日本語の情報量の上限の推定

森 信介† 山地 治†

本論文では，形態素単位の n-gram モデル（$1 \leq n \leq 16$）による日本語の情報量の上限の推定方法とその結果を示す．各 n-gram モデルは，データスパースネスの問題に対応するため，低次の n-gram モデルとの補間を行ってある．補間係数は，最も有効であると考えられている削除補間法により求める．実験では EDR コーパスの約 9 割からモデルのパラメータを推定し，残りの 1 割に対して情報量を計算した．その結果，$n = 16$ のときに 1 文字あたりの情報量は最小の 4.30330 ビットであった．また，学習コーパスの大きさとモデルの次数による情報量の変化を調べた結果，モデルの次数を上げることによる情報量の減少量は微小であるが，学習コーパスを大きくすることによる情報量の減少量はかなりあるということが分かった．さらに，パラメータ数とエントロピーの関係についても議論する．これは，実際の日本語処理に n-gram モデルを応用する際に，適切に n の値を選ぶ指標となる．

An Estimate of an Upper Bound for the Entropy of Japanese

SHINSUKE MORI† and OSAMU YAMAJI†

In this paper we present an estimate of an upper bound for the entropy of Japanese by morpheme n-gram model ($1 \leq n \leq 16$). Each n-gram model is interpolated with lower order n-gram models. The deleted interpolation method is applied for estimating interpolation coefficients. We estimated the parameters from 90% of the EDR corpus and calculated the entropy on the rest 10%. As the result, the minimum entropy was 4.30330 [bit] a character with $n = 16$. The relation between the size of learning corpus or the order of model and entropy showed that incresing the order decreases entropy slightly and increasing the size of learning corpus decreases it noteworthily. In addition, we discuss the relation between the number of parameters and entropy. This is usefull to select the value of n to apply n-gram model to the practical Japanese processing.

1. はじめに

自然言語を計算機に認識させる方法として，言語を確率的な現象としてとらえる方法が提案されている．この方法では，入力（音素列や画像）に対応する文字列の中で，出現確率が最も高い文字列を出力する．この出現確率は，一般に，入力と文字の対応を表す確率と，対象としている言語における文字列の出現確率の積に分解できる．このうちの後者の確率を推定するモジュールを確率的言語モデルと呼ぶ．このモジュールは応用における入力とは無関係であるので，独立に研究することができる．この性質は確率を用いる方法の利点の 1 つである．また，良い確率的言語モデルは，自然言語の文字列の出現確率を高い精度で推定できるので，自然言語のテキストに対する高性能な圧縮器を構成することを可能にする．

† 京都大学工学研究科電子通信工学専攻
Department of Electronics and Communication, Kyoto University

確率的言語モデルの良否は，文字予測における曖昧性を表す情報量（エントロピー）で測られる．英語に対する情報量の推定は，情報理論の草創期にすでに行われているが，当時は大規模な機械可読のコーパスもなく，大量のデータを高速に処理する計算機も存在しなかったため，十分な実験を行うことができなかった[1]．近年，計算機技術の発達と大規模な機械可読の言語データの出現により，信頼できる推定が行われている．Cover と King[2]は広範な文献紹介を行っている．また，Brown ら[3]は約 6 億文字のコーパスから推定されたトークン単位の tri-gram モデルで約 600 万文字のコーパスの情報量を計算した結果として，文字あたり 1.75 ビットの上限を報告している．

このように，英語を対象とした研究は多々あるが，日本語の情報量の推定に関しては，あまり報告されていない．浅井[4]は，シャノンと同様の人間による予測の結果を統計的に処理することで，日本語のエントロピーの上限を推定している．これは，今日的視点からは，様々な点で不十分である．また，音声認識などの言

語モデルの評価として，単語単位のパープレキシティ（エントロピーに一意に換算できる）を報告している文献もあるが，未知語の扱いなどが厳密でないので，言語モデルの比較という目的を超えた絶対的な値としての意味はない．さらに，これらの文献では経験的に有効とされている単語 tri-gram を用いているが，より長い先行文脈を用いることで，エントロピーがどのように変化するかが明確ではない．

本論文では，形態素（単語と品詞の直積）単位の n-gram モデルによる日本語の文字あたりの情報量を計算する方法を説明し，EDR コーパス[5]を用いて行った実験の結果を報告する．単語間に空白を置かない日本語の場合は，単語の定義が明らかではないが，コーパスで与えられている単語の定義をそのまま用いた．コーパスを 9 対 1 に分割し，パラメータ推定とエントロピーの計算を行った．その結果，$n = 16$ のときに 1 文字あたりの情報量は最小の 4.30330 ビットであった．また，学習コーパスの大きさとモデルの次数による情報量の変化を調べた結果，モデルの次数を上げることによる減少量は微小であるが，学習コーパスを大きくすることによる減少量はかなりあるということが分かった．さらに，パラメータ数とエントロピーの関係についても議論する．これは，実際の日本語処理に n-gram モデルを応用する際に，適切に n の値を選ぶ指標となる．

2. 方　　法

我々が用いた，自然言語の情報量の上限の推定の方法は，ある情報源が与えられたとき，その出力を予測するモデルを作成し，そのモデルによる情報源のクロスエントロピーがその情報源の情報量の上限を与えるという事実に基づいている．この章では，関係する概念を簡単に述べる．

