OCR 入力された日本語文の誤り検出と自動訂正†

伊東伸泰†† 丸山宏††

著者らは既存の日本語印刷文書をハイパーメディアなどのデータベースに効率よく入力・運用する目的で文書理解システム (DRS: Document Recognition System) を開発中であるが，その機能の1つとして OCR で認識された文字中から日本語文の制約を利用して誤りを検出，オペレータに警告し，可能な場合はより確からしい候補に置き換える後処理を実現した．本後処理は日本語辞書と品詞間接続テーブルを参照して文法的に成立する文字列の候補を生成した後，各単語の品詞，出現頻度，遷移確率，および認識の確からしさに基づいてコストを計算しその値が最良のものから一定値以内の候補パスを選び出す．そして各カラムの文字候補について，自分自身を通る候補パスに付随するコストと他の文字を通るパスのそれから確信度を計算し，その値により当該候補の入れ替えや，オペレータに対する警告を行う．実験によれば後処理なしで 95% 程度の認識率であったデータで認識率が約 99% に向上し，検出されなかった（言い換えれば入れ替え，警告のいずれも行われなかった）誤認識文字は 0.2% 程度にとどまった．候補パスを見出す探索には動的計画法とビームサーチを用いることで，80386(25 MHz) のパーソナルコンピュータ上で約 27 文字/秒の実行速度が得られた．

1. はじめに

既存の日本語文書を入力する方法として OCR はきわめて有力である．しかしながら認識誤りを完全に避けることはほとんど不可能であり，入力後の確認・修正が不可欠である．したがって，入力文書が帳票ではなく1ページ当たり 2,000 字程度は普通であるような一般雑誌等になると，OCR の認識速度よりもこの確認・修正作業の時間で全体の入力効率が左右されることになる[1]．そこでオペレータによるこの作業を補助および(半)自動化する試みが行われてきた．その中で比較的基本的なものは認識結果の確信度を識別時の距離等から算出し，結果が唯一に決められない場合はリジェクトとしてオペレータに警告すると同時に，その前後および周辺の文字から得られる制約をもとに候補文字の中から正解を推定するものである．利用する制約としては文字単位での連接情報[2]や単語としての成立可能性，さらに単語間の接続規則[3]などが報告されている．ところが入力文書の品質がよほどよい場合を除けば，きわめて多くのリジェクトが出力されてしまい，これらの手法が適用し難い場合も多い．そこで認識結果のあいまい度が大きい（候補文字が1個に絞れないことが多い）場合にも適用可能な方法として，各候補文字を組み合わせてできるパスを日本語辞書と単語（言い換えれば品詞）間の接続規則を利用して探索する手法が提案された[4)~6)]．この手法を適用するにあたって考慮しなければならない点としては，つぎのようなことが考えられる．

1. 適用する日本語文法：文字認識では対象となる文書を極端に絞ることは実用的ではないため，なるべく広い範囲の日本語文を受理できることが望ましい．ところが文字認識結果に対する制約として利用する場合には“ゆるい”文法であるほど，その効果が低下すると考えられる[4]．
2. 処理速度：現在の OCR の認識速度は 10 文字/秒から 100 文字/秒程度であり，多くの場合そのエラー修正はパーソナルコンピュータ上で行われるであろう．したがってパーソナルコンピュータで上記の速度に大きく遅れない程度の処理を認識と同期して行うことが要求される[6]．
3. 得られたパスの評価：候補文字の組合せから得られる（少なくとも文法的には正しい）パスは多くの場合複数存在する．そこで何らかの評価値（以下ではコストと呼ぶ）によって“より良い”パスを選択し，オペレータに提示する必要がある．そして高尾ら[6]が将来の課題として指摘しているように後処理によって如何に認識率が向上するとしても 100% になることはあり得ないのでオペレータによる確認は欠かせない．したがって後処理自身がその結果を評価し誤りらしい個所を提示できることが全体としての入力速度向上のために必要である．

特にこの第3の側面（認識誤りの検出・指摘機能）は従来のこの分野の研究においてあまり検討されたことがなかったが OCR 自体の認識率が向上するにつれて重要になると思われる．OCR の認識率が比較的低

† A Method of Detecting and Correcting Errors in the Results of Japanese OCR by NOBUYASU ITOH and HIROSHI MARUYAMA (Tokyo Research Laboratory, IBM Japan Ltd.).

