．さらに，単語ユニグラムによるモデルおよび以前の提案による手法[6] との性能比較を行う．

新たに提案する手法では，HMM における各状態は，1 つの文章を確率的に識別するものとする．この手法は，各状態が段落や話題を表す点は従来手法と同じだが，すなわち，分割対象のテキストについて，1 文ごとに各状態での受け入れ確率が求められるものとし，テキ

スト全体において最も受け入れ確率が高い状態遷移系列を求め，互いに違う状態への遷移が段落の切れ目であるとする手法である．状態遷移確率は一般の HMM と同じ扱いができ，それぞれの文に対する受け入れ確率の和が，それぞれの状態で 1 となるならば，本手法においても Baum-Welch アルゴリズムを利用することができる．

評価実験として，複数のウェブニュースが連なったテキストファイルに対してニュースの記事ごとへの分割を行った．その結果，本手法により従来手法よりも高い性能を得ることができ，特にランダムに話題が移り変わるようなテキストデータに対しては，大きな性能向上となることが確認できた．

2. HMM による段落分割と状態における言語モデル

実数の集合を R とする．離散型隠れマルコフモデル (HMM) を状態の有限集合 Q，出力記号の集合 B，状態間の遷移確率 $a : Q \times Q \to R$，各状態における出力確率 $b : Q \times B \to R$ にて定義する．任意の $i \in Q$ について，$a(i, \cdot)$ および $b(i, \cdot)$ は確率分布である．初期状態確率分布を $i \in Q$ について $a(0, i)$ と表す．

テキスト t は単語の列 $w_1 w_2 \cdots w_n$ であり，扱うすべてのテキストに出現するすべての単語の集合を W と表す．一般に HMM を用いたテキスト分割は，学習データであるテキスト集合 T を用いて単語の出力モデルである HMM を構成し，分割対象のテキストに対し最適な状態遷移系列を求め，その状態の移り変わりが話題の移り変わりであると見なし，分割位置を決定する (図 1)．

すなわち，以下のように対応付けている．

- テキストにおける段落：$q \in Q$
- テキストを構成する単語：$w \in B$
- 段落 q_1 から段落 q_2 に移り変わる確率：$a(q_1, q_2)$

すなわち，テキストに現れる話題を 1 つの状態とし，その話題を述べる場合に出現しやすい単語の分布を出力記号の分布としてモデル化する手法である．この場合，HMM の各状態は，対応する話題に対する単語ユニグラムの言語モデルを表現していると言える．

HMM の各パラメータの推定には，EM アルゴリズムである Baum-Welch アルゴリズムがよく知られている．この学習アルゴリズムは，教師なし学習アルゴリズムでありサンプルデータの集合から直接 HMM の各パラメータを推定することができる．

