自然言語処理 128－1
（1998. 11. 5）

文字情報縮退方式を用いた帰納的学習による べた書き文のかな漢字変換手法の性能評価

松原 雅文 † 荒木 健治 †† 桃内 佳雄 † 栃内 香次 ††
† 北海学園大学工学部，†† 北海道大学大学院工学研究科

本手法は，文字情報縮退方式で入力された数字列を漢字かな混じり文に変換するものである．文字情報縮退方式において，0〜9,*,#の１２個それぞれの文字には複数のかなが割り当てられており，読みがな１文字の入力を１ストロークで行うことが出来る．さらに入力された数字列と校正済みの変換結果から帰納的学習により語を獲得するので，辞書が空の状態からでも対象分野に適応した辞書が自動生成される．そして，母音情報縮退により失われた情報を，帰納的学習による適応能力に加え，隣接文字情報と位置推測処理により回復している．

Performance Evaluation for Non-Segmented Kana-Kanji Translation Method Using Inductive Learning with Degenerated Keyword Input

Masafumi Matsuhara† , Kenji Araki†† , Yoshio Momouchi † and Koji Tochinai††
† Hokkai-Gakuen University, †† Hokkaido University

Our proposed method translates the strings of numbers inputed by the degenerated keywords into Japanese sentences. Each keyword in 0, 1, ..., 9, * and # means some Kana characters. Therefor we can input a Kana character by only one stroke. This automatically generates the dictionary adapted to the target field, even if it is empty at first. Because this obtains words using inductive learning to compare the strings of numbers and proofread translation results. And this recovers the lost information for degeneration of a vowel by inductive leaning, information of neighboring character strings and position presuming process.

1 はじめに

近年，携帯端末の性能が飛躍的に進歩している．携帯端末は大別すると携帯性を犠牲にして操作性を重視したものと，その逆に操作性を犠牲にして携帯性を重視したものに分けられる．この後者をさらに発展させ主にインターネット，電子メールを意識した端末として，携帯電話一体型のものが登場している．その大きさは一般的な携帯電話本体を一回り大きくした程度であり，携帯性は非常に優れている [1]．一方，その大きさの制約から，入力に用いられるキーは数字を入力するための１０個のキーとその補助を行うわずかな特殊キーのみである．

これらのキーを用いて日本語文の入力を行うものの中には，携帯電話などでのデータ入力によく用いられる「文字循環指定方式」[2] やポケットベルへのメッセージの送信に使われる「文字コード化方式」[2] を使ったものもあるが，１文字の入力に数回の手数を必要とし，ユーザの負担が大きい．これらの操作性が優れない原因は，本体の大きさの制約から多数のキーを備えることが出来ないことに起因している．これとは別に「音声入力」[3] もあるが，現在の技術では小さな携帯端末に搭載するのは困難であり，その性質上秘密性に問題が残る．携帯端

表 1: 数字とかなの対応関係

1:あいうえおー	2:かきくけこ	3:さしすせそ
4:たちつてと	5:なにぬねの	6:はひふへほ
7:まみむめも	8:やゆよゃゅょ	9:らりるれろ
*:（半）濁音	0:わをん	#:句読点

末で電子メールの送受信など需要が高いことから，少数のキーで日本語文の迅速な入力が可能な手法が望まれる．

そこで我々はこの問題を解決するために「文字情報縮退方式を用いた帰納的学習によるべた書き文のかな漢字変換手法」[4] を提案している．まず，５０音のかな文字を0〜9，#，*の１２個のキーそれぞれに複数対応させ，日本語のあ行を１，か行を２，...のように数字の列で入力する [2]．そして，入力された数字列を漢字かな混じり文に変換する手法である．かなを入力して漢字かな混じり文に変換する「かな漢字変換」に対して，入力が数字列である本手法の変換を「数字漢字変換」と呼ぶ．数字漢字変換において，読みがなはかなに対応した数字である．数字とかなの対応関係を表１に，実行例を図１に示す．

