

連続音声認識・理解システムのための構文解析法の比較・検討[†]

中川聖一^{††} 大黒慶久^{††}

音声理解システムにおいて、単語ラティスから最適単語列の解釈を見つけるアルゴリズムとしていくつかの方法が提案されている。本論文では文脈自由文法を用いた文音声認識アルゴリズムとして、left-to-right & top-down 型構文解析法と island-driven & bottom-up 型構文解析法とを種々の観点から比較検討した。両者の差異を明らかにするために、単語ラティスの質、音素認識率、解析時間などと文認識率との関係を論じた。実験では、連続音声認識システムにおける音響分析・音韻認識部をシミュレートすることによって、音韻認識率、ビーム幅、ワードラティスなどを変化させた場合における特質を調べた。その結果ビーム探索のビーム幅が小さい場合や文頭が noisy な場合には island-driven & bottom-up 法が文認識率が高いが、比較的処理時間が長くなること、left-to-right & top-down 法は広いビーム幅を要しながらも、効率よく探索が行えること、そして不要語の検出が可能であると仮定すれば、探索空間をより広くとることによって対処できることと思われることがわかった。処理時間を考えれば left-to-right 法が優れていると結論できる。

1. はじめに

本論文では文脈自由文法を用いた文音声認識手法として、left-to-right & top-down 型構文解析法と island-driven & bottom-up 型構文解析法とを種々の観点から比較検討し、両者の差異を明らかにする。

音声理解システムにおいて、単語ラティスから未知の最適単語列の解釈を見つけるアルゴリズムとしていくつかの方法が提案されている。

HWIM システムで開発された Woods の short-fall method¹⁾は、ヒューリスティックな評価関数を用いた Hart らの A*-method²⁾と類似な最適解を得る方法であり、island-driven 的な構文解析法に適用された。これは将来においても現在の仮説よりもよくならないものを枝刈りしており、Paxton による left-to-right の best-first 法³⁾よりもよいとされているが能率的でない。またこれの変形として island-driven shortfall density method (単語長によってスコアを正規化したもの、平均フレーム間距離の利用)、および left-to-right (厳密にいえば left-hybrid shortfall density) も試みられ 10 文章において比較されている。それによると island-driven 法は途中生成される theory (部分単語列) が多くなり過ぎ、枝刈りを行わざるをえない結果、準最適な left-to-right 法との差はほとんどなかった。ただし Woods は、文頭付近が

noisy な場合などに island-driven (middle-out) 法の有効性を指摘している。

これに対して Harpy システム⁴⁾や LITHAN システム⁵⁾では HWIM の left-to-right 法と類似なビームサーチ、best-few 法を採用していた (ただし LITHAN では文尾の述語から先に同定している)。

しかしながら island-driven 法と left-to-right 法との十分な比較検討は行われていない。カーネギ・メロン大学の Stern らは半自動的な音響・音韻処理部を用いて、言語の統語モデル表現法と解析法の比較などを行った^{13), 14)}。それによると、ワードラティス中に未検出単語 (missing word) が多い場合には、完全な文法を用いるよりもワードクラスの三つ組 (trigram) の出現確率の使用の方が有効であること、left-to-right 法よりも island-driven 法による解析の方が有効であることを見い出している。しかし言語モデルとして trigram を用いていることと、評価基準として単語認識率を用いていること等により、彼らのシミュレーション結果は本論文で採用する評価基準である文認識率とでは異なる結果になる可能性があると思われる。

本論文では単語ラティスの質、音素認識率、解析時間などの点で両解析法を比較検討した結果について述べる。

さらにこれら 2 種の解析法を文頭が noisy な場合や、不要語 (「えーっと」等の間投詞、「助詞の引き伸ばし」など) を含んだ連続音声に適用した場合についても考察した。以下第 2 章では文認識問題の定式化、第 3、4 章では比較した構文解析法、第 5 章では比較検討結果について述べる。

[†] Considerations on Syntactic Analysis for Continuous Speech Recognition or Understanding System by SEIICHI NAKAGAWA and YOSHIHISA OHGURO (Faculty of Engineering, Toyohashi University of Technology).

^{††} 豊橋技術科学大学工学部情報工学系

2. ワードラティスからの文の認識問題⁷⁾

なんらかの方法で得られたワードラティス中の単語候補が、(始端位置、終端位置、単語名、スコア) の四つ組からなっているとする。これらから次の条件を満たす単語列 $w = w_1 w_2 w_3 \cdots w_n$ を検出し、文認識結果とする。

(a) 候補単語 w_i を候補単語 w_j に接続するためには、 w_i の終端位置 e_i と w_j の始端位置 b_j に次の関係が成立しなければならない。

$$-gap \leq b_j - e_i \leq gap$$

gap 1: 両単語が時間的に離れていてもよい最大幅
gap 2: 両単語がオーバラップしていてもよい最大幅

(b) $w = w_1 w_2 w_3 \cdots w_n$ は与えられた文法で受理されねばならない。

(c) 以上の(a), (b)の条件を満たす単語列の内で、最大スコアをもつものを認識結果とする。

ここで単語列の評価基準は単に単語スコアの和で行わず、それぞれの単語列長を乗じたスコアの和で定義する。もちろん単語結合の際には、ギャップやオーバラップに応じてスコアを補正する必要がある。

3. left-to-right & 下降型構文解析法⁷⁾

Earley のアルゴリズムに基づき、時間軸に沿って発話の先頭から順序よく処理する方法⁷⁾であり、以下簡単に述べる。

日本語の文を図 1 のような文脈自由文法で表現する。図 1 の文法は後述する「計算機ネットワーク」タスクの文法の一部である（部分文「ジョブ 3 を止めよ（始めよ）」の生成に関係する部分だけ抜粋したもの）。@で始まるストリングは非終端記号、*で始まる記号は非終端記号の一種であるがワードクラスと呼び、終端記号のように扱う。各文法およびその書換え規則中の語の位置は、便宜上数字で表す。例えば番号 “16” は @s 1, “17” は @q 3 を表す。

