音声言語情報処理 27－5
（1999．7．23）

大語彙連続音声認識エンジン Julius における A* 探索法の改善

李 晃伸　　河原 達也

京都大学大学院 情報学研究科 知能情報学専攻

あらまし　　大語彙連続音声認識エンジン Julius における解探索アルゴリズムの種々の改善手法を提案し，その評価を行う．ヒューリスティックの非適格性から生じる探索誤りや探索失敗を解消するために，第2パスで探索が幅優先に陥るのを防ぐ enveloped best-first 探索を提案するとともに，第1パスで単語間 triphone を近似計算することで探索の高精度化を図る．高速化の面からは，第2パスの音響尤度計算におけるスコアでのビーム設定と，第1パスでの 1-gram 確率に基づく factoring を導入する．JNAS の 20,000 語タスクでの評価実験の結果探索誤りの多くが解消され，またその精度を落とさずに計算量を削減することができた．最終的に，実時間の 12.9 倍で 94.9%，monophone ではほぼ実時間で 84.2% の単語認識精度を得ることができた．

キーワード　大語彙連続音声認識，A* 探索，最適性

Improvements of A*-based search algorithm in LVCSR engine Julius

Akinobu Lee　　Tatsuya Kawahara

Graduate School of Informatics, Kyoto University

Abstract　　The recent improvements in our LVCSR engine "Julius" are shown. To ease search errors and search failure caused by dis-optimality of heuristics, an enveloped best-first search algorithm and an approximation of the inter-word context dependency on the 1st pass are proposed. More, score-based envelope beam for aoustic scanning on the 2nd pass and 1-gram factoring are introduced to decrease computational costs. Experiments on 20,000-word JNAS task show that most of the search errors are dissolved and the costs can be efficiently cut with little accuracy loss. The system achieved word accuracy of 94.9% in a real-time factor of 12.9, and using monophone model, 84.2% in nearly real-time.

key words　Large Vocabulary Continuous Speech Recognition, A* search, optimality

1 はじめに

大語彙連続音声認識，すなわち任意語彙発声の自動書き落し（ディクテーション）は，近年最も盛んに研究されている音声研究分野の一つである [1][2]．音声ワープロ・放送の自動書き起こしなどの応用が考えられるほか，そこで得られる要素技術や知見は対話処理や音声インタフェースなどの関連する音声研究を推進する基盤ともなる．国内でも研究機関どうしで共通のプラットフォームの整備が進み [3]，共有可能な数万語クラスの大規模モデルが提供されるなど [4]，研究のための環境が整いつつある．しかし日常の人間の自然な発話の認識を実現するには，言語モデル・音響モデル・認識アルゴリズムのそれぞれの面においてさらなる研究が必要である．

我々はその中で，大語彙連続音声認識のための認識アルゴリズムについて研究を行なっている．これまでに A* 探索に基づく 2 パスの解探索アルゴリズムを提案し [5]，認識エンジン Julius として公開してきた．しかし，その認識誤りの中に探索の誤りに起因するものが少なからず存在することが指摘されている [6]．また音声インタフェースなどの現実の応用を考えると，効率よく処理を行なう高速な探索を行なうことも重要である．

本稿ではこの Julius における A* 探索法について，探索誤りの解消および高速化の両面から改善法をいくつか提案・検討する．そして語彙数 20,000 の JNAS タスクにおいて評価を行なった結果を報告する．以下，第 2 節で Julius における A* 探索アルゴリズムと現在の問題点を述べた後，第 3 節では A* 探索の誤りを解消する高精度化手法について，第 4 節では高速化手法について述べ，それぞれ評価した結果を示す．第 5 節ではそれらを統合した認識システムとしてのトータルの性能を示す．

