単文内での共起情報を用いた同音語処理

高橋雅仁† 吉村賢治† 首藤公昭†

仮名漢字変換において，同音語に起因する複数の変換候補から正しい候補を選択する方法として，単文内で格関係を持つ名詞と動詞の共起情報を利用する方法を提案する．本方式で用いる名詞と動詞間の共起情報はコーパスから比較的容易に収集可能であることからシステムの実現性も高く，従来用いられてきた動詞の意味的選択制限を利用する方法と異なり，動詞の入力を待たずに変換を行うことができるという特徴を持つ．本方式による変換実験を1,129文の単文に対して行った結果，14.7%の文について変換候補の優先順位を決定でき，それらの文については81.9%の文で第一候補に正解を得ることができた．入力単文中の動詞が名詞候補と共起しうる動詞の集合に含まれていなければならないという制約を取り除いて，名詞の変換のみを行った場合には，93.3%の文について変換候補の優先順位を決定でき，それらの63.0%で第一候補に正解を得ることができた．単語の使用頻度に基づく同音語処理方式との比較を行った結果，本方式の正解率が文単位で40.3%上回っており，本方式の有効性を確認できた．

Processing Japanese Homonyms Using Information about the Word Co-occurrence in the Simple Sentence

MASAHITO TAKAHASHI,† KENJI YOSHIMURA† and KOSHO SHUDO†

Kana-to-*kanji* (phonogram-to-ideogram) conversion technology is nowadays common in Japanese word processor development. However, the correct conversion without human interaction is still quite difficult because of the existence of many homonyms. We propose, in this paper, a new method to process homonyms on the basis of the co-occurrence relation between a noun and a verb in a sentence. Our method is based on the idea that nouns which co-occur in a simple sentence share the sentence-final verb as a governor, therefore, the most feasible candidates of *kanji* nouns in an input simple sentence are those each of which co-occurs with an identical verb in a simple sentence with the highest frequency. An experiment of *kana*-to-*kanji* conversion by our new method for 1,129 input simple sentences has shown that the conversion is carried out in 93.3% of the input sentences and the accuracy rate is 63.0%. It is clarified that our method is more effective than the ordinary method based on the word occurence frequency.

1. はじめに

日本語入力方式として一般に用いられている仮名漢字変換における問題のひとつに，同音語の存在に起因する複数個の変換候補の中から正しい候補を選択するための同音語処理がある．従来の仮名漢字変換における同音語処理では，直前に使用した単語を優先するLast Used - First Out方式や，単語の使用頻度の学習により使用頻度が高い候補を優先する方法が広く用いられている[1]．しかし，これらの方法では文の意味をまったく考慮していないため，人間が見て不自然と感じる誤った変換結果を生じるという問題がある．この問題に対処するため，用言の格フレームを用いて単文内の意味的な整合性を調べる方法[2]や，隣接単語間の共起関係を調べる方法[3]が提案されている．しかし，用言の格フレームを利用する方法では，用言の格要素となる名詞に対する制約を意味的な階層構造における上位概念で記述するため，個々の単語間の直接的な共起情報が失われてしまう．文献2)で報告されている実験結果では，格フレームの意味的制約による変換精度の改善は1%程度にとどまっている．また，この方法では用言が入力されるまで変換を行うことができないため，用言が文末に位置する日本語の場合，文が最後まで入力されないと変換が開始できない．隣接単語間の共起関係を調べる方法においても，共起関係を意味的な階層構造における上位概念で記述しており，用言の格フレームを利用する方法と同様な問題を持つ．ま

† 福岡大学工学部
Faculty of Engineering, Fukuoka University

た，共起情報の利用が隣接する2単語に限られている．

最近の研究では，文脈の中での意味的な整合性を考慮し，ニューラルネットワークを用いて単語の共起関係を記述する方法[4]や，特定の話題ごとに共起グループ辞書を作成し，それぞれの話題を一意に決定するキーワードを検出すると対応する話題の共起グループ辞書に含まれる単語を優先する方法[5]が提案されている．これらの方法では，共起情報の質的・量的な拡大，および，共起情報の利用範囲の拡大を図っている．すなわち，個々の単語間の共起関係を陽に記述するとともに，共起情報を複数の文，すなわち，文脈の中で取り扱うことを可能としている．しかし，これらの方法によって実用的な仮名漢字変換システムを実現するためには，5万語から10万語におよぶ単語を扱うのに十分なニューラルネットワークや共起グループ辞書を構築することが必要となり，実現性の面で大きな困難がともなうことが予想される．

本論文では，単文内で格関係を持つ名詞と動詞間の共起情報を利用した同音語処理の方法を提案し，その有効性を確認するために行った実験の結果について報告する．本方式で利用している単文内で格関係を持つ名詞と動詞の共起情報はコーパスから収集可能であり，実現性の高い方式であるといえる．また，名詞と動詞間の共起情報を利用しているが，動詞の入力を待たずに変換を行うことができる点も本方式の特徴のひとつである．

以下，2章で本方式で利用する共起情報の定義，および，共起情報を用いた同音語処理における基本的な考え方を示し，3章で同音語処理のアルゴリズムを示す．続いて，4章で実験内容とその結果を示し，実験結果について考察する．

