音声言語情報処理 19－14
(1997. 12. 12)

予測単位の変更による n-gram モデルの改善

森 信介　山地 治　長尾 眞
京都大学工学研究科
〒606-01 京都市左京区吉田本町
{mori,oyamaji,nagao}@kuee.kyoto-u.ac.jp

あらまし
本論文では、文字 n-gram モデルや形態素 n-gram モデルの予測単位を文字列や形態素列に拡張した連文字 n-gram モデルや連語 n-gram モデルを定義し、予測力という観点でモデルを改善する方法を提案する。モデルの探索における目的関数は、形態素クラスタリングで有効性が示されている平均クロスエントロピーである。これは、削除補間のように、評価用のコーパスとモデルの推定用のコーパスとを別に用意するというアイデアに基づいている。日本語コーパスを用いた実験の結果、クロスエントロピーを計算すると、連文字 n-gram モデルは 4.3791 であり文字 n-gram モデルの 5.4105 より低く、連語 n-gram モデルは 4.4555 であり形態素 n-gram モデルの 4.6053 より低く、モデルの改善が観測された。
キーワード n-gram モデル 確率的言語モデル 連文字 連語 EDR コーパス

An Improvement of n-gram Model by a Change of the Prediction Unit

Shinsuke Mori　Osamu Yamaji　Makoto Nagao
Department of Electrical Engineering, Kyoto University
Yoshida-honmachi, Sakyo, Kyoto, 606-01 Japan
{mori,oyamaji,nagao}@kuee.kyoto-u.ac.jp

Abstract
In this paper, we define a string-based n-gram model and a phrase-based n-gram mode as expansions of character n-gram model and word-based n-gram model, and we propose a method to improve an n-gram model in terms of prediction. The objective function in model search is the average cross entropy, which is proven to be effective for word clustering. This criterion is, like deleted interpolation, based on the idea of separation of the corpus for evaluation and the corpus for model estimation. As an experimental result on a Japanese corpus, we obtained the entorpeis as follows: the string-based n-gram model had 4.3791, which is less than the character n-gram model's 5.4105, and the phrase-based n-gram mode had 4.4555, which is less than the word-based n-gram model's 4.6053.
Key Words n-gram model, stochastic language model, string, phrase, EDR corpus

1 はじめに

情報理論の初期の言語モデルは、文字列の頻度統計の結果から推定される文字 n-gram モデル [1] であるが、統計的手法を用いた音声認識の成功に端を発して、今日ではこの予測単位を単語とした単語 n-gram モデルが、様々な言語処理に用いられている。したがって、さまざまな応用の精度向上を図るために、単語 n-gram モデルを改善する方法が提案されている。これらの中で重要なのは、予測単位を単語の集合とするクラスに基づく n-gram モデル [2] や場合に応じて参照する文脈を可変長とする可変記憶長マルコフモデル [3] や予測単位としての単語数を変更するモデルがある。

本論文では、文字 n-gram モデルや形態素 n-gram モデルの予測単位を文字列や形態素列に変更するという自然な拡張としての連文字 n-gram モデルや連語 n-gram モデルを定義し、これによりモデルの改善が図られることを示す。連文字 n-gram モデルや連語 n-gram モデルにおいて問題となるのは、確率的言語モデルという意味で最適な連文字や連語を求めることである。この課題に対して、政瀧ら [4] は、n-gram モデルの推定に用いるコーパスのエントロピーを最小にするという条件で、任意の 2 単語を連語として特殊化していくことを提案している。しかし、この基準による最適解は、各文におけるすべての単語が連語として特殊化されたモデルである。したがって、探索を終了させるために、特殊化する単語組の数をあらかじめ恣意的に設定しておく必要がある。この方法の最大の問題点は、確率言語モデルの最終的な評価基準であるクロスエントロピーとは異なる基準を用いているため、特殊化する単語組の数によっては最終的に得られる連語 n-gram モデルが形態素 n-gram モデルよりも良くなることが必ずしも期待できないことである。他の先行研究として、文献 [5] では、単語クラスタリングの方法として提案された leaving-one-out 法 [6] を用いて、特殊化する連語の選択を行っている。これを日本語コーパスへ適用した結果は文献 [7] に報告されている。しかし、確率的言語モデルの最適化の基準としては、削除補間法を拡張した平均クロスエントロピーがより良いことが、クラスに基づく n-gram モデルのクラスの推定方法として報告されている [8]。本論文では、これらの報告を踏まえて、特殊化する単語列の選択基準として平均クロスエントロピーを用いることを提案する。実験の結果、文字 n-gram モデルや形態素 n-gram モデルの改善が観測された。