†† 日本アイ・ビー・エム(株)東京基礎研究所

い間はいずれにせよ全文を読みなおす必要があるが，ほとんどが正解という状況ならば“あやしい”部分だけをチェックすることが（ユーザにとって）望ましいからである．

筆者らは現在印刷文書を効率的にデータベース化するための文書理解システム（DRS: Document Recognition System）を開発中であるが，そのために必要な機能の1つとして，これらの要求を考慮するとともに DRS の目的に適した後処理を含む文字認識機能を実現したので報告する．最後に次章で DRS の概略を述べ，第3章で後処理の実現している機能および手法について説明する．さらに第4章では後処理の効果および速度についての実験結果を提示し，最後にまとめを行う．

2. DRS（文書理解システム）の概要

前述のように DRS の目的は印刷文書（特に需要が大きいと思われる科学技術文献）をハイパーメディアなどのデータベースに効率よく入力することであり以下のような機能をもっている[7]．

1. レイアウト理解：文書のレイアウト構造を与えられた文書モデルに基づいて解析し，書誌情報の抽出・読み順の決定を自動的に行う[8]．これには図を自動的に検出しイメージとして取り出す機能も含まれる．
2. 文字認識機能
3. 認識誤りの検出・自動修正を行う後処理機能
4. キーワード候補の抽出：文字認識すると同時に，文書検索に欠かすことのできないキーワードの候補を抽出する．
5. 後処理と同時並列的に実行可能なエラー修正のためのユーザインタフェース

この中で 2. のみが漢字 OCR アダプタカード（マイクロプロセッサ 68020，3 Mbyte のメモリーおよび専用ハードウェアからなる）上で実行され，その他はすべてパーソナルコンピュータ（80386，25 MHz）上で OS/2 のもとに実現されている．文字認識単独の速度は約 30 文字/秒である．

3. 後処理方式

3.1 日本語文法

池田ら[4]は OCR の後処理という立場から形態素レベルでの日本語文法を考察し，カテゴリー数 86 にのぼる品詞分類とその接続規則を提案している．しかしながら，“はじめに”で述べたように，より多くの文を受理することとより強い制約となることは相反する要求である．この事実と第1章で述べた1，3の要求を考慮すれば，すべての接続規則を対等に扱うのではなく，文法自身に確率を付与することによってパス選択のときに利用するコストの1つとして取り入れることが必要である．機械翻訳の前処理としての形態素解析では，解が多くなり過ぎて次のステップである係り受け・構文解析に負担がかかるのを防ぐため，単語間の接続に対してその出現頻度や共起確率に基づいたコストを導入し，それぞれの解に付随するコストで解を序列化しようというコスト付き形態素解析の試みが報告されている（たとえば久光[9]）．この場合でも単語をどのように分類するかは大きな問題となる．つまり分類がより細かい方が制約としてはより効果的であるが，信頼できる共起確率を求めるためには Bigram の場合でもカテゴリー数の二乗に比例して学習データ量を増やさなければならない．実用的な立場から言えばごく簡単な品詞分類の Bigram でも十分な制約になり得る場合もあれば，個々の単語の Trigram さらには複数文節間の関係を評価（言い換えれば構文解析）しなければ妥当なコストを付けれられない場合も存在するわけで，最も困難な場合にすべてを合わせることは実際的ではない．文字認識の後処理においてもっとも救済が困難（言い換えれば強い制約を持たない）ものは1文字単語である[3]．そこで原則は仮名漢字変換向けに開発された品詞分類[10]を用い，誤認識されやすくかつその分類で同じカテゴリーに属している1文字単語については必要に応じより詳細な分類および接続コストを記述することにした．すなわち詳細分類のための辞書（以下詳細辞書と呼ぶ）を別に用意しそこに記述されていなければ各品詞間の接続ごとに定義されているデフォールトのコストを用いることになる．現在この辞書に記述されているものには句読点（，．）や助詞（か，が）などがある．それ以外の辞書は次のとおりである．

