

信頼度基準デコーディングを用いた高効率な単語グラフ生成法

李 晃 伸[†] 河原 達 也^{††} 鹿野 清 宏[†]

音声認識結果の単語集合をグラフ形式で表現する方法が広く用いられている。本報告では、2パストリートレリスデコーダーにおいてオンラインで得られる単語信頼度を用いて単語グラフを効率よく生成する手法を提案する。後段パスのスタックデコーディング中に、(1) 同一個所に現れる同一単語候補をマージすることで探索中に動的に単語グラフを生成し、(2) それまでに得られた仮説パスと重複する部分文仮説の探索を中断する。さらに、(3) 探索中に得られる単語信頼度に基づく枝刈りにより、信頼度の低い単語の展開を抑制する。これらの手法により、不要な探索を抑えて効率よく単語グラフを生成する。認識実験の結果、グラフの動的生成により N-best 候補に比べてより少ない計算量で多くの単語仮説を残すことができ、最小の GER を改善できることが示された。信頼度による枝刈りは一定の効果が確認できたが、探索中の信頼度を用いることによる信頼度値の不安定さの影響が大きいことが分かった。

An Efficient Word Graph Generation using Confidence Scored Decoding

AKINOBU LEE,[†] TATSUYA KAWAHARA^{††}
and KIYOHRO SHIKANO[†]

An efficient search algorithm to generate word graph on tree-trellis decoder using search-time confidence score is investigated. In the 2nd pass of tree-trellis search, the stack decoding is modified to fixate word candidates on-the-fly in order to generate word graph dynamically by merging paths by sharing same words at the same time frame. The hypotheses will be terminated if the last words has already been in the current partial word graph and expanded so far. Furthermore, a word pruning by confidence score is applied at word expansion to suppress unpromising words to appear. Experimental results show that the proposed method can generate a word graph faster than using N-best sentence list, and also can generate more word graphs to improve minimal GER. Pruning by confidence score showed a certain effect to reduce the number of word expansion, but has little effect on the total search time due to the unstableness of the search-time confidence scoring especially for a short words.

1. はじめに

音声認識システムや音声対話システムを我々の日常環境でよりストレス無く動作させようとする場合、ユーザの自由で自然な発話を認識する必要がある。このような自発的な発話は ill-formed であり、発話様式やスタイルも様々であるため、音声認識において曖昧性が存在し常に最良の結果を一意に得ることが難しい。そのため、候補を複数出力してそれらを意味解釈などする必要がある。

音声認識において複数の認識結果を表現する形式として最も単純なのは、上位 N 個の文候補を並べた N-best リスト形式であるが、近年は単語グラフ形式が

よく用いられている。単語グラフ形式は N-best リストに比べてより多くの候補をコンパクトに表現でき、またビーム探索や単語対近似といった主要な 1パスの大語彙連続音声認識手法とのアルゴリズムの親和性が高く、その生成法や後処理の方法について盛んに研究がなされている¹⁾²⁾。

後段での言語処理や意味解釈を考慮すると、最終的には正解の可能性の高い単語の並びをなるべく小さい単語グラフとして得られることが望ましい。しかし一方で、近年広く用いられている単語事後確率に基づく単語信頼度の付与³⁾では、一般に同時刻に存在する他の競合候補からの距離や尤度比を基準とするため、一位以外の多くの仮説候補を出力する必要があり、精度良く求めるためには大きなグラフが必要とされる。このため、典型的な単語グラフ生成系は、(1) 認識処理による巨大な単語グラフの生成、(2) グラフ上での単語信頼度の算出、(3) 後処理 (リスクアリング、pruning⁴⁾、最小化⁵⁾ など) といった複数の手順を経る。

本研究では、2パスのトリートレリスデコーダーに

[†] 奈良先端科学技術大学院大学情報科学研究科
Graduate School of Information Science, Nara Institute
of Science and Technology

^{††} 京都大学学術情報メディアセンター
Academic Center for Computing and Media Studies,
Kyoto University

において、単語グラフを効率よく生成するアルゴリズムを提案する。後段のスタックデコーディング中に、探索木の枝上の同一区間に存在する同一単語をマージすることで、探索過程において動的にグラフを生成する。また、その部分文仮説の末端が他の枝と完全に重複する場合、それ以上の仮説展開を中断する。これにより、類似した仮説の展開を抑えて、N-best リストよりも多くの単語仮説をグラフ形式で得る。さらに、探索中に得られる単語信頼度¹¹⁾に基づいて展開単語の枝刈りを行うことで、探索終了と同時に効率良く単語グラフを生成する。

以下、2節で音声認識における単語グラフの生成方法について概観し、3節でスタックデコーディングにおける単語グラフの動的生成方法を提案する。4節で探索中の単語信頼度に基づく枝刈りによるデコーディングの高効率化について述べる。5節で評価実験を行った結果を示し、6節でまとめる。

2. 既存の単語グラフ生成法

2.1 N-best 文リストからの構築

単語グラフを生成する最も単純な方法は、認識結果のN-best 候補リストを圧縮する方法である。音声認識器の出力する上位N個の文候補のうち、同じ位置、あるいは似た位置の単語どうしを共有化することで単語グラフを形成できる。N-best リストに比べて単語数を少なくでき、かつ単語の共有による単語間接続の制約緩和によってより多くの候補を表現できる。

ただし、一般にN-best 候補リストは似通った候補が多くなる傾向がある。十分な単語候補の種類をもつグラフを得るにはNを大きくする必要があり、多くの認識処理量を要する。特に、信頼度計算においては多くの単語候補が必要なため、大量のNを求める必要があり、計算効率が非常に悪い。

2.2 1パス単語グラフ探索

単語グラフの生成法として最も広く用いられている手法は、1パス認識処理の過程で可能な単語仮説と接続関係を動的に生成・保存することで単語グラフを構成する方法である¹⁾。典型的な方法は、1パスのフレーム同期ビーム探索において、出現した単語候補をその接続関係とともに保存する。認識処理終了後に、最終フレームにおいて残った単語から単語単位でバックトレースを行い、探索中に出現した単語仮説全体を単語グラフの形で出力する。単語対近似などの単語履歴近似を行うことで、履歴が似た仮説の束ねが行われ、結果として仮説空間は、コンテキストと出現位置の同じ候補が束ねられた、単語仮説のグラフとして表現される。この単語グラフは、単語ごとに始端フレームと言語・音響スコアが保存されている。

このようにして生成される単語グラフは多くの場合巨大であり、様々な後処理手法が研究されている。最終的なベスト候補を得るために、より高次のモデルや

表1 探索方法の比較
Table 1 Comparison of search methods

項目	探索手法	
	ツリートレリス	単語グラフ+ リスコアリング
前段パス計算量	小 (1-best 近似)	大 (単語対近似)
中間表現	単語トレリス	単語グラフ
展開単語	ある時刻上の全単語	制約あり
音響スコア再計算	必須	必ずしも必要ない

単語信頼度に基づいたグラフ上での最尤仮説探索が行われる⁶⁾⁷⁾。また、グラフのサイズを縮小してより最適なグラフを構築する方法として、最尤スコアとのスコア比が一定以上の単語を除外する graph pruning¹¹⁾⁴⁾や、単語誤り最小化基準により単語グラフを効率よく最小