2. 共起情報と同音語処理

2.1 共起情報

筆者らは文における単語の共起のうち，単文中での動詞とその格要素である名詞の共起を“近い共起”，近い共起以外の一般の単語の共起を“遠い共起”とよんで区別しているが[6]，本論文で提案するアルゴリズムでは近い共起だけを利用しているので，以下の記述では特に区別する必要がない限り，近い共起のことを単に共起とよぶ．また，単文中で動詞とその格要素として共起する名詞の関係を格関係とよぶ．

単文において格関係を持つ名詞と動詞間の共起情報を，名詞 n，格助詞 c および格助詞 c をともなった名詞 n と単文内で共起する動詞の集合 S_v からなる3項組 (n, c, S_v) の集合 Σ_N で与える．ここで，c は格助詞“が”，“を”，“に”，“へ”，“と”，“から”，“より”，“で”のいずれかである．また，

$$S_v = \{(v_1, f(n, v_1)), (v_2, f(n, v_2)), \cdots, (v_l, f(n, v_l))\}$$

で，$v_i\,(i=1,2,\cdots,l)$ は動詞，$f(n, v_i)\,(i=1,2,\cdots,l)$ は格助詞 c をともなった名詞 n が動詞 v_i と共起する出現頻度で，格助詞 c をともなった名詞 n と動詞 v_i がコーパス中の単文に出現した回数 $k_i\,(i=1,2,\cdots,l)$ から次の計算式で求めたものである．

$$f(n, v_i) = k_i / K \tag{1}$$

$$K = \sum_n \sum_c \sum_{j=1}^{l} k_j \tag{2}$$

ここで，K はコーパスから抽出した共起データの総数である．

2.2 同音語に対する優先順位の計算

本論文では，仮名漢字変換の対象とする文は次のような構造を持つ単文とする．

$$N_1 \cdot c_1 \quad N_2 \cdot c_2 \quad \cdots \quad N_m \cdot c_m \quad V$$

ここで，$N_i\,(i=1,2,\cdots,m)$ は仮名表記の名詞，$c_i\,(i=1,2,\cdots,m)$ は格助詞，V は仮名表記の動詞を表す．

この節では，本論文で提案する同音語処理方式の基本的な考え方を，簡単のため $m=2$ の場合について説明する☆．入力として

$$N_1 \cdot c_1 \quad N_2 \cdot c_2 \quad V$$

の形式の単文を考え，名詞 N_1，N_2 と動詞 V の変換候補の組合せの1つを

$$[n_1, n_2, v]$$

と書く．このとき，単文中に格助詞 c_1 をともなった名詞 n_1，格助詞 c_2 をともなった名詞 n_2 および動詞 v が同時に現れる確率を格助詞を省略して $P(n_1, n_2, v)$ と書くと，

$$P(n_1, n_2, v) = P(n_1, v) P(n_2 | n_1, v) \tag{3}$$

である．なお，以下では“格助詞 c_i をともなった名詞 n_i”を単に“名詞 n_i”と記述する．ここで，$P(n_1, v)$ は名詞 n_1 と動詞 v が同時に出現する確率，$P(n_2|n_1, v)$ は名詞 n_1 と動詞 v が同時に出現したときに名詞 n_2 が出現する条件付き確率である．基本的には式(3)の値が大きい候補の組合せに高い優先順位を与えるのが望ましいと考えられるが，2語以上で条件の付いた条件付き確率，ここでの $P(n_2|n_1, v)$ などを実際に求めるのは容易ではない．そこで，$P(n_2|n_1, v)$ の代わりに $P(n_2|v)$ を用いた次式(4)を評価の基本式とする．

☆ 同様な議論は一般の m についても展開できるが，実際の単文における格要素の個数は2〜3個の場合が多い．

$$Q_1(n_1,n_2,v)=P(n_1,v)P(n_2|v) \quad (4)$$

すなわち，動詞を n 項述語とし，名詞のシソーラスなどを用いてその格要素に対する選択制限を行う方式に対して，n 項関係を名詞と動詞間の単語としての2項関係（共起）に分解して，確率的にとらえるのが本論文の立場である．ただし，ここでは N_2 の n_2 以外の変換候補 n_2' について，$P(n_2|n_1,v)>P(n_2'|n_1,v)$ かつ $P(n_2|v)<P(n_2'|v)$ となる場合などについては無視している．