2.1 ナイーブな文章生成モデル

本研究では，以下の視点にもとづき HMM を用いた段落分割手法を改善する．

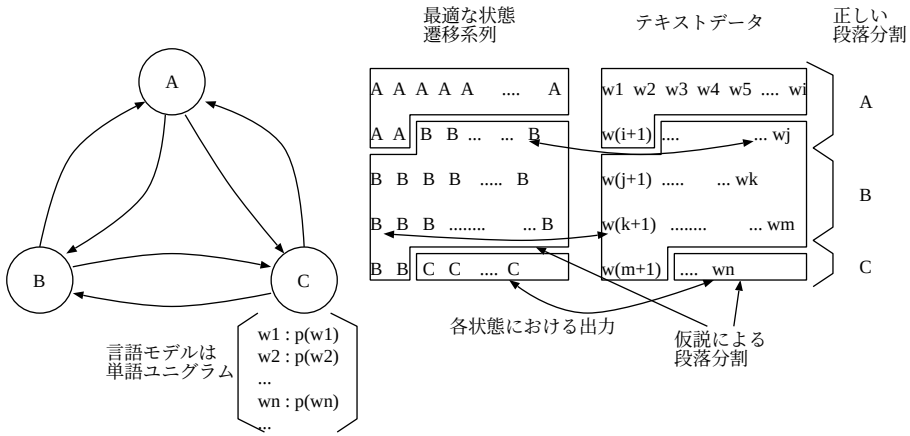

図 1 HMM による段落分割

- 段落の切れ目は必ず文の終わりであり，文の途中で区切られることはない．
- 複数の単語の組み合わせで特徴的な用語となる場合がある．

以上の点から，一文を分割できない範囲と見ることにより，分割性能の向上が期待できる．

一般に，複数の単語を含む範囲を取り扱うには n-gram を出力記号とする HMM とすることが考えられる[7]．しかし，n-gram の出現確率は，単語 1 つの出現率 p の場合に比べ p^n となるため，より多くの学習データが必要であり，学習時間も増加する．本論文では 1 つの文章に対してその出現率を求める方法を定めることにより，1 文を出力記号とする手法を提案する．

まず，1 つの文章 s は単語列 $w_1 w_2 \cdots w_n$ から成ると仮定し，確率変数 x は単語 w について $x = w$ もしくは $x = \neg w$ の 2 値をとるものとする．ある状態 q が 1 文を出力する際にその中に w が含まれている確率を $p_q(x = w)$ とする．以後，確率変数を省略し $p_q(x = w)$ を $p_q(w)$ と表す．この確率は後に述べる学習アルゴリズムの中で再推定される．

以上を用いて，ある文 $s = w_1 w_2 \cdots w_n$ に対するある状態 $q \in Q$ における出力確率 $p_q(s)$ を以下のように 2 通り提案する．

- 手法 1：文章に含まれる単語の出現確率の総積を文章の出現確率とする方法．すなわち，

$$p_q(s) = \prod_{i=1,\cdots,n} p_q(w_i)$$

である．

- 手法 2：文章に含まれない単語も考慮した方法．すなわち，

$$p_q(s) = \prod_{i=1,\cdots,n} p_q(w_i) + m(\prod_{u \neq w_i (i=1,\cdots,n)} (1 - p_q(u))$$

である．ここで，m は，第 1 項と第 2 項の重みを調整する係数であり，1 文の平均単語数 l と学習データすべての単語の異なり数 v より l/v を基準として，予備実験より定める．

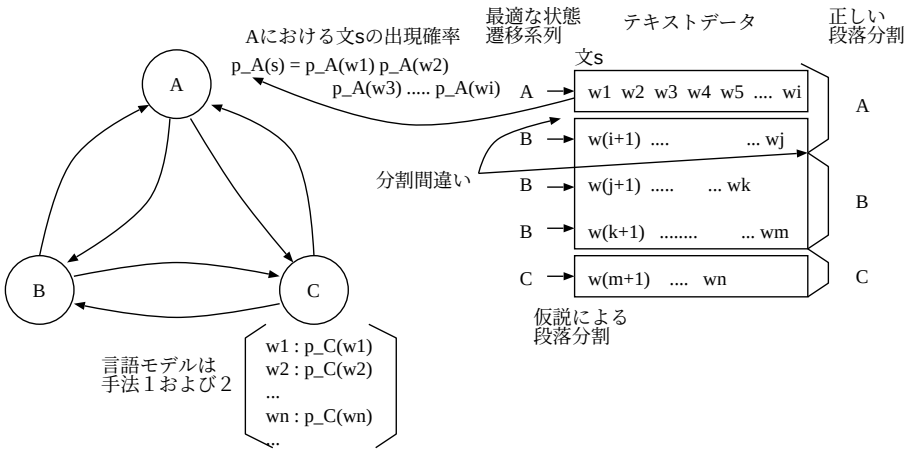

図 2 ナイーブな文章生成モデルと HMM

すなわち，HMM における状態遷移確率は従来と同じく，状態 q, r について $a(q, r)$ と表され，各状態は話題を表すが，出力記号は 1 つの文章となる．図 2 に各状態における言語モデルと HMM との関連を示す．

状態 $q \in Q$ における 1 文に対する出力確率が定まると，t 番めの文 s_t を出力するまでの前向き確率 $\alpha_t(q)$，後向き確率 $\beta_t(q)$，および $\gamma(q \in Q, r \in Q)$ を以下のように定められる．