2 大語彙連続音声認識における A* 探索

2.1 A* 探索に基づく音声認識

A* 探索に基づく音声認識を概観する．A* 探索は best-first 探索の一種であり，評価値の最も高い仮説を展開することによって探索を進める．仮説の評価値には，未探索部分のスコアのヒューリスティックな推定値を加える．すなわち，仮説 n について，その評価値 $\hat{f}(n)$ を次のように定義する．

$$\hat{f}(n) = g(n) + \hat{h}(n)$$

ただし，$g(n)$ は既に展開された仮説のスコア（対数尤度），$\hat{h}(n)$ は未展開部分の推定スコアである．

探索時に現れ得る全ての仮説に対して $\hat{h}(n)$ をあらかじめ計算しておく必要があるため，処理は典型には 2 パス構成となる．

A* 探索を実現するために $\hat{h}(n)$ に求められる条件は以下の通りである．

(1) 適格性（実行可能性条件）

最適解を得られることが保証されるためには，推定スコア $\hat{h}(n)$ を実際のスコア $h(n)$ より厳しくしてはならない．

$$|\hat{h}(n)| \leq |h(n)| \tag{1}$$

(2) 推定関数の品質

推定スコアが実際のスコアに近いほど，無駄な仮説を展開せずに素早く最適解を見つけることができ，ヒューリスティックとして強力である．

$$|\hat{h}(n)| \approx |h(n)| \tag{2}$$

(3) 処理量

全体の処理効率は，探索だけでなくヒューリスティック計算も含めて評価される．ヒューリスティックの計算量が探索の際の計算量と同等ではパスを分ける意義は薄いものとなる．

このように，A* 探索の性能や探索の成否は使用するヒューリスティックに大きく依存する．

2.2 大語彙連続音声認識エンジン Julius

A* 探索に基づく大語彙連続音声認識エンジン Julius の概要を述べる．図 1 に全体の構成図を示す．

認識処理全体は 2 パスで構成される．まず第 1 パスはヒューリスティック計算であり，入力（ポーズまで）を完全に処理して中間結果をトレリスの形で出力する．第 1 パスのモデルとして，処理の簡便さから 2-gram と単語 HMM の木構造化辞書を用いる．triphone 音素環境依存モデルを用いる場合は，単語内の環境依存のみを考慮して木を構築する．探索は left-to-right にフレーム同期ビーム探索を行う．

第 2 パスでは 3-gram を用いて探索を行う．逆向きに探索を行うことで，仮説の評価に第 1 パスで計算された尤度を未探索部の先読みとして反映させることができ，仮説の入力全体に対する評価を得ながら探索を進めることができる．第 1 パスで探索空間はかなり絞られており，かつ完全な先読みができるため，単語単位の best-first なスタックデコーディングを行う．

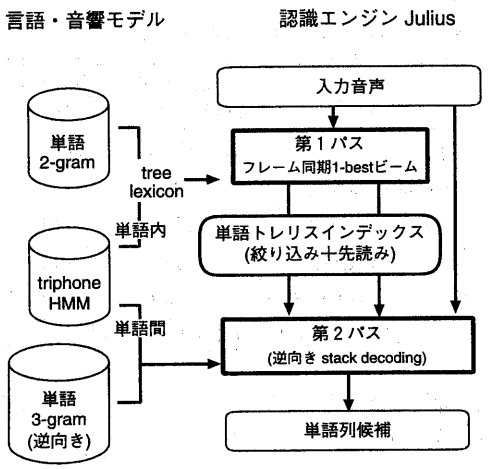

図 1: Julius の構成

2.3 問題点

Julius における探索上の課題について考察する．そもそも A* 探索の観点からは，ヒューリスティックとして用いる単語 2-gram 言語確率は単語 3-gram に対して適格性の式 1 を満たさない．また音響モデルについても単語内のみ依存を考える triphone モデルと単語間を考慮した triphone ではその音響スコア差は大きい．このため以下のような誤りを生じる．