この文法中の書換え規則を有限回適用して文の構造を解析していく。このときワードラティス中の単語は、現在適用中の規則が正しいか否かを確認するために用いられる。

例えば、いま図 1 の例で、部分文 “zyobu saN o” の右側に隣接可能な単語を予測する場合を考える。この部分文の解析過程は次のような書換え規則の適用履歴によって表現される。

	0	1	2	3	4	5
8	@s 0 → @s 1					
16	@s 1 → @q 3	@x 1				
24	@s 1 → *z b	@w s	@r 4	@p 5		
32	@x 1 → @r 4	@p 5				
40	@x 1 → *j 1	*g s	@r 4	*6 p		
48	@w s → *n 1					
56	@p 5 → *5 p					
64	@r 4 → *j q					
72	@q 3 → *k s	*k 1				
	*k s → keisaNki					
	*k 1 → gazoo					
	*k 1 → oNsei					
	*z b → zyobu					
	*n 1 → ichi					
	*n 1 → ni					
	*n 1 → saN					
	*n 1 → yoN					
	*n 1 → go					
	*5 p → hazimeyo					
	*5 p → tomeyo					
	*6 p → tunage					
	*g s → gaiseN					
	*j 1 → ni					
	*j 1 → e					
	*j q → o					

図 1 文脈自由文法の例

Fig. 1 An example of context free grammar.

“8(@s 0)”

→“8(@s 0) 9(@s 1)”

→“8(@s 0) 9(@s 1) 24(@s 1)”

→“8(@s 0) 9(@s 1) 24(@s 1) 25(*z b : zyobu)”

→“8(@s 0) 9(@s 1) 24(@s 1) 26(@w s)”

→“8(@s 0) 9(@s 1) 24(@s 1) 26(@w s)
48(@w s)”

→“8(@s 0) 9(@s 1) 24(@s 1) 26(@w s)
48(@w s) 49(*n 1 : saN)”

→“8(@s 0) 9(@s 1) 24(@s 1) 27(@r 4)”

→“8(@s 0) 9(@s 1) 24(@s 1) 27(@r 4)
64(@r 4)”

→“8(@s 0) 9(@s 1) 24(@s 1) 27(@r 4)
64(@r 4) 65(*j q : o)”

→“8(@s 0) 9(@s 1) 24(@s 1) 28(@p 5)”

→“8(@s 0) 9(@s 1) 24(@s 1) 28(@p 5)
56(@p 5)”

→“8(@s 0) 9(@s 1) 24(@s 1) 28(@p 5)
56(@p 5) 57(*5 p)”

→*5 p の予測

隣接可能な単語は *5 p から “hazimeyo” または “tomeyo” であることがわかる。そしてこれら二つの単語が接続条件を満足してワードラティス中に存在しているならば、これらをそれぞれ連結する。

以下にこの方法による文認識アルゴリズムを示す。これは入力パターンの第*i*フレーム目で終端しているすべての部分文について、その付近で始端をもつ候補単語を文法の制限をチェックしながら接続していく方法である。

- (1) $i=1$ 、部分文は空とし文開始記号から始める。
- (2) 部分文の規則適用の履歴を調べ、前述した単語予測の方法にしたがって隣接可能な単語を予測する。
- (3) 予測単語がワードラティス中にあり接続条件を満たすならば部分文と接続し、接続後の部分文のフレーム長($i + \text{単語フレーム数}$)に応じたテーブルに登録する。
- (4) 登録後、部分文の仮説数がビーム幅を越えないよう仮説のスコアによりソートする。なお文法的に同じ単語列(後続の単語予測が同じになる単語列)は最適なものだけ残す。
- (5) $i=i+1$ として全発声区間にわたって(2)～(4)の処理を繰り返す。全発声区間の処理が終わった後、スコアの最もよいものを認識結果とする。

4. island-driven & 上昇型構文解析法^{9), 10)}

4.1 island-driven 方式に必要な機能

人間はおそらく確からしい単語を中心に理解を進めていると思われる。また、発話の先頭から順にその内容を確定できるという保証はない。そこで信頼度の高い部分(島)から処理を始める方法が考えられており island-driven(島駆動)方式と呼ばれている。

この方式は文法規則により与えられた言語情報を用いて、認識された単語列(島)の右または左に接続可能な単語を予測する。また二つの単語列の結合可能性についても調べ、可能ならば結合しより大きな島にする。今回比較に用いたものでは、単語予測のアルゴリズムとしては BUP(Bottom-Up Parser in Prolog)で用いられた左隅解析法を応用しており⁹⁾、処理を島駆動に行うために必要である次の三つの機能をもっている¹⁰⁾。

- (a) 島の生成：ワードスポットされた単語から新しい島(seed)を作る。
- (b) 島の拡張：島の左右に構文的に接続可能な単語を予測し、ワードスポットティング部でスポットされた単語を接続して新しい島(theory: 部分文)を作る。
- (c) 島の結合：二つの島が構文的に結合可能であるかを調べ、可能であれば二つを結合した新しい島を作成する。

い島を作る。

以下本方式を簡単に説明する。

4.2 単語予測のアルゴリズム

まず、文法規則が与えられた時点での非終端記号間のリンク関係を求める。ここでいうリンク関係とは BUP で定義しているリンク関係を拡張したもので、左リンク関係・右リンク関係の 2 種があり、それぞれある非終端記号 X がどの非終端記号 Y の構文木の左端(右端)の子孫となり得るかを示したものであり、