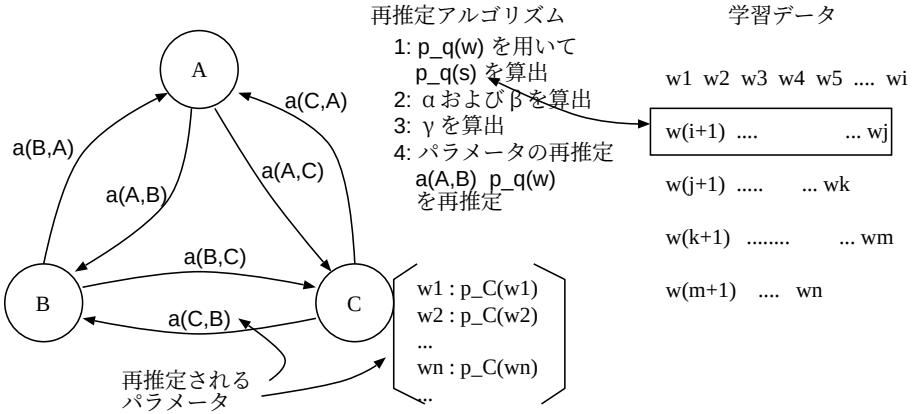

図 3 手法 1,2 における Baum-Welch アルゴリズム

$$\alpha_1(q) = \pi(q) p_q(s_1)$$
$$\alpha_t(q) = p_q(s_t) \sum_{q' \in Q} \alpha_{t-1}(q') a(q', q)$$
$$\beta_T(q) = 1$$
$$\beta_{t-1}(q) = \sum_{q' \in Q} a(q, q') p_{q'}(s_t) \beta_t(q')$$
$$\gamma_t(q, r) = \frac{\alpha_t(q) a(q, r) p_r(s_{t+1}) \beta_{t+1}(r)}{\sum_{q' \in Q} \alpha_T(q')}$$
$$\gamma_t(q) = \sum_{r \in Q} \gamma_t(q, r)$$

ここで T は学習テキストにおける文の数であり，$\pi(q)$ は状態 q に対する初期確率である．文 s_t に出現するすべての単語の集合を W_t とすると，各パラメータの再推定も Baum-Welch のアルゴリズムが適用できる．

$$\pi(q) = \gamma_1(q)$$
$$a(q, r) = \frac{\sum_{t=1}^{T} \gamma_t(q, r)}{\sum_{t=1}^{T} \gamma_t(q)}$$
$$p_q(w) = \frac{\sum_{t: w \in W_t} \gamma_t(q)}{\sum_{t=1}^{T} \gamma_t(q)}$$

特に $p_q(w)$ の再推定については，一般的な記号出力確率ではなく，文 s_t が単語 w を含む確率として，本論文における定義と矛盾なく定められる．さらに，ある文においてある状態に到達する確率を分母とし，そのときの文に単語 w が含まれている割合を再推定値としているため，すべての文を高い確率で出力するような極値に収束する可能性も低く，EM アルゴリズムとしての動作も引き継がれている．図 3 に手法 1,2 における学習アルゴリズムと Baum-Welch アルゴリズムとの対応を示す．

2.2 ポアソン分布による文章生成モデル

1 つの文章においてある単語 w が含まれるか否かの確率 p は微小な確率であり，文章の長さ n との積 np は一定であると仮定できる．すると，ポアソン分布で表すことができる．すなわち，$u = p_q(w)$ を期待値とするポアソン分布

$$PO(k) = \frac{u^k}{k!}\exp(-u)$$

を用いて，ある状態から出力される文に単語 w が k 個含まれる確率を求めることができる．以上より，ある状態 $q \in Q$ における文章出力確率の定め方を以下のように定める．

- 手法 3 : 文章に含まれる単語の出現数をポアソン分布で推定する方法．すなわち，文 s に出現するすべての単語の集合を W_s とし，s に出現する単語 w の個数を k_w すると，

$$p_q(s) = \prod_{w \in W_s} PO(k_w)$$

である．

この場合も，文に対する出力確率 $p_q(s)$ が定義できるため，前節と同じ再推定アルゴリズムが利用できる．

図 4 にポアソン分布による文章生成モデルの構成要素を示す．

2.3 ナイーブベイズ識別器を用いた手法

以前我々は，テキストに段落のラベルを付けた学習データから，その段落ラベルを出力記号とする HMM を構成し，段落分割を行う手法を提案した[6]．この手法では，学習データを 1 文ごとに分割し，文とラベルとの関係から 1 文に対してどのラベルを割り当てるべきかを決定する分類器を構成する．さらに，学習データであるテキストを 1 文ごとに 1 つのラベルが付いたラベルの記号列に変換し，そのラベルの記号列を出力する HMM を構成する．分割対象のテキストに対しては，分類器を用いて 1 文ごとにラベルを推定し，ラベルの列を作成する．次に作成したラベルの列を生成する最適な状態遷移系列を学習により構成した HMM を用いて推定し，各文がどの話題であるかを決定し，段落分割を行う．この手法

