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 ≤ n ≤ 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 increasing the order decreases entropy slightly and increasing the size of learning corpus decreases it notably. In addition, we discuss the relation between the number of parameters and entropy. This is useful 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，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．このモデルは
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．
言語モデルの評価として、単語単位のバープレキシティ（エントロピーに近い値を算出できる）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して目指すべき新しい絶対的な値としての意味はない。さらに、これらの文献では実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（単語と品詞の組合せ）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、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、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 ビットであつた。また、学習コーパスの大きさとモデルの次元による情報量の変化を調べた結果、モデルの次元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。この段階では、関係する概念を簡単な述べる。

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 \( X = \{ X_0, X_1, X_2, X_3, \ldots \} \) として、\( P \) 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、次のように表される。

\[
H(P) = \lim_{n \to \infty} -\frac{1}{n} \log P(X_{-1},X_{-2}, X_{-3}, \ldots, X_{n-1})
\]

もし、この確率過程がエントロピー的であれば、Shannon-McMillan-Breiman の定理 \(5) \) により次の式が成り立つ。

\[
H(P) = \lim_{n \to \infty} \frac{1}{n} \log P(X_1 X_2 \ldots X_n)
\]

よって、エントロピー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表せる。

\[
H(P,M) = \lim_{n \to \infty} -\frac{1}{n} E_P \log M(X_0|X_{-1},X_{-2}, \ldots, X_{n-1})
\]

よって、\( D(P||M) \) を \( P \) と \( M \) の Kullback-Leibler 距離 \(7) \) として、以下の式が成り立つ。

\[
H(P,M) - H(P) = \lim_{n \to \infty} \frac{1}{n} \log \frac{P(X_1, X_2, \ldots, X_n)}{M(X_1, X_2, \ldots, X_n)}
\]

Kullback-Leibler 距離は、文脈に非負であるので、この式から

\[
H(P) \leq H(P,M)
\]

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いければ、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きさ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でモデルを構成し、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可能である。

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。

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

ここで、$$I(X_1, X_2, \ldots, X_n)$$ 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、クロスタントロピーは、情報源 $$P$$ からのアルファベット列をモデル $$M$$ で用いて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。

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

等号が成り立たないのは、実際に符号化する際に整数値の符号を用いないなければならないという制約に応える。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、我々は実験によって説明された。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、簡単にというと形態素列のコーパスにおける頻度に基づき、ある形態素列を後続する形態素の出現頻度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、あたかもあらわれた形態素をモデル（既知語）に応答する状態に加えて、これ以外の形態素（未知語）すべてに応答する状態を持つ。未知語は、この状態から現れると考える。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記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野し、次記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 $$s_1, s_2, \ldots, s_n$$ 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 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。

$$P(s_1, s_2, \ldots, s_t) = \prod_{i=1}^{t} P(s_i | s_{i-k}, \ldots, s_{i-2}),$$

ここで $$s_i (i \leq 0)$$ と $$s_{i+1}$$ は、文頭と文末に対応する特別な記号である。そ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 $$P(s_i | s_{i-k}, \ldots, s_{i-2})$$ は同一（類似）の情報源から記号列の（コーパス）を用いて推定する。記号列の状態への還元が単独である場合に、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 $$N$$ とするとき、以下の式で表される。

$$P(s_i | s_{i-k}, \ldots, s_{i-2}) = \frac{N(s_{i-2}, \ldots, s_{i-1}, s_i)}{N(s_{i-k}, \ldots, s_{i-2})}$$

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下す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。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、次式で表される。

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

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 $$P(s_i | s_{i-2}) = P(s_i)$$ 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 $$(\lambda_1, \lambda_2, \ldots, \lambda_k)$$ 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。

$$(\lambda_1, \lambda_2, \ldots, \lambda_k) = \text{argmax}_{(\lambda_1, \lambda_2, \ldots, \lambda_k)} \prod_{i=1}^{h} P'(s_i)$$