2.1 情報量，クロスエントロピー

ある有限のアルファベット $\mathcal{X}$ の定常確率過程を $X = \{\ldots, X_{-2}, X_{-1}, X_0, X_1, X_2, \ldots\}$ とし，P を X の確率分布とすると，X の情報量は次の式で定義される．

$$H(P) = -E_P \log P(X_0|X_{-1}, X_{-2}, \ldots)$$

ここで，E_P は P による期待値を表す．一般に対数の底は 2 であり，このとき情報量の単位はビットである．$H(P)$ はまた，次の式でも表せる．

$$\begin{aligned} H(P) &= \lim_{n\to\infty} -E_P \log P(X_0|X_{-1}, X_{-2}, \ldots, X_{-n}) \\ &= \lim_{n\to\infty} -\frac{1}{n} E_P \log P(X_1, X_2, \ldots, X_n) \end{aligned}$$

もし，この確率過程がエルゴード的であれば，Shannon-McMillan-Breiman の定理[6]により次の式が成り立つ．

$$H(P) = \lim_{n\to\infty} -\frac{1}{n} \log P(X_1 X_2 \cdots X_n)$$

よって，エルゴード的な確率過程の情報量は確率分布 P に従って無作為に抽出された十分長いアルファベット列のサンプルに対する知識から得ることができる．もし P が知られていないとしても，$H(P)$ の上限は P の近似から求めることができる．P をモデル化する定常確率情報源を M とすると，P の M によるクロスエントロピーは以下の式で定義される．

$$H(P, M) = -E_P \log M(X_0|X_{-1}, X_{-2}, \ldots)$$

情報量の場合と同様に，クロスエントロピーは次の式でも表せる．

$$\begin{aligned} &H(P, M) \\ &\quad = \lim_{n\to\infty} -E_P \log M(X_0|X_{-1}, X_{-2}, \ldots, X_{-n}) \\ &\quad = \lim_{n\to\infty} -\frac{1}{n} E_P \log M(X_1, X_2, \ldots, X_n) \end{aligned}$$

よって，$\mathrm{D}(P||M)$ を P と M の Kullback-Leibler 距離[7]として，以下の式が成り立つ．

$$\begin{aligned} &H(P, M) - H(P) \\ &\quad = \lim_{n\to\infty} \frac{1}{n} E_P \log \frac{P(X_1, X_2, \ldots, X_n)}{M(X_1, X_2, \ldots, X_n)} \\ &\quad = \lim_{n\to\infty} \frac{1}{n} \mathrm{D}(P(X_1, X_2, \ldots, X_n)|| \\ &\qquad\qquad M(X_1, X_2, \ldots, X_n)) \end{aligned}$$

Kullback-Leibler 距離はつねに非負であるので，この式から

$$H(P) \le H(P, M)$$

である．よって，確率分布 P に従って無作為に抽出された十分長いアルファベット列のサンプルに対して計算されたモデル M によるクロスエントロピーは，情報量 $H(P)$ の上限の推定値となる．また，クロスエントロピーが小さければ，Kullback-Leibler 距離を尺度としてモデル M が P により近いことを意味する．したがって，クロスエントロピーの大小はモデルの良否を測る尺度として用いることができる．

以上に述べたことから，日本語を確率過程と見なしてモデルを構成し，クロスエントロピーを計算することで，日本語の情報量の上限を推定することができる．また，クロスエントロピーの大小によってモデルの良否を測ることができる．

2.2 テキスト圧縮

情報量とクロスエントロピーは，テキスト圧縮という視点からも重要な値である．情報量は，この情報源からのアルファベット列を一意に復号できる符号で記

述するために必要な平均ビット数の期待値の下限を与える．これを式で表すと以下のようになる．

$$\lim_{n\to\infty}\frac{1}{n}E_P l(X_1,X_2,\ldots,X_n)\geq H(P)$$

ここで，$l(X_1,X_2,\ldots,X_n)$ はアルファベット列 X_1, $X_2,\ldots,X_n$ に対応する符号のビット数を表す．同様に，情報源 P をモデル M でモデル化した場合，クロスエントロピーは，情報源 P からのアルファベット列をモデル M を用いて一意に復号できる符号で記述するために必要な平均ビット数の期待値の下限を与える．これを式で表すと以下のようになる．

$$\lim_{n\to\infty}\frac{1}{n}E_P l_M(X_1,X_2,\ldots,X_n)\geq H(P,M)$$

等号が成り立たたないのは，実際に符号化する際に整数個の符号を用いなければならないという制約による．しかし，算術符号[7),8)]を用いれば上式の等号が成り立つことが示される．

3. 言語モデル

この章では，我々が実験に用いた言語モデルについて述べる．これは，基本的には音声認識などの応用で一般的に用いられている，形態素を単位とした n-gram モデルである．これは，簡単にいうと，形態素列のコーパスにおける頻度に基づき，ある形態素列に後続する形態素の出現確率を計算するというモデルである．形態素単位の n-gram モデルは，あらかじめ与えられた形態素集合（既知語）に対応する状態に加えて，これら以外の形態素（未知語）すべてに対応する状態を持つ．未知語は，この状態から出現すると考える．このときの確率は，文字 n-gram モデルによって実現される未知語モデルにより与えられる．