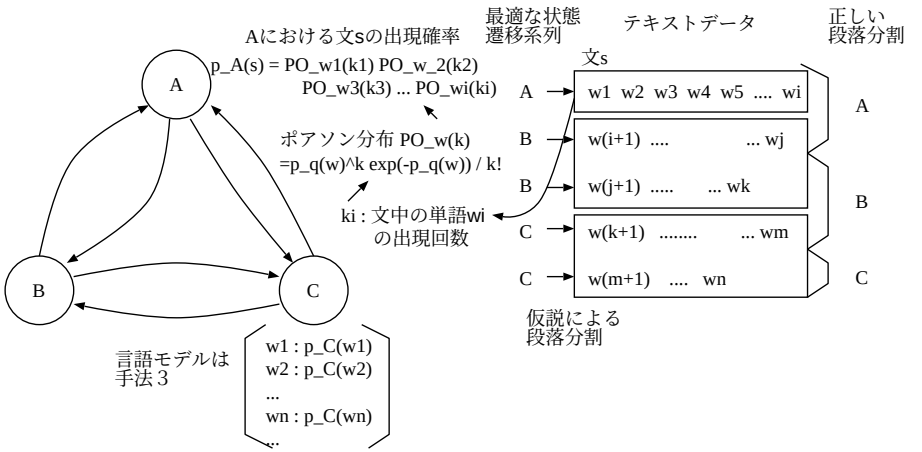

図 4 ポアソン分布による文章生成モデルと HMM

では，学習に段落のラベルが付いた正解の学習データが必要であり，本論文における手法とは別の枠組みとなるが，参考のため評価実験を行った．

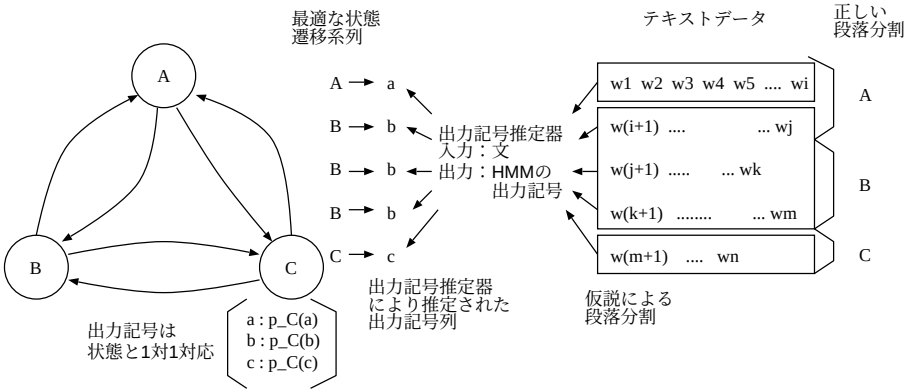

図 5 識別器を用いたモデルと HMM

図 5 に識別器を用いた手法の概要を示す．

3. 評価実験

3.1 実験データ

評価実験として，ウェブのニュース記事をつなげたものを 1 つのテキストとし，このテキストに対して段落分割を行った．学習データは，ランダムに話題が転換するデータセットとした．ニュース記事の素材の仕様は以下の通りである．

(1) ウェブニュースの記事のジャンル：5 ジャンル (社会，国内，国際，娯楽，スポーツ)
(2) 各ジャンル平均 1493 記事づつ，計 7467 記事
(3) 1 つの記事の平均長 (単語数)：301 単語
(4) 記事の最小長および最大長：最小 14 単語，最大 2501 単語
(5) 記事集合全体で使われている単語の種類：21943

