自然言語処理 125－10
音声言語情報処理 21－8
（1998. 5. 29）

確率的モデルによる仮名漢字変換

森 信介
日本IBM 東京基礎研究所
〒242-8502 大和市下鶴間1623-14
mori@trl.ibm.co.jp

土屋 雅稔 山地 治 長尾 真
京都大学大学院情報学研究科
〒606-8317 京都市左京区吉田本町
{tsuchiya,oyamaji}@kuee.kyoto-u.ac.jp

あらまし
本論文では、確率的モデルによる仮名漢字変換を提案する。これは、従来の規則とその重みに基づく仮名漢字変換と異なり、入力に対応する最も確率の高い仮名漢字混じり文を出力とする。この方法の有効性を確かめるため、片仮名列と仮名漢字混じり文を有するコーパスを用いた変換実験を行ない、変換精度を測定した。変換精度は、第一変換候補と正解の最長共通部分列の文字数に基づく再現率と適合率である。この結果、我々の提案する手法による再現率は95.07%であり、適合率は93.94%であった。これは、市販の仮名漢字変換器の一つであるWnn6の同じテストコーパスに対する再現率(91.12%)と適合率(91.17%)を有意に上回っており、確率的モデルによる仮名漢字変換の有効性を示す結果となった。

キーワード 仮名漢字変換 確率的手法 コーパス 最長共通部分列 Wnn6

Kana-Kanji Conversion by A Stochastic Model

Shinsuke Mori
Tokyo Research Laboratory, IBM Japan
1623-14 Shimotsuruma Yamatoshi
Kanagawaken 242-8502 Japan
mori@trl.ibm.co.jp

Tsuchiya Masatoshi, Osamu Yamaji, Makoto Nagao
Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo
Kyoto, 606-8317 Japan
{tsuchiya,oyamaji}@kuee.kyoto-u.ac.jp

Abstract
In this paper, we present a *kana-kanji* converter by a stochastic model. Given an input this method returns the most probable *kana-kanji* character sequence. For its evaluation, we converted *kana* sequences of a corpus containing *kana-kanji* sequences. The criterion we used is the ratio of the length of longest common subsequece. The recall and precision of our method are 95.07% and 93.94% respectively. This result is much better than that of Wnn6 (recall: 91.12%; precision: 91.17%).

Key Words *Kana-kanji* converter, Stochastic Method, Corpus, Longest Common Subsequence, Wnn6

1 はじめに

計算機に日本語を入力することを考えた場合、日本語の記述に用いられるアルファベット数は通常のキーボードのキーの数を遥かに上回っているので、英語などの入力のようにアルファベットにキーを直接対応させることはできない。したがって、日本語の入力には多数のアルファベットを少数のキーによって入力する方法が必要である。このような方法として、ユーザーが意図する日本語の文(仮名漢字混じり文)の読みに相当する仮名文字列を入力し、これを計算機が仮名漢字混じり文に変換するという仮名漢字変換方式が一般的である。仮名文字列に対応する仮名漢字混じり文は複数存在するので、計算機はユーザーの意図した文を推定する必要があり、これを実現するための様々な方法が提案されている。多くの先行研究では、まず仮名文字列を文節単位に分割し、次に文節を構成する単語を適切に選ぶという手順によって仮名漢字変換を行っている。文節単位に分割するための方法としては、2文節最長一致法[1]や文節数最小法[2]などが提案されている。また、単語の選択については前後に連続している文字や単語による方法[3, 4]や格フレームを用いる方法[5]などが提案されている。これらの方法では、いずれも開発者の言語的直感に基づいた規則とその重み付けによって仮名漢字変換の精度向上を図っている。しかしながら、このような方法では、規則の増加にともなって仮名漢字変換システムが複雑になると、規則の重みの調整が極めて困難になる。また、どの程度の規則を追加すれば、どの程度の精度向上が期待できるかを見積もることができないなどの欠点が指摘される。

計算機に日本語を入力する他の方法として、音声認識が最近にわかに脚光を浴びている。精度やインターフェイスなどの点では、キーボードを用いた仮名漢字変換には遠く及ばないが、確率を利用するというパラダイムは注目に値する。つまり、根拠のない規則や重み付けに頼らずに、現実に存在する膨大な数の文(コーパス)や大量の音響特徴量列からそれぞれ確率的言語モデルと確率的音響モデルを構築しておく。そして、未知の入力をこれらのモデルに照らし合わせて、最も確率の高い文字列(仮名漢字混じり文)を出力する。これは、確率的モデルによる音声認識と呼ばれており、現在の音声認識の主流となっている方法である。