3.1 n-gram モデル

過去に観測された記号列に基づいて次の記号を予測するためのモデルの1つとして，マルコフモデルがある．これは，過去に観測された記号列を直前の記号列で分類し，次の記号を予測するというモデルである．直前の記号列の長さが k のとき，k 重マルコフモデルと呼ぶ．マルコフモデルによる記号列 $s_1s_2\cdots s_n$ の出現確率は，以下の式で与えられる．ただし，状態は k 個の記号の直積と1対1に対応しているので，表記においてはこれらを区別していない．

$$P(s_1s_2\cdots s_l)=\prod_{i=1}^{l+1}P(s_i|s_{i-k}\cdots s_{i-2}s_{i-1})$$

ここで $s_i\,(i\leq 0)$ と s_{l+1} は，文頭と文末に対応する特別な記号である．これらを導入することによって，すべての可能な記号列に対する確率の和が1となることが保証される．

言語モデルとして用いる場合，状態遷移確率 $P(s_i|s_{i-k}\cdots s_{i-2}s_{i-1})$ は同一（類似）の情報源からの記号列（コーパス）を用いて推定する．記号から状態への写像が単射である場合には，コーパスにおける状態列の頻度を計数した結果から最尤推定することができる．これは，コーパスにおける頻度を N とすると，以下の式で表される．

$$P(s_i|s_{i-k}\cdots s_{i-2}s_{i-1})=\frac{N(s_{i-k}\cdots s_{i-2}s_{i-1}s_i)}{\sum_s N(s_{i-k}\cdots s_{i-2}s_{i-1}s)}$$

このように，このモデルはコーパスにおける $n=k+1$ 個の記号列の頻度統計の結果に基づくので n-gram モデルとも呼ばれる．

対象とする事象の頻度が低い場合には，推定値の信頼性が低くなるという問題がある．この問題に対処する方法として，補間と呼ばれる方法が用いられる[9)]．より信頼性が高いことが期待される，より低次の n-gram モデルの遷移確率を一定の割合で足し合わせるという操作を施すことをいう．これは，次の式で表される．

$$P'(s_i|s_{i-k}\cdots s_{i-2}s_{i-1})=\sum_{j=0}^{k}\lambda_j P(s_i|s_{i-j}\cdots s_{i-2}s_{i-1})$$

$$\text{ただし } 0\leq\lambda_j\leq 1,\ \sum_{j=0}^{k}\lambda_j=1$$

ここで，$j=0$ のときは $P(s_i|s_{i-j}\cdots s_{i-2}s_{i-1})=P(s_i)$ であるとする．これは，過去に観測された記号列によらない確率分布であり，状態遷移確率と同様に，以下の式を用いてコーパスから最尤推定する．

$$P(s_i)=\frac{N(s_i)}{\sum_s N(s)}$$

補間係数 $(\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_j)$ の値は状態頻度の計数に用いたコーパスとは別のコーパス（Held-Out Data）の出現確率が最大になるように決定する[10)]．

$$(\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_k)=\mathop{\mathbf{argmax}}_{(\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_k)}\prod_{i=1}^{h}P'(\boldsymbol{s}_i)$$

ここで，$\boldsymbol{s}_i$ は補間係数推定用コーパスの i 番目の記号列であり，h は補間係数推定用コーパスに含まれる文の数である．この補間係数は，状態の関数とすることも可能である．次の章で述べる実験では，先行事象の学習コーパスにおける頻度が0の場合と1以上場合で補間係数を以下のように区別した．

$$P'(s_i|s_{i-k}\cdots s_{i-2}s_{i-1}) = \sum_{j=0}^{h} \lambda_j^h P(s_i|s_{i-j}\cdots s_{i-2}s_{i-1})$$

ただし，h はそれぞれの先行事象について頻度が1以上となる最長の先行記号数である．

$N(s_{i-h}\cdots s_{i-2}s_{i-1}) > 0$ かつ

$N(s_{i-h-1}\cdots s_{i-2}s_{i-1}) = 0$

以上のようにすることで，式(1)の値が不定となる場合を参照することを避けられる．このとき，n-gram モデルの補間係数の数は $1+2+\cdots+(n-1)$ となる．

補間係数を求めるための最も優れた方法として，削除補間法と呼ばれる方法がある．削除補間法では，まず学習コーパス L を m 個の互いに素な部分集合 $L_1, L_2, \ldots, L_m$ に分割する．そうしておいて，状態頻度の計数を L_k を除いた学習コーパスに対して行い，L_k を用いて補間係数を推定するということを k を変えながら m 通り行い，それぞれの補間係数の平均値を最終的な補間係数とする．

3.2 形態素 n-gram モデル

前節で説明した n-gram モデルの記号を形態素と考えることで，自然言語の文を形態素の連接と見なすモデルが構成できる．これを形態素 n-gram モデルと呼ぶ．この場合に問題となるのは，記号に対応する形態素（既知形態素）の選択である．ただし，どのような形態素の集合を選択したとしても，テストコーパスに出現する可能性のあるすべての形態素が，学習コーパスに出現することは望めない．このため，未知形態素の扱いが避けられない問題となる．この問題に対処するため，未知形態素に対応する特別な記号を用意し，既知の形態素以外はこの記号から次節で述べる未知語モデルにより与えられる確率で生成されることとする．未知形態素に対応する特別な記号は，かならずしも唯一である必要はなく，品詞などの情報を用いて区別される複数の記号であってもよい．以下の説明では，各品詞に対して未知形態素に対応する記号を設ける．たとえば，以下のような形態素列（文）が与えられたとする．