この記事データから，ランダムに 10 記事を選び結合したものを 1 つの評価テキストとする．学習データとして，評価テキストを 100 テキスト準備し，さらに別の 100 テキストを評価データとしたもを 1 セットとする．4 セットの学習データをそれぞれ data1, data2, data3, data4 と呼び，4 セットの評価データをそれぞれ test1, test2, test3, test4 と呼ぶ．それぞれのデータの詳細は以下の通りである．

(学習データ)

- テキストの最大行数：232
- テキストの最小行数：51
- テキストの平均行数：116.23
- 1 行の最大単語数：240
- 1 行の最小単語数：1
- 1 行の平均単語数：30.41

(評価データ)

- テキストの最大行数：281
- テキストの最小行数：57
- テキストの平均行数：114.79
- 1 行の最大単語数：282
- 1 行の最小単語数：1
- 1 行の平均単語数：30.10

3.2 評価方法

評価テキスト data1, data2, data3, data4 に対してそれぞれの学習データを用いて HMM を構成する．その後，得られた HMM を用いて評価データ test1, test2, test3, test4 を段落分割し，分割位置の正しさを評価する．評価は，以下の値を比較した．

- テキスト中の 2 つの文に対する誤分類 (2 文評価)．これは，文献[2] における評価尺度である．
- 分割位置の一致に関する精度と再現率および F 値．
- 正しいジャンルに分類されている文章の割合 (分類率)．

3.3 2 文評価による結果

正しく段落分割がなされているテキスト (正解データ) を t_r と表し，同じテキストを分割アルゴリズムで分割したもの (仮説データ) を t_h と表す．ともに長さは，n 文であるとする．2 文評価は，t_r における i 番めと j 番めの文章 r_i, r_j と t_h における i 番めと j 番めの文章 h_i, h_j について，段落への分割が一致しているか否かを測る尺度である．すなわち，以下の値 $P_D(t_r, t_h)$ を求める．

$$P_D(t_r, t_h) = \sum_{1 \le i \le j \le n} D(i,j)(\delta_r(i,j) \bar{\oplus} \delta_h(i,j))$$

ここで，$\delta_r(i,j)$ は，t_r において，r_i と r_j が同一の段落に含まれていれば 1 そうでなければ 0 をとる関数であり，$\delta_h(i,j)$ は，t_h において，h_i と h_j が同一の段落に含まれていれば 1 そうでなければ 0 をとる関数である．また，$\bar{\oplus}$ は排他的論理和の否定である．すなわち，両辺が同一の値の場合のとき，かつそのときに限り 1 となる．関数 $D(i,j)$ は，i 番めの文と j 番めの文の位置に対する価値を与える関数である．一般には i と j が遠く離れている場合は低い値をとり，近い場合は高い値を返す．本論文では，以下の 2 種類の関数を用いた．

- 定数 k に対して，

$$D_k(i,j) = \begin{cases} 1 & |i-j| < k \\ 0 & otherwise \end{cases}$$

とし，$k = 2, 4, 6, 8, 16$ の 5 種類に付いて実験を行う．この定数は，段落の平均の長さの半分の場合，ベースラインのアルゴリズム (段落の切れ目を，ランダムにする場合，すべての文の境界とする場合，一定間隔とする場合，テキスト全体を 1 つの段落とする場合) のいずれにおいても 0.5 付近の値となる[2]．本実験では，およそ $k = 6$ であるた

め，上記の 4 種類とした．

- 定数 $u = 0.2$ として，
  $D_e(i, j) = u \exp(-u|i - j|)$
  とする．これは，i と j の差が 3 で 0.1，差が 5 で 0.07 の値となる，緩やかな定数を選んだ．

表 1 に 2 文評価の結果を示す．評価対象としたアルゴリズムは，以下の通りである．

- 単語ユニグラムを用いた HMM (従来手法)．従来法は，最適状態遷移系列を求めたときに 1 文の中で最も多く留まった状態を文全体の状態と判定し段落の切れ目を求めた．
- 手法 1(ナイーブな生成モデル) を用いた HMM.
- 手法 2(文章に出現しない単語も考慮したモデル) を用いた HMM．(重みパラメータ $m = 10^{-4}$，これは (1 文の平均単語数 12) / (データ全体の単語の異なり数 22000) の値に基づく)
- 手法 2(文章に出現しない単語も考慮したモデル) を用いた HMM．(重みパラメータ $m = 10^{-3}$)
- 手法 3(ポアソン分布モデル) を用いた HMM.
- ナイーブベイズ識別器を用いた手法 (正解データが必要なモデル)．