$\text{left-link}(X, Y)$: 左リンク関係

$\text{right-link}(X, Y)$: 右リンク関係

と表す。

次にこのリンク関係を用いた単語予測の方法について述べる。例としてスポットされた単語の右側にくる単語を予測する場合を示す。

スポットされた単語がワードクラス(3章参照)c で、c を右辺に含む書換え規則

$$A \rightarrow b c D \quad (1)$$

が与えられたとすれば、(1)より c の右側には D という構文が予測され、その D から導出される最左端の単語が求める予測単語である(図 2(a) 参照)。

この単語予測はトップダウン予測と呼ばれ、D の左リンク関係のテーブルより

$\text{left-link}(x, D)$ (x : ワードクラス)

を探索することに相当する。つまり c の右側にはこの関係を満足するワードクラスに属する単語がくる。

次に上で求めた x がワードクラス f であり、属する規則が

$$E \rightarrow f g \quad (2)$$

であるならば、この書換え規則において f の右側にはワードクラス g に属する単語がくることがわかる。この時点で E の構文木の解析は終了してしまい、g の右側を予測するには E の属する、さらに上位の構文木の

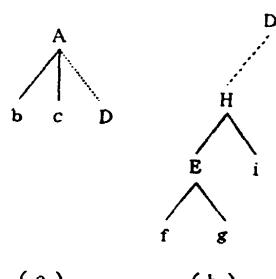


図 2 島の拡張

(a) トップダウン予測のある場合

(b) トップダウン予測のない場合

Fig. 2 Extension of island.

書換え規則を用いなければならない。このとき上位の構文木と E との間、そしてこの上位の構文木と D との間には左リンク関係がなくてはならない（図 2 (b) 参照）。

例えばそれが

$$H \rightarrow E \ i \quad (3)$$

であったとすれば g の右側にはワードクラス i に属する単語がくることがわかる。

以上のように、現在適用中の書換え規則においてトップダウン予測があるならば右側にくる単語を左リンク関係を用いて予測し、右辺のすべての非終端記号の解析が終了したならば左辺の表す構文木は完成したことになる。そして完成した構文木と左リンク関係にある親構文木の書換え規則を適用し、同様な処理を続け左から右に、下位から上位に解析を進める。

これまで右方向への単語予測であるが、左方向についても右リンク関係を用いて同様に考えることができる。

さらに以上の単語予測のアルゴリズムを島駆動を行うためには左右の規則適用が矛盾なく行われねばならない。したがって単語予測の際は反対方向の規則適用履歴も参照し、現在解析中の構文木より下位の構文木の解析が終了したときには単語予測は行わず、反対方向のみの予測を行う。左右両方向の予測が今までの規則適用履歴を用いて行えなくなった場合（左右ともに下位の構文木が完成）には、今完成している構文木を右辺に含む任意の書換え規則（リンク関係は問わない bottom-up）を適用する。

4.3 島の結合

前節の単語予測アルゴリズムによって作られた島が別の島と構文的・時間的に結合可能であるならば結合し、より大きな島を作る。

二つの島の結合には次の 2 種類がある。

(1) 結合する島が結合される島をトップダウン予測する場合（図 3 (a) 参照）

単語を予測する場合と同様に、現在適用中の書換え規則においてトップダウン予測される非終端記号の表す島があるならば結合する。

結合される島は一方の構文木の下の部分構分木となる。

(2) 同一の構文木の下で結合する場合（図 3 (b) 参照）

両方の島がトップダウン予測が不可能で（部分構文木が完成）、その両者が隣接するような書換え規則が

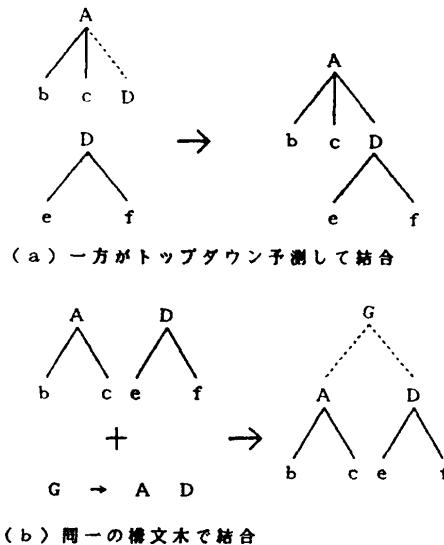


図 3 島の結合
Fig. 3 Concatenation of islands.

あるならばこの規則の下で結合する。

4.4 単語列生成戦略

今まで述べてきた island-driven 的構文解析法を次のように文音声認識アルゴリズムに組み込んでいる。

(1) ワードラティス中よりスコアのよいものをビーム幅の許す限り seed 単語として選ぶ。このとき「の、を」等の助詞など極端に発声長の短い単語は除外している。

(2) 島の長さが短いものから（最初は seed 単語）島を拡張し、接続後の部分単語列長（島の長さ）に応じたテーブルに登録する。なお文法的に等しいものは最適なもののみ残す。

(3) 登録後、記憶しておく島の仮説数がビーム幅個を越えないようにスコアによりソートする。

(4) 一単語拡張するごとに現在までに生成されている島の仮説間の結合可能性を調べ、結合可能であれば両者を結合し結合後の島の長さに対応するテーブルに登録する。結合後の島についても常に島の仮説数がビーム幅個を越えないようにスコアによりソートする。

(5) 全発声長の島の処理が終了するまで(2)～(4)を繰り返す。全発声をカバーする島のうち最もスコアのよい仮説を認識結果とする。

5. 音響・音韻・単語処理部のシミュレーション方法

シミュレーションが実際の音声認識に即したもので

	a	i	u	e	o	N	y	w	b	d	g	r	z	m	n	p	t	k	s	h
a	P																			
i		P																		
u		P																		
e			P																	
o			(1-P) × 0.8	P																
N					5															
y						P														
w						P														
b						P														
d						P														
g						P														
r						P														
z						P														
m						P														
n						P														
p						P														
t						P														
k						P														
s						P														
h						P														

図 4 コンフュージョンマトリクス
(Pは平均音韻認識率)

Fig. 4 Confusion matrix between phonemes
(P denotes the average phoneme
recognition rate).