漱石/名詞　です/助動詞　．/記号

「漱石/名詞」が未知形態素とすると，この文の n-gram モデル（$n=3$）による生成確率は，以下の式で与えられる．ただし，UM は未知形態素を表し，BT は文末と文頭に対応する特別な記号である．

$$\begin{aligned} &P(\text{漱石/名詞, です/助動詞, ./記号}) \\ &\quad = P(\texttt{UM/名詞} \mid \texttt{BT, BT}) P(\text{漱石} \mid \texttt{UM/名詞}) \\ &\qquad \times P(\text{です/助動詞} \mid \texttt{BT, UM/名詞}) \\ &\qquad \times P(\text{./記号} \mid \texttt{UM/名詞, です/助動詞}) \\ &\qquad \times P(\texttt{BT} \mid \text{です/助動詞, ./記号}) \end{aligned}$$

このように，未知の形態素は，形態素 n-gram モデルでその品詞の未知形態素に対応する記号を生成してから，未知語モデルで個々の表記（文字列）をある確率（この例では $P(\text{漱石} \mid \texttt{UM/名詞})$）で生成する．こうすることで，すべての記号列の生成確率の和が1となることが保証される．ただし，未知語モデルが既知の形態素も生成する場合は，既知の形態素を含む文は複数の導出を持つので，これらの導出にわたる和も計算しなければならない．

以上に述べた形態素 n-gram モデル M_m による，形態素列 $m_1 m_2 \cdots m_h$ の出現確率は以下の式で表される．ただし $\mathcal{M}_k$ は既知形態素の集合を表し，pos は m_i の品詞を表す．また $m_i = \texttt{BT}$ $(i \leq 0 \vee i = h+1)$ である．

$$M_m(m_1 m_2 \cdots m_h) = \prod_{i=1}^{l+1} P_m(m_i|m_{i-k}\cdots m_{i-2}m_{i-1})$$

$$P_m(m_i|m_{i-k}\cdots m_{i-2}m_{i-1}) = \begin{cases} P(m_i|m_{i-k}\cdots m_{i-2}m_{i-1}) & \text{if} \quad m_i \in \mathcal{M}_k \\ P(\texttt{UM}_{pos}|m_{i-k}\cdots m_{i-2}m_{i-1}) M_{x,pos}(m_i) & \text{if} \quad m_i \notin \mathcal{M}_k \end{cases}$$

この式の中の $M_{x,pos}$ は，次節で述べる未知語モデルであり，品詞が pos であることを条件として，引数で与えられる文字列の生成確率を値とする．

前節で述べたように n-gram モデルの確率値は，コーパスの頻度から最尤推定するのが一般的である．形態素 n-gram モデルの場合もほぼ同じである．唯一の違いは，アルファベットの定義が明確でないことであり，これを何らかの方法でパラメータ推定の前に決定しなければならない．これには，何らかの辞書の見出し語を用いることや，学習コーパスに高頻度で出現する形態素とすることなどが考えられる．既知形態素集合を定義した後は，これに未知形態素に対応する特別な記号を加えてアルファベットとし，学習コーパスの未知形態素をこれらの記号に置き換えて頻度を計数することで形態素 n-gram モデルの確率値を推定する．補間係数の推定なども同様に行うことができる．

3.3 未知語モデル

未知語モデルは，表記から確率値への写像として定

義され，既知形態素以外のあらゆる形態素の表記を0より大きい確率で生成し，この確率をすべての表記にわたって合計すると1以下になる必要がある．このような条件を満たすモデルの1つとして，文字単位の n-gram モデルがある．日本語の表記に用いられる文字は有限と考えられるので，文字単位の n-gram モデルは，すでに説明した n-gram モデルの記号を言語の文字と見なすことで容易に定義できる．形態素 n-gram モデルの場合と同様に，文字集合をコーパスに現れる既知文字と現れない未知文字に分類し，未知文字はこれを表す特別な記号から生成されるものとすることもできる．文字の使用頻度には大きな偏りがあることが予測されるので，これらを1つのグループと見なすことで，モデルが改善されると考えられる[11]．文字は有限であるから，未知語モデルの場合と異なり，各未知文字の生成確率は等確率とすることができる．たとえば，未知形態素の表記として「漱石」が与えられ，このうち「漱」のみが未知文字とすると，この未知形態素の文字 n-gram モデル（$n=3$）による生成確率は以下の式で与えられる．ただし，UX は未知文字を表し，BT は形態素の区切りに対応する特別な記号である．

$$P(\text{漱石}) = P(\texttt{UX} \mid \texttt{BT}, \texttt{BT})P(\text{漱} \mid \texttt{UX}) \times P(\text{石} \mid \texttt{BT}, \texttt{UX}) \times P(\texttt{BT} \mid \texttt{UX}, \text{石})$$

このように，未知の文字は，文字 n-gram モデルで未知文字に対応する記号を生成してから，一定の確率で生成される．このときの確率は，未知文字の集合を $\mathcal{X}_u$ とすると以下の式で与えられる．

$$P(x \mid \texttt{UX}) = \frac{1}{|\mathcal{X}_u|} \quad \text{ただし } x \in \mathcal{X}_u$$