教師なし学習の枠組みのアルゴリズムの中では，全体的に，1. 手法 1(ナイーブ)，2. 手法 2($m = 10^{-3}$ or $m = 10^{-4}$)，3. 手法 3(ポアソン分布)，4. 単語ユニグラム (従来法) の順で性能が低下していることがわかる．特に，テストデータの 1 つの段落の平均行数が 12 行であることから，その半分の D_6 における実験では，本論文による工夫を行った分割アルゴリズムは，いずれも従来法よりも高性能であった．D_6 においては，ベースラインはおよそ 0.5 である．これは，従来法も含めたいずれの手法も大きく上回っている．しかし，教師あり学習の枠組みである識別器を用いた手法に対しては，いずれも下回る性能となった．

D_k の k 値は大きくなるほど離れた位置の文章を比較対象とするため，性能は低下する傾向にある．特に，平均段落行数の 12 を超えると，2 つ以上の分割位置をまたいだ評価となる．単語ユニグラム (従来法) は，$k = 16$ における性能が低下していないが，これは頑強さというよりも，よりランダムな分割を行っていると解釈することができる．D_e に関しても，その他の関数における結果と同様の性能順位となった．

特に手法 1 は，教師あり学習である識別器を用いた手法に近い性能が得られており，本論文における提案手法の有効性が示せたと言える．

3.4 分割位置と分類率による結果

正しく段落分割されているテキストデータ t_r において，段落の切れ目の直前の文番号の集合を B_r とする．同様に分割アルゴリズムを用いて分割したテキストデータ t_h の段落の切れ目の直前の文番号の集合を B_h とする．B_r, B_h の一致について，精度，再現率，F 値を調べる．これを完全一致の結果と呼ぶ．さらに $i \in B_r$ および $j \in B_h$ について，$|i - j| \leq 1$ ならば一致であるとみなし，同様に精度，再現率，F 値を調べる．これを前後許容による結果と呼ぶ．

さらに，t_r と t_h で同じ段落に分類されている文章の割合を分類率とする．

表 2 に分割位置一致に関する性能を，表 3 に分類率の性能を示す．比較対象とするアルゴリズムは，2 文評価の場合と同じである．

この評価尺度でも，教師なし学習の枠組みのアルゴリズムの中では全体的に，1. 手法 1(ナイーブ)，2. 手法 2($m = 10^{-3}$ or $m = 10^{-4}$)，3. 手法 3(ポアソン分布)，4. 単語ユニグラム (従来法) の順で性能が低下していることがわかる．本実験では，各テキストは 10 個の段落を含むため，分割位置はすべて 9 個である．したがって，分割位置の一致では，性能評価を計算する際の分母が 9 と少ないため，結果にばらつきが大きくなる．また，教師あり学習の枠組みを用いたアルゴリズムの性能には，提案手法のいずれも到達していないことも 2 文評価における結果と同様である．

本結果で特徴的な点は，単語ユニグラム (従来法) による結果において，精度が低く，再現率が高いことが挙げられる．これは，より多くの位置を分割位置として提示していることを示している．すなわち，細かく分割された段落が多数提示されている．この結果は，2 文評価において，k の値が $6, 8, 16$ と変化しても性能が低下していないことの裏付けとなっている．これに対し提案手法では，精度と再現率のバランスはある程度とれており，2 文評価による結果とも矛盾しない．

分類率による評価でも，性能の順位は他の評価尺度と比較して変化はないことがわかる．

4. おわりに

テキストセグメンテーションを HMM を用いて行う手法において，各状態が表す言語モデルを複数の単語が扱えるモデルとする方法を提案した．その結果，従来の単語ユニグラムを言語モデルとする手法に比べ，高性能であることが確認できた．提案手法の特徴として，複数の単語を扱い，1 文に対する出力確率を求めることができるが，n-gram による手法に比べて計算の負荷が低いことが挙げられる．実際，学習に要する計算時間，評価に要する計