探索失敗 探索が入力終端まで達せず，探索をあきらめてしまう場合．ヒューリスティックのスコアが部分的に低くなる部分で探索が幅優先に陥り，前へ進まなくなる（この場合第 1 パスのベスト解で代用）．特に長い入力で顕著であった．

探索誤り 最尤でない文候補が解として得られてしまう場合．ヒューリスティックが非適格なため，最適でない解が先に得られてしまい，真の最適解が得られる前に探索が終了してしまう．

これまでの Julius ではこのようなヒューリスティックの非適格性に対して，2-gram と 3-gram で異なる重みを与えるたり，単語間で生じる音響スコア差を埋めるために単語間遷移に対して insertion penalty を課すなどのスコア調整や，最適解を得るために 10 程度の複数文候補を得てからソートして 1 位を出力するなどの方法で対処してきた．

しかし，南條ら [6] がこれまでの 5,000 語彙での評価実験において誤り原因を同定したところ，正解文よりもスコアの低い仮説が認識結果として出力されるといった明らかに探索に起因する誤りが，認識誤り全体の約半数にのぼることが示された．この傾向は第 1 パスのビーム幅を大きくした場合や語彙数が 20,000 の条件下でも同様であり，さらに高精度な探索を目指すには探索アルゴリズムにさらなる改良が必要であることを示唆している．

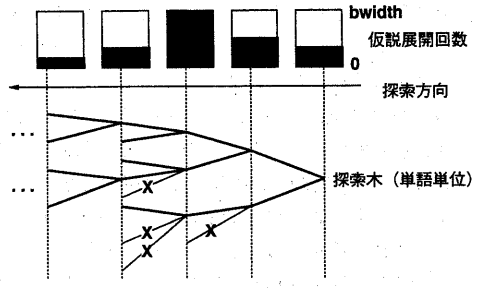

図 2: Enveloped best-first 探索

また高速性についても，効率よく探索を行なうために処理を簡略化し，また不要な処理を除外することは重要である．

3 探索アルゴリズムの高精度化手法

大語彙連続音声認識における A* 探索の高精度化手法を提案する．戦略としては，まず探索失敗になるケースを防いだうえで，探索誤りを軽減するためにヒューリスティック自身を高精度にする．以下，二つの高精度化手法を述べる．

3.1 Enveloped best-first 探索

認識精度向上の面からは，探索が失敗するよりは，最適でなくても部分的に正しいであろう何らかの解がとにかく得られるほうがよいと考える．このためには，探索の幅に何らかのビームを設けることで，探索を幅優先に陥らせずに強制的に探索が進むようにすることが有効である．

しかし，通常のヒューリスティックビーム探索のように単語同期に仮説を展開する場合，これまで best-first に正しい解が得られていたサンプルに対しては処理の無駄が多い．単語同期に展開することで best-first に比べて無駄な仮説が展開されることになる．

そこで，探索は best-first のままで，ビーム幅を仮説長（＝単語数）ごとに設定することを提案する．具体的には，探索において仮説スタックから取り出され

た仮説の数を仮説長ごとにカウントし，その数がある長さにおいて上限値に達したら，そこより短い仮説をスタックから破棄する（図2）．

本稿ではこの探索法を enveloped best-first と呼ぶ．探索が幅優先に陥りそうになったときに，それより短い仮説を全て枝刈りすることで探索を強制的に前に進め，探索失敗を防ぐ効果と共に，うまくヒューリスティックに導かれる場合にはなんら影響を与えない．A* 探索の best-first 性を損なわずに，長い入力に対しても最悪でも何らかの解が得られることを保証するものである．

3.2 第1パスでの単語間 triphone の近似計算

ヒューリスティック最適性を改善することを考える．特に音響モデルについては，音素環境依存性を単語内のみを考慮することで全体のスコアは異なるため，ヒューリスティックとしての性能が損なわれている部分がある．探索の精度の面からは，第1パスから音響モデルとして triphone を使用すればヒューリスティックが抜本的に改善されると考えられる．