- **自立語辞書**：約 115,000 語，自立語を 39 に分類
- **付属語辞書**：約 900 語，付属語を 70 に分類
- **ユーザ辞書**：約 1,000 語，現在は主としてコンピュータ関係の用語を格納

よく知られているように自立語（特に名詞）はその語数の多さの割には日本語文法における分類項目が少ない上，より詳細な分類（言い換えれば接続規則による差別化）が困難である．そこで自立語については出

現頻度を（品詞ごとではなく）各単語ごとに計算し，その対数値に基づいたコストを辞書の各エントリーに記述した．

学習データには JICST 科学技術データベースの電気工学編 (Vol. 26)（約 40 万字），および朝日新聞昭和 62 年度の約1か月分（約300万字），学習方式は丸山らの手法[11]を用いた．

各辞書は TRIE 構造を採用しており，辞書引きを行う位置から前方の文字ラティスの要素のいずれかと適合するすべての長さの単語が高速に抽出できる．

3.2 パスの探索戦略とあいまい度の評価

最初にコスト付き形態素解析を記号を用いて形式的に表現し，次にその拡張としての後処理手法を示した後，最も重要なあいまい性の評価について述べる．

用いられる文字集合を *Cset* とすると単語 (*W*)，文 (*Sent*) は *Cset* の要素から構成される文字列として定義できるので，コスト付き形態素解析とは $Sent$ ($=s_1s_2,\cdots,s_l\ (s_i\in Cset)$) を単語列として $Sent=W_1W_2,\cdots,W_m$ のように分解したときその単語列から決まるコスト関数 $g(W_1,\cdots,W_m)$ を算出し，その値が最小コストから一定値以内であるか上位 N 位までに属するものを求める作業である．ここで s_i を文字ではなく順序付けられた候補文字集合 $S_i=[s_{i1},s_{i2},\cdots,s_{in}]$（[ ] は順序付けられていることを示すために用いる）に置き換えれば，文は文字列から文字ラティスとなる．通常の形態素解析では各文字位置 (i) ごとにその先の部分文字列 $s_is_{i+1},\cdots,s_l$ について辞書引きが行われるわけであるが，その替わりに部分ラティス $S_iS_{i+1},\cdots,S_l$ から得られる文字の組合せについて辞書引きを行い候補単語を生成する手続き ($DC(i)$) があればその分場合の数は増加するが上記作業は容易に文字ラティスからコストの低い順にパスを求める OCR の後処理手法に拡張できる．高尾らの後処理[6]はこの探索に A^* アルゴリズムを用い，最良のパスを見出すコスト付き形態素解析の1つと考えることができる．著者らは，（後に説明するように）得られたパスのあいまい性を評価するため最適なパスに加え最小コスト g_{opt} から一定値 α 以内の複数パスを求める必要があった．そこで動的計画法に基づく非巡回グラフの最短経路探索アルゴリズムとビームサーチを併用したつぎのような探索手法を採用した．

各ノード D を位置 (P：単語の末尾文字の位置で表す)，そのノードに至る直前のノード ($\bar{D}$)，単語 (W)，その位置までの累積コスト (C) および品詞 (H) の5項組 $\langle P,\bar{D},W,C,H\rangle$ により記述し，同一位置 (i) をもつノード集合を $NS(i)$ とすると

Step 1 $NS(i)\leftarrow\emptyset$ $(i=1,\cdots,l)$.
$NS(0)\leftarrow\langle 0,D_\emptyset,W_\emptyset,0,B\rangle$.
$i\leftarrow 0$.
ただし $\emptyset, D_\emptyset, W_\emptyset, B$ はそれぞれ空集合，直前ノードが存在しないこと，長さ0の仮想的単語，および文節先頭であることを意味する．