2 言語モデル

この章では、我々が実験に用いた文字 n-gram モデルと形態素 n-gram モデルについて説明する。これらは予測単位が異なる点を除けば本質的には同じである。つまり、文字列や形態素列のコーパスにおける頻度に基づき、ある予測単位列に後続する予測単位の出現確率を計算するというモデルである。

2.1 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 $s_1 s_2 \cdots s_l$ 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。

$$P(s_1 s_2 \cdots s_l) = \prod_{i=1}^{l+1} P(s_i | s_{i-k} \cdots s_{i-2} s_{i-1})$$

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。

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

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]。

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

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 以上の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。

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

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_i を除いた学習コーパスに対して行い、L_i を用いて補間係数を推定するということを i 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。

2.2 文字 n-gram モデル

前節で説明した n-gram モデルの記号を文字と考えることで、自然言語の文を文字の連接と見なすモデルが構成できる。これを文字 n-gram モデルと呼ぶ。この場合に問題となるのは、記号に対応する文字 (既知文字) の選択である。これを言語のすべての文字とすることも可能であるが、日本語のようにアルファベット数が大きい場合には、学習コーパスにすべての文字が出現するとは限らず、データスパースネスの問題をより重大にする。これを避けるために、アルファベットを既知文字集合 $\mathcal{X}_k$ と未知文字集合 $\mathcal{X}_u$ に分割し、未知文字を 1 つのグループとみなし、これを 1 つの未知文字記号 `UX` で代表し、すべての未知文字は未知文字記号から等確率に生成されることとする。

$$M_{ux}(x) = \frac{1}{|\mathcal{X}_u|}$$

したがって、以上で説明した文字 n-gram モデルは、アルファベット $\mathcal{X}_k \cup \{\texttt{UX}\}$ 上の n-gram モデル $M_{x,n}$ を用いて以下の式のように表される。ただし、式中の $f(\boldsymbol{x})$ は、文字列 $\boldsymbol{x}$ の未知文字を未知文字記号に置き換えた記号列を返す関数である。

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

$$P(x_i|x_{i-k}\cdots x_{i-2}x_{i-1}) = \begin{cases} \text{if } x_i \in \mathcal{X}_k \\ \quad M_{x,n}(x_i|f(x_{i-k}\cdots x_{i-2}x_{i-1})) \\ \text{else} \\ \quad M_{x,n}(\texttt{UX}|f(x_{i-k}\cdots x_{i-2}x_{i-1}))M_{ux}(x_i) \end{cases}$$

パラメータの推定は、既知文字を設定した後に、学習コーパスの未知文字を未知文字記号に置き換えることで得られる $\mathcal{X}_k \cup \{\texttt{UX}\}$ 上の n-gram 統計の結果から推定される。

2.3 形態素 n-gram モデル

文字 n-gram モデルの場合と同様に、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。この場合にも問題となるのは、記号に対応する形態素 (既知形態素 $\mathcal{M}_k$) 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文字 n-gram モデルからなる未知語モデル M_{um} 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 ($\mathtt{UM}_{pos}$)。

したがって、以上で説明した形態素 n-gram モデルは、アルファベット $\mathcal{M}_k \cup \{\mathtt{UM}_{pos_1}, \mathtt{UM}_{pos_2}, \cdots, \mathtt{UM}_{pos_k}\}$ 上の n-gram モデル $M_{m,n}$ を用いて以下の式のように表される。ただし、式中の $f(\boldsymbol{m})$ は、形態素列 $\boldsymbol{m}$ の未知形態素を未知形態素記号に置き換えた記号列を返す関数である。

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

$$P(m_i | m_{i-k} \cdots m_{i-2} m_{i-1}) = \begin{cases} \text{if } m_i \in \mathcal{M}_k \\ \quad M_{m,n}(m_i | f(m_{i-k} \cdots m_{i-2} m_{i-1})) \\ \text{else} \\ \quad M_{m,n}(\mathtt{UM}_{pos} | f(m_{i-k} \cdots m_{i-2} m_{i-1})) M_{um}(m_i) \end{cases}$$