ベイズの定理より

$$Q_1(n_1,n_2,v)=\frac{P(n_1,v)P(n_2,v)}{P(v)} \quad (5)$$

となるので，動詞の出現確率は単語によらず一定であると仮定すると

$$Q_1(n_1,n_2,v)\propto P(n_1,v)P(n_2,v) \quad (6)$$

となる☆．この式から，変換候補の組合せ $[n_1,n_2,v]$ の尤度を計算する評価関数 $e_1(n_1,n_2,v)$ を次のように定義する．

$$e_1(n_1,n_2,v)=P(n_1,v)P(n_2,v) \quad (7)$$

$e_1(n_1,n_2,v)$ の値が大きい変換候補の組合せ $[n_1,n_2,v]$ に高い優先順位を与える．

次に，入力仮名列から文末の動詞を取り除いた場合について考える．2組の名詞・格助詞対からなる入力仮名列

$$N_1\cdot c_1 \quad N_2\cdot c_2$$

における名詞 N_1 と N_2 の変換候補の組合せの1つを

$$[n_1,n_2]$$

と書く．このとき，単文中に名詞 n_1 と名詞 n_2 が同時に現れる確率 $P(n_1,n_2)$ は

$$P(n_1,n_2)=\sum_v P(n_1,n_2,v) \quad (8)$$

であるから，評価の基本式を

$$Q_2(n_1,n_2)=\sum_v Q_1(n_1,n_2,v) \quad (9)$$

とし，動詞の出現確率は単語によらず一定であると仮定すると，式(6)より

$$Q_2(n_1,n_2)\propto\sum_v P(n_1,v)P(n_2,v) \quad (10)$$

となる．そこで，変換候補の組合せ $[n_1,n_2]$ の尤度を計算する評価関数 $e_2(n_1,n_2)$ を次のように定義する．

$$e_2(n_1,n_2)=\sum_{j=1}^{r}P(n_1,v_j)P(n_2,v_j) \quad (11)$$

ここで，$v_j(j=1,2,\cdots,r)$ は名詞 n_1 と名詞 n_2 をそれぞれ格助詞 c_1 と c_2 が指定する格要素として取りうる動詞である．$e_2(n_1,n_2)$ の値が大きい変換候補の組合せ $[n_1,n_2]$ に高い優先順位を与える．

以上の式(7)，(11)における確率 $P(n_i,v)\,(i=1,2)$ も正確に求めることは難しいが，2.1節で定義した Σ_N で与えられる $f(n_1,v)$，$f(n_2,v)$ はそれらの粗い近似となっている．したがって，実際には式(7)，(11)は近似式(12)，(13)として適用する．

$$e_1(n_1,n_2,v)\cong f(n_1,v)\cdot f(n_2,v) \quad (12)$$

$$e_2(n_1,n_2)\cong\sum_{j=1}^{r}f(n_1,v_j)\cdot f(n_2,v_j) \quad (13)$$

なお，式(2)で定義される K は定数であるため，式(12)における $f(n_1,v)$ と $f(n_2,v)$，式(13)における $f(n_1,v_j)$ と $f(n_2,v_j)$ は，それぞれ，k_1 と k_2，k_{1_j} と k_{2_j} に置き換えてもかまわない．

3. 同音語処理アルゴリズム

3.1 アルゴリズム

ステップ1 $m=0$，$T_i=\varepsilon\,(i=1,2,\cdots)$ とする．

ステップ2 入力仮名列が空であればステップ4に進む．そうでなければ入力仮名列の左側から1文節分の仮名列，すなわち $N\cdot c$ または V を読み込み，読み込んだ仮名列を入力仮名列から削除する．

ステップ3 ステップ2で読み込んだ1文節分の仮名列に対して，$N\cdot c$ の N または V に対するすべての変換候補 $w_k\,(k=1,2,\cdots)$ を求める．

m の値を1増やす．

すべての同音語の候補 $w_k\,(k=1,2,\cdots)$ について以下の処理を行う．

(1) w_k が名詞であれば，共起情報 Σ_N から (w_k,c,S_k) を検索し，T_m に2項組 (w_k,S_k) を追加する．

(2) w_k が動詞であれば，T_m に2項組のデータ $(w_k,\{(w_k,1)\})$ を追加する．

ステップ2に進む．

ステップ4 $T_i\,(i=1,2,\cdots,m)$ より，$|\bigcap_{i=1}^{m}S_i|$ の値が最大となる組合せ

$$(w_1,S_1)(w_2,S_2)\cdots(w_m,S_m)$$
$$(w_i,S_i)\in T_i\,(i=1,2,\cdots,m)$$

を求める．このとき単語列 $w_1,w_2,\cdots,w_m$ が，入力仮名列に対する最も確からしい仮名漢字変換結

☆ 本研究の目的は同音語処理における動詞と名詞の共起情報の有効性を確認することであるため，ここで示す方式では動詞単独の出現確率は利用していないが，これらを評価関数 Q_1 に反映する方法も考えられる．

果となる．ここで $\bigcap_{i=1}^{m} S_i$ は次のように定義される集合で，

$$\bigcap_{i=1}^{m} S_i = \left\{ \left(v, \prod_{i=1}^{m} f(w_i, v) \right) \middle| \right.$$
$$(v, f(w_1, v)) \in S_1 \wedge \cdots$$
$$\left. \wedge (v, f(w_m, v)) \in S_m \right\}$$

$|\bigcap_{i=1}^{m} S_i|$ は，集合 $\bigcap_{i=1}^{m} S_i$ のすべての要素について求めた第 2 項目の総和を表しており，$F = \prod_{i=1}^{m} f(w_i, v)$ とすると次のように定義される．

$$\left| \bigcap_{i=1}^{m} S_i \right| = \sum_{(v,F) \in \bigcap_{i=1}^{m} S_i} F$$