あることに努めつつ、その方法が複雑にならない程度に簡略化している。

5.1 音韻カテゴリ

音韻認識部の動作をシミュレートする際に、扱う音韻のカテゴリは次のとおりである。母音/a, i, u, e, o/, 撥音/N/, 半母音/y, w/, 有声子音/m, n, b, d, g, r, z/, 無声子音/s, h, p, t, k/, 促音/q/, 長母音は同母音を2個続けて表現する。

5.2 音響分析・音韻認識のシミュレーション

タスクの入力文を音韻単位の文字列に変換し、図4のコンフュージョンマトリクスに基づいて誤りを含んだ音韻列を発生させることによって音響分析・音韻認識部をシミュレートする。

図4は実際の音韻認識の能力を模擬的に表現したものである。すなわち各音韻を母音(撥音を含む)、有声子音、無声子音の3カテゴリに分類し、各音韻の認識率は一律と仮定し、誤認識の場合には自カテゴリ内での混同が多く、他カテゴリとは混同しにくいという状況を表したものである。例えば母音は、その認識誤りの80%が他の母音との混同であり、10%が有声子音カテゴリ、残りの10%が無声子音カテゴリと混同することを示している。母音と子音を同一の認識率に設定したのは日本語音声ではやや非現実的であるが

(日本語では母音の認識の方が易しく、英語では子音の認識の方が易しい)、それらの認識率はシステムに依存するので便宜上同一に仮定した。

ただし、挿入・脱落エラーは各音韻でそれぞれ5%，促音は誤認識率10%で“p, t, k”とのみ混同とした。

つまり音韻*i*が音韻*j*に識別される確率を $p(j|i)$ ($\sum p(j|i)=1$) とし、挿入誤り・脱落誤りをコンテキストによらず各々 p_i , p_o とすれば音韻*i*の置換誤りは次のようになる。なお、本論文では $p(i|i)$ を音韻*i*の認識率と呼ぶ。

$$\text{置換誤り: } (1-p_i-p_o) \cdot p(j|i)$$

この(誤りを含んだ)音韻列に対して、次に示すワードスポットティング法を施しワードラティスを得る。ワードラティスとは入力音声中で存在しそうな単語候補をすべて出力した表現方法で、通常は(始端位置、終端位置、単語名、スコア)の四つ組で示される。

なおコンフュージョンマトリクスは平均音韻認識率60%および80%の2種を用いた。

5.3 ワードラティス生成部⁷⁾

ワードスポットティングの手法について説明する。まず使用する記号の定義を述べる。

$Q^*(i, j)$: 入力パターンの $m \sim i$ フレームと標準パターン n の $1 \sim j$ フレームとの最大累積対数尤度の m についての最大値

J^* : 標準パターン n のフレーム長(単語長、音韻数)

$B^*(i, j)$: $Q^*(i, j)$ に対応する入力パターンの始点位置

$p^*(i, j)$: 入力パターンの i フレームの音韻が標準パターンの j フレームの音韻に置換する確率

なお本実験では $p^*(i, j)$ の対数をとったものを音韻間の尤度とする。各単語標準パターンに対して、入力パターンの各フレームで終端する最適な照合位置と累積尤度が求まればワードスポットティングが実現できる。すなわち $Q^*(i, J^*)$ があるしきい値より大きければ入力パターンの $B^*(i, J^*) \sim i$ フレームに単語 n が存在可能と判断すればよい。可能性の程度は $Q^*(i, J^*) / J^*$ の値で評価でき、あらかじめ設定してあるしきい値以上ののみを残す。本実験では、このしきい値を単語長に応じて3段階に設定している。

これらは動的計画法を用いて容易に求めることができる(DPマッチングによるワードスポットティング

グ法)⁷⁾.

また本実験ではラティスのスコアを次式で定義した(最大スコアは1000で、正解単語に対しては大半が850~950のスコア)。

$$\text{スコア} = 1000 + 10 \times \text{累積尤度} / \text{単語パターン長}$$

6. 実験

6.1 タスク

認識の対象となるタスクは「計算機ネットワーク」⁶⁾に関するもので、以下に示す5種類がある。

なお(4)の混合においては「デパート案内」タスク¹¹⁾も使用している。

(1) 小語彙(104語)

(2) 中語彙(250語)

(3) 文頭が noisy 状態

「小語彙」ラティスにおいて文頭の正解2単語のみのスコアを減じたもの)

(4) 混合(小語彙+「デパート案内」: 計247語)

(5) 不必要語を考慮したパーサ用タスク

a) 不必要語のある文

b) 不必要語のない文

各場合において50文に対して認識実験を行った。

「計算機ネットワーク」で使用した文例を以下に示す。

(1) 計算機中央の磁気ディスク装置3番から計算機画像データ4をロードせよ。

(2) ジョブ8を止めよ。

(3) 計算機交換の磁気テープ装置7番を巻き戻せ。

不要語を含んだ文

(1) あのー、計算機中央の磁気ディスク装置3番からー、計算機画像データ4をロードせよ。

(2) えーっと、ジョブ8を止めよ。

(3) じゃあ、計算機交換の磁気テープ装置7番をー、巻き戻せ。

6.2 比較した4種の解析方法

比較する解析法は次の4種である。

(1) left-to-right & top-down 法

(2) island-driven & bottom-up 法

(3) BUP(4.1節参照: 横向き上昇型)

左→右に解析(left-to-right & bottom-up法)

(4) BUP

右→左に解析(right-to-left & bottom-up法)

いずれの解析法でもビームサーチ⁴⁾を採用した。なお不要語を含む文に対しては以下のような対処を行った。

会話音声中に含まれる間投詞などの不要語は、代表的な4種類で90%が占められ、種類によらず定常的な母音を含むことが多い⁶⁾。よって不要語辞書にこれらを登録しておき、通常単語のように扱う。また語尾の長音化も定常的な母音を含み、ワードスポットティングにより検出可能であり、同様に扱える。

前述した両解析法とも検出された不要語を読み飛ばす機能を付加している。ただし今回の実験では、不要語は文頭または文節間(助詞の後)にのみ現れると仮定している。

6.3 シミュレーション結果

音響分析・音韻認識部をシミュレートすることによって得られたワードラティスの質を表1(a), (b)に示す。ここでいうワードラティスの質とは、ワードラ

表1 ワードスポットティング結果の評価

Table 1 Evaluation results of word-spotting or word-lattice.