このような確率的モデルの音声認識における成功を踏まえて、本論文では、確率的モデルによる仮名漢字変換を提案する。音声認識との主な違いは、入力が音響特徴量の列ではなく、読みなどのキーボードからの入力が可能な記号の列である点である。しかし、確率的手法を採用すれば、この違いは、単語などの単位で音響特徴量の列を表記の列と対応させる音響モデルの代わりに、キーボードからの入力が可能な記号の列を表記の列と対応させるモデルを用いることで吸収される。したがって、どのような日本語文が出現しやすいかを記述する確率的言語モデルには、今日までに蓄積された研究成果を用いることができる。この確率的言語モデルとしては、クラス n-gram モデル[6]を用いることとした。これは、実用となっている音声認識において頻繁に用いられている単語 n-gram モデルの改良であり、単語をクラスと呼ばれるグループに分類することで、言語の記述精度を有意に向上させている。このクラス分類の算出には、削除補間を応用することで得られる平均クロスエントロピーを目的関数とした単語クラスタリング[7]を用いた。

確率的モデルによる仮名漢字変換の有効性を確かめるため、EDR コーパス[8]を用いた変換実験を行なった。コーパスを9対1の比率で分割し、前者から言語モデルを推定し、後者に対して変換精度の計算を行った。変換精度は、第一変換候補と正解の最長共通部分列(longest common subsequece)[9]の文字数に基づく再現率と適合率である。この結果、クラス 2-gram モデルによる再現率は 95.07% であり、適合率は 93.94% であった。これは、市販の仮名漢字変換器の一つである Wnn6 の同じテストコーパスに対する再現率(91.12%)と適合率(91.17%)を有意に上回っており、確率的モデルによる仮名漢字変換の有効性を示す結果となった。

2 仮名漢字変換の定式化

仮名漢字変換は、キーボードから直接入力することが可能な記号 $\mathcal{Y}$ の正閉包 $y \in \mathcal{Y}^+$ からの、日本語のアルファベット $\mathcal{X}$ の正閉包 $x \in \mathcal{X}^+$ への対応である。仮名漢字変換の入力の記号列は、一般的に、ユーザーが計算機に入力したい日本語文の読みである。このとき、複数の日本語文が同一の読みを共有する状況が頻繁に発生する。つまり、仮名漢字変換の読みに対応する日本語文(変換候補)が複数あるという状況である。このような場合には、入力効率を最大にするために、ユーザーが意図している日本語列に近いと推測される変換候補を順次出力する。したがって、仮名漢字変換は、キーボードから直接入力することが可能な記号列(読み)から日本語文(変換候補)の列への写像である。これは、以下の式のように示される。

$$\mathcal{Y}^+ \mapsto (\mathcal{X}^+, \mathcal{X}^+, \cdots, \mathcal{X}^+)$$

ここで、右辺の変換候補の数は入力の記号列に依存し、それらはユーザーが意図している日本語文に近いと推測される順に左から右へ並んでいるとする。

2.1 確率的モデルによる仮名漢字変換

上で定義したような仮名漢字変換を実現する方法の一つとして、大量の日本語文(コーパス)から推定された確

率的モデルを用いる方法を提案する。これは、基本的には音声認識と同じであるが、入力が音響特徴量の列ではなくキーボードから入力される記号列である点と最尤解だけでなくすべての候補をその尤度順に出力する点が異なる。この尤度は、キーボードからの入力の記号列が与えられたときの日本語文の条件付確率 $P(\boldsymbol{x}|\boldsymbol{y})$ である。したがって、確率的モデルによる仮名漢字変換器 wnm は、以下のような写像である。

$$wnm(\boldsymbol{y}) = (\boldsymbol{x}_1,\ \boldsymbol{x}_2,\ \ldots,\ \boldsymbol{x}_n)$$

$$\text{ただし}\quad i \leq j \Leftrightarrow P(\boldsymbol{x}_i|\boldsymbol{y}) \geq P(\boldsymbol{x}_j|\boldsymbol{y})$$