ここで，ステップ 3 の (2) で T_m に追加する 2 項組のデータ $(w_k, \{(w_k, 1)\})$ は，入力仮名列の末尾に動詞が存在する場合の処理と存在しない場合の処理を同一のアルゴリズムで記述するために導入したものである．

入力仮名列の末尾に動詞が存在しない場合，ステップ 4 の集合 $\bigcap_{i=1}^{m} S_i$ の要素の第 2 項目 $\prod_{i=1}^{m} f(w_i, v)$ は，式 (13) 右辺の $f(n_1, v_j) \cdot f(n_2, v_j)$ を一般化したものに相当するので，$|\bigcap_{i=1}^{m} S_i|$ は，式 (13) を一般化したものに一致している．また，入力仮名列の末尾に動詞が存在する場合は，ステップ 3 の (2) で T_m に $(w_k, \{(w_k, 1)\})$ が追加されており，$|\bigcap_{i=1}^{m} S_i|$ の定義における v が固定されるので，$|\bigcap_{i=1}^{m} S_i|$ は $\prod_{i=1}^{m} f(w_i, v)$ となり，式 (12) を一般化したものに一致する．

3.2 同音語処理の例

上記のアルゴリズムを用いた同音語処理の例を示す．仮名漢字変換の対象とする入力仮名列を

“かわに　はしを”

とする．ここで，“かわ”，および，“はし” は，それぞれ，以下の同音語を持つものとする．

“かわ”： 川
　　皮
“はし”： 橋
　　箸

また，助詞 “に” をともなった “川”，“皮” および助詞 “を” をともなった “橋”，“箸” に対する共起情報としては以下のものが準備されているものとする．

(川，に，{(行く，8/66)，
　　(架ける，6/66)，
　　(落とす，5/66)})
(皮，に，{(塗る，6/66)，
　　(触る，3/66)})
(橋，を，{(渡る，9/66)，
　　(架ける，7/66)，
　　(作る，6/66)，
　　(落とす，4/66)})
(箸，を，{(使う，7/66)，
　　(落とす，3/66)，
　　(作る，2/66)})

前節のアルゴリズムを適用し，それぞれの同音語候補の組合せに対する評価値を求めると次のようになる．

川に橋を　0.014{ 架ける，落とす }
川に箸を　0.003{ 落とす }
皮に橋を　0.0{}
皮に箸を　0.0{}

したがって，入力仮名列に対する最も確からしい単語の組合せは，“川に橋を” となる．

4. 実　　験

4.1 単語辞書と共起データの作成

3 章で示したアルゴリズムの有効性を確認するための実験を時事問題に関連した単文を対象に行った．ここでは，実験に使用した単語辞書と共起データの作成について述べる．

4.1.1 名詞ファイルの作成

時事問題に関して記述した文章から名詞を抽出してファイルを作成した．ここで抽出した 190 個の名詞の中で同音語が存在するものについて，それらをファイルに追加した．この段階で名詞の総数は 323 語である．このファイルを名詞ファイルと呼ぶ．

4.1.2 共起ファイルの作成

名詞ファイル中の名詞 323 語に対する共起情報を 4 通りの方法で作成した．得られた共起情報のファイルをそれぞれ共起ファイル A，B，C，D とよぶ．共起情報のレコード形式は次のとおりである．

［名詞，格助詞，動詞，出現頻度］

これを共起レコードとよぶ．

各共起ファイルの作成方法を次に示す．

(1) 共起ファイル A（レコード総数=15,856 レコード，名詞 1 個あたり平均 49 レコード）
名詞ファイルに含まれるすべての名詞に対して，8 種類の格助詞と組み合せてできる名詞・格助詞対を格要素としてとる動詞を思いつくだけ列挙する作業を 10 人が独立に行い，その結果をマージした．そのときに等しい共起データがあ

表1 実験1（動詞を含む場合の変換）の結果
Table 1 Results for sentences with a verb.

	共起ファイルA	共起ファイルB	共起ファイルC	共起ファイルD
変換文数	111/1129 (9.8%)	166/1129 (14.7%)	218/1129 (19.3%)	383/1129 (33.9%)
正解文数	90/111 (81.1%)	136/166 (81.9%)	167/218 (76.6%)	243/383 (63.4%)
変換文節数	334/3444 (9.7%)	500/3444 (14.5%)	658/3444 (19.1%)	1157/3444 (33.6%)
正解文節数	312/334 (93.4%)	469/500 (93.8%)	603/658 (91.6%)	982/1157 (84.9%)

れば，それらの個数を出現回数として記録した．

(2) 共起ファイルB（レコード総数=25,665レコード，名詞1個あたり平均79レコード）
名詞ファイルに含まれるすべての名詞に対して，共起ファイルAと同様の共起レコードをEDR電子化辞書日本語共起辞書第2版[7]から抽出し(11,294レコード)，共起ファイルAとマージして作成した．