この式の中の $M_{um}(m)$ は未知語モデルであり、未知形態素に対応する記号から形態素 m の表記が生成される確率を返す。この際、未知形態素記号を品詞ごとに区別している場合には、未知形態素の品詞情報を用いてもよい。未知語モデルとしては様々なモデルが考えられるが、前節で説明した文字 n-gram モデルを用いることとした。

文字 n-gram モデルの場合と同様に、パラメータの推定は、既知形態素を設定した後に、学習コーパスの未知形態素を未知形態素記号に置き換えることで得られる $\mathcal{M}_k \cup \{\mathtt{UM}_{pos_1}, \mathtt{UM}_{pos_2}, \cdots, \mathtt{UM}_{pos_k}\}$ 上の n-gram 統計の結果から推定される。ただし、形態素に分割されたコーパスが必要である点に注意しなければならない。未知語モデルのパラメータは、学習コーパスにおける未知形態素を実例として品詞ごとに推定される [11]。

以上で説明した形態素 n-gram モデルは、文字 n-gram モデルの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、文字 n-gram モデルと相似の構造である。

3 予測単位の変更

前章で説明した文字 n-gram モデルと形態素 n-gram モデルの相違は、未知文字モデルと未知語モデルを同一視すれば、予測単位のみである。形態素 n-gram モデルにおいて品詞を無視すれば、この予測単位はそれぞれ文字と文字列である。形態素 n-gram モデルにおける予測単位は学習コーパスにおける単語区切りに依存している。この区切りは、人間の言語直感に基づいているが、確率的言語モデルという観点からは、必ずしも最良であるとは考えられない。したがって、文字列予測という観点から導出された予測単位 (文字列) を用いて n-gram モデルを構築することで、文字 n-gram モデルを改善することが可能であると考えられる。このようなモデルを連文字 n-gram モデルと呼ぶ。形態素解析などのように、応用によっては形態素が最小単位であることが望ましいことがある。このような場合には、形態素 n-gram モデルの予測単位を形態素列に変更した連語 n-gram モデルを用いることができる。この章では、記号列が予測単位となる n-gram モデルを説明する。次に、連文字や連語の選択の基準 (目的関数) について述べる。最後に、連文字や連語の探索方法について述べる。

3.1 連文字モデルと連語モデル

品詞を無視した形態素 n-gram モデルは、文字列に対する n-gram モデルとなっている。文字も文字列と考えることができるので、未知文字を含む文字列を予測単位としないことを条件に、このようなモデルは、アルファベット $\mathcal{X}_k^+ \cup \{\mathtt{UX}\}$ 上の n-gram モデル $M_{x,n}$ を用いて以下の式のように表される。ただし、式中の $f(\boldsymbol{x})$ は、文字列の列 $\boldsymbol{x}$ の未知語を未知語記号に置き換えた記号列を返す関数であり、$x_1 x_2 \cdots x_l = \boldsymbol{x}_1 \boldsymbol{x}_2 \cdots \boldsymbol{x}_{l'}$ であるとする。

$$P(x_1 x_2 \cdots x_{l+1}) = \prod_{i=1}^{l'+1} P(\boldsymbol{x}_i | \boldsymbol{x}_{i-k} \cdots \boldsymbol{x}_{i-2} \boldsymbol{x}_{i-1})$$

$$P(\boldsymbol{x}_i | \boldsymbol{x}_{i-k} \cdots \boldsymbol{x}_{i-2} \boldsymbol{x}_{i-1}) = \begin{cases} \text{if } \boldsymbol{x}_i \in \mathcal{X}_k^+ \\ \quad M_{\boldsymbol{x},n}(\boldsymbol{x}_i | f(\boldsymbol{x}_{i-k} \cdots \boldsymbol{x}_{i-2} \boldsymbol{x}_{i-1})) \\ \text{else} \\ \quad M_{\boldsymbol{x},n}(\mathtt{UT} | f(\boldsymbol{x}_{i-k} \cdots \boldsymbol{x}_{i-2} \boldsymbol{x}_{i-1})) M_{ux}(\boldsymbol{x}_i) \end{cases}$$

同様に、連語 n-gram モデルはアルファベット $\mathcal{M}_k \cup \{\mathtt{UM}_{pos_1}, \mathtt{UM}_{pos_2}, \cdots, \mathtt{UM}_{pos_k}\}$ 上の n-gram モデル $M_{\boldsymbol{m},n}$ と形態素 n-gram モデルと同様の未知語モデルを用いて定義される。