(a) ラティスの質(単語検出率)

	小語彙 104	文頭が noisy	中語彙	タスク混合
音韻認識率	60%	80%	80%(1)	80%(2)
1位	55.4	80.8	74.6	72.4
2位まで	69.1	88.3	82.8	78.7
5位まで	86.3	95.9	95.9	88.8
10位まで	91.8	97.5	97.5	92.8
missing	6/489	0/489	0/489	0/489
検出単語数	857	931	931	1605
				1745

(b) 不要語を含むラティスの質(単語検出率)

“有文・有語彙”: 語彙に不要語有り・不要語のある文を認識

“無文・有語彙”: 語彙に不要語有り・不要語のない文を認識

	有文・有語彙	無文・有語彙
音韻認識率	60%	80%
1位	56.6	80.4
2位まで	67.1	86.1
5位まで	80.8	92.2
10位まで	87.2	95.7
missing	11/541	2/541
検出単語数	1665	1695
		1521
		1562

“小語彙 104”: 計算機網タスク語彙数104語

“文頭が noisy”: “小語彙 104” 音韻認識率80% ラティスの文頭正解2単語のスコアを50(80%(1))または25(80%(2))減じたもの

“中語彙”: 計算機網タスクを104語から250語にしたもの

“タスク混合”: 計算機網+デパート案内の計247語のタスク

ク

“missing”: ラティス中にはない単語数/ラベル数

“検出単語数”: 一文当たりのワードラティス中の単語数

表 2 不必要語を含まず、かつ考慮しないバーサによる実験 (“小語集 104” 使用)
Table 2 Experimental results in the case of no interjectional words.

	ビーム幅	音韻認識率=60%				音韻認識率=80%			
		SR %	PW	BF	T sec	SR %	PW	BF	T sec
left-to-right & top-down	5	50	391	5.7	6.4	66	505	6.4	7.7
	10	66	801	6.8	8.9	78	886	6.7	10.1
	20	66	1351	6.9	12.4	84	1430	5.9	14.0
island-driven & bottom-up	5	54	513	3.5	16.2	70	511	3.5	17.0
	10	66	1005	3.7	25.7	78	1063	3.6	27.7
	20	68	1935	3.7	50.1	84	2007	3.6	52.5
BUP left-to-right	5	50	409	6.7	9.4	66	525	6.1	10.8
	10	66	804	6.8	11.7	82	994	5.9	13.8
	20	66	1308	5.9	15.3	84	1617	5.2	18.5
BUP right-to-left	5	62	321	5.1	13.1	74	348	5.4	14.4
	10	64	632	4.9	15.6	82	664	5.2	17.0
	20	68	1166	4.9	20.1	84	1225	5.1	21.7

SR: 文認識率, PW: 予測単語数, BF: 平均分岐数, T: 一文当たりの処理時間

ティス中において正解単語が存在しているべきところ(付近)で、正解単語がどの程度の順位で検出されているかを表したものである。この質は文認識率に大きな影響を与える。

これによると、音韻認識率が低いと単語が検出されない場合があること(表中の missing 欄)、認識語彙数に比例してラティスを構成する単語数が増えているがその割には検出率が落ちないことがわかる。missing 単語には単語長の短い「を、は」などの助詞が多く、これらはその後の解析では大きな問題(文型そのものを誤る)となる。

6.4 実験結果

6.4.1 音韻認識率、ビーム幅による比較

4種類の解析法による文認識結果を表2に示す。表中の SR は文認識結果の正解率、PW は予測単語数、BF は平均分岐数(一つの部分文から予測される平均単語数)、T は1文当たりの処理時間(Apollo DOMAIN 4000, C 言語による), をそれぞれ示している。

これによると音韻認識率が低い場合には文認識率も低く、いかに構文情報を用いるといえども、高精度のワードスポットティング部が望まれることがわかる。

left-to-right 法と island-driven 法を比べると、island-driven 法が両方向に予測するので予測単語数が多く、処理時間が長い。これは island-driven 法の処理が複雑なことと、文の途中から解析を始めるために生成される仮説(部分文)の数が多いことによる。

平均分岐数 BF が、left-to-right 法に比べて island-

driven 法では少ないので次の理由によると考えられる。left-to-right 法の場合には、文頭から単語列を同定していくため、入力文に依存した構文に対する分岐数になるのに対し、island-driven 法では bottom-up に予測を行うために、予測時に適用する書換え規則が一意に決まらず必然的に多くの書換え規則を適用することになり、結果として書換え規則の集合つまり文法の平均分岐数に近くなってしまい、たまたまこの方が小さかったからである。

またビーム幅が小さいときは、island-driven の方が平均予測単語数が多いにもかかわらず、文認識率がよいことがわかる。これは left-to-right 法が左から右に処理を進めるために、文頭付近で正単語列がビーム幅内に残らなくなる可能性が高いからである。一方、island-driven 法は、信頼度の高い部分から任意に解析を開始でき島を成長させるため、一部のスコアが悪くとも島としてのスコアへの影響は少ない。

BUP において、left-to-right に解析を進めた場合と right-to-left に進めた場合とを比較すると、right-to-left の方が文認識が高いことがわかる。これは日本語の場合には、述語によって文の構造がかなり制限されるからである¹²⁾。すなわち、文尾の述語を先に同定してやれば探索すべき仮説(部分文)が少なくて済み、これは予測単語数が少なくなることを意味し(表2の PW, BF 欄参照)、結果的に小さいビーム幅でも十分な探索が行えることを示している。

表 3 (表 2 で) 文頭が noisy な場合 (“小語彙 104” タスク中で文頭正解 2 単語のスコアを減じた場合)
Table 3 Experimental results in the case of noisy words in the head part of input sentence.