ここで、$$s_i$$ 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式に区別した。
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。

$$N(s_{i-k} \cdots s_{i-2}s_{i-1}) > 0 \text{ かつ}$$
$$N(s_{i-k} \cdots s_{i-2}s_{i-1}) = 0$$
以上のようになることで，式 (1) の値が一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。そして，学習コーパスに対して，すべての記号の出現頻度を数える。実際に，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築できる。これを形態素 n-gram モデルと呼ぶ。この場合，形態素 n-gram の出現頻度を $m$ 個の互いに素な部分集合に観察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を望まない。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，未知形態素に対して特別な記号を用意し，未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。未知形態素に対して特別な記号は，かならずしも唯一である必要はなく，品詞などの情報を利用し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
```

この記号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。

$$P(\text{漱石/名詞，です/助動詞，./記号})$$
$$= P(\text{UM/名詞 | BT，BT})P(\text{漱石 | UM/名詞}) \times P(\text{です/助動詞 | BT，UM/名詞}) \times P(\text{./記号 | UM/名詞，です/助動詞}) \times P(\text{BT | です/助動詞，./記号})$$

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）を生成する（この行列では $P(\text{漱石 | UM/名詞})$ 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，未知語モデルが既に形態素が生成される場合は，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計算を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，$M_k$ は既知形態素の集合を表し，pos は $m_i$ の品詞を表す。また $m_i = BT$ （$i \leq 0 \vee i = h + 1$）である。

$$M_m(m_1 m_2 \cdots m_h)$$

$$= \prod_{i=1}^{h+1} P(m_i | m_{i-k} m_{i-k+1} \cdots m_{i-2} m_{i-1})$$

$$P(m_i | m_{i-k} m_{i-k-1} \cdots m_{i-2} m_{i-1})$$

$$= \begin{cases} P(\text{UM/pos | m_{i-k} m_{i-k-1} \cdots m_{i-2} m_{i-1})M_k/pos(m_i) & \text{if } m_i \in M_k \\ P(m_i | m_{i-k} m_{i-k-1} \cdots m_{i-2} m_{i-1}) & \text{if } m_i \notin M_k \end{cases}$$

この式中の $M_k/pos$ 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先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。例のそれぞれの文は，アルファベットの定義が明確でないことである，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内容を用いることや，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别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算することことで形態素 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
字と見なし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 \( \mathcal{M}_d \) に現れる既知文単位と現れない未知文単位に分類し、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と、モデルが改善されると考えられる。文字
は有限であるから、未知文単位の場合と異なり、各未知文単位の生成確率は等確率となることができる。た
とえば、未知形態素の表記として「激しめ」が与えられ、
このうち「激しめ」のみが未知文単すると、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、\( \mathcal{U} \) は未知文単
を表し、\( \mathcal{B} \) は形態素の区切りに対応する特別な記号
である。

\[
P(\text{激しめ}) = P(\mathcal{U} \mid \mathcal{B}, \mathcal{R}) P(\text{激しめ} \mid \mathcal{U})
\times P(\mathcal{B} \mid \mathcal{R}, \mathcal{U}) P(\mathcal{R} \mid \mathcal{U})
\]

このように、未知の文字は、文字 n-gram モデルで未
知文単位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、未知文単位の集合を \( \mathcal{X}_u \)
とすると以下の式で与えられる。

\[
P(x \mid \mathcal{U}) = \frac{1}{|\mathcal{X}_u|} \quad \text{ただし } x \in \mathcal{X}_u
\]

このような未知文単位モデル \( M_x \) は、未知形態素だけでな
く既知形態素の表記 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。

\[
\sum_{m \in \mathcal{M}_u} M_x(m) + \sum_{m \in \mathcal{M}_k} M_x(m) = 1 - \sum_{m \in \mathcal{x}} M_x(m) < 1
\]

これは、言語モデルとしての条件を満たしているが、
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
的である。

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

（1）

本論文では、辞書の出し方を変えることで与えられる未
知形態素の部分集合に等しく配分する方法を提案する。
つまり、ある形態素の集合が与えられたとき、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 \( M_d \) の要素数で割った値の和