このように定義される連文字 n-gram モデルや連語 n-gram モデルを構成する上で問題となるのは、文字列に区切ったコーパスが必要となることである。このような区切りを求めるための目的関数とアルゴリズムを以下で述べ

る。

3.2 目的関数

すでに述べたように、予測単位の変更の目的は文字列予測という観点からより良い言語モデルを構成することである。一般に、確率的言語モデルの予測力は、テストコーパスに対して計算されるクロスエントロピー [10] で評価される。これは、確率的言語モデル M とテストコーパス $L_{test} = \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_l$ の関数であり、以下の式で定義される。

$$H(L_{test}, M) = \frac{1}{\sum_{i=1}^{l} |\boldsymbol{x}_i|} \sum_{i=1}^{l} -\log M(\boldsymbol{x}_i)$$

この式の中で、本論文の枠組みにおける可変部分は、確率的言語モデルの予測単位だけである。したがって、クロスエントロピーを最小にする予測単位を、テストコーパスを参照せずに算出することが問題となる。そのための基準として、形態素 n-gram モデルをクラス n-gram モデルへ改善することを目的とした形態素クラスタリングの基準として提案されている平均クロスエントロピー [8] を用いる。これは、削除補間 [10] の考え方を拡張して、テストコーパスを参照することなく、学習コーパス内でテストコーパスを模擬する。つまり、学習コーパスを m 個に分割し、$m-1$ 個の部分でモデルを推定し、残りの部分をテストコーパスと見なしてクロスエントロピーを計算するということをすべての組合せにわたって行い、その平均値を全体の評価関数の値とする。

$$\overline{H} = \frac{1}{m} \prod_{i=1}^{m} H(M_i, L_i) \qquad (2)$$

ここで、M_i は i 番目以外の $m-1$ の部分コーパスから推定された n-gram モデル (補間係数の推定も含む) であり、L_i は i 番目の部分コーパスを表す。

本論文で問題としているのは、確率的言語モデルとして連文字 n-gram モデルや連語 n-gram モデルを用いた場合の予測単位の変更である。この場合、コーパス (文の列) は一定であり、確率的言語モデルは予測単位にのみ依存する。したがって、平均クロスエントロピーは、予測単位の関数と見なすことができる。クロスエントロピーの値域は正の実数であるから、平均クロスエントロピーの値域も正の実数であり、これにより予測単位 (学習コーパスの区切り方) に全順序関係を与えることができる。定義から明らかなように、この値がより小さいほうが、未知のコーパスに対してより良い言語モデルであることが予測される。以上の議論から、予測単位の推定の目的は、式 (2) で定義される平均クロスエントロピーを最小化する予測単位を求めることであるといえる。

政瀧ら [4] も、連語探索の基準としてエントロピーを用いているが、計算の対象とするコーパスは言語モデルの推定に用いるコーパスと同じである。我々は、この先行研究と異なり、削除補間法を応用して、連文字や連語の探索のためのコーパスを言語モデル推定用のコーパスとは別に用意することとしている。この利点は、テストコーパスを学習コーパス内で模擬しているので、テストコーパスに対する予測力の改善がより確実に期待できることである。

3.3 連語や連文字の探索アルゴリズム

予測単位の変更の解空間は、学習コーパスの文の文字列分割のすべての組合わせである。しかし、この数は非常に大きいため、これらすべてに対して平均クロスエントロピーを計算し、これを最小化する予測単位を選択するということは、計算量という観点から不可能である。平均クロスエントロピーは予測単位の一部分の変更が全体に影響するという性質を持っているので、分割統治法や動的計画法を用いることもできない。以上のことから、我々は最適解を求めることをあきらめ、貪欲アルゴリズムを用いることにした。連文字 n-gram モデルに対するアルゴリズムは以下のとおりである。

アルゴリズム

1. 文字 n-gram モデルを構築する (補間も行う)。
2. 学習コーパスのすべての部分コーパス (L_i) に出現する 2 文字以上の文字列をこの状態での平均クロスエントロピーの減少量の絶対値の降順に並べ $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ とする。
3. この順に $\boldsymbol{x}_i$ をひとつの予測単位とした場合の平均クロスエントロピーの変化を計算し、これが減少する場合には、学習コーパスに出現するすべての $\boldsymbol{x}$ をひとつの予測単位とし、補間係数を再推定する。