ビーム幅		音韻認識率=80%							
		スコアを 50 減じた場合				スコアを 25 減じた場合			
		SR %	PW	BF	T sec	SR %	PW	BF	T sec
left-to-right & top-down	5	26	502	6.3	7.6	62	505	6.4	7.6
	10	26	876	6.6	10.0	72	883	6.7	10.0
	20	26	1431	5.8	14.0	78	1429	5.9	14.0
island-driven & bottom-up	5	64	561	3.5	17.2	68	627	3.8	18.6
	10	66	1107	3.6	27.7	78	1242	4.0	30.6
	20	70	2032	3.7	52.0	82	2354	4.1	59.6
BUP left-to-right	5	22	514	5.5	10.8	68	543	5.9	10.9
	10	28	999	5.6	13.9	78	1000	5.9	13.8
	20	28	1614	5.0	18.6	80	1617	5.1	18.5
BUP right-to-left	5	64	348	5.4	14.4	70	348	5.4	14.4
	10	70	663	5.2	17.0	78	664	5.2	16.9
	20	70	1225	5.1	21.7	80	1225	5.1	21.7

6.4.2 文頭が noisy な場合

文頭が noisy な状態にある文をシミュレートするために、文頭の正解 2 単語に対しスコアの劣化操作を行った。これによって、正解単語はラティス内で劣化前と比べて相対的に悪い順位で検出されたことになる。表 1 (a)によると 1 位の検出率が 6~8% 落ちている。表 3 (a), (b) に文認識結果を示す。表より left-to-right に解析を行った場合には認識率が悪いことがわかる。left-to-right に解析を進める場合、文頭から順に単語列を同定していくために、文頭でスコアが悪ければビーム幅から漏れてしまうからである。逆に right-to-left に解析を進める場合には、文頭のスコア劣化部分に至るまでに正解部分文が高スコアで同定されていれば、単語結合の際のスコア補正により(劣化操作後の) 正解単語を選び得るから、文認識率はあまり落ちない。island-driven 法の文認識率があまり落ちない理由も同様で、時間軸によらずスコアの高い部分を優先して処理するため、結果的に劣化部分の同定を遅らすこととなり、正解仮説の島がビーム幅から漏れることが少ないからである。

これらの結果は文献1)で紹介した Woods の考察に一致している。

6.4.3 語彙数・タスクによる比較

表 4 に語彙数を 2.5 倍にした場合の結果を示す。平均分岐数・予測単語数は 2 倍以上増えているが、文認識率は表 2 と比べて、それほど落ちていない。これは単語検出率(表 1 (a)) が 2~3% 程度しか落ちていな

表 4 表 2 の語彙を 104 から 250 にした場合

の結果 (“中語彙” 使用)

Table 4 Experimental results for the vocabulary size of 250 words.

ビーム幅		音韻認識率=80%			
		SR %	PW	BF	T sec
left-to-right & top-down	5	68	1249	16.7	22.5
	10	72	2384	18.1	28.5
	20	82	3933	15.0	38.2
island-driven & bottom-up	5	68	1019	7.5	41.1
	10	74	2109	8.2	59.5
	20	82	4008	8.4	98.0

表 5 タスクを混合した場合の結果 (計算機網 & デパート案内) (“タスク混合” 使用)

Table 5 Experimental results in the case of combination of two tasks.

ビーム幅		音韻認識率=80%			
		SR %	PW	BF	T sec
left-to-right & top-down	5	66	649	5.6	22.4
	10	82	1042	5.4	25.8
	20	84	1612	5.3	31.4
island-driven & bottom-up	5	68	469	3.4	39.3
	10	82	967	3.5	53.0
	20	84	1741	3.4	80.9

いために、解析過程で選ばれていく仮説群は表 2 とあまり変わらないからだと考えられる。

表 5 に別のタスク (「デパート案内」) を混合し、語

彙数を 247 語にした場合の結果を示す。left-to-right 法では、表 2 と比較すると平均分岐数 BF が減り、予測単語数 PW は増えている。文認識率は落ちていないのは、「デパート案内」タスクが混入したもの、「計算機」タスクと共にワードクラスがなかったことによると思われる。一方、island-driven 法においては BF , PW ともにわずか減少している。これは seed 単語として異種タスクの単語が選ばれることが少なかったうえに、生成される仮説が多いため異種タスクの仮説が当該タスクの仮説との競合の結果棄却されてしまったからだと考えられる。

表 2, 3, 4 よりビーム幅が小さいときは island-driven 法の方が left-to-right 法よりも文認識率がよく、ビーム幅が大きい場合は両者に差がないことが語彙数やタスクに関係なく言える。

6.4.4 island-driven 法において best-first サーチを行った場合

表 6 に island-driven & bottom-up 法において、4.4 節で述べた単語列長の順 (breadth-first サーチ) ではなくスコアのよい順に単語列仮説を優先して解析 (best-first サーチ) した場合の結果を示す。表中 “最終結果一つ” とは、全発声区間をカバーする単語列仮説が一つ生成されれば処理を終了することを表す。表より best-first サーチによって探索された最初の文認識結果 (“最終結果一つ” の場合) が必ずしも最も仮説ではないことがわかるが、最終結果の文仮説数を 4 個にすれば breadth-first サーチと比べて文認識率は若干落ちる程度である。best-first サーチは処理に要する時間を短縮でき(表 2 と比較)、音韻認識率がある程度高くて生成される単語列仮説が十分確からしい場合には有効な方法であろう。