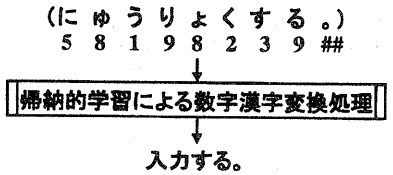

図 1: 実行例

この入力方式を用いることにより，１文字の入力を１ストロークで行うことができるので、入力回数が少なくてすみ、入力ミスを最小限に抑えることができる．また，システムが帰納的学習により語を自動的に獲得するので初期辞書の作成は必要なく，ユーザ個人に合わせた辞書が自動的に生成される特徴がある [5]．ユーザに合わせた辞書が生成されるので，必要最小限の語だけが登録され，変換候補重複の減少にもつながる．

文字情報縮退方式による入力は，母音情報が縮退しており，数字１文字の情報量がかなのそれと比べて少ない．そのため，縮退により失われた情報をいかにして回復するかが問題となる．この問題に対して，本手法では帰納的学習による語の獲得 [5] に加えて，隣接文字列情報と位置推測処理を利用することとした [4]．位置推測処理により，語の獲得時，情報縮退のために発生する曖昧さを極力解消する．隣接文字列情報は n-gram 統計 [6][7] により抽出される．これにより変換候補となる語単体の尤度だけでなく，先行，後続の文字列とのつながりを考慮に入れた変換が可能となっており，同文中に出現する同音異義語にも対処できる．

ここで人間が，ある文字列 A をほかの文字列 B に変換することを考える．

$$A = a_1 \cdot a_2 \cdots a_x \cdots a_n \quad a_x : \text{部分文字列}$$

$$B = b_1 \cdot b_2 \cdots b_y \cdots b_m \quad b_y : \text{部分文字列}$$

$(a_x : b_y)$ の対応関係は統計的な知識によりある程度分かっているものとする．まず，$(a_x : b_y)$ の対応で変換候補単体の尤度の高いものから変換を行う．

$$\cdots a_{x-1} \cdot b_y \cdot a_{x+1} \cdots$$

次に，変換した文字列 b_y との隣接を考慮して文字列 a_{x-1}, a_{x+1} の変換を行う．

$$\cdots b_{y-1} \cdot b_y \cdot b_{y+1} \cdots$$

このようにして人間は変換処理を行っていると考えられ，本手法はこれを模倣したものとなっている．

2 概要

本手法においてユーザは１２個のキーによる文字情報縮退方式を用いて，日本語入力を行

う．この入力方式は，１ストロークがかな文字１文字に対応しているので迅速な入力が可能である．

入力された数字列は，変換処理で語情報辞書と隣接文字列情報辞書を用いて漢字かな混じり文に変換される．語情報辞書は階層構造を持っており，上位階層の語から優先的にあてはめて変換を行う．変換候補が重複した場合，正変換率，誤変換率，隣接文字列情報を利用して，最適な語を決定する．このように，変換は単純な語のあてはめだけではなく，隣接する文字列を考慮したものとなっている．

変換が正しく行われなかった場合，校正処理を行う．人手により変換結果を訂正する．

学習処理では，校正済み変換結果と数字列との比較から，語を抽出する．抽出された語は複数の語から構成されている可能性があるので，さらにそれらを共通，差異部分に分解し，語として辞書に登録する．同時に全文字列を隣接文字列情報辞書に登録する．この情報により，隣接する文字列を考慮した変換が可能となっている．

フィードバック処理では，正変換，誤変換された語はその情報を語情報辞書に持ち，次回からの変換に役立てられる．また，正変換率により語が所属する階層を移動し，辞書の活性化を図っている．このように，変換処理，学習処理，フィードバック処理を繰り返し，変換精度が向上するのと同時に，ユーザに合わせた辞書が生成されていく．

3　処理過程

本手法における処理過程を図２に示す．変換処理，校正処理，学習処理，フィードバック処理の順である．

3.1　変換処理

入力は１２個のキーを用いて，べた書きの数字列で行う．入力例を図３に示す．

本処理過程では入力された数字列を漢字かな混じり文に変換する．入力された数字列に対して，語情報辞書に登録されている語をあてはめて変換を行う．語のあてはめは上位の階層より