連語 n-gram モデルに対するアルゴリズムは、上述のアルゴリズムの文字を形態素とすることで得られる。

4 実験結果

我々は、前章で説明した予測単位の変更方法によって得られるモデルを評価するために、同じ学習コーパスから推定された文字 2-gram モデルと連文字 2-gram モデルと形態素 2-gram モデルと連語 2-gram モデルを構成し、テストセットパープレキシティで比較した。この章では、この実験の条件と結果を提示し、考察を行う。

表 1: コーパス

コーパス	文数	文字数	形態素数
学習コーパス	187,022	7,252,558	4,595,786
テストコーパス	20,780	802,576	509,261

表 2: 連文字 2-gram モデルの候補数とその例

文字数	候補数	例
2	30,265	こと
3	40,753	という
4	25,863	システム
5	13,818	ことができ
6	6,481	ネットワーク
7	2,861	コンピューター
8	1,214	なければならない
9	526	いるにもかかわらず
10	234	ことを明らかにした。
11	105	スーパーコンピューター
12	55	を開発、販売を開始した。
13	25	日の株主総会後の取締役会で
14	10	吹上御所でご療養中の天皇陛下
15	4	ゴルバチョフ・ソ連共産党書記長
合計	122,214	

4.1 実験の条件

実験には EDR コーパス [12] を用いた。まず、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。前章で述べたように、予測単位の推定では、この 9 個の学習コーパスのうちの 8 つから n-gram モデルを推定し、残りの 1 つのコーパスに対してクロスエントロピーを求めるということを 9 とおり行って得られる平均クロスエントロピーを評価基準とする。それぞれのコーパスに含まれる文数と文字数と形態素数は表 1 の通りである。既知文字や既知形態素は、2 個以上の部分学習コーパスに現れる文字や形態素とした。文字 2-gram モデルは既知文字に対応する状態の他に、未知文字に対応する状態と文区切りに対応する状態を持つ。形態素 2-gram モデルは、既知形態素に対応する状態の他に、各品詞の未知語に対応する状態 (15 個) と文区切りに対応する状態を持つ。連文字 2-gram モデルや連語 2-gram モデルは、これらに加えて、探索によって得られた連文字や連語に対応する状態を持つ。

表 3: 連語 2-gram モデルの候補数とその例

形態素数	候補数	例
2	20,649	い/動詞 う/語尾
3	12,184	て/助詞 い/動詞 る/語尾
4	6,498	に/助詞 つ/動詞 い/語尾 て/助詞
5	3,368	ば/助詞 な/動詞 ら/語尾 な/助動詞 い/語尾
6	1,509	の/助詞 で/助動詞 は/助詞 な/形容詞 い/語尾 か/助詞
7	544	な/助動詞 けれ/語尾 ば/助詞 な/動詞 ら/語尾 な/助動詞 い/語尾
8	154	わけ/名詞 に/助詞 は/助詞 い/動詞 か/語尾 な/助動詞 い/語尾 。/記号
9	40	を/助詞 開発/動詞 、/記号 販売/名詞 を/助詞 開始/動詞 し/語尾 た/助動詞 。/記号
合計	44,946	

それぞれのモデルを比較するために、これらを同じ学習コーパスから構成し、同じテストコーパスに対してクロスエントロピーを計算した。それぞれの言語モデルの構成の手順は以下のとおりである。

- 文字 2-gram モデル
  1. 削除補間により補間係数を推定
  2. すべての学習コーパスを対象に文字 2-gram と文字 1-gram を計数
- 連文字 2-gram モデル
  1. 削除補間により補間係数を推定
  2. 前章で述べた方法で予測単位を推定
  3. すべての学習コーパスを対象に連文字 2-gram と連文字 1-gram を計数
- 形態素 2-gram モデル
  1. 削除補間により補間係数を推定
  2. すべての学習コーパスを対象に形態素 2-gram と形態素 1-gram を計数
- 連語 2-gram モデル
  1. 削除補間により補間係数を推定
  2. 前章で述べた方法で予測単位を推定

表 4: 文字 2-gram モデルと連文字 2-gram モデルの比較

言語モデル	状態数	平均文字数	エントロピー
文字 2-gram	3,119	1.0000	5.4105
連文字 2-gram	15,554	1.6786	4.3791