ただし，探索の最初のパスで単語間の音素環境依存性を扱うには非常に煩雑な処理が必要である．

- 単語の先頭では左コンテキストに
- 単語終端では右コンテキストに

応じて異なる複数の音素モデルを別々に扱わなければならない．特に第1パスでは，認識処理中は単語終端において右コンテキストが音響的に常に未知であるため，単語終端ごとに登場し得る全コンテキストに対して別々のモデルを並列に処理するなどの機構が必要であり，大語彙では処理が複雑化する．

そこで，第1パスにおいても単語間で triphone モデルを近似的に適用した音響スコアを与えることを考える．ただし処理量を抑えるため，単語の末尾ではその左コンテキストで出現し得る全 triphone の音響スコアの最大値を用いることとする．単語の先頭では，直前単語の最終音素ごとに音素モデルを切り替えてその音響スコアを適用する．

単語先頭では，1-best 近似による直前単語の誤差が存在するため，常に正しい triphone が適用されるとは限らない．また Viterbi パスはベストのもののみを扱う．このためスコアは真の単語間 triphone に比べて誤差を含むが，ヒューリスティックの性能としては十分改善されると見込まれる．また単語終端については可能な最大値を与えるため真の音響スコアよりも常に厳しくない値となり，ヒューリスティックの適格性の面からも好ましい．

表 1: Julius における探索の高精度化手法の評価

探索手法	単語誤り率 total (pass1)	平均時間 (秒)
(a) base	11.1 (21.4)	117.2
(b) base.fixed	8.8 (21.1)	52.8
(c) +enveloped	8.0 (21.1)	49.9
(d) +IWCD1	6.5 (14.9)	89.5
(e) +IWCD2fix	6.0 (14.9)	218.3

1サンプル当たり平均長＝5.8秒

3.3 評価実験および考察

これらの高精度化手法を実装し，ディクテーション実験により評価した．タスクは JNAS の語彙数 20k の新聞記事読み上げ音声のディクテーションである．モデルは「日本語ディクテーション基本ソフトウェア 1998 年度版」[4] のなかから，言語モデルは 75ヶ月分の新聞記事で学習した cutoff-1,1 の単語 3-gram，音響モデルは 2000 状態・16 混合分布の性別依存 triphone モデルを用いた．テストセットは IPA-98-TestSet[4] のうち，男性話者のサンプル（23 名・計 100 文発声）を用いた．未知語率は 0.44% である．集計は自動集計ツール [7] で機械的に行なっており，人間が目視で照合した場合と比べて 0.5% 程度誤り率が増加する．実験は UltraSPARC 300MHz 上にて行なった．なお実験ではビーム幅や言語重みなどの探索時のパラメータは精度に最適化してある．

認識実験の結果を表1に示す．(a) は改善前の Julius[1] である．(b) はアルゴリズムは同じままで実装上およびコード上の改善・最適化を施したものである．

Enveloped best-first 探索を行なったところ，認識精度に 0.8% の改善が認められた (c)[2]．特に長い入力に対して，100 文中 12 文存在した探索失敗が 1 個に減少した．このことから，特に enveloped best-first 探索は非適格なヒューリスティックに対しても失敗の少ない頑健な探索が行なえることが示された．

次に第1パスで単語間 triphone の近似計算を導入したところ，第1パスの認識誤りが 21.1% から 14.9% に減少した (d)．その結果最終的な単語誤り率も 8.0%