Step 2 $NS(i)$ に属するすべてのノード (D_{ij}) について，
—D_{ij} と同じ W,H をもつノード ($\in NS(i)$) の中で最小の累積コスト (C) をもつものを選択しその値を C_{min} とおく．
—$C_{ij}>C_{min}+\alpha$ ならば D_{ij} を $NS(i)$ から削除する（α がビーム幅を決定）．

Step 3 $i<l$ ならば $DC(i+1)$（辞書引き）
$i=l$ ならば終了．

Step 4 辞書引きされた N_w 個の単語 (W_k ($k=1,\cdots,N_w$)) すべてについて，
—$NS(i)$ に属するノード
$(D_{ij}=\langle i,\bar{D}_{ij},W_{ij},C_{ij},H_{ij}\rangle)$ を1つ選択（すべて選択し終わったならば Step 4 を終了）．
—つぎのようなノード (D) を対応するノード集合 $NS(i+len)$ に追加する（ただし len は W_k の単語長で $i+len\leq l$ を満たしていること）．
$P\leftarrow i+len$.
$\bar{D}\leftarrow D_{ij}$.
$W\leftarrow W_k$.
$C\leftarrow g(W_{ij},W_k)+C_{ij}$ (g はコスト関数).
$H\leftarrow W_k$ の品詞.
$g(W_{ij},W_k)$ が ∞，言い換えれば W_k が当該ノードに接続し得ない場合は当然この段階で排除しておく．

Step 5 $i\leftarrow i+1$ として Step 2 から繰り返す．

このような手続きの結果得られる $NS(l)$ の中のノードから経路 $\bar{D}_{ij}$ を逆にたどれば全コストが最小から一定値 (α) 以内のパスを求めることができる．

[証明]

仮に位置 i のあるノード D_{ij1} で $C_{ij1}>C_{min}+\alpha$ であって，かつこのノードを削除せずに残した結果，W_k がその後に接続し，ノード $D_{i+len,j2}$ が作成されたとする．問題は $NS(i+len)$ に属するノードの中で

の最小累積コストを C_{opt} としたときに $C_{i+len,j2} \leq C_{opt}+\alpha$ になることがあるかということである．仮定より累積コストが C_{min} でありかつ単語・品詞がそれぞれ W_{ij1}, H_{ij1} である（言い換えれば D_{ij1} のそれに等しい）ノード D_{min} が $NS(i)$ 中に存在する．本形態素解析では単語の接続が直前の単語・品詞によってのみ決定されるので，単語 W_k は明らかにノード D_{min} にも接続し，新たにできるノード（$\in NS(i+len)$）の累積コストは $C_{min}+g(W_{ij1}, W_k)$ となる．したがって

$$\begin{aligned} C_{opt}+\alpha &\leq C_{min}+g(W_{ij1}, W_k)+\alpha \\ &< C_{ij1}+g(W_{ij1}, W_k) \\ &= C_{i+len,j2}. \end{aligned}$$

となり，ビームサーチの結果削除されるノードを延長したパスで $C_{i+len,j2} \leq C_{opt}+\alpha$ となるものは $NS(i+len)$ に存在しない．以上の過程を繰り返し適用すれば全コストが最小値から一定値 α 以内のパスは途中で削除されることなく，$NS(l)$ 内に存在することが言える．

コスト関数 $g(W_{ij}, W_k)$ としては各単語（W_k）の頻度（Unigram），品詞間（詳細辞書に記述があるときは単語間）の遷移確率（Bigram），および単語 W_k を構成する各文字の正解確率（認識時の距離から実験的に求めた関数により得る）の積（実際の計算では各項の逆数の対数和）で表現している．

解析の範囲は句読点（と認識された文字）を区切り記号として得られる部分を1単位とした．この方法では句読点以外の文字が句読点に誤認識された場合，悪影響が予想されるがOCRの性質として（逆に句読点が句読点以外の文字に誤ることは無視できないものの）この種の誤りはきわめて少ないため実用上の問題はないと考えられる．事実われわれの用いた実験データではそのような認識誤りはなかった．

次に誤りらしい個所を指摘する機能を実現するため各候補文字のあいまい性について考える．

文書について　発明の詳細な説明
害　を
(a)　(b)

図 1 認識結果のあいまい性
Fig. 1 Ambiguity of recognition results.