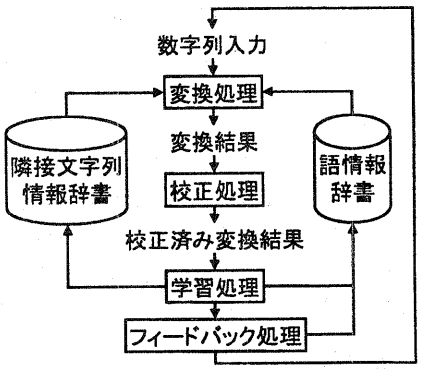

図 2: 処理過程

明日、お伺いします。 ⇒ 134#1122*1373##
１２個のキー ⇒ 3*8152521

図 3: 入力例

読み数字が２文字以上の語から行う．これは１文字語の持つ情報量が２文字以上の語のそれと比べて極端に少ないからである．この情報量の少なさを補うために，１文字語の変換はその両側の語が確定済みのときにのみ行うものとする．変換候補が重複した場合，語の正変換率，誤変換率，隣接文字列存在確率をもとにして尤度評価関数により優先度を決定して最適な語を適用する．

3.2　校正処理

変換処理において変換が正しく行われなかった場合，校正処理が行われる．人手により変換結果を訂正する過程である．

3.3　学習処理

本処理過程では，校正済み変換結果と入力された数字列から語を語情報辞書に獲得する．まず，校正された変換結果と数字列との比較から共通部分，差異部分を抽出する．この抽出された語を S1(Segment One) と呼ぶ．この際，右から左方向の解析，左から右方向の解析を行い，一致したものだけを抽出する．これは誤って獲得される語を極力少なくするためである．抽出された S1 は，さらに共通部分，差異部分に分解され，語として辞書に獲得される．これは抽出された語が複数の語から構成されている可能

性があるからである．共通部分をCS(Common Segment)，差異部分をRS(Remained Segment)と呼ぶ．獲得された語は，それぞれ階層CS, S1, RSに登録され，変換時の語のあてはめもこの順序で行われる．CS, RSに分解されたS1は辞書中より削除される．同時に校正済み変換結果，数字列中の全文字列を獲得し，隣接文字列情報辞書に登録する．この情報は変換処理での尤度評価の際に利用される．

数字列
⇒20281344*2028139##
校正済み変換結果
⇒研究室で研究する。
S1として抽出される共通、差異部分
⇒(2028134:研究室)(4*:で)
(20281:研究)(39:する)
S1同士の共通、差異部分
共通部分
⇒(20281:研究)
差異部分
⇒(34:室)

図 4: 語の獲得例

図4に語の獲得例を示す．数字列と校正済み変換結果において，下線部分が共通部分となり，差異部分とともにS1として抽出される．次にS1同士の共通部分，差異部分を抽出する．共通部分は実線，差異部分は破線である．最終的に獲得される語の読み数字と表記は以下の通りである．

CS (20281 : 研究)
S1 (4* : で) (39 : する)
RS (34 : 室)

3.3.1 位置推測処理

本処理は，学習処理の一部である．学習処理において獲得できない語が存在した場合，平均読み文字数を利用して，対応する共通部分の位置を推測し，語を獲得する．これにより，情報縮退により発生する学習時の曖昧さの解消を目指している．平均読み文字数は，語情報辞書に登録されているすべての語から表記文字数ごとに求められる．

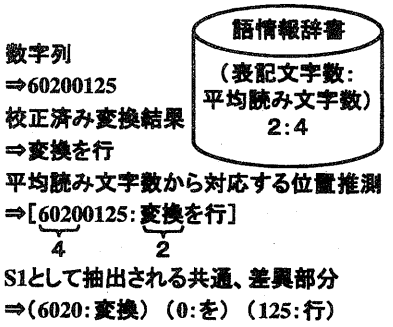

図 5: 位置推測処理例

図5に位置推測処理例を示す．校正済み変換結果の「を」に対応する「0」は数字列中の3箇所に存在するので，一意に決定することができない．そこで「変換」の表記文字数2の語の平均読み文字数を語情報辞書から求める．平均読み文字数4が得られるので，「を」に対応する「0」の位置が決定され，語が獲得される．このようにして，獲得される語数の増大を図り，学習時の曖昧さの解消を目指している．