[1] IPA'97 に含まれる julius-1.1

[2] IPA'98 に含まれる julius-2.1

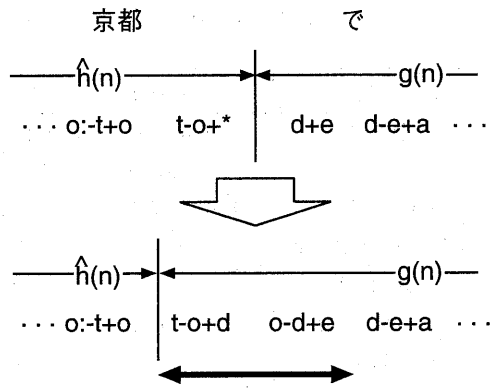

図 3: ヒューリスティック接続部の音素環境依存性

から 6.5% に改善され，ヒューリスティックの改善による探索精度の向上が確認できた．ただし，単語終端で扱うモデル数の増大と単語先頭での動的な triphone 切り替えのオーバーヘッドのため，処理時間は全体で 2 倍近くに増大している．

ここでさらに，第 2 パスの前向き尤度 $g(n)$ とヒューリスティック $\hat{h}(n)$ の接続部についても，音素環境依存性を考慮した評価値計算を導入した．通常は図 3 上部のように単純に接続して評価値を求め，あとで $g(n)$ の更新時に依存を考慮して再計算するが，これを単語展開の時点で厳密に計算を行なう．結果は表 (e) にあるように，認識率に 0.5% と僅かな改善が見られた．しかし単語展開ごとに全ての仮説候補について triphone を参照して音響マッチングを行なうためコストが非常に高い処理であり，(d) からさらに 2 倍以上の計算時間を要した．

4 探索処理の高速化手法

次に，探索における処理の高速化手法について，第 2 パスの尤度計算におけるスコアでのビーム設定と，第 1 パスでの 1-gram 確率に基づく factoring の 2 手法を検討する．

4.1 1-gram 確率に基づく factoring

第 1 パスの木構造化辞書を用いた認識では，図 4 のように，本来単語の終端（= 木の末端ノード）でしか与えられない言語スコアを木の途中のノードにおいても最大値を順に清算していくことで前倒しに与える factoring[8] が探索精度の面で有効である．しかし

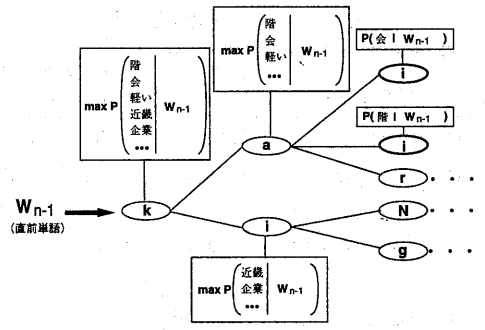

図 4: 2-gram factoring

単語の先頭 (= 木の根元のノード) では常に全単語に対する言語スコアを計算する必要があり，コストが大きい．特に単語 N-gram 確率は直前単語に依存するため，単語終端が現れるたびに全単語の 2-gram 確率を計算することになり，特に大語彙では計算量が問題となる．

この factoring を単語 1-gram 確率で行うことを考える．1-gram 確率は仮説のコンテキストに依存しないので，探索前に木構造化辞書の全ノードに対して factoring スコアをあらかじめ計算しておくことができ，言語確率の計算量を抑えることができる．正確な 2-gram 確率は，探索時に単語終端に達したときにはじめて計算して与える．ただし factoring の最適性は失われるため，認識精度は低下すると考えられる．

4.2 スコアに基づくビームの設定

Julius では第 2 パスで入力に対して音響モデルを再照合しているが，この仮説の前向きスコア計算において，スコアに基づく枝刈りを導入することで，不要な音響マッチングを抑えることができる．入力全体の各フレームごとにその時点での最大スコアを保存しておき，前向き尤度計算の際にはそこから一定のスコア幅内に収まる部分のみについて計算を行なう．主に単語単位の仮説展開を行なう 1 パス探索で用いられる手法である [9] が，Julius の best-first 探索においても同様に有効であると考えられる．