この式から、仮名漢字変換器の主要な役割は、各変換候補の確率値 $P(\boldsymbol{x}|\boldsymbol{y})$ の順序関係の算出であることがわかる。逆にこの順序関係を保持している限りにおいて、実際にはこの確率値以外の他の値を用いてもよいと結論できる。我々は、この点を考慮に入れて、確率的音声認識における式変形と同様に、以下の式のように確率的言語モデルの分離を行なうことを提案する。

$$\begin{aligned} & P(\boldsymbol{x}_i|\boldsymbol{y}) \geq P(\boldsymbol{x}_j|\boldsymbol{y}) \\ \Leftrightarrow\ & \frac{P(\boldsymbol{y}|\boldsymbol{x}_i)P(\boldsymbol{x}_i)}{P(\boldsymbol{y})} \geq \frac{P(\boldsymbol{y}|\boldsymbol{x}_j)P(\boldsymbol{x}_j)}{P(\boldsymbol{y})} \\ & \qquad (\because \text{ベイズの公式}) \\ \Leftrightarrow\ & P(\boldsymbol{y}|\boldsymbol{x}_i)P(\boldsymbol{x}_i) \geq P(\boldsymbol{y}|\boldsymbol{x}_j)P(\boldsymbol{x}_j) \qquad (1) \\ & \qquad (\because P(\boldsymbol{y})\text{は}\boldsymbol{x}_i\text{や}\boldsymbol{x}_j\text{によらない}) \end{aligned}$$

この式において、日本語文 $\boldsymbol{x}$ の出現確率を表す $P(\boldsymbol{x})$ が確率的言語モデルである。残りの $P(\boldsymbol{y}|\boldsymbol{x})$ は、日本語文 $\boldsymbol{x}$ が与えられたときのキーボードからの入力の記号列（読み）の確率を表す。これを仮名漢字モデルと呼ぶことにする。仮名漢字変換のための確率的モデルが、確率的言語モデルと仮名漢字モデルという互いに独立なモデルに分割されている点に注意しなければならない。以下の節では、これらのモデルについて述べる。

2.2 確率的言語モデル

日本語の確率的言語モデルは、日本語のアルファベット列 $\mathcal{X}^*$ が出現する確率値を記述する。これは、以下のように表される。

$$P : \mathcal{X}^* \mapsto [0,\ 1]$$

確率的モデルであるので、確率値をすべてのアルファベット列に渡って合計すると 1 以下になる必要がある。

$$\sum_{\boldsymbol{x} \in \mathcal{X}^*} P(\boldsymbol{x}) \leq 1$$

このようなモデルとしては、文字 n-gram モデル [10] や形態素 n-gram モデル [11] やクラス n-gram モデル [7] などの入力の長さ n に対して $O(n)$ の解析アルゴリズムが知られている確率正規文法に属するモデルや係り受けモデル [12] などの入力の長さ n に対して $O(n^3)$ の解析アルゴリズムが知られている確率文脈自由文法に属するモデルなどを用いることができる。本論文では、予測力が高いという長所のみならず記憶領域が小くて済むという利点も兼ね備えたクラス n-gram モデルを用いることとした。文字 n-gram モデルを用いれば記憶領域はより小さくなるが、未知語の登録などの実際的な側面に支障を来たす。係り受けモデルを用いれば、形態素の連接という局所的な現象のみならず、位置的に離れた形態素の呼応や共起などを考慮する必要があるような変換候補の曖昧性の解消を行なうことができると考えられるが、本論文では対象としていない。これは、照応などのより複雑な言語現象を記述する確率的言語モデルの研究とともに今後の課題である。

クラス n-gram モデル [6] では、既知形態素 $\mathcal{M}_k$ をあらかじめクラスと呼ばれるグループに分類し、未知形態素を品詞ごとの未知語記号 ($\mathtt{UM}_t$) で代表されるクラスに分類しておき、先行するクラスの列を直前の事象とみなして分類する。そして、各時点の形態素を直接予測するのではなく、クラスを予測した後、そのクラスから形態素を予測する。未知形態素の場合は、後述する品詞ごとの未知語モデル $M_{x,t}$ を用いてその表記を予測する。入力の形態素列を $\boldsymbol{m} = m_1 m_2 \cdots m_h$ とし、形態素 m_i のクラスを c_i とすると、クラス n-gram モデル $M_{c,n}$ による形態素列の出現確率は、以下の式で与えられる。