学習処理において獲得できなかった語は，ほかの曖昧性のない文中から獲得されるが，情報縮退によりこういった獲得されない語は増加すると考えれるので，本処理により対応する．ここで獲得される語は必ずしも正しいとは限らないが，誤って獲得された語はフィードバック処理によりその尤度が下がり，次第に淘汰されていく．

3.4 フィードバック処理

語を語情報辞書に獲得する際に，その語の情報を共に獲得する過程である．獲得する情報は，正変換，誤変換のそれぞれの度数である．これらをそれぞれCF(Frequency of Correct translation)，EF(Frequency of Erroneous translation)と呼ぶ．語の正変換率CR(Rate of Correct translation)，誤変換率ER(Rate of Erroneous translation)を，式(1)(2)に示す．

$$CR = \frac{CF}{CF + EF} \tag{1}$$

$$ER = \frac{EF}{CF + EF} \quad (2)$$

ここで，辞書中の語のうち，正変換率が95%以上になったものをMS(Most certain Segment)，40%未満のものをLS(Least certain Segment)と呼び，それぞれの階層に移動する．また，MS階層の語のうち正変換率が95%未満，LS階層の語のうち正変換率が40%以上となったものは，元の階層に戻される．このように語情報辞書の階層は，上位の階層より，MS, CS, S1, RS, LSとなっており[5]，辞書の効率化，活性化を図っている．階層構造を図6に示す．

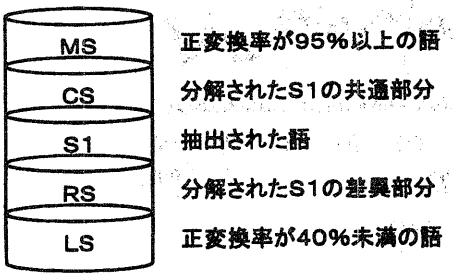

図 6: 語情報辞書の階層

変換時に変換候補が重複した場合，フィードバックされた情報から正変換率，誤変換率，さらに隣接文字列存在確率を用いて尤度評価関数により評価し正しい語を決定する．正変換率，誤変換率は語情報辞書から，隣接文字列存在確率は隣接文字列情報辞書から得られる．隣接文字列存在確率NR(Rate of Neighboring character strings)を，式(3)に示す．

文字列　$\cdots a_{x-1} \cdot a_x \cdot a_{x+1} \cdots$
対応関係　$(a_x : b_y)$

$$\begin{aligned} NR(b_y) &= P_l(b_y) + P_r(b_y) \quad (3) \\ &= P(a_{x-1} \mid b_y) + P(a_{x+1} \mid b_y) \\ &= \frac{P(a_{x-1} \cdot b_y)}{P(b_y)} + \frac{P(b_y \cdot a_{x+1})}{P(b_y)} \end{aligned}$$

尤度評価関数CEF(Credibility Evaluation Function)を式(4)に示す．

$$CEF = \alpha \times NR + \beta \times CR - \gamma \times ER \quad (4)$$
$$\alpha, \beta, \gamma : \text{係数}$$

この式により先行，後続の文字列に隣接する確率と正変換率が高く，誤変換率が低い語の尤度が高くなり，優先されて選択される．

4 評価実験

4.1 実験方法

実験に用いたデータはUNIXのオンラインマニュアルより無作為に抽出した項目である．表2に示す．

表 2: 実験データ

項目名	文字数
ftp	11,000
mail	15,000
cc	8,000
csh	29,000
ld	12,000
ed	15,000
dbx	10,000
合計	100,000