表 5: 形態素 2-gram モデルと連語 2-gram モデルの比較

言語モデル	状態数	平均文字数	エントロピー
形態素 2-gram	59,973	1.5534	4.6053
連語 2-gram	63,825	1.9309	4.4555

3. すべての学習コーパスを対象に連語 2-gram と連語 1-gram を計数

表2と表3は、連文字 2-gram モデルと連語 2-gram モデルの探索の候補数とひとつの予測単位となった文字列や形態素列の例である。

文字 2-gram モデル以外によるテストコーパスの生成方法は複数存在する可能性があるが、形態素への分割は予めコーパスに付加されたものを用い、連文字や連語への書き換えは学習コーパスの書き換えと同じ順序で行った。したがって、これらのモデルのクロスエントロピーは、テストコーパスの真の生成確率でもなければ、確率最大の生成確率でもない。

形態素 2-gram モデルと連語 2-gram モデルに含まれる未知語モデルは、品詞を区別した上での文字 2-gram モデルである。既知形態素集合が共通なので、この部分のクロスエントロピーへの寄与は一定であり、クロスエントロピーの差を考える限りにおいては影響がない。

4.2 結果と考察

表4と表5はそれぞれ、文字 2-gram モデルと連文字 2-gram モデルの比較と、形態素 2-gram モデルと連語 2-gram モデルの比較である。それぞれの場合において、予測単位の変更にって n-gram モデルの予測力が改善していることが分かる。なお、これらすべての中で最良なモデルは連文字 2-gram モデルであるが、この比較はあまり意味をなさないであろう。これは、探索の目的関数が真の生成確率から計算される平均クロスエントロピーではないことによる。

表6と表7はそれぞれ、文字 n-gram モデルと連文字 2-gram モデル及び連文字 3-gram モデルの比較と、形態素 n-gram モデルと連語 2-gram 及び連語 3-gram モデルの比較である。ただし、連文字 3-gram モデルと連文字 3-gram モデルの予測単位は、連文字 2-gram モデルや連語 2-gram モデルと同じであり、 3-gram モデルを仮定して探索した結果ではない。これらの結果から以下のことが分かる。連文字 2-gram モデルの予測力は文字 3-gram モデルと文字 4-gram モデルの間に位置する。連文字 3-gram モデルの予測力は、文字 16-gram モデルよりも良い。文献 [13] によると $n = 16$ 付近での n に対するクロスエントロピーの変化は非常に小さく、先行事象をさらに長くしても連文字 3-gram モデルの予測力を超えない。これは、情報理論的な観点から日本語の単位として連文字 (単語) を導入することの絶対的な優位性を示すと考えられる。その一方、連語 3-gram モデルの予測力は形態素 6-gram モデル相当であり、連文字 3-gram モデルほどの予測力の向上が達成されていない。連語 3-gram モデルを用いて探索した結果ではないので、拙速な判断である可能性は否めないが、この理由は、コーパスが与える形態素の定義が、確率的予測という観点では足枷になっていることであると考えられる。

表 6: 連文字 2-gram モデルと文字 n-gram モデルの比較

言語モデル	エントロピー	カバー率
文字 2-gram モデル	5.4152	98.837%
文字 3-gram モデル	4.4652	88.953%
文字 4-gram モデル	4.2544	68.851%
文字 16-gram モデル	4.1863	0.237%
連文字 2-gram モデル	4.3791	90.429%
連文字 3-gram モデル *	4.1769	51.047%

* 予測単位は連文字 2-gram モデルの結果を使用

表 7: 連語 2-gram モデルと形態素 n-gram モデルの比較

言語モデル	エントロピー	カバー率
形態素 2-gram モデル	4.6053	91.807%
形態素 3-gram モデル	4.4129	65.073%
形態素 6-gram モデル	4.3725	12.922%
形態素 16-gram モデル	4.3649	0.034%
連語 2-gram モデル	4.4555	85.861%
連語 3-gram モデル *	4.3725	46.444%