(3) 共起ファイルC（レコード総数=177,063レコード，名詞1個あたり平均548レコード）
共起ファイルBに含まれるすべての共起レコード中の動詞について分類語彙表[8]からその動詞と同位の概念に属するすべての動詞（等しい分類番号と段落番号を持つ動詞）を取り出し，それらの動詞を元の共起レコード中の動詞と置き換えて新たな共起レコードを生成し，共起ファイルBとマージして作成した．

(4) 共起ファイルD（レコード総数=876,618レコード，名詞1個あたり平均2,713レコード）
共起ファイルCの作成と同様にして，共起ファイルBの共起情報を機械的に増加させた．ただし，新たな共起レコードの生成は，同位ではなく1つ上位の概念に属する動詞（等しい分類番号を持つ動詞）に対して行った．

4.1.3 変換実験用単文ファイルの作成

共起ファイルの作成に関与していない3人の作業者により，名詞ファイルに含まれる323語の名詞と8種類の格助詞および任意の動詞を用いて変換実験用の単文を作成した．作成した単文の総数は1,129文で，1つの単文の文節数は平均3.1である．単文は単語単位で分かち書きし，変換実験の入力データとして使用する仮名表記のファイルと正解データとして使用する漢字仮名混じり表記のファイルを作成した．

4.1.4 単語辞書の作成

仮名漢字変換用の自立語辞書☆中の動詞23,912語と名詞ファイル中の名詞323語に対して，単語の仮名表記，漢字仮名混じり表記および品詞で構成される単語レコードを作成し，変換実験用の単語辞書を作成した．

4.2 実験結果

実験としては，本論文で提案した方式の変換精度を確認するための実験として，仮名表記の単文ファイルに含まれるすべての単文（3,444文節）を変換対象とした実験と各単文から動詞を取り除いたもの（2,315文節）を変換対象としたとした実験をそれぞれ4種類の共起ファイルを用いた場合について行った．また，共起情報の有効性を確認するために，単語に付与した同音語間の優先順位だけを利用する方式と変換精度の比較実験を行った．各実験の結果を以下に示す．

4.2.1 動詞を含む場合（実験1）

実験の結果を**表1**に示す．たとえば，共起ファイルBを用いて変換を行った場合，1,129文（3,444文節）中166文［14.7%］(500文節［14.5%］)で同音語に対する優先順位の決定を行うことができた．残りの963文については，共起情報の不備のために優先順位の決定を行うことができなかった☆☆．優先順位を決定できた166文（500文節）のうち136文［81.9%］(469文節［93.8%］)については第一候補に正しい変換結果が得られた．

4.2.2 動詞を含まない場合（実験2）

実験の結果を**表2**に示す．たとえば，共起ファイルBを用いて変換を行った場合，1,129文（2,315文節）中1,053文［93.3%］(2,155文節［93.1%］)で同音語に対する優先順位の決定を行うことができた．残りの76文については，共起情報の不備のために優先順位の決定を行うことができなかった☆☆☆．優先順位を決定できた1,053文（2,155文節）のうち663文［63.0%］(1,716文節［79.6%］)については第一候補に正しい変換結果が得られた．

4.2.3 単語の優先順位だけを用いる方式との比較（実験3）

共起情報を用いた本方式で同音語に対する優先順位の決定を行うことができた動詞を含む166文に対して，

☆ (株)エー・アイ・ソフト製WX2の自立語辞書

☆☆ 単文中の名詞の変換候補と動詞の変換候補の共起に関する情報が共起ファイルにないため評価関数 e_1 を適用できなかった．

☆☆☆ 単文中の名詞の変換候補と共起しうる動詞の集合の積集合が空集合になったため評価関数 e_2 を適用できなかった．

表2 実験2（動詞を含まない場合の変換）の結果
Table 2 Results for sentences without a verb.

	共起ファイルA	共起ファイルB	共起ファイルC	共起ファイルD
変換文数	1032/1129 (91.4%)	1.053/1129 (93.3%)	1071/1129 (94.9%)	1104/1129 (97.8%)
正解文数	651/1032 (63.1%)	663/1053 (63.0%)	667/1071 (62.3%)	711/1104 (64.4%)
変換文節数	2112/2315 (91.2%)	2155/2315 (93.1%)	2191/2315 (94.6%)	2261/2315 (97.7%)
正解文節数	1688/2112 (79.9%)	1716/2155 (79.6%)	1741/2191 (79.5%)	1819/2261 (80.5%)

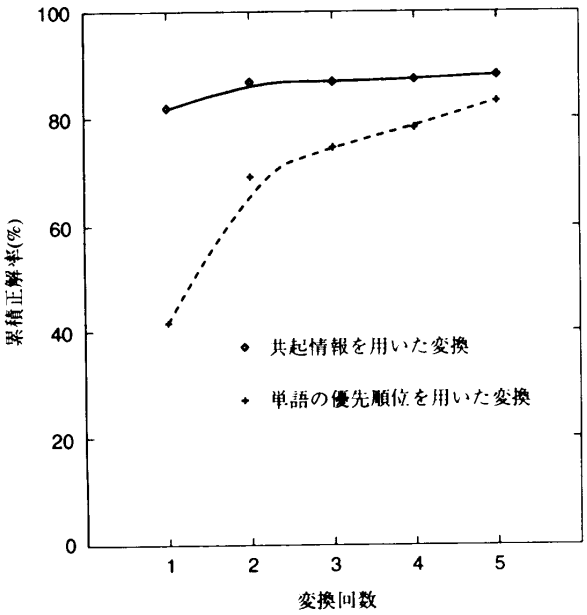

図1 動詞を含む入力文の累積正解率
Fig. 1 Accuracy rate for sentences with a verb.