このような未知語モデル M_x は，未知形態素だけでなく既知形態素の表記も0より大きい確率で生成する可能性がある．この場合には，以下の式が示すように，未知形態素の生成確率の合計は1未満となる．

$$\sum_{m\in\mathcal{M}_u} M_x(m) + \sum_{m\in\mathcal{M}_k} M_x(m) = \sum_{m\in\mathcal{X}^*} M_x(m) = 1$$
$$\Leftrightarrow \sum_{m\in\mathcal{M}_u} M_x(m) = 1 - \sum_{m\in\mathcal{M}_k} M_x(m) < 1$$
$$(\because M_x(m) > 0,\ \exists m \in \mathcal{M}_k)$$

これは，言語モデルとしての条件を満たしてはいるが，クロスエントロピーという点で改善の余地がある．つまり，既知形態素の生成確率を何らかの方法で未知形態素に分配することで，未知形態素の生成確率が大きくなり，テストコーパスにそのような未知形態素が出現した場合に，テストコーパスの出現確率が大きくなる．既知形態素の生成確率の分配には，様々な方法が考えられるが，以下の式が表すように，すべての未知形態素にその生成確率に比例して分配する方法が一般的であろう．

$$M'_x(m) = \frac{M_x(m)}{\sum_{m\in\mathcal{M}_u} M_x(m)} = \frac{M_x(m)}{1 - \sum_{m\in\mathcal{M}_k} M_x(m)} \quad (m \in \mathcal{M}_u) \tag{1}$$

本論文では，辞書の見出し語などとして与えられる未知形態素の部分集合に等しく配分することを提案する．つまり，ある形態素の集合が与えられたとして，ここから既知形態素を除いた集合を $\mathcal{M}_d$ として，この要素の生成確率を文字 n-gram モデルによる確率と既知語の生成確率の合計を $\mathcal{M}_d$ の要素数で割った値の和とする．

$$M'_x(m) = M_x(m) + \frac{1}{|\mathcal{M}_d|} \sum_{m\in\mathcal{M}_k} M_x(m) \quad (m \in \mathcal{M}_d) \tag{2}$$

これは，既知形態素の生成確率を，学習コーパスには現れないが辞書などから形態素であると考えられる文字列に優先的に分配し，それらの生成確率を相対的に高くすることを意味する．このような文字列の集合を外部辞書と呼ぶ．品詞ごとに未知語モデルを持つ場合には，外部辞書には文字列のほかにその品詞が記述されている必要がある．

以上に述べた未知語モデル M'_x による，文字列 $m = x_1x_2\cdots x_l$ の出現確率は以下の式で表される．ただし $\mathcal{X}_k$ は既知文字を表す．また $x_i = \texttt{BT}\ (i \le 0 \vee i = l+1)$ である．

$$M'_x(x_1x_2\cdots x_l) = \begin{cases} 0 & \text{if} \quad m \in \mathcal{M}_k \\ M_x(x_1x_2\cdots x_l) & \text{if} \quad m \in \mathcal{M}_u \wedge m \notin \mathcal{M}_d \\ M_x(x_1x_2\cdots x_l) + \frac{1}{|\mathcal{M}_d|}\sum_{m\in\mathcal{M}_k} M_x(m) & \text{if} \quad m \in \mathcal{M}_u \wedge m \in \mathcal{M}_d \end{cases}$$

$$M_x(x_1x_2\cdots x_l) \prod_{i=1}^{l+1} P_x(x_i|x_{i-k}\cdots x_{i-2}x_{i-1})$$

$$P_x(x_i|x_{i-k}\cdots x_{i-2}x_{i-1}) = \begin{cases} P(x_i|x_{i-k}\cdots x_{i-2}x_{i-1}) & \text{if} \quad x_i \in \mathcal{X}_k \\ P(\texttt{UX}|x_{i-k}\cdots x_{i-2}x_{i-1})\frac{1}{|\mathcal{X}_u|} & \text{if} \quad x_i \notin \mathcal{X}_k \end{cases}$$

以上で説明した文字 n-gram モデルは，未知文字を等確率で生成するモジュールを「未知文字モデル」と考えると，形態素 n-gram モデルと相似の構造である．

文字 n-gram モデルの確率値は，形態素 n-gram モデルの場合と同様に，アルファベットを定義してから，未知形態素の実例における文字列の頻度から推定される．未知形態素の実例の収集の方法として，我々は，削除補間法を応用した以下の方法を提案する．

学習コーパスを k 個の部分コーパスに分割し，
i 番目の部分コーパスの未知形態素の実例を，
i 番目の部分コーパス以外を学習コーパスとし
i 番目の部分コーパスをテストコーパスと見た
場合の未知形態素とする．

対案としては，学習コーパスに含まれるすべての形態素とすることや，学習コーパスにおける頻度が1である形態素とする[12]などが考えられるが，我々が提案する方法は，削除補間法を応用して，実際のテストコーパスにおける未知形態素と類似した実例を得ているので，他の方法よりも優れていると予測される．

アルファベットの定義には，何らかの辞書の見出し語の文字を用いることや，学習コーパスまたは未知語の実例に高頻度で出現する文字とすることなどが考えられる．既知文字集合を定義した後は，これに未知文字に対応する特別な記号を加えてアルファベットとし，未知語の実例の未知文字をこれらの記号に置き換えて頻度を計数することで文字 n-gram モデルの確率値を推定する．補間係数の推定なども同様に行うことができる．

4. 実験結果

我々は，前章で説明した言語モデルを用いて，日本語の情報量の上限を推定した．この章では，この結果を提示し，それに対する考察を述べる．

4.1 実験の条件

実験には EDR コーパス[5]を用いた．まずこれを 10 個に分割し，このうち 9 個を学習コーパスとし，1 個をテストコーパスとした．これはすべての実験を通して不変である．**表 1** はコーパスの大きさである．なおアルファベットの数は 6,879 とした．これは，我々の計算機環境で表示可能であった全角文字に文区切り記号を合わせた数である．

表 1 コーパス
Table 1 Corpus.