$$M_{c,n}(\boldsymbol{m}) = \prod_{i=1}^{h+1} P_c(m_i|c_{i-k}c_{i-k+1}\cdots c_{i-1}) \qquad (2)$$

$$\begin{aligned} & P_c(m_i|c_{i-k}\cdots c_{i-2}c_{i-1}) \\ & = \begin{cases} \text{if} \quad m_i \in \mathcal{M}_k \\ \qquad P(c_i|c_{i-k}\cdots c_{i-2}c_{i-1})P(m_i|c_i) \\ \text{if} \quad m_i \notin \mathcal{M}_k \\ \qquad P(\mathtt{UM}_t|c_{i-k}\cdots c_{i-2}c_{i-1})M_{x,t}(m_i) \end{cases} \qquad (3) \end{aligned}$$

この式の中の c_j $(j \leq 0)$ は、文頭に対応する特別な記号である。これを導入することによって式が簡便になる。また、c_{h+1} は、語末に対応する特別な記号であり、これを導入することによって、すべての可能な文字列に対する確率の和が 1 となる。確率 $P(c_i|c_{i-k}c_{i-k+1}\cdots c_{i-1})$ の値および、確率 $P(m_i|c_i)$ の値は、形態素に分割されたコーパスから以下の式を用いて最尤推定することで得られる。

$$\begin{aligned} & P(c_i|c_{i-k}c_{i-k+1}\cdots c_{i-1}) \\ & = \frac{N(c_{i-k}c_{i-k+1}\cdots c_i)}{N(c_{i-k}c_{i-k+1}\cdots c_{i-1})} \qquad (4) \end{aligned}$$

$$P(m_i|c_i) = \frac{N(m_i, c_i)}{N(c_i)} \quad (5)$$

形態素 n-gram モデル同様、データスパースネスの問題に対処する方法として、補間 [6] を用いることができる。

$$P'(c_i|c_{i-k}c_{i-k+1}\cdots c_{i-1}) = \sum_{j=0}^{k} \lambda_j P(c_i|c_{i-j}c_{i-j+1}\cdots c_{i-1})$$

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

以上で説明したクラス n-gram モデルによる日本語文 $\boldsymbol{x}$ の出現確率は、以下の式で表されるように、表記の連接 $\boldsymbol{w}(\boldsymbol{m})$ が $\boldsymbol{x}$ と等しくなるような形態素列の出現確率の総和である。

$$P(\boldsymbol{x}) = \sum_{\boldsymbol{w}(\boldsymbol{m})=\boldsymbol{x}} M_{c,n}(\boldsymbol{m})$$

しかし、この場合には文字列の等価性の判断を必要とし、結果として解探索の計算速度が著しく低下すると考えられる。この問題を回避するために、本論文では、以下の式が示すように、条件を満たす形態素列を区別し、その出現確率でこの値を近似することとした。

$$P(\boldsymbol{x}) \approx M_{c,n}(\boldsymbol{m}) \quad (6)$$

この近似により、出力の日本語文の形態素解析結果が得られるという副次的効果があることを付言しておく。

2.3 確率的仮名漢字モデル

確率的仮名漢字モデル $P(\boldsymbol{y}|\boldsymbol{x})$ は、日本語文 $\boldsymbol{x}$ が与えられたときのキーボードからの入力の記号列 $\boldsymbol{y}$ の確率を表す。あらゆる可能な日本語文に対する入力記号列の確率を推定することは不可能であるから、日本語文を文字や形態素や文節などの単位に分割し、それらの入力記号列との対応関係がそれぞれ独立であると仮定する。この単位としては、現実的な最小単位としての形態素を選択することとした。文字を単位とした場合には、読みが文字ごとに分割できない熟語の扱いが困難になり、文節やそれ以上に特殊化された単位では、データスパースネス問題が生じると考えたからである。その他に、形態素単位の言語モデルを採用していることに由来する探索の容易さや、一般的に利用できるコーパスが形態素に分割され読みが振られているという点も、確率的仮名漢字モデルの単位として形態素を選択した理由である。