なお，入力に用いた数字列，校正済み変換結果の1文字あたりのエントロピーは数字列が3.5[bit]，校正済み変換結果が8.0[bit]であった．

辞書が空の状態からftp, mail, cc, csh, ld, ed, dbxの順番に実験を行った．式(4)の尤度評価係数は，予備実験より以下の値を用いた．
$\alpha = 10$, $\beta = 1$, $\gamma = 5$

実験の評価に用いた正変換率，誤変換率，未変換率を式(5)(6)(7)に示す．

$$\text{正変換率 [\%]} = \frac{\text{正変換文字数}}{\text{入力文字数}} \times 100 \quad (5)$$

$$\text{誤変換率 [\%]} = \frac{\text{誤変換文字数}}{\text{入力文字数}} \times 100 \quad (6)$$

$$\text{未変換率 [\%]} = \frac{\text{未変換文字数}}{\text{入力文字数}} \times 100 \quad (7)$$

尤度評価係数の最適値を決定するため，予備実験を行った．実験に用いたデータは，UNIXのオンラインマニュアルよりftpの項目の10,000文字である．$\beta = 1$, $\gamma = 5$としてαの値を変化させて実験を行った．αの値に対する平均正変換率を表3に示す．

表 3: 予備実験結果

α	1	2	5	10	20	30	50
平均正変換率 [%]	52.0	51.9	52.7	53.6	52.8	52.7	52.6

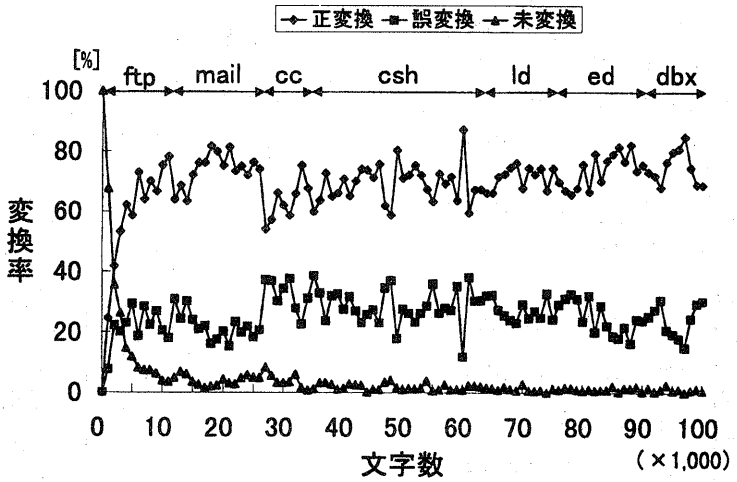

図 7: 変換率の推移

表 4: 平均正変換率

項目名	平均正変換率 [%]
ftp	60.7
mail	74.1
cc	63.5
csh	69.4
ld	71.6
ed	74.0
dbx	75.0
合計	70.2

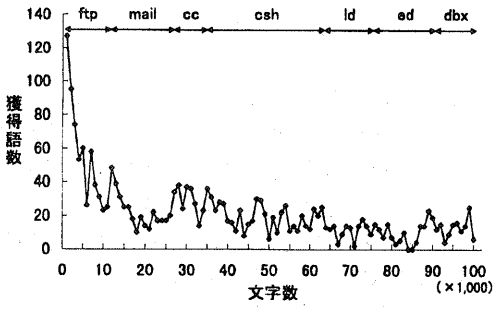

図 8: 獲得語数の推移

4.2 実験結果

変換率の推移を図 7 に，獲得語数の推移を図 8 に，平均正変換率を表 4 に示す．入力文字数の増加に伴い，正変換率が徐々に上昇している．項目の変わり目において，正変換率は一旦下降し，その後また上昇している．

5 考察

図 7 から分かるように，辞書が空の状態から語を獲得することにより，徐々に正変換率が上昇している．項目の変わり目では，図 8 の獲得語数が増えていることから分かるように辞書に未登録の語が多く存在するため，正変換率が下降している．しかしその後，その項目の語を学習し次第に正変換率が上昇している．このように，UNIX のオンラインマニュアルの項目ごとの狭い対象における適応も確認された．正しく変換できない場合として，次のような語が挙げられる．