4.3 評価実験と考察

これまでに述べた 2 つの高速化手法を認識実験を通して評価した．実験条件は 3.3 節と同一である．前出の表 1 の (c) ("+enveloped") をベースとして，前節で述べた高速化手法を導入した結果を表 2 に示す．

表 2: 探索の高速化手法の評価

探索手法	単語誤り率 total (pass1)	平均時間 (秒) pass1 / pass2
(a) base2	8.0 (21.1)	39.1 / 10.8
(b) +1factor	8.8 (26.1)	35.7 / 10.0
(c) +scorebeam	8.5 (26.1)	35.7 / 5.7

注：(a) base2 は 表 1 の (c) に対応

1-gram 確率に基づく factoring を行なったところ (b)，第 1 パスの計算時間を実時間分（発話長 ×1）ほどが削減できた．triphone では第 1 パスでは音響モデルの照合が処理量の多くを占めることから，言語スコアの計算量を大きく削減できたと考えられる．一方，認識誤りは第 1 パスで 5.0% と大幅に増加したが，第 2 パスでは 0.8% の増加にとどまり，また第 2 パスの探索時間にも大きな変化はみられなかった．これは Julius が単語トレリスインデックスを中間表現として用いているため，第 1 パスの誤りを第 2 パスで回復可能であるためと考えられる．このことから，1-gram 確率に基づく factoring は，厳密な 2-gram を用いる factoring に比べて言語確率の計算量を削減でき，かつ誤差についても Julius では第 2 パスでほぼ回復可能であることが分かった．

またスコアに基づいてビームを設定したところ (c)，第 2 パスの処理時間が半分近くに短縮された．これまでは，第 2 パスで，全入力フレームに対して音響的照合を行ない仮説の（前向き）尤度を求めていたのに対して，各フレームごとにビームを設定することで，結果的に必要な範囲だけを照合できるようになったためである．認識精度に関しても，適切なビーム幅であれば，低下させずに処理量を抑えることができた．

以上から，ここで導入した 2 つの高速化手法は，認識精度を大きく下げずに計算量を削減できる有効な手法であることが示された．

5 システム性能

これまでに述べた改善手法を全て施した認識エンジン Julius の，最終的な認識システムとしての性能を評価する．テストセットは男性話者に加えて，女性話者に対しても評価を行なった．結果を表 3 に示す．

「高精度」は本稿で提案した全ての探索改善手法を施した精度重視の設定である．認識精度は男女平均で 94.9% を達成した．特に処理時間に関しては，第

表 3: システム性能一覧

システム設定	高精度	高効率
音響モデル	triphone	monophone
	2000×16	192×16
探索改善手法	高精度 + 高速	高速
探索打ち切り候補数	10-best	1-best
認識時間 (秒)	74.6	6.6
認識時間 (×RT)	12.9	1.1
認識率 (男性)	94.3	82.6
認識率 (女性)	95.4	85.7
認識率 (平均)	94.9	84.2

認識率= word accuracy

3.2 節で探索の高精度化手法のみを評価した際に顕著だった第 2 パスの単語間 triphone の音響照合時間の増加が，スコアに基づくビームの設定の影響でほとんど抑えられるという効果が認められた．

「高効率」は処理効率を重視して monophone を使用し，改善手法も高速化のみを適用する設定である．特に 1-gram factoring の効果から，ほぼ実時間処理で 84.2 % の認識精度を得ることができた．monophone 使用時は，探索処理全体に占める言語確率の計算量の割合が相対的に高いため，実時間探索の実現には 1-gram factoring が相対的に大きな意味を持つ．

6 まとめ

大語彙連続音声認識エンジン Julius における解探索アルゴリズムについて，高精度化と処理の高速化の観点から種々の改善手法を提案し，20,000 語の読み上げ音声ディクテーションタスクでそれらの評価を行なった．