上述のように、形態素列と入力記号列の対応において、各形態素と各入力記号列が互いに独立であると仮定すると、形態素列 $\boldsymbol{m}$ が与えられたときの入力記号列 $\boldsymbol{y}$ の確率的仮名漢字モデル M_{kk} による出現確率は以下の式で表される。

$$M_{kk}(\boldsymbol{y}|\boldsymbol{m}) = \prod_{i=1}^{h} P(\boldsymbol{y}_i|m_i) \quad (7)$$

ここで、入力記号部分列 $\boldsymbol{y}_i$ は形態素 m_i に対応する入力記号列であり、以下の条件を満たす。

$$\boldsymbol{y} = \boldsymbol{y}_1\boldsymbol{y}_2\cdots\boldsymbol{y}_h$$

確率 $P(\boldsymbol{y}_i|m_i)$ の値は、形態素ごとに（入力記号列）読みが振られたコーパスから以下の式を用いて最尤推定することで得られる。

$$P(\boldsymbol{y}_i|m_i) = \frac{N(\boldsymbol{y}_i, m_i)}{N(m_i)} \quad (8)$$

2.4 確率的言語モデルと確率的仮名漢字モデルの統合

すでに述べたように、確率的モデルによる仮名漢字変換において、変換候補に順序関係を与える尤度は、確率的言語モデルによる確率値と確率的仮名漢字モデルによる確率値の積で与えられる。したがって、式 (1) 中の $P(\boldsymbol{y}|\boldsymbol{x})P(\boldsymbol{x})$ は式 (2)(6)(7) から以下のようになる。

$$\begin{aligned} P(\boldsymbol{y}|\boldsymbol{x})P(\boldsymbol{x}) &\approx M_{kk}(\boldsymbol{y}|\boldsymbol{m})M_{c,n}(\boldsymbol{m}) \\ &= \prod_{i=1}^{h+1} M_{kk}(\boldsymbol{y}_i|m_i)M_{c,n}(m_i|c_{i-k}\cdots c_{i-2}c_{i-1}) \end{aligned}$$

クラス n-gram モデルの既知語と未知語の場合分けの式 (3) と最尤推定の式 (4)(5)(8) を代入することで、この積の繰り返しの対象である式は予測される形態素が既知か未知かに応じて以下のように計算される。

1. 既知形態素の場合 ($m_i \in \mathcal{M}_k$)

$$\begin{aligned} &M_{kk}(\boldsymbol{y}_i|m_i)M_{c,n}(m_i|c_{i-k}\cdots c_{i-2}c_{i-1}) \\ &= \frac{N(c_{i-k}\cdots c_{i-1}c_i)}{N(c_{i-k}\cdots c_{i-2}c_{i-1})}\frac{N(m_i, c_i)}{N(c_i)}\frac{N(\boldsymbol{y}_i, m_i)}{N(m_i)} \\ &= \frac{N(c_{i-k}\cdots c_{i-1}c_i)}{N(c_{i-k}\cdots c_{i-2}c_{i-1})}\frac{N(\boldsymbol{y}_i, m_i)}{N(c_i)} \end{aligned}$$

ここで、形態素とクラスの対応関係が多対一なので $N(m_i, c_i) = N(m_i)$ であることを用いている。

2. 未知形態素の場合 ($m_i \notin \mathcal{M}_k$)

$$\begin{aligned} &M_{kk}(\boldsymbol{y}_i|m_i)M_{c,n}(m_i|c_{i-k}\cdots c_{i-2}c_{i-1}) \\ &= \frac{N(c_{i-k}\cdots c_{i-1}c_i)}{N(c_{i-k}\cdots c_{i-2}c_{i-1})}M_{x,t}(m_i)M_{kk}(\boldsymbol{y}_i|m_i) \end{aligned}$$

クラス n-gram モデルでは、未知形態素の表記の出現確率を品詞ごとに計算するための未知語モデルをアルファ

ベット $\mathcal{X}$ 上の文字 n-gram モデルを用いて実現するのが一般的である。このような未知語モデルと形態素よりも小さい単位を用いた仮名漢字モデルとを組み合わせることで、未知語に対しても変換候補を列挙する仮名漢字変換器を構成することが可能である。しかし、未知語に対する正確な仮名漢字変換は困難であり、十分大きな学習コーパスを用いれば実際の使用における未知語率は極めて低いので、未知語に対して変換候補を列挙できるか否かは全体の精度にほとんど影響しないと考えられる。以上の理由から、我々は、アルファベット $\mathcal{X}$ 上の未知語モデルとアルファベット $\mathcal{Y}$ とアルファベット $\mathcal{X}$ の対応関係を記述する仮名漢字モデルをまとめることで得られるアルファベット $\mathcal{Y}$ 上の未知語モデル $M_{y,t}$ を上式の $M_{x,t}$ と M_{kk} の積の代わりに用いることとした。これは以下の式で与えられる近似である。