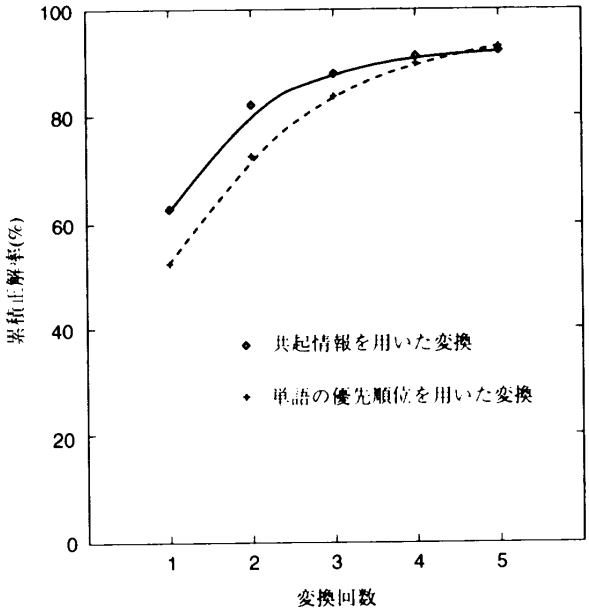

図2 動詞を含まない入力文の累積正解率
Fig. 2 Accuracy rate for sentences without a verb.

共起ファイルBを用いて変換した場合と，共起情報を用いずに単語に付与した同音語間の優先順位だけを利用した場合について変換精度の比較を行った．同音語間の優先順位については，普段使用して単語の使用頻度をある程度学習しているWnnで個々の単語を変換することによって調査した．変換精度の比較は，優先順位の高いものからi番目までの変換結果に正しい変換が含まれているときに正解としたときの正解率（累積正解率）を$i \leq 5$に対して比較した．iは仮名漢字変換における変換キーの操作回数に相当する．同音語間の優先順位だけを利用する方式では，優先順位が1位の単語の組合せを第一候補とし，以下の順番は，左から順番に1位の単語を1つだけ2位の単語で置き換えた組合せ，1位の単語を2つだけ2位の単語で置き換えた組合せというように機械的に付けた．変換結果の個数と文単位での累積正解率の変化を**図1**に示す．また，同様の比較を，動詞を取り除いた1,053文に対して行った場合の文単位の累積正解率の変化を**図2**に示す．なお，5個以上の変換候補を持つ文は1,053文中277文で，共起情報を用いて5位までの優先順位を付けることができた文はそのうちの164文であった．

共起情報を用いて変換を行った場合の第一変換候補の正解率は，動詞を含む単文の変換で81.9%，動詞を含まない場合で63.0%であるが，同音語間の優先順位だけで変換候補の優先順位を決定した場合の第一変換候補の正解率は，動詞を含む単文の変換で41.6%，動詞を含まない場合で52.4%であった．共起情報を用いた変換を行う場合の方が動詞を含む単文の変換で40.3%，動詞を含まない場合で10.6%正解率が高く，共起情報を用いた同音語処理方式が有効であることが確認された．このことは，共起情報を用いる本方式の方が正しい変換結果を得るまでに必要な変換操作が少なくて済むことを意味している．これは，共起情報を用いることによって，入力仮名列に対する変換候補の中で意味的に整合性の高い候補が優先的に出力されるためであると考えられる．**表3**に，共起情報を用いて変換を行った場合と，共起情報を用いずに同音語間の優先順位に基づいて変換を行った場合の変換結果の例を示す．この例からも共起情報を用いた場合には，変換誤りの不自然さが少ない候補から優先的に出力されていることが分かる．

表3 変換結果の例
Table 3 An example of conversion.

	共起情報使用	頻度情報使用
第1候補	"国連 で 協議 を"	"国連 で 競技 を"
第2候補	"国連 で 教義 を"	"国連 で 協議 を"
第3候補	"国連 で 競技 を"	"国連 で 狭義 を"
第4候補	"国連 で 狭義 を"	"国連 で 教義 を"

表4 格助詞の照合に関する比較実験の結果
Table 4 Effect of matching case particles.