6.4.5 不必要語を考慮した場合

表 7 (a) に不需要語対処機能を付加したパーサを不需要語を含む 50 文 (「えー」 26 個、「あのー」 14 個、

「じゃ」 12 個、「助詞の引き伸ばし」 21 個) に適用した結果を、表 7 (b) には含まれない文 (表 2 と共通) に適用した結果を各々示す。

表 2 と表 7 (a) を比較すると平均分岐数 BF が増えており、それについて予測単語数 PW も増加している。これは明らかに不需要語の仮説を考慮したためであり、文認識率が落ちているのは、生成される仮説数の増大によって正解単語列よりスコアのよい単語列が生成されやすくなるからである。しかし文認識率は大幅な低下ではなく、不需要語の仮説を考慮しつつ、正解単語列を選んでいるといえる。

表 7 (b) は表 2 と共通な 50 文について認識した結果であるが、 BF , PW ともに増えているが予測単語数は表 7 (a) よりも少ない。それゆえ表 2 の文認識率と比較すると文認識率の低下は小さく、2% の低下にとどまった。

これらの結果より、不需要語に対してはそれがある程度検出されれば、十分対処可能であると思われる。またここでも、ビーム幅が小さい場合には island-driven 法が文認識率が高いが処理時間が長いこと、一方 left-to-right 法は広いビーム幅を必要としながらも効率よく探索を行える等の前述した特徴が見られる。

6.4.6 助詞の欠落を補った場合

6.3 節で述べたように「を、は」などの助詞は単語長が短く認識が困難であり、検出されない場合がある。助詞が検出されなくても解析が進められるように、検出された助詞とともにデフォルト値で代替した助詞も仮説に加えた場合の結果を表 8 (a) に示す。また助詞の発声は前提とするが、すべての助詞をデフォルト値で代替した場合(すなわち、助詞は認識しない)の結果を表 8 (b) に示す。表 2 と表 8 (a) を比較すると、音韻認識率 60% でビーム幅が小さい場合に文認識率の向上が見られるが、ビーム幅が 20 の文認識率は同程度である。また表 2 と表 8 (b) を比較すると

表 6 island-driven & bottom-up パーサで best-first サーチによる結果
Table 6 Experimental results by best-first search of island-driven & bottom-up parser.

	ビーム幅	音韻認識率=60%				音韻認識率=80%			
		SR %	PW	BF	T sec	SR %	PW	BF	T sec
最終結果 1 つ	5	52	232	4.8	11.6	70	259	7.0	12.0
	10	58	334	5.4	13.3	72	332	7.3	13.5
	20	58	421	5.4	16.4	72	367	7.3	15.4
最終結果 4 つ	5	54	498	4.3	15.3	74	482	4.9	15.8
	10	66	798	4.6	21.7	78	831	5.0	22.8
	20	66	1085	4.7	30.8	82	1100	4.9	32.3

表 7 不必要語を考慮した場合の文認識結果

Table 7 Experimental results of spoken sentence recognition in the existence case of interjectional words.

(a) 不必要語のある文の認識結果 (“有文・有語彙” 使用)

ビーム幅		音韻認識率=60%				音韻認識率=80%			
		SR %	PW	BF	T sec	SR %	PW	BF	T sec
left-to-right & top-down	5	46	743	10.1	17.3	60	891	10.5	18.8
	10	58	1433	10.8	24.4	76	1626	10.2	26.8
	20	62	2453	10.1	38.9	80	2954	9.7	44.8
island-driven & bottom-up	5	52	972	5.5	42.7	66	1073	6.3	41.9
	10	62	2010	5.4	68.3	78	2107	6.1	67.0
	20	62	3951	5.0	130	80	3996	5.7	126

(b) 不必要語のない文の認識結果 (“無文・有語彙” 使用)

ビーム幅		音韻認識率=60%				音韻認識率=80%			
		SR %	PW	BF	T sec	SR %	PW	BF	T sec
left-to-right & top-down	5	50	661	8.3	14.8	64	776	10.1	16.8
	10	62	1276	10.4	21.5	70	1417	9.8	23.8
	20	64	2368	10.7	36.4	82	2618	9.4	39.9
island-driven & bottom-up	5	52	810	6.3	38.0	68	853	8.7	37.3
	10	64	1668	6.4	60.0	78	1724	8.0	60.5
	20	66	3400	6.4	119	82	3353	7.4	116

表 8 助詞の欠落を考慮したパーサによる結果

Table 8 Experimental results of spoken sentence recognition in the case of missing postposition.

(a) 助詞の欠落をデフォルト値で補った場合 (デフォルト値=900)

ビーム幅		音韻認識率=60%				音韻認識率=80%			
		SR %	PW	BF	T sec	SR %	PW	BF	T sec
left-to-right & top-down	5	48	423	6.3	6.9	60	534	4.6	8.3
	10	60	904	7.1	10.3	72	999	4.6	11.6
	20	64	1510	7.1	15.3	84	1658	4.6	17.3
island-driven & bottom-up	5	66	614	4.1	16.8	80	658	4.6	17.7
	10	66	1245	4.3	28.0	86	1359	4.8	30.1
	20	66	2501	4.4	56.9	88	2608	4.7	59.7

(b) 助詞を認識せずすべてデフォルト値で代替した場合 (デフォルト=900)

ビーム幅		音韻認識率=60%				音韻認識率=80%			
		SR %	PW	BF	T sec	SR %	PW	BF	T sec
left-to-right & top-down	5	22	277	5.0	6.1	40	404	4.9	7.5
	10	34	744	5.9	9.3	50	786	5.7	10.0
	20	64	1263	6.7	13.1	80	1278	5.7	14.3
island-driven & bottom-up	5	62	535	4.1	15.9	76	617	4.2	17.5
	10	64	1128	4.2	26.8	80	1183	4.4	28.0
	20	64	2315	4.3	53.9	84	2355	4.3	55.9