図 1(a) の文字ラティスを見れば（誰でも直感的に）“文書について”が正しい元の文ではないかと考えるであろう．しかしながら“文害”という言葉も“文”（名詞）+“害”（名詞）と考えれば文法規則を満たしている．さらに図 1(b) ではより広い範囲の情報がなければ人の目にもいずれが正しいかさだかではない．言い換えれば各候補文字におけるあいまい性の程

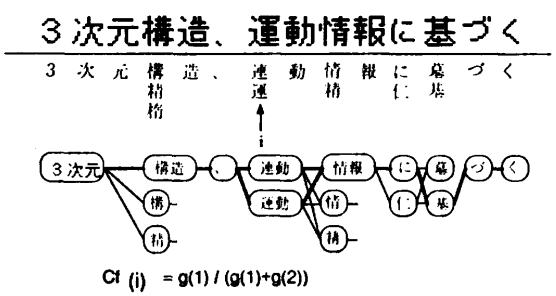

図 2 複数パスの探索とあいまい性の評価
Fig. 2 Estimation of ambiguity by multiple-path search.

度とはその候補文字を通るパス（複数の場合もある）と，当該カラムにおいて他の候補文字を通るパスとの生起確率の比で実現するのが妥当であると考えられる．そこで各文字ごとのあいまい性の評価が可能な後処理として，次のような手法を採用した（図 2）．

1. 認識結果（$\langle S\rangle = S_1, \cdots, S_l$）について上記の拡張コスト付き形態素解析を行い，最小コスト g_{opt} から一定値 α 以内のコストをもつパスをすべて求める．
2. 求まったパスを $P(j)=s_{j1}s_{j2}, \cdots, s_{jl}$ $(j=1, \cdots, N)$，さらに各文字位置 (i) について最適パスがその位置で採用した S_i の要素を s_{opt}，最適パス同様その位置 (i) において s_{opt} を選択したパスを $P(j')$ $(1 \leq j' \leq N)$ とするとき，位置 i での確信度 $(Cf(i))$ をつぎの式で定義する．

$$Cf(i)=\sum_{j'} g(j') \Big/ \sum_{j=1}^{N} g(j)$$

ただし $g(j)$ $(j=1, \cdots, N)$ はパス $P(j)$ のコスト，左側の Σ は $P(j')$ に対応するコストの総和を求めることを意味する．

したがって Cf は $(0, 1]$ の変数で1に近いほど確信度が高いことになる．g の定義から Cf は文節単位の文脈を考慮したときの，当該文字の生起確率比を近似していると考えることができる．たとえば図2の例では，可能なパスとして“三次元構造，運動情報に基づく”と“三次元構造，運動情報に基づく”の2つが残り，2つのパスで文字が異なる7文字目で確信度が計算される．ここで複数パスが存在することと，あるカラムで複数の可能性があることは必ずしも一致しない．つまり“北大西洋”という複合語を例にとると（“北大”

という単語が辞書に存在する場合)“北大＋西洋”および“北＋大西洋”という2つのパスが出力されることになるが，文字列としては同一であり，認識結果という立場からみればあいまい性は存在しないからである．

3. Cf の値と s_{opt} が1位の認識結果と一致するか否かによりつぎのように候補の入れ替えや当該文字を Marking（候補入れ替え，警告）することによるオペレータへの通知を行う．
   - s_{opt} が1位の認識結果と一致し，かつ $Cf>\delta$ ならば何も行わない（δ は閾値，以下同様）．
   - s_{opt} が1位の認識結果と一致せず，かつ $Cf>\delta$ ならば1位候補を s_{opt} に入れ替える．この場合もオペレータに対して（入れ替えたことを）通知する．
   - s_{opt} が1位の認識結果と一致し，かつ $Cf\leq\delta$ ならば警告を行う．
   - s_{opt} が1位の認識結果と一致せず，かつ $Cf\leq\delta$ ならば1位候補を s_{opt} に入れ替えた上で警告を行う．