$$M_{x,t}(m_i)M_{kk}(\boldsymbol{y}_i|m_i) \approx M_{y,t}(\boldsymbol{y}_i)$$

このようなモデルは、学習コーパスの未知語を $\mathcal{Y}^+$ に変換しておき、通常のパラメータ推定を行なうことで容易に得られる。このようなモデルによる未知語の変換結果は入力の記号列と同じである。未知語の多くは入力記号列から一意に変換できる片仮名列であると考えられるので、未知語の変換結果は片仮名で出力することとした。実際、後述する実験において、最大の学習コーパスを用いたモデルから見たテストコーパスの未知語を構成する文字の 33.0% が片仮名であった。これは、我々の選択の妥当性を示す。

3 評価

以上で説明した確率的言語モデルによる仮名漢字変換器を実装し、その変換精度を評価した。この節では、実験の条件とその結果を提示し、それに対する考察を述べる。

3.1 実験の条件

実験には EDR コーパス [8] を用いた。このコーパスの各文は、以下のように、入力記号列（読み）が振られた形態素に分割されている。

１９８７/１９８７/数字 ネン/年/名詞 ノ/の/助詞
アタラシ/新し/形容詞 イ/い/語尾
ケイコウ/傾向/名詞 ハ/は/助詞 、/、/記号
ＩＢＭ/ＩＢＭ/名詞 ガ/が/助詞 ドレ/どれ/名詞
ダケ/だけ/助詞 セイヒン/製品/名詞 ノ/の/助詞
（後略）

まず、コーパスを 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。それぞれのコーパスに含まれる文数と形態素数と文字数は表 1の通りである。

表 1: コーパス

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

確率的言語モデルとしては、文字 2-gram モデルからなる未知語モデルを備えたクラス 2-gram モデルを用いた。これは、文献 [7] に従って構築した。つまり、既知形態素を 2 個以上の部分学習コーパスに現れる形態素とし、これらを平均クロスエントロピーを基準にクラスタリングすることで得られる単語のグループをクラスとした。唯一異なるのは、品詞を区別してからクラスタリングを行なった点である。上述の基準で選択された 59,956 個の既知形態素をクラスタリングした結果、6,156 個のクラスが得られた。未知語モデルは、2節で述べたように、学習コーパスの未知語の入力記号列から文献 [11] に従って構成した。

3.2 評価基準

我々が用いた評価基準は、各文を一括変換することで得られる最尤解と正解の最長共通部分列 (longest common subsequece)[9] の文字数に基づく再現率と適合率である。EDR コーパスに含まれる文字数を N_{EDR} とし、仮名漢字変換結果に含まれる文字数を N_{SYS} とし、これらの最長共通部分列の文字数を N_{LCS} とすると、再現率は N_{LCS}/N_{EDR} と定義され、適合率は N_{LCS}/N_{SYS} と定義される。例として、コーパスの内容と変換結果が以下のような場合を考える。

コーパス

私が長尾真です。

変換結果

渡しが長尾マコトです。

この場合、最長共通部分列は「が長尾です。」の 6 文字であるので、$N_{LCS} = 6$ となる。コーパスに含まれる文字数は 8 であり、変換結果に含まれる文字数は 11 であるので、$N_{EDR} = 8, N_{SYS} = 11$ である。よって、再現率は $N_{COR}/N_{EDR} = 6/8$ となり、適合率は $N_{COR}/N_{SYS} = 6/11$ となる。

3.3 変換精度の評価

仮名漢字変換の精度を評価するために、本論文で提案するクラス 2-gram モデルによる仮名漢字変換エンジンを用いて、テストコーパスの入力記号列の変換候補を列挙し、その第一候補に対して、再現率と適合率を計算した。また、同一のテストコーパスを用いて、市販の仮名漢字変換器の一つである Wnn6[13] による変換の第一候補を同一