用途	文数	形態素数	文字数
学習	187,022	4,595,786	7,252,558
評価	20,780	509,261	802,576

補間係数の推定には削除補間法を用いた．すなわち，9 個の学習コーパスのうちの 8 個で状態列の頻度を計数し，残りの 1 個の出現確率が最大になる補間係数の推定を 9 通り行い，その平均値を補間係数とした．この補間係数とすべての学習コーパスに対して計数した状態列の頻度をパラメータとする n-gram モデルを構成し，クロスエントロピーの計算をテストコーパスに対して行った．テストコーパスの単語区切りは，コーパスにあらかじめ付加されたものを用いた．したがって，テストコーパスに含まれる文字列の出現確率は，その文字列のすべての生成方法による確率を合計した値ではなく，コーパスに示された生成方法のみによる値である．なお，単語区切りが明示されていない文に対しては，動的計画法を用いたアルゴリズムにより，出現確率が最大となる状態遷移列（単語区切り）を，文に含まれる文字数に比例した時間で求めることができる[13]．

4.2 未知語モデルの評価実験

前章で説明した未知語モデルを実装し，この部分でのテストコーパスの生成確率の対数値を計算した．文字 n-gram モデルの n を 2 とし，既知文字は 9 個に分割された学習コーパスの 2 個以上に現れる文字とした．以下の 2 点に関しては，前章で述べた他の方法との比較実験を行った．

(1) パラメータ推定のための未知語の実例の収集方法
(2) 既知形態素の生成確率の分配方法

以下では，それぞれの結果を提示し検討を加える．

4.2.1 未知語の実例の収集方法

実験を行った未知語の実例の収集方法は以下のとおりである．

方法 1 学習コーパスにおける頻度が 1 である形態素
方法 2 学習コーパスに含まれるすべての形態素
方法 3 分割された学習コーパスの 1 個にのみ出現する形態素

方法 1 は，永田[12]が未知語の収集に用いた方法である．方法 2 の長所は，非常に多くの実例が得られることである．方法 3 は，我々が提案する方法である．この方法の長所は，実際の未知語の性質を最もよく反映した実例が得られることである．

これらの方法を実装し，学習に用いた実例（方法により異なる）の文字あたりの情報量と，テストコーパスの未知語（方法によらず同じ）の情報量を計算した．**表 2** は，この結果である．なお，ここでの実験に用いた未知語モデルは，品詞を区別していない．この結果から，この実験では方法 3 が最も良い未知語モデ

表2 未知語の実例の収集方法の比較
Table 2 Comparison between methods to collect unknown words.

方法	補間係数の値		情報量	
	1-gram	2-gram	学習セット	テストセット
1	0.193	0.807	6.303	6.075
2	0.007	0.993	4.113	6.905
3	0.191	0.809	6.307	6.040

ルであったことが分かる．方法2は，他の方法よりも2-gramの補間係数が非常に高く，学習に用いた実例の情報量は非常に低い．その一方，テストコーパスの未知語の情報量は相対的に高い．これは，未知語の性質とすべての形態素の性質が大きく異なることを意味する．方法1と方法3では結果に大差はないが，学習に用いた実例の情報量では方法1がより良く，テストコーパスの未知語の平均情報量では方法3がより良いという結果である．これも，学習に用いた形態素とテストコーパスの未知語の性質の類似性によると考えられる．

4.2.2 既知形態素の生成確率の分配方法

実験を行った既知形態素の生成確率の分配方法は以下のとおりである．未知語の実例の収集方法としては，上述の方法3を用いた．

方法A すべての未知形態素にその生成確率に比例して分配（式(1)）

方法B 特定の未知形態素に等しく配分（式(2)）

学習コーパスには出現しないが，辞書等に記載されている単語は多数ある．これらが，テストコーパスに未知語として多数出現する場合には，方法Bが方法Aよりも良い結果となることが容易に想像される．実際に問題になるのは，既知形態素の生成確率を分配する対象となる形態素集合の選択である．テキストの分野を反映した形態素が多く含まれているほど良い結果を導くと考えられる．また，未知語モデルを品詞等で分ける場合には，このための情報が記述されていなければならない．ここで述べる実験では，以下の2つの形態素集合の和集合とした．

- EDR日本語単語辞書[5]の見出し語
- 学習コーパスに出現する未知形態素

この結果，テストコーパスの未知語の文字あたりの情報量は，方法Aでは5.9338であり，方法Bでは5.1403であった．この差は十分有意であるので，既知形態素の生成確率をすべての表記に分配するモデルよりも形態素と考えられる特定の表記に配分するモデルが優れていると結論できる．