5.1 辞書に未登録の語

語情報辞書に未登録の語が出現した場合，変換されないで未変換となるか，誤った語があて

はめられ誤変換となる．数字漢字変換は，かな漢字変換よりも曖昧性が高いため，誤変換される割合が高い．しかし，帰納的学習により文脈に適合した語を学習できるので，次第に誤変換の割合は減少する．対象に適応していくことにより，図8から分かるように未登録語は減少し，それに伴い正変換率は上昇している．

5.2 尤度が同程度になる語

読み数字が重複し，出現確率が同程度であり，さらに隣接する文字列が類似する語については本手法の変換処理では対処できない．例えば，(394：すると) (394：された) などである．これらの語は今回用いたデータにおいて一様に出現し，先行，後続する文字列も類似している．出現回数はそれぞれ212回，221回であった．それぞれの語に先行文字列「省略」，後続文字列「ファイル」が隣接する確率は式(3)を用いて0.18，0.16であった．このように，尤度評価の値が同程度となり正しく変換できない場合がある．このような語を正しく変換するためには，離れた位置にある文字列との共起情報を使うことが考えられる．

5.3 表記が重複する語

例えば，(2*81：行) (125：行) などである．これらは読みが「ぎょう」「おこな」と異なるため，変換時のあてはめはまったく別の位置で行われる．しかし，表記が同一であるため，隣接文字列存在確率が同一となる．すなわち，式(3)の分母が2つを合わせたものとなり，尤度を下げているのである．この問題を解決するためには，隣接文字列情報で読みを考慮に入れる必要がある．しかし，学習処理から分かるように，表記と読みを一意に決定できない場合もあり，今後の課題である．

6 おわりに

本稿では，携帯性に優れた小型の携帯端末を想定し，少数のキーで迅速な日本語入力を可能にするため，「文字情報縮退方式を用いた帰納的学習によるべた書き文のかな漢字変換手法」を提案した．数字漢字変換の際に，縮退した情報をいかにして回復するかが問題となるが，帰納的学習による高い適応能力に加え，隣接文字列情報，位置推測処理を利用することにより，評価実験結果の通り，縮退した情報を回復できることを確認した．本手法においては，動的に対象に適応することができるので，汎用性を失うことはない．最終的に80%以上の正変換率が得られ，本手法の有効性が確認された．

今後は，多くの他の分野に対する適応能力の調査を行う予定である．また，さらなる情報の縮退，すなわち本手法では５０音のかな文字を１２個の数字に割り当てているが，さらに少ない文字種に割り当てた場合について研究を進める予定である．

謝辞 なお，本研究の一部は文部省科学研究費 (No.09878070，No.10680367) 及び北海学園大学ハイテク・リサーチ・センター研究費による補助のもとに行われた．

参考文献

[1] PCが握るモバイル活用の鍵，日経バイト５月号，日経BP社，pp.142-165(1998).

[2] 佐藤，東田，林，奥，村上：ＰＢ電話機を利用した日本語入力方式，電子情報通信学会総合大会，D-6-6(1997).

[3] 圧縮／認識／合成を追求する音声処理の徹底研究，Interface ８月号，CQ出版社，pp.81-133(1998).

[4] 松原，荒木，桃内，栃内：文字情報縮退方式を用いた帰納的学習によるべた書き文のかな漢字変換手法，情報処理学会第５７回全国大会，1R-1(1998).

[5] 荒木，高橋，桃内，栃内:帰納的学習を用いたべた書き文のかな漢字変換，電子情報通信学会誌，Vol.J79-D-Ⅱ，No.3，pp.391-402(1996).

[6] 長尾：自然言語処理，岩波書店 (1996).

[7] 森，長尾：ｎグラム統計によるコーパスからの未知語抽出，情報処理学会論文誌，Vol.39，No.7，pp.2093-2100(1998).