	照合あり	照合なし
変換文数	166/1129 (14.7%)	306/1129 (27.1%)
正解文数	136/167 (81.9%)	213/306 (69.6%)
変換文節数	500/3444 (14.5%)	929/3444 (27.0%)
正解文節数	469/500 (93.8%)	822/929 (88.5%)

4.3 共起情報の不備を補う方法

実験1の結果から入力仮名列に動詞を含む場合，その動詞が共起情報に含まれていないために変換候補の決定が行えない場合が多いことが分かる．実用のシステムに本論文の方式を採用するには共起情報をより充実させなければならないが，収集それ自体は容易でも，自然言語の持つ量的な性質から，完璧な共起情報を準備しておくことが難しいのも事実である．そこで，この不備を補う方法を考えておく必要があるが，ここでは，以下の3種類の方法について予備的な実験を行い，検討した．

(1) 入力仮名列の動詞を変換対象に含めない方法
(2) 共起情報中の格助詞の照合を行わない方法
(3) シソーラスを用いて共起情報の拡張を行う方法

その結果，以下のようにこれらの方法では，入力仮名列に動詞を含めずに変換処理を施すことが最も有効であることが分かった．なお，本節の記述における正解率と変換結果が得られる割合は文単位で計算したものである．

4.3.1 入力仮名列の動詞を変換対象に含めない方法

共起ファイルBを用いて行った実験1と実験2の結果を比較すると次のことがいえる．

入力仮名列に動詞を含む場合，変換結果が得られる割合は14.7%，得られた変換結果の正解率は81.9%であり，入力文全体に対する正解率は12.0%である．これに対して，入力仮名列に動詞を含まない場合，得られた変換結果の正解率は63.0%となり18.9%低くなるが，変換結果が得られる割合は93.3%となって大幅に高くなり，入力文全体に対する正解率も58.7%となり大きく向上している．

このことから，入力仮名列に動詞を含む場合であっても優先順位を決定できない場合には，入力仮名列に動詞を含まない場合の変換処理を試みることが有効であると予想される☆．

4.3.2 共起情報中の格助詞の照合を行わない方法

実験1では，入力仮名列に対する変換候補と共起情報との照合処理を名詞，格助詞，動詞の3項すべてについて行っている．これに対して，格助詞の照合を行わない場合は，共起情報中の格助詞を無視するため，利用できる共起情報が増加する．共起ファイルBを用いて動詞を含む入力仮名列に対して実験した結果を**表4**に示す．表4から，格助詞の照合を行わない場合，変換結果が得られる割合は27.1%，得られた変換結果の正解率は69.6%となり，入力文全体に対する正解率は18.9%で，照合を行う場合の約1.5倍となっている．

このように，共起情報を用いた同音語処理において，格助詞の照合を行わないようにすることも，共起情報の不備を補う上で多少の効果があることが分かる．

4.3.3 シソーラスを用いて共起情報の拡張を行う方法

上記の動詞を含まない場合の変換を行う方法，および，格助詞を照合せずに変換を行う方法は，共起情報を用いた同音語処理の制限を緩めることにより，変換処理で使用できる共起情報を増加させ，共起情報の不備を補う方法であった．一方，シソーラスを用いて既存の共起情報を拡張させる方法が考えられる．すなわち，実験1，実験2において共起ファイルCおよびDを使用した場合がこれにあたる．実験1の動詞を含む場合について，共起ファイルBを用いた場合との比較を行うと，まず，共起ファイルCを使用した場合，共起レコード数は約7倍（名詞1個あたり平均550レコード程度）になっているが，このとき，変換結果が得られる割合は19.3%となりやや増加し，得られた変換結果の正解率は76.6%となって5.3%低くなっている．入力文全体に対する正解率も14.8%で，やや向上する程度である．共起ファイルDを使用した場合，共起レコード数は約34倍（名詞1個あたり平均2,700レコード程度）になる．このとき，変換結果が得られる割合は33.9%となりほぼ倍増する．また，得られた変換結果の正解率は63.4%となり18.5%低くなるが，入力文全体に対する正解率は21.5%となり，1.8倍になっている．

この結果から，共起情報の不備を補う方法として，シソーラスを用いて共起情報を増加させる方法は，格

☆ ただし，この場合には入力仮名列中の動詞に対する変換をどうするかという問題が残る．

助詞を照合せずに変換を行う方法と同程度の効果があることが分かる．しかし，シソーラスを用いて名詞1個あたり2,700レコードの共起情報を準備しても文単位で33.9%の変換結果しか得られていない．このことは，シソーラスを用いて共起情報を拡張することにも限界があることを予想させる．これは，今回の実験に利用した分類語彙表における動詞の分類が，本論文の方式で必要としている動詞の格構造に基づいて行われていないことが原因であると考えられる．

4.4 変換誤りの分析

実験1で共起ファイルBを用いた場合に第一候補が誤っていた30文について，変換誤りのパターンを分析した結果，以下のことが分かった．

(1) 変換誤りの約40%は，格助詞“が”を含む文節において発生している．これは，主格となる名詞は，他の格要素となる名詞と比較して，より広い範囲の動詞と共起し，共起情報の網羅的な収集が困難となるためであると考えられる．

(2) 変換誤りを生じた文の約50%は，“ある”，“なる”，“行う”，“見る”等，名詞と共起しないと意味が限定できない基本動詞を含んでいる（“中止に なる”，“終結を 見る”等）．このような動詞に対しては，名詞まで含んだ表現をまとめて動詞として扱うなどの対策が必要になる．これは仮名漢字変換における慣用表現データの利用とも関連する．