3.3 キーワード抽出と誤認識検出の補助

前節で述べたように，本後処理はコスト付き形態素解析の拡張となっているので，後処理で行った時点で副産物として単語の切れ目，および品詞が分かることになる．そこで（複合語を含め）名詞を検出すればキーワードの候補が得られるがこれを表示する機能を付けている．加藤[12]によればOCRの誤りは1つの文書内では著しく偏在し特定の文字が繰り返し間違う傾向があり，名詞でのそのような誤りは得られたキーワードにも波及する可能性が高い．したがってこの機能はデータベースへの文書入力を目的とする場合に必要なばかりでなく，最終的にオペレータが誤りを見付けるための補助手段としても重要である．

4. 認識実験

本手法の効果を確かめるため認識実験を行った．用意したテストデータは**A**．電子情報通信学会論文誌（D分冊）の論文フロントページ（コピー：計9,455文字），**B**．コンピュータに関する顧客研修用資料（ワープロ出力をオフセット印刷したもの：計4,129文字），および**C**．電気工学分野の特許公報（計2,393文字）である．前2者は通常使用される程度の印字品質の代表として，**C**は低印字品質の代表として比較的つぶれ，かすれが多く見られるものを選んだ．これらの文書にはコンピュータ関連用語が頻出するが，その多くはわれわれの自立語辞書に含まれていないため，ユーザ辞書に約300語登録した．後処理前の認識率と処理後の認識率および誤認識に対する検出率との関係をページ単位で**図3**に示す．ただし検出（図中 Detect）とは当該文字に対して誤りの可能性があると識別し，候補の入れ替え，警告のいずれかが行われた（言い換えればMarking された）ことの意味で用いている．さらに対象文書ごとの平均値をとったものが**表1**である．誤認識には切り出し誤りによるものも含まれている．その割合は全文字中 0.14% であった．本手法の効率を評価するため後処理後認識率のほか以下に示す3つの尺度を用いる．

未検出率：検出されなかった誤認識文字の数（図3の Undetected に相当）/全文字数

過剰検出率：認識結果が正しいにもかかわらず警告された文字数/全文字数

表1 対象別にみた後処理の効果
Table 1 Performance of post-processing for each document.

文書	認識率(%)		未検出率(%)	過剰検出率(%)	総検出率(%)
	後処理前	後処理後			
A	96.50	99.26	0.18	1.81	5.35
B	94.74	99.03	0.17	1.50	6.64
C	87.46	94.61	1.09	4.80	16.38

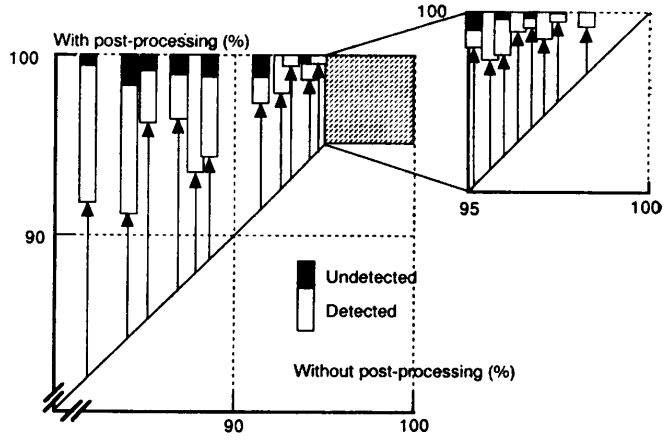

図3 後処理あり・なしにおける認識率および誤認識文字の検出率
Fig. 3 Recognition rate with or without post-processing and detection rate of misrecognized characters.