とする。

\[
M'_x(m) = M_d(m) + \frac{1}{|\mathcal{M}_d|} \sum_{m \in \mathcal{M}_d} M_x(m)
\quad (m \in \mathcal{M}_d)
\]

（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。

以下に示された未知語モデル \( M'_x \) による、文字列 \( v_1 v_2 \cdots v_l \) の出現確率は以下の式で表される。ただし
\( \mathcal{X}_k \) は既知文単位を表す。また \( x_i = \mathcal{B} \) \( (i \not\leq 0 \vee i = l+1) \) である。

\[
M'_{x_1} (v_1 v_2 \cdots v_l)
= \begin{cases}
0 & \text{if } m \in \mathcal{M}_k \\
M_x(v_1 v_2 \cdots v_l) & \text{if } m \in \mathcal{M}_u \land m \not\in \mathcal{M}_d \\
M_x(v_1 v_2 \cdots v_l) + \frac{1}{|\mathcal{M}_d|} \sum_{m \in \mathcal{M}_d} M_x(m) & \text{if } m \in \mathcal{M}_u \land m \in \mathcal{M}_d
\end{cases}
\]

\[
P_x(v_1 | v_{i-k} \cdots v_{i-2} v_{i-1})
= \begin{cases}
P(x_i \mid x_{i-k} \cdots x_{i-2} v_{i-1}) & \text{if } x_i \in \mathcal{X}_k \\
P(\mathcal{U} \mid x_{i-k} \cdots x_{i-2} v_{i-1}) & \text{if } x_i \not\in \mathcal{X}_k
\end{cases}
\]

以上で説明した文字 n-gram モデルでは、未知文字を等確率で生成するモデルを「未知文字モデル」と考えると、文脈文 n-gram モデルと似た構式である。

文字 n-gram モデルの確率値は、文脈文 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、未知文脈文の実例における文字列の頻度から推定される。未知文脈文の実例の収集方法として、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、
1番目の部分コーパスの未知文脈文の実例を、
0番目の部分コーパス以外を学習コーパスとし、
1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文脈文とする。

対象としては、学習コーパスに含まれるすべての文脈文とすることや、学習コーパスにおける頻度が 1 である文脈文とする文脈文の選定方法は、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文脈文と類似した実例を得ているので、他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文脈文の実例に頻度で出現する文字をとることなどが考えられる。実際の集合を定義した後、これら未知文脈文に対応する特別な記号を加えてアルファベットとし、未知文脈文の実例の未知文脈文をその記号に置き換えて頻度を計数することで未知文脈文のモデルの確率値を推定する。補間係数の推定方法も同様に行うことができる。

4. 実験結果

我々は、前章で説明した文脈文モデルを用いて、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とした。これはすべての実験を通じて変化である。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を文圏切り記号を合わせた数である。

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 1 個の出現確率が最大になる補間係数の定義を用いた。その平均値が補間係数とした。この補間係数とすべての学習コーパスに対して計測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利用した。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も、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列に比例した時間を求めることができ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数した。文字 n-gram モデルの n を 2 とし、既知文字が 9 個に分割された学習コーパスの 2 個以上の現れる文字とし、以下の2点に関しては、前章で述べた他の方法と比較実験を行った。

（1）パラメータ推定のための未知語の実例の収集方法

（2）既知文脈文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通りである。

方法 1 学習コーパスにおける頻度が 1 である文脈文

方法 2 学習コーパスに含まれるすべての文脈文

方法 3 分割された学習コーパスの 1 個にのみ出現する文脈文

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がある。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がある。

これらの方法を実装し、学習に用いた実例（方法に依存する）の文字あたりの情報量を、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、文脈を区別していない。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

<table>
<thead>
<tr>
<th>表 1 コーパス</th>
<th>Corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td>用途</td>
<td>表 2</td>
</tr>
<tr>
<td>学習</td>
<td>文数</td>
</tr>
<tr>
<td>187,022</td>
<td>4,955,786</td>
</tr>
<tr>
<td>評価</td>
<td>20,750</td>
</tr>
</tbody>
</table>
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.

<table>
<thead>
<tr>
<th>方法</th>
<th>補間係数の値</th>
<th>情報量</th>
<th>学習セット</th>
<th>テストセット</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-gram</td>
<td>0.193</td>
<td>0.607</td>
<td>6.503</td>
<td>6.075</td>
</tr>
<tr>
<td>2-gram</td>
<td>0.007</td>
<td>0.993</td>
<td>4.113</td>
<td>6.905</td>
</tr>
<tr>
<td>3-gram</td>
<td>0.191</td>
<td>0.809</td>