5. おわりに

本論文では，共起情報を用いた同音語処理方式とその実験結果について報告した．本方式は，コーパスから比較的容易に収集可能な単文内の名詞と動詞間の共起情報を利用して処理を行うことを特徴としている．

1,129文の単文に対して，名詞1個あたり平均80レコードの名詞，動詞間の共起情報を用いて変換実験を行った結果，14.7%の文について変換候補の優先順位を決定でき，それらの文については81.9%の文で第一候補に正解を得ることができた．また，入力単文中の動詞が共起動詞として名詞候補に共有されなければならないという制約を取り除いた場合には，93.3%の文について変換候補の優先順位を決定でき，それらの63.0%で第一候補に正解を得ることができた．単語の頻度情報に基づく同音語処理方式に比べ，いずれの場合も共起情報を用いた同音語処理方式の方が優れていることが確認できた．また，正しい変換結果を得るまでの変換回数も少なくて済むことも分かった．

なお，共起情報が完全でない場合には，動詞を含む単文の変換に本方式を適用できる割合が低くなるという問題点がある．この問題への対応策としては，格助詞の照合を行わないことにより共起情報の適用に関する制約を緩める（4.3.2節）方法やシソーラスを用いて共起情報を拡張する（4.3.3節）方法が考えられるが，これらの方法を用いた場合，2項関係自体の質が低下したり，式(4)の導出において無視したような現象が顕著になって2項関係による n 項関係の近似が悪くなり，十分な効果が得られないことが今回の実験結果から分かった．この問題への処理アルゴリズム上の対応策としては，動詞を含む単文に対して変換結果が得られなかった場合には，動詞を含まない場合の変換処理を施し，名詞の変換候補を得るという方法が有効である．このとき，動詞に対する変換処理については別途検討を行う必要がある．

本方式で変換が不能なものについては，単語単独の出現頻度や慣用表現を考慮に加えた方式への発展が考えられる．また，本方式を実際の文に即した文構造へ対応できるよう拡張すること，仮名漢字変換システムに共起情報の学習機能を取り入れて，システムを使用する過程でユーザの利用状況に適応した共起情報を収集・利用すること，さらに，単文の範囲を超え，文脈の中での意味的な整合性を考慮した共起情報を用いる方式等についても検討を進める予定である．

謝辞 本研究を行うにあたり，実験システムの作成などで協力をいただいた株式会社エー・アイ・ソフトの山村隆志氏や福岡大学大学院工学研究科電子工学専攻修士課程2年生の新中剛君をはじめ，共起データや実験用単文の作成に協力していただいた福岡大学工学部電子制御実験室の方々に感謝の意を表します．

参考文献

1) 長尾真監修：日本語情報処理，電子通信学会(1984).
2) 大島義光，阿部正博，湯浦克彦，武市宣之：格文法による仮名漢字変換の多義解消，情報処理学会論文誌，Vol.27, No.7, pp.679-687 (1986).
3) 本間 茂，山段正樹，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).
4) 小林 勉，中里茂美，長崎秀紀：ニューロかな漢字変換の実現，東芝レビュー，Vol.47, No.11, pp.868-870 (1992).
5) 山本喜大，久保田淳市：共起グループを用いたかな漢字変換，情報処理学会第44回全国大会論文集，pp.189-190 (1992).
6) Takahashi, M., Onomatsu, T., Yoshimura, K. and Shudo, K.: Processing Homonyms Us-

ing the Co-occurrence Relation Between a Noun and a Verb, *Proc. NLPRS'93*, pp.128–135 (1993).
7) 日本電子化辞書研究所：共起辞書（第2版）(TR-043) (1994).
8) 国立国語研究所：分類語彙表［フロッピー版］解説書 (1993).

(平成7年6月30日受付)
(平成8年3月12日採録)

高橋　雅仁（正会員）

昭和31年生．昭和54年福岡大学工学部電子工学科卒業．昭和56年九州工業大学大学院工学研究科情報工学専攻修士課程修了．同年九州松下電器（株）入社．平成5年福岡大学大学院工学研究科博士後期課程（情報・制御システム工学専攻）に社会人学生として進学．日本語ワープロ，機械翻訳システムの研究開発に従事．電子情報通信学会，言語処理学会各会員．

吉村　賢治（正会員）

昭和30年生．昭和53年九州大学工学部電子工学科卒業．昭和55年同大学院工学研究科電子工学専攻修士課程修了．昭和58年同大学院工学研究科電子工学専攻博士後期課程修了．福岡大学工学部教授．工学博士．自然言語処理に関する研究に従事．電子情報通信学会，人工知能学会，言語処理学会，認知科学会各会員．

首藤　公昭（正会員）

昭和18年生．昭和40年九州大学工学部電子工学科卒業．昭和42年同大学院工学研究科電子工学専攻修士課程修了．昭和45年同大学院工学研究科電子工学専攻博士後期課程単位取得後退学．福岡大学工学部教授．工学博士．自然言語処理に関する研究に従事．電子情報通信学会，人工知能学会，言語処理学会，認知科学会，ACL各会員．