総検出率：検出されたすべての文字数（過剰検出を含む）/全文字数

これらの図表から，後処理の効率は元の認識率に強く依存することが明らかであるが，後処理なしで 95% 程度の認識率が確保できればそれを 99% 程度まで引き上げ，かつ誤認識の見逃し（未検出率）を 0.2% 程度に押えることができることがわかる．これはワープロ検定試験の1級が正解率 98.9% である[1]ことと比較すればほぼ十分な精度であると言える．さらにその場合の過剰検出も通常の印字品質である **A**, **B** では 1〜2% であり，文節単位での処理としては十分少ないと言える．

未検出および過剰検出の要因について述べる．まず未検出はそのすべてが候補内に正解が存在せずかつ当該文字部分で1位候補のみが唯一のパスとして選択されたことによるものであった．たとえば「多量」の「量」が候補になくかつ1位候補が「重」で「多重」というパス1個が有効となった場合があげられる．その他アルファベットで記述されたリスト番号（例："h."(正解は "b.")），接辞（例：「仕様善」(正解は「仕様書」)，「谷曲面」（正解は「各曲面」)）などが多い．これらの内，接辞については特定の漢語との共起確率に応じたコストを詳細辞書にもち，接続確率が低い（言い換えればコストが高い）ならば警告するなどの手法が考えられるが，前2者については構文レベルでの意味関係を評価しない限り，検出は困難である．

過剰検出（もともと認識が正解であったにもかかわらず過剰に警告されたもの，計 345 個）についてその原因を分類したものが**図 4** である．

もっとも数が多いのは当該文字で複数パスが存在した場合で，その約半数が句読点（"," など）である．

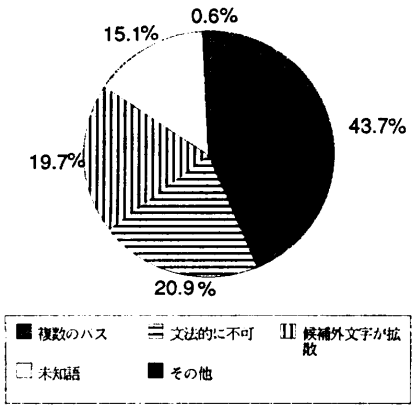

図 4 過剰検出の要因
Fig. 4 Reasons of over-detection.

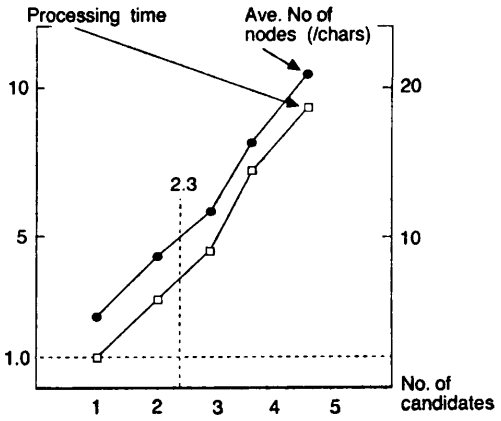

図 5 平均ノード数と処理速度
処理速度は候補文字が1の場合で正規化したもの．
Fig. 5 Processing time and average number of nodes against average number of candidate characters. Processing time is normalized by the case that the number of candidate characters is one.

特に文節が名詞で終了しているときに，句点，読点のいずれもがある程度の確率（コスト）で接続し得るため（そのどちらか）区別できない場合が多い．文法によるものの大半はわれわれの文法で「研究/開発」などの「/」が名詞に接続しないと記述されていたことによる．

処理速度は第2章で述べた環境で実行して約 27 文字/秒であった（ただし1文字当たりの平均候補数は 2.3 個，最大候補数は5個の場合）．本手法は探索時間という観点からみると動的計画法の性質上，ノード数に対して線形に増加するだけであると考えられるが，そのノード数が候補文字の増加にともなってどのように増えるか，さらに辞書引きなどの要素を含めた全処理時間は単純なモデル化が難しい．そこで候補文字数を増減させた場合のノード数および処理速度の変化を実験的に求めたのが**図 5** である（ただしビーム幅 (α) はすべて同じ値とした）．これによれば少なくとも平均候補文字数 (n) が5程度までならば n に比例し，より認識率が低く候補文字が絞れない場合にも対応できることがわかる．