表 1 2 文評価の結果

	D_2					D_4					D_6				
	test1	test2	test3	test4	平均	test1	test2	test3	test4	平均	test1	test2	test3	test4	平均
単語ユニグラム (従来法)	0.861	0.869	0.861	0.867	0.869	0.773	0.791	0.770	0.784	0.780	0.750	0.771	0.745	0.761	0.757
手法 1(ナイーブ)	0.919	0.948	0.943	0.953	0.941	0.835	0.879	0.871	0.889	0.869	0.796	0.842	0.831	0.853	0.831
手法 2($m = 10^{-4}$)	0.943	0.941	0.947	0.946	0.944	0.867	0.873	0.860	0.872	0.868	0.823	0.827	0.812	0.828	0.823
手法 2($m = 10^{-3}$)	0.956	0.958	0.956	0.958	0.957	0.882	0.886	0.883	0.888	0.885	0.824	0.827	0.826	0.833	0.828
手法 3(ポアソン分布)	0.914	0.916	0.905	0.914	0.912	0.817	0.826	0.802	0.818	0.816	0.773	0.787	0.756	0.774	0.773
識別器を用いた手法 (教師あり学習)	0.967	0.969	0.966	0.969	0.968	0.922	0.927	0.923	0.929	0.925	0.897	0.905	0.898	0.906	0.902
	D_8					D_{16}					D_e				
	test1	test2	test3	test4	平均	test1	test2	test3	test4	平均	test1	test2	test3	test4	平均
単語ユニグラム (従来法)	0.742	0.766	0.737	0.753	0.750	0.745	0.774	0.740	0.755	0.754	1.87e-2	1.91e-2	1.87e-2	1.87e-2	1.88e-2
手法 1(ナイーブ)	0.775	0.821	0.808	0.831	0.809	0.761	0.804	0.786	0.807	0.790	1.96e-2	2.05e-2	2.04e-2	2.05e-2	2.03e-2
手法 2($m = 10^{-4}$)	0.798	0.785	0.802	0.800	0.794	0.764	0.774	0.750	0.767	0.764	2.00e-2	2.00e-2	1.98e-2	1.98e-2	1.99e-2
手法 2($m = 10^{-3}$)	0.782	0.782	0.783	0.793	0.785	0.700	0.700	0.698	0.716	0.704	1.93e-2	1.92e-2	1.94e-2	1.93e-2	1.93e-2
手法 3(ポアソン分布)	0.753	0.771	0.737	0.754	0.754	0.737	0.763	0.727	0.745	0.743	1.91e-2	1.94e-2	1.89e-2	1.89e-2	1.91e-2
識別器を用いた手法 (教師あり学習)	0.882	0.893	0.884	0.892	0.888	0.862	0.881	0.867	0.874	0.871	2.18e-2	2.19e-2	2.19e-2	2.17e-2	2.18e-2

表 2 分割位置の結果

	完全一致の精度					完全一致の再現率					完全一致の F 値				
	test1	test2	test3	test4	平均	test1	test2	test3	test4	平均	test1	test2	test3	test4	平均
単語ユニグラム (従来法)	0.200	0.202	0.194	0.195	0.198	0.763	0.745	0.767	0.768	0.761	0.317	0.318	0.310	0.384	0.332
手法 1(ナイーブ)	0.258	0.372	0.351	0.390	0.264	0.506	0.502	0.477	0.482	0.492	0.342	0.428	0.404	0.431	0.401
手法 2($m = 10^{-4}$)	0.339	0.355	0.306	0.321	0.330	0.444	0.463	0.402	0.421	0.433	0.385	0.402	0.347	0.364	0.375
手法 2($m = 10^{-3}$)	0.393	0.362	0.381	0.392	0.382	0.242	0.233	0.217	0.247	0.235	0.300	0.284	0.277	0.303	0.291
手法 3(ポアソン分布)	0.253	0.245	0.229	0.230	0.239	0.564	0.573	0.535	0.559	0.558	0.349	0.343	0.321	0.326	0.335
識別器を用いた手法 (教師あり学習)	0.541	0.568	0.537	0.558	0.551	0.739	0.746	0.742	0.726	0.738	0.625	0.645	0.623	0.631	0.631
	前後許容の精度					前後許容の再現率					前後許容の F 値				
	test1	test2	test3	test4	平均	test1	test2	test3	test4	平均	test1	test2	test3	test4	平均
単語ユニグラム (従来法)	0.246	0.255	0.244	0.240	0.246	0.939	0.953	0.967	0.956	0.954	0.390	0.402	0.390	0.384	0.392
手法 1(ナイーブ)	0.371	0.496	0.492	0.535	0.473	0.732	0.670	0.691	0.660	0.688	0.493	0.570	0.570	0.591	0.556
手法 2($m = 10^{-4}$)	0.498	0.489	0.460	0.470	0.479	0.647	0.628	0.611	0.615	0.625	0.563	0.550	0.525	0.533	0.543
手法 2($m = 10^{-3}$)	0.541	0.490	0.554	0.528	0.528	0.345	0.312	0.323	0.345	0.331	0.421	0.381	0.408	0.418	0.407
手法 3(ポアソン分布)	0.342	0.330	0.314	0.313	0.325	0.761	0.761	0.752	0.748	0.756	0.472	0.460	0.443	0.441	0.454
識別器を用いた手法 (教師あり学習)	0.620	0.647	0.619	0.651	0.634	0.848	0.855	0.863	0.857	0.856	0.716	0.737	0.721	0.740	0.729