ビーム幅が小さい場合には文認識率は落ちているが、ビーム幅が大きい場合の劣下は小さい。このように助詞は必ずしも認識・検出されなくても、十分解析ができることがわかる。

7. おわりに

連続音声システムにおける音響分析・音韻認識部をシミュレートすることによって、解析手順としては代表的な2種の方法、left-to-right & top-down法とisland-driven & bottom-up法について、音韻認識率、ビーム幅、ワードラティスなどを変化させた場合における文認識率との関係を調べた。その結果ビーム幅が小さい場合や文頭がnoisyな場合にはisland-driven & bottom-up法が文認識率が高いが、比較的処理時間が長くなること、left-to-right & top-down法は広いビーム幅を要しながらも、効率よく探索が行えること、そして不必要語の検出が可能であると仮定すれば、本方式で対処できると思われることなどがわかった。結論としては構文知識をベース（意味情報も構文知識へ組み込める意味文法を使用する場合も含む）とする連続音声認識では処理時間と認識精度の点からleft-to-right解析法の方がisland-driven解析法よりも優れているといえよう。本シミュレーションで用いたタスクに限り、このことはタスクの複雑さや語彙数に関係なく言えた。しかしタスクの複雑性と文認識率等の関係は別の機会に報告する^{15),16)}。

しかし、連続音声（特に会話音声）には構的には全くおかしい文が多く存在し構文知識のみでは対応できない場合があり、時間的に離れた複数の仮説を様々なレベルの知識を用いて処理できるisland-driven的処理も必須であろう。これに関してはSternらの研究^{13),14)}は興味ある。

今後はisland-driven的処理を行う意味主導型の構文・意味解析による会話音声理解を検討していく予定である。

参考文献

- 1) Woods, W. A.: Optimal Search Strategies for Speech Understanding Control, *Artif. Intell.*, Vol. 18, pp. 295-326 (1982).
- 2) Hart, P. et al.: A Formal Basis for the Heuristic Determination of Minimum Cost Paths, *IEEE Trans. Syst. Science & Cybernetics*, Vol. SSC-4, No. 2, pp. 100-107 (1968).
- 3) Paxton, W. H.: A Best-First Parser, *IEEE Trans. Acoust. Speech & Signal Process.*, Vol. ASSP-23, No. 5, pp. 426-432 (1975).
- 4) Lowerre, B. T.: The HARPY Speech Recognition System, Ph. D. thesis, Department of Computer Science, Carnegie Mellon University (1976).
- 5) Sakai, T. and Nakagawa, S.: A Speech Understanding System of Simple Japanese sentences in a Task Domain, *Trans. Inst. Elect. Comm. Engrs.*, Vol. 60-E, No. 1, pp. 13-20 (1977).
- 6) 本間 茂, 中津良平: 連続音声認識のための会話音声の特性解析, 音響学会講演論文集, 3-5-8 (1987. 3).
- 7) 中川聖一: 文脈自由文法のフレーム同期型構文解析法による連続音声認識, 信学論, Vol. 70-D, No. 5, pp. 907-916 (1987).
- 8) 大黒慶久, 中川聖一: left-to-right & top-down構文解析法とisland-driven & bottom-up構文解析法による連続音声認識の比較・検討, 信学会言語処理とコミュニケーション技報, NLC 87-12 (1987. 10).
- 9) 松本裕治, 田中穂積: Prologに埋め込まれたbottom-up parser: BUP, 情報処理学会自然言語処理研究会資料, 34-4 (1982. 8).
- 10) 渡原 茂, 小林 豊, 新美康永: 音声理解システムにおける単語予測方式—island-driven法について, 情報処理学会知識工学と人工知能研究会資料, 48-10 (1986).
- 11) 浮田輝彦, 石川憲洋, 中川聖一, 坂井利之: 音声による対話システムにおける発話の確認方法, 情報処理学会論文誌, Vol. 22, No. 6, pp. 589-595 (1981).
- 12) 中川聖一, 坂井利之: 音声自動認識に関する情報工学の諸考察, 情報処理学会論文誌, Vol. 21, No. 5, pp. 407-417 (1980).
- 13) Stern, R. M. et al.: Sentence Parsing with Weak Grammatical Constraints, *Proc. Int. Conf. Acoust. Speech & Signal Process.*, pp. 381-383 (1987).
- 14) Ward, W. H. et al.: Parsing Spoken Phrases Despite Missing Words, *Proc. Int. Conf. Acoust. Speech & Signal Process.*, pp. 275-278 (1988).
- 15) 大黒慶久, 中川聖一: 音韻認識率・文発声法・Perplexityおよび文認識率との相互関係, 信学会, 音声技報, SP 88-113 (1988. 12).
- 16) 大黒慶久, 中川聖一: 単語認識率・Perplexityおよび文認識率との相互関係, 音響学会全国大会講演論文集 (1989. 3).

(昭和63年5月25日受付)
(平成元年5月9日採録)



中川 聖一（正会員）

昭和 51 年京都大学大学院 博士課程修了。同年京都大学情報助手。昭和 55 年豊橋技術科学大学情報工学系講師。昭和 58 年助教授。工学博士。昭和 60~61 年カーネギー・メ

ロン大学客員研究員。音声情報処理、自然言語処理、人工知能の研究に従事。昭和 52 年電子通信学会論文賞受賞。著書：「情報基礎学詳説」（分担執筆、コロナ社）、「確率モデルによる音声認識」（電子情報通信学会）など。電子情報通信学会、日本音響学会、人工知能学会、IEEE、INNS 各会員。



大黒 廉久

昭和 62 年豊橋技術科学大学情報工学系卒業。平成 1 年同大学院修士課程情報工学専攻修了。現在リコー(株)中央研究所勤務。在学中は音声情報処理の研究に従事。