* 予測単位は連語 2-gram モデルの結果を使用

探索の基準の良否を調べるために、連語 2-gram モデルの連語探索の途中でのテストコーパスに対するクロスエントロピーを、連語探索の基準として平均クロスエントロ

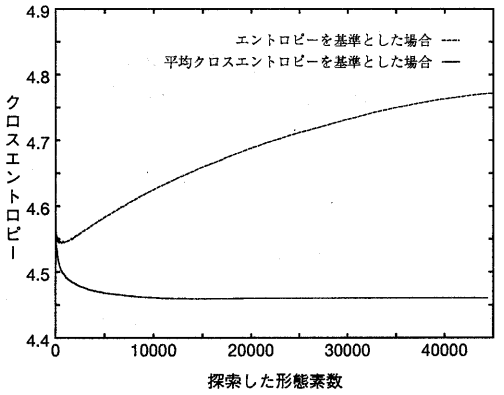

図 1: 単語数とクラス数の関係

ピーを用いた場合とエントロピーを用いた場合の双方について計算した。図1は、この結果である。双方共に、探索の初期の段階ではテストコーパスに対するクロスエントロピーの減少が観測されるが、エントロピーを用いた場合は途中からこれが上昇に転じる。これは、形態素 3-gram モデルとの比較から分かるように、初期状態である形態素 2-gram モデルが盲目的な特殊化によっても改善されるが、ある程度の特殊化の後には、適切な特殊化のみが真に有効であることによると考えられる。エントロピーを目的関数として形態素の品詞からの特殊化と長さ 2 の連語の特殊化を行った研究 [4] においてもモデルの改善が報告されているが、これは特殊化の回数を 1000 としていることが幸していると考えられる。このような方法に対する我々の方法の優位性は、このように特殊化の回数に恣意的な制限を設ける必要がないことに加えて、実験的にではあるが、最終的に得られるモデルの予測力がより高いことが示されていることである。

5 おわりに

本論文では、文字 n-gram モデルや形態素 n-gram モデルの予測単位を変更することで得られる連文字 n-gram モデルや連語 n-gram モデルを仮定して、準最適な予測単位を求める方法について述べた。この方法は、予測単位を推定するのためのコーパスをモデルの推定用のコーパスとは別に用意するという削除補間のアイデアを応用している。このアルゴリズムを実装し、EDR コーパス [12] を用いて、実験した結果、連文字 n-gram モデルや連語 n-gram モデルの予測力が文字 n-gram モデルや形態素 n-gram モデルよりも高くなることが観測された。

今後の課題として、探索の対象となるモデル空間を、参照する履歴 (条件付き確率の条件部分) を可変長とする可変記憶長マルコフモデル [3] や、予測単位を単語の集合とするクラスに基づく n-gram モデル [2] [8] のモデル空間を含むように変更することが考えられる。

参考文献

[1] C. E. Shannon. Prediction and Entropy of Printed English. *Bell System Technical Journal*, Vol. 30, pp. 50–64, 1951.

[2] Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jenifer C. Lai, and Robert L. Mercer. Class-Based n-gram Models of Natural Language. *Computational Linguistics*, Vol. 18, No. 4, pp. 467–479, 1992.

[3] Dana Ron, Yoram Singer, and Naftali Tishby. The power of amnesia: Learning probabilistic automata with variable memory length. In *Machine Learning Special Issue on COLT94*, 1996.

[4] 政瀧浩和, 松永昭一, 匂坂芳典. 連続音声認識のための可変長連鎖統計言語モデル. 電子情報通信学会技術研究会報告 SP95-73, pp. 1–6, 1995.

[5] Klaus Ries, Finn Dag Buø, and Alex Waibel. Class Phrase Models For Language Modeling. In *International Conference on Speech and Language Processing*, 1996.

[6] R. Kneser and H. Ney. Improved Clustering Techniques for Class-Based Statistical Language Modelling. In *Eurospeech*, pp. 21–23, 1993.

[7] Laura Mayfield Tomokiyo and Klaus Ries. What makes a word: Learning base units in Japanese for speech recognition. In *Proceedings of the Computational natural Language Learning*, pp. 60–69, 1997.

[8] 森信介, 西村雅史, 伊東伸泰. クラスに基づく言語モデルのための単語クラスタリング. 情報処理学会論文誌, Vol. 38, No. 11, 1997.

[9] F. Jelinek. 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] 森信介, 山地治. 日本語の情報量の上限の推定. 情報処理学会論文誌, Vol. 38, No. 11, 1997.

[12] 日本電子化辞書研究所. EDR 電子化辞書仕様説明書, 1993.

[13] 森信介. テキストコーパスからの確率的言語モデルの推定. 博士論文, 京都大学工学研究科, 1997.