表 2: 変換精度

変換エンジン	再現率	適合率
クラス 2-gram モデル	95.07%	93.94%
Wnn6	91.12%	91.17%

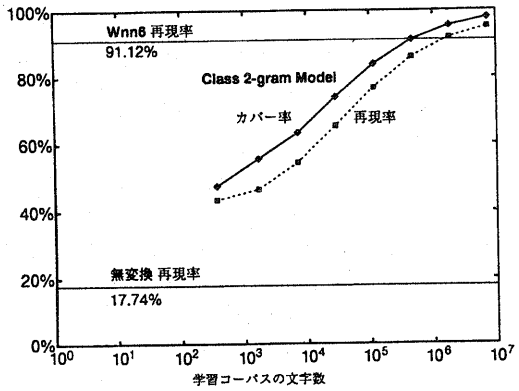

図 1: 学習コーパスの大きさと再現率の関係

の基準で評価した。この結果を表 2に掲げる。この結果から、クラス 2-gram モデルによる仮名漢字変換エンジンの精度が Wnn6 の精度を再現率と適合率の双方で有意に上回っており、確率的モデルによる仮名漢字変換が有効であると結論できる。

図 1と図 2は、学習コーパスの大きさと再現率および適合率およびカバー率の関係である。両方のグラフに、Wnn6 の第一変換候補の精度と入力記号列を第一変換候補とした場合の精度を計算した結果も加えてある。全般的に再現率よりも適合率が低くなっているのは、未知語を入力記号列のまま残すことにしているので、変換候補の文字列が平均的に長くなる傾向があることによる。これらのグラフから以下のことがわかる。学習コーパスの増加により再現率と適合率の双方が有意に向上し、その増加量は最大の学習コーパスを用いた場合の 10^7 文字付近でもまだ大きい。このことから、学習コーパスをさらに大きくすることで容易に精度向上を図れることがわかる。このコストは決して安くはないが、仮名漢字変換市場の大きさを考慮すると、有望な精度向上の方法と考えられる。また、精度とコストの関係が見積もれるという点が、開発者の言語直観を頼りに試行錯誤的に精度向上を図る他ない規則を用いる方法と比較した場合の特筆に値する長所である。また、このグラフに顕在化されているように、再現率と適合率の両方とカバー率とに強い相関関係が見られる。このことから、学習コーパスを大きくする方法以外に、機械可読の百科辞典や人名辞典などの見出し語から、学習コーパスに現れない形態素を外部辞書として付加することによる精度向上が見込めることがわかる。

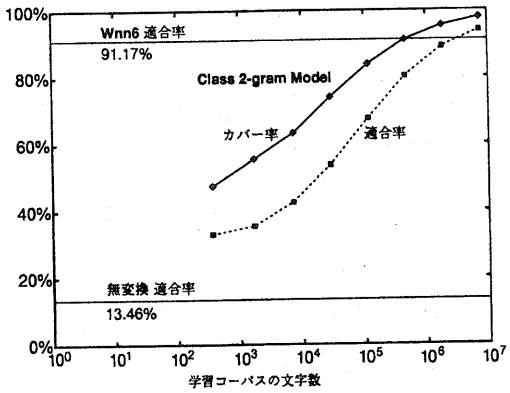

図 2: 学習コーパスの大きさと適合率の関係

3.4 誤変換の例

今後のさらなる精度向上に必要な改良点を明らかにするために、目立った不正解の理由を例を添えて以下に提示する。なお、推定結果と正解とが異なる部分の内の注目している理由による部分には下線を引いてある。

1. クラス 2-gram を越える文脈による同音意義語の選択

   変換結果 １０００平方メートルの敷地に床面積４０００平方メートルのビルが立っていれば、容積率は４００％ということになる。

   コーパス １０００平方メートルの敷地に床面積４０００平方メートルのビルが建っていれば、容積率は４００％ということになる。

   この種の誤りは、言語モデルとしてより長い n-gram モデルを採用することで部分的には解決できる。しかし、本質的な解決には、係り受けなどの連続しない要素間の関係を記述するモデルを採用する必要がある。