探索誤りの改善については，

- 第 2 パスで Enveloped best-first 探索
- 第 1 パスで単語間に triphone を近似的に適用

の 2 つのアプローチで探索誤りおよび探索失敗の解消を図った．前者は探索の安定化に寄与し，best-first 性を保ちながら探索が幅優先に陥るのを防いで探索を強制的に前へ進める効果が得られた．後者については，ヒューリスティック最適性の改善による探索精度の向上が確認できた．両手法によって認識誤りは

8.8% から 6.0% へ改善され，探索に起因する誤りの多くを解消できることか示された．

探索処理の高速化手法については，

- 第 1 パスで 1-gram 言語確率に基づく factoring を導入する
- 第 2 パスの音響尤度計算時にスコアにビームを設定する

の 2 手法を導入し評価した．その結果，前者は第 1 パスの言語モデルの計算量を，後者は第 2 パスでの音響モデルの計算量を，それぞれ削減することができた．実験の結果，認識誤りの増加は両手法を合わせて 0.5% と非常に小さくて済み，これらが計算量の削減手法として有用であることが示された．

これらの改善を施した認識エンジン Julius を用いた認識システムの最終的な単語認識精度は，最も高精度な設定で 94.9%，高速性を重視した設定では，ほぼ実時間で 84.2% を達成した．

今後は，主に高速度化について，引き続いて音響モデルの出力確率計算の高速化や tied-mixture モデルへの対応を行なう予定である．

謝辞

本研究は情報処理振興事業協会 (IPA) の「日本語ディクテーションの基本ソフトウェアの開発」(代表者：鹿野清宏教授) の援助を受けて行われた．モデルを提供して頂いた関係各位に感謝します．

参考文献

[1] 西村雅史, 伊東伸泰. 単語を認識単位とした日本語ディクテーションシステム. 電子情報通信学会論文誌, Vol. J81-D-II No.1, pp. 10–17, 1998.

[2] 堀貴明, 岡直生, 加藤正治, 伊藤彰則, 好田正紀. 大語彙連続音声認識のための音素グラフに基づく仮説制限法の検討. 情報処理学会論文誌, Vol. 40, No. 4, pp. 1365–1373, 1999.

[3] 河原達也, 李晃伸, 小林哲則, 武田一哉, 峯松信明, 伊藤克亘, 伊藤彰則, 山本幹雄, 山田篤, 宇津呂武仁, 鹿野清宏. 日本語ディクテーション基本ソフトウェア (97 年度版). 日本音響学会誌, Vol. 55, No. 3, pp. 175–180, 1999.

[4] 河原達也, 李晃伸, 小林哲則, 武田一哉, 峯松信明, 伊藤克亘, 山本幹雄, 山田篤, 宇津呂武仁, 鹿野清宏. 日本語ディクテーション基本ソフトウェア (98 年度版) の性能評価. 情報処理学会研究報告, 99-SLP-26-6, 1999.

[5] 李晃伸, 河原達也, 堂下修司. 単語トレリスインデックスを用いた段階的探索による大語彙連続音声認識. 電子情報通信学会論文誌, Vol. J82-D-II No.1, pp. 1–9, 1999.

[6] 南條浩輝, 李晃伸, 河原達也. 大語彙連続音声認識における認識誤り原因の自動同定. 情報処理学会研究報告, 99-SLP-27-12, 1999.

[7] 伊藤克亘, 山本俊一郎, 鹿野清宏, 中村哲. ディクテーションにおける日本語の特質を考慮した単語正解率判定ツール. 日本音響学会春季論文集, Vol. 3-Q-19, , 1999.

[8] J.J.Odel, V.Valtchev, P.C.Woodland, and S.J.Young. A one pass decorder design for large vocabulary recognition. In *Proc. ARPA Human Language Technology Workshop*, pp. 405–410, 1994.

[9] P.S.Gopalakrishnan, L.R.Bahl, and R.L.Mercer. A tree search strategy for large-vocabulary continuous speech recognition. In *Proc. IEEE-ICASSP*, pp. 572–575, 1995.