4.3 形態素 n-gram の評価実験

前項で述べた未知語モデルを持つ形態素 n-gram モデルを実装し，以下の項目を調べる実験を行った．

- 学習コーパスの大きさとクロスエントロピーの関係
- n の値（先行事象の長さ）とクロスエントロピーの関係

以下では，これらの結果を提示し，考察を加える．

表3 と**図1** は，n-gram モデル（$n = 1, 2, 3, 4$）による学習コーパスの大きさとクロスエントロピーの関係である．グラフから分かるように，1-gram モデル以外のクロスエントロピーは，学習コーパスの文字数が 10^7 の付近でもかなり減少している．このことは，1-gram モデル以外は，学習コーパスを大きくするだけでより良い言語モデルが得られることを意味する．ただし，グラフの横軸は学習コーパスの文字数の常用対数値であり，横軸を1目盛右に移動した結果を得るには10倍の学習コーパスが必要であるという点に注意しなければならない．

表4 と**図2** は，n の値（先行事象の長さ）とクロスエントロピーの関係である．ただし，未知語モデルは一定である．また，補間係数の推定の繰返し計算は，小数点以下6桁が変化しなくなるまで行った．この結果から，n の値をさらに大きくすることでモデルの予測力が増すことが分かる．しかし，その変化量はきわめて微小であり，変化量自身も減少している．よって $n = 16$ での情報量4.30330ビットを，本論文での日本語の情報量の上限の推定値とする．グラフおよび表から，先行事象の長さを長くすることでこの値が減少することは容易に予測される．また，未知語モデルを文字2-gramモデルからより次数の高い文字 n-gram モデルに変更することで，この値が減少することは容易に予測される．しかし，すでに述べた学習コーパスの大きさとクロスエントロピーの関係を考慮すると，先行事象の長さを長くすることによる減少量よりも，より大きな学習コーパスを用いることによる減少量の方が十分大きい．このことから，先行事象の長さを長くすることよりは，学習コーパスの大きさを増大することが言語モデルの改善により貢献すると考えられる．

Brown ら[3]は約6億文字の学習コーパスを用いて英語に対して行った実験の結果として，1.75ビットを報告している．アルファベット数は96としているので，これを考慮して換算すると，日本語を96の文字で表した場合の文字あたりの情報量は $4.30330 \times \frac{\log_2 96}{\log_2 6879} = 2.22$ となる．Brown らが用いた学習コーパスの大きさが $10^{8.78}$ 文字であること考慮すると，日本語の情報量と

表3 学習コーパスの大きさとクロスエントロピーの関係
Table 3 The relation between the size of training corpus and cross entropy.

学習コーパスの大きさ	2.56	3.23	3.85	4.45	5.06	5.66	6.26	6.86
1-gram モデル	11.9769	10.3021	8.7935	7.6972	7.0243	6.6247	6.3905	6.2586
2-gram モデル	11.3875	9.3900	7.7457	6.5839	5.8277	5.2788	4.8565	4.5437
3-gram モデル	11.3986	9.3708	7.7079	6.5316	5.7651	5.1958	4.7337	4.3514
4-gram モデル	11.4034	9.3673	7.7031	6.5239	5.7547	5.1817	4.7139	4.3217

学習コーパスの大きさは文字数の常用対数値

表4 n の値とクロスエントロピーの関係
Table 4 The relation between the value of n and cross entropy.

n	1	2	3	4	5	6	7	8
情報量	6.25854	4.54374	4.35139	4.32174	4.31302	4.31098	4.30956	4.30765
n	9	10	11	12	13	14	15	16
情報量	4.30575	4.30450	4.30386	4.30356	4.30342	4.30338	4.30333	4.30330

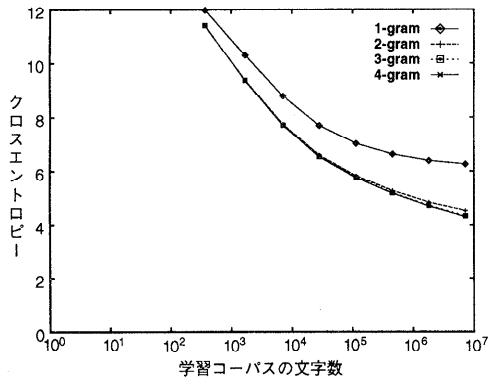


図1 学習コーパスの大きさとクロスエントロピーの関係
Fig. 1 The relation between the size of training corpus and cross entropy.

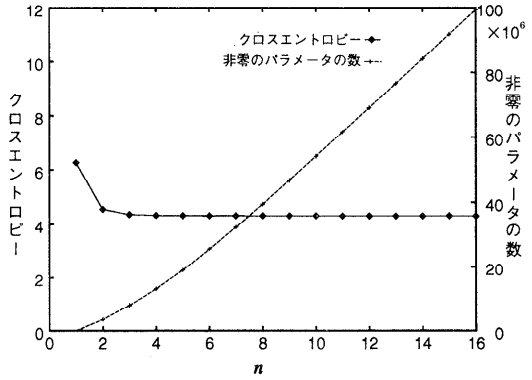


図2 n の値とクロスエントロピーの関係
Fig. 2 The relation between the value of n and cross entropy.