5. ま と め

以上 DRS の文脈後処理機能とその実験結果について述べた．通常の使用環境においてはほぼ十分な精度，および速度で実行できる後処理機能が実現できた．特に後処理自身がその結果に対してあいまい性を

評価し警告できることが（オペレータによる確認・修正を含めた）トータルな処理速度に貢献すると考えられる．ただし何らかの Marking （候補の入れ替え，または警告）が行われる率が約 5% (20 個に 1 個程度）というのはまだ多過ぎるという評価もできる．これについては過剰検出をさらに減らすと同時に，候補の入れ替えを行った場合でも十分な確信度ならばMarkingしないことが良いと考えられるが未検出（見逃し）との Trade off でありより多くの実験が必要と考えている．また候補パスの選び方も現在は最良値から一定以内のコストをもつものという基準で選んでいるが，上位一定個数をとるということも考えられる．これについては探索手法と関連付けて検討が必要であろう[9]．今後は上記の課題に加え，文章の対象を広げオペレータの作業時間と（確認後を含む）入力精度を実験的に明らかにしたい．

謝辞 日頃ご指導いただいている豊川担当をはじめ，パターン認識，日本語処理両グループの方々，および記事データを提供してくださった朝日新聞社に深謝いたします．

参考文献

1) 宮原：文書情報の蓄積検索システムに関する検討，情報処理学会ヒューマンインタフェース研究会，29-3，pp. 1-10 (1990).
2) 杉村，斉藤：文字連接情報を用いた読み取り不能文字の判定処理—文字認識への応用—，電子通信学会論文誌，Vol. J 68-D, No. 1, pp. 64-71 (1985).
3) 新谷，梅田：文字認識における複合後処理法の能力評価，電子通信学会論文誌，Vol. J 68-D, No. 5, pp. 1118-1124 (1985).
4) 池田，大田，上野：手書き原稿における語彙および構文の検定，情報処理学会論文誌，Vol. 26, No. 5, pp. 862-869 (1985).
5) 杉村：候補文字補完と言語処理による漢字認識の誤り訂正処理法，電子情報通信学会論文誌，Vol. J 72-D-II, No. 7, pp. 993-1000 (1989).
6) 高尾，西野：日本語文書リーダ後処理の実現と評価，情報処理学会論文誌，Vol. 30, No. 11, pp. 1394-1401 (1989).
7) 天野ほか：マルチメディア文書入力のための文書画像認識システム：DRS，情報処理学会マルチメディア通信と分散処理研究会，48-6, pp. 41-48 (1991).
8) Yamashita, A., Amano, T., Takahashi, H. and Toyokawa, K.: A Model Based Layout Understanding Method for the Document Recognition System, *Proc. of the first ICDAR*, Saint-Malo (France), pp. 130-140 (1991).
9) 久光，新田：接続コスト最小法による形態素解析の提案と計算量の評価について，信学技法，NLC 90-8 (1990).
10) 大河内：仮名漢字変換のための形態素接続規則，IBM リサーチレポート，N: G 318-1560 (1981).
11) 丸山，荻野，渡辺：確率的形態素解析，日本ソフトウェア科学会第 8 回大会論文集，pp. 177-180 (1991).
12) 加藤，高橋：階層的辞書配置によるマルチフォント漢字認識，電子情報通信学会論文誌，Vol. J 74-D-II, No. 1, pp. 8-18 (1991).

(平成 3 年 12 月 26 日受付)
(平成 4 年 3 月 12 日採録)

伊東 伸泰（正会員）

1958 年生．1982 年大阪大学基礎工学部生物工学科卒業．1984 年同大学院修士課程修了．同年日本アイ・ビー・エム(株)入社．東京基礎研究所に勤務．文字認識について認識アルゴリズムおよび言語後処理の研究・開発に従事．

丸山 宏（正会員）

1958 年生．1981 年東京工業大学理学部情報科学科卒業．1983 年同大学大学院修士課程修了．同年日本アイ・ビー・エム(株)入社．東京基礎研究所に勤務．自然言語理解，論理型言語，日英機械翻訳の研究に従事．