<td>6.307</td>
<td>6.040</td>
</tr>
</tbody>
</table>

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の付近で僅かに減少している。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も良い逆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対数値であり、縦軸を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、補間係数の推定の順番と計算は、小数点以下6桁を変化させることで行う。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えることが分かる。しかし、その変化量はわずかであったが、変化量も減少している。よって n = 16 での情報量が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。ただし、未知語モデルを文字2-gram モデルからより下位の高い文字 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により、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重要と考えられる。

Brownら(9)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 × 10^{-96} = 2.22 となる。Brownらが用いた学習コーパスの大きさが10^{6.79}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ーズエントロピーの関係
Table 3 The relation between the size of training corpus and cross entropy.

<table>
<thead>
<tr>
<th>学習コーパスの大きさ</th>
<th>2.56</th>
<th>3.23</th>
<th>3.85</th>
<th>4.45</th>
<th>5.06</th>
<th>5.66</th>
<th>6.26</th>
<th>6.86</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-gram モデル</td>
<td>11.97</td>
<td>10.30</td>
<td>8.79</td>
<td>7.69</td>
<td>7.02</td>
<td>6.24</td>
<td>6.39</td>
<td>6.28</td>
</tr>
<tr>
<td>2-gram モデル</td>
<td>11.38</td>
<td>9.39</td>
<td>7.75</td>
<td>6.58</td>
<td>5.83</td>
<td>5.28</td>
<td>4.86</td>
<td>4.54</td>
</tr>
<tr>
<td>3-gram モデル</td>
<td>11.38</td>
<td>9.37</td>
<td>7.70</td>
<td>6.53</td>
<td>5.76</td>
<td>5.19</td>
<td>4.75</td>
<td>4.36</td>
</tr>
<tr>
<td>4-gram モデル</td>
<td>11.40</td>
<td>9.36</td>
<td>7.70</td>
<td>6.53</td>
<td>5.75</td>
<td>5.18</td>
<td>4.75</td>
<td>4.32</td>
</tr>
</tbody>
</table>

学習コーパスの大きさは文字数の対数値

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

<table>
<thead>
<tr>
<th>n</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
</tr>
</tbody>
</table>

図1 学習コーパスの大きさとクローズ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の値とクローズエントロピーの関係
Fig.2 The relation between the value of n and cross entropy.

して我々が提案している値は、Brownらの結果と同様に提案する。

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、この場合においては、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、学習コーパスに実装する文字列の順序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 に示す。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わかる。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、クローズエントロピーという基準で約42.3%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。モーエの表を基にした場合、これらナー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、このようなにして計算したクローズ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ーズエントロピーには、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わかる。よって、短期的に導入した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重要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要素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要素や文節文脈依存文法などの 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.

<table>
<thead>
<tr>
<th>モデルの部分</th>
<th>クロスエントロピー</th>
</tr>
</thead>
<tbody>
<tr>
<td>形態素予測</td>
<td>3.92692</td>
</tr>
<tr>
<td>未知語の文字予測</td>
<td>0.37038</td>
</tr>
<tr>
<td>合計</td>
<td>4.30330</td>
</tr>
</tbody>
</table>

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3)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

4) 浅井清浩：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).
7) 韓大舜，小林貞吾：情報と符号化の数理，岩波書店 (1994).
10) 北研二，中村哲，永田昌明：音声言語処理，森北出版 (1996).

(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。