表 3 分類率の結果

	分類率				
	test1	test2	test3	test4	平均
単語ユニグラム (従来法)	0.713	0.750	0.723	0.733	0.730
手法 1(ナイーブ)	0.725	0.772	0.749	0.775	0.755
手法 2($m = 10^{-4}$)	0.698	0.716	0.679	0.724	0.704
手法 2($m = 10^{-3}$)	0.523	0.548	0.516	0.569	0.539
手法 3(ポアソン分布)	0.691	0.730	0.691	0.718	0.708
識別器を用いた手法 (教師あり学習)	0.804	0.847	0.828	0.835	0.829

算時間ともに単語ユニグラムを用いた従来手法にくらべ大差はなく，いずれも数時間から十数時間程度である．

提案手法の分割性能は，いずれも複数の評価尺度において従来手法を上回り，有効性が示せたと言える．しかし，教師あり学習の枠組みで処理を行うアルゴリズムの性能には及ばなかった．単語ユニグラムによる従来法では，分割数が多くなり，小さな段落が多数作られる傾向があったが，提案手法では解決されていることが，評価実験から明らかとなった．

今後の課題として，教師あり学習の手法に本提案手法を取り込むことが考えられる．教師あり学習の枠組みでは，時系列データに対する処理は，CRF などの生成モデルが様々な分野で高い性能を示しており，本手法による複数の単語をまとめて取り扱う方法を応用できれば，より高性能な分割器を作ることができると思われる．

参 考 文 献

1) Hearst, M. A.: Texttiling: segmentaing text into multi-paragraph subtopic passages, Computaional Linguistics, Vol. 23, pp.33-64 (1997)
2) Beeferman, D., Berger, A. and Lafferty, J.: Statistical models for text segmentation, Machine Learning, Vol. 34, Nos.1-3, pp.177-210 (1999)
3) McCallum, A., Freitag, D., Pereira, F.: Maximum entropy markov models for information extraction and segmentation, Proc. of ICML'00, pp.591–598 (2000)
4) Ostendorf, M., Digalakis, V. V. and Kimball, O. A.: From HMM's to segment models: a unified view of stochastic modeling for speech recognition, IEEE Transactions on speech and audio processing, Vol. 4, No.5, pp.360–378 (1996)
5) Yamron, J.P., Carp, I., Gillick, L., Lowe, S., van Mulbregt, P.: A hidden markov model approach to text segmentation and event tracking, Proc. of IEEE conf. on Acoustics, Speech and Signal Processing, vol.1, pp.333-336 (1998)
6) 但馬康宏，北出大蔵，中林智，藤本浩司，小谷善行：HMM とテキスト分類器による対話の段落分割，情報処理学会論文誌　数理モデル化と応用，vol.2, no.2, pp.70–79 (2009)
7) 長野雄，鈴木基之，牧野正三：HMM を用いた複数 n-gram モデルによる言語モデルの構築，情報処理学会研究報告 SLP 40-26, pp.151–156 (2002)