2. 未知語

   変換結果 ２月のルーブル合意の基本は、米国の大幅な貿易赤字の シュクゲン が急務だ、という点にある。

   コーパス ２月のルーブル合意の基本は、米国の大幅な貿易赤字の 縮減 が急務だ、という点にある。

   本論文で提案する仮名漢字変換では、未知語は片仮名で出力することにしているため、片仮名以外の文字から構成される未知語に対しては変換を誤ることになる。この

種の誤りに対しては、未知語に対する変換を実現するための未知語モデルを構築することである程度改善できると考えられる。しかし、未知語の仮名漢字変換は困難であると考えられるので、登録語の増加や外部辞書の付加が最良の解決方法であろう。

3. 漢字と平仮名やアラビア数字と漢数字の表記の揺れ

変換結果 2号が海王星の裏側に回り込んだときに電波を出し、その屈折の仕方で海王星の大気やプラズマの状態、ワの有無を調べる。

コーパス 2号が海王星の裏側に回りこんだ時に電波を出し、その屈折のしかたで海王星の大気やプラズマの状態、輪の有無を調べる。

この種の不正解は、一般的に誤りとは考えられない。このような不正解は正解とすべきかもしれないが、表記の揺れとして受理されるか否かには個人差がある。また、どちらも誤りでないにしてもユーザーが意図するのは一つであり、それを出力することが要求される。以上の点を考慮して我々は、コーパスとの文字列比較という人手を介さず客観的な評価方法を採用した。

以上の種類に属さない誤りもあるが、それらは非常に稀である。したがって、短期的な精度向上には上述の1と2の誤りに対処することが最良であると考えられる。

4 おわりに

本論文では、確率的モデルによる仮名漢字変換手法を提案した。具体的に提案した仮名漢字変換エンジンは、確率的言語モデルとしては形態素クラスタリングの結果得られるクラス2-gramモデルを持ち、仮名漢字モデルは形態素単位で入力記号列と日本語表記の対応を記述する。未知語モデルには入力記号の2-gramモデルを用いており、未知語に対しては仮名漢字変換は諦め片仮名列として出力することとした。このような変換エンジンを実装し、テストコーパスに対する第一変換候補を、コーパスに予め与えられた正解との最長共通部分列の文字数に基づく再現率と適合率を計算した結果、確率的モデルによる再現率は95.07%であり、適合率は93.94%であった。これは、市販の仮名漢字変換器の一つであるWnn6の同じテストコーパスに対する再現率(91.12%)と適合率(91.17%)を有意に上回っており、確率的モデルによる仮名漢字変換の有効性が実験的に示された。

謝辞

本研究は文部省科学研究費補助金(課題番号00093069)の助成を受けている。ここに謝意を表する。

参考文献

[1] 牧野寛, 木澤誠. べた書き文の分かち書きと仮名漢字変換 — 二文節最長一致法による分かち書き —. 情報処理学会論文誌, Vol. 20, No. 4, pp. 337–345, 1979.

[2] 吉村賢治, 日高達, 吉田政. 文節数最小法を用いたべた書き日本語文の形態素解析. 情報処理学会論文誌, Vol. 24, No. 1, pp. 40–46, 1983.

[3] 栃内香次, 伊藤太亮, 鈴木康広. 前後連接文字を利用した同音語選択機能を有するかな漢字変換システム. 情報処理学会論文誌, Vol. 27, No. 3, pp. 313–320, 1986.

[4] 本間茂, 山階正樹, 小橋史彦. 連語解析を用いたべた書きかな漢字変換. 情報処理学会論文誌, Vol. 27, No. 11, pp. 1062–1067, 1986.

[5] 高橋雅仁, 吉村賢治, 首藤公昭. 単文内での共起情報を用いた同音語処理. 情報処理学会論文誌, Vol. 36, No. 6, pp. 998–1006, 1995.

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

[7] 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, pp. 2200–2208, 1997.

[8] 日本電子化辞書研究所. EDR 電子化辞書仕様説明書, 1993.

[9] Alfred V. Aho. 文字列中のパターン照合のためのアルゴリズム. コンピュータ基礎理論ハンドブック, I: 形式的モデルと意味論, pp. 263–304. Elseveir Science Publishers, 1990.

[10] 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. PhD thesis, 京都大学工学研究科, 1997.

[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, pp. 2191–2199, 1997.

[12] 森信介, 長尾真. 係り受けを用いた確率的言語モデル. 情報処理学会研究報告, 第96-NL-122巻, 1997.

[13] 日本サン・マイクロシステムズ株式会社. Wnn6 ユーザーマニュアル, 1995.