して我々が提示している値は，Brown らの結果と符合する．

n-gram モデルの実装に必要な頻度（確率）表は，配列で実装するのが簡潔かつ高速であるが，この場合には，$O(e^n)$ の記憶領域が必要である．このため，リストやハッシュなどのデータ構造を用いて，学習コーパスに実際出現する文字列の頻度だけを記憶するという方法がとられる．この場合の記憶領域の目安となる n 文字以下の文字列の種類数を図 2 に加えてある．このグラフから，リストやハッシュなどのデータ構造を有効利用すればおおよそ $O(n)$ の記憶領域で n-gram モデルが実装できることが分かる．また，表 4 と図 2 は，実際に n-gram モデルを応用する際に，適切に n の値を選ぶ指標となる．たとえば，2-gram モデルを 3-gram モデルに変更した場合，クロスエントロピーという基準で約 4.23%の改善となるが，頻度表の記述に必要となる記憶領域は約 2.27 倍となる．

3 章で述べたように，形態素 n-gram モデルは，形態素を予測する部分と未知語の文字列を予測する部分からなる．モデルを表す式から分かるように，これらの部分の全体への寄与を独立に計算することができる．**表5** は，このようにして計算したクロスエントロピーの各部分の内訳である（$n=16$）．この結果を見ると，テストコーパスに対するクロスエントロピーには，形態素を予測する部分がかなり大きく寄与していることが分かる．よって，短期的により良い言語モデルを構成するためには，この部分を改良することが近道であると考えられる．そのためには，品詞や文節などの文法的概念を用いることが有効であると考えられる．また，n-gram モデルよりも抽象度の高いモデルを用いることも考えられる．長期的には，未知語の文字列を予測する部分も改良することが望ましい．この場合にも，文字のクラスなど文法的概念や確率文脈自由文法などの n-gram モデルよりも抽象度の高いモデルを用いることが考えられる．また，より根本的に，文字予測という観点から形態素（単語）の定義を見直すこと

表5 クロスエントロピーの内訳
Table 5 Component contributions to the cross entropy.

モデルの部分	クロスエントロピー
形態素予測	3.92692
未知語の文字予測	0.37638
合計	4.30330

も興味深い課題である.

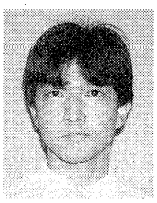
5. おわりに

本論文では,形態素を単位とした n-gram モデルによる日本語の文字あたりの情報量を計算する方法を説明し,EDR コーパス[5]を用いて行った実験の結果を報告した.コーパスの9割から推定した形態素 16-gram モデルを用いて,残りのコーパスの文字あたりの情報量を計算した結果,日本語の情報量の上限(クロスエントロピー)として 4.30330 ビットという値を得た.また,学習コーパスの大きさとモデルの次数によるクロスエントロピーの変化を調べた結果,モデルの次数を上げることによる減少量は微小であるが,学習コーパスを大きくすることによる減少量は形態素 2-gram モデルでもかなりあるということが分かった.

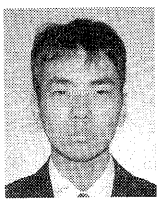
参考文献

1) Shannon, C.E.: Prediction and Entropy of Printed English, *Bell System Technical Journal*, Vol.30, pp.50–64 (1951).
2) Cover, T.M. and King, R.C.: A Convergent Gambling Estimate of the Entropy of English, *IEEE Trans. Information Theory*, Vol.24, No.4, pp.413–421 (1978).
3) Brown, P.F., Pietra, S.A.D. and Mercer, R.L.: An Estimate of an Upper Bound for the Entropy of English, *Computational Linguistics*, Vol.18, No.1, pp.31–40 (1992).
4) 浅井清朗:日本語の Entropy について,計量国語学, pp.4–7 (1965).
5) 日本電子化辞書研究所:EDR 電子化辞書仕様説明書 (1993).
6) Algeot, P. and Cover, T.: A Sandwich Proof of the Shannon-McMillan-Breiman Theorem, *Annals of Probability*, Vol.16, No.2, pp.899–909 (1988).
7) 韓 太舜,小林欣吾:情報と符号化の数理,岩波書店 (1994).
8) Nelson, M.: データ圧縮ハンドブック,トッパン (1994).
9) Jelinek, F.: Self-Organized Language Modeling for Speech Recognition, Technical Report, IBM T.J. Watson Research Center (1985).
10) 北 研二,中村 哲,永田昌明:音声言語処理,森北出版 (1996).
11) Brown, P.F., Pietra, V.J.D., deSouza, P.V., Lai, J.C. and Mercer, R.L.: Class-Based n-gram Models of Natural Language, *Computational Linguistics*, Vol.18, No.4, pp.467–479 (1992).
12) 永田昌明:単語頻度の期待値に基づく未知語の自動収集,情報処理学会研究報告, Vol.96-NL-116 (1996).
13) Ney, H.: The Use of One-Stage Dynamic Programming Algorithm for Connected Word Recognition, *IEEE Trans. Acoustics, Speech, and Signal Processing*, Vol.32, No.2, pp.263–271 (1984).

(平成9年6月 2 日受付)
(平成9年9月10日採録)



森 信介(学生会員)
1970年生.1995年京都大学大学院工学研究科電気工学第二専攻修士課程修了.同年,同大学大学院博士後期課程進学.計算言語学の研究に従事.言語処理学会会員.



山地 治
1973年生.1996年京都大学工学部電気系学科卒業.現在,同大学大学院工学研究科修士課程在学中.自然言語処理の研究に従事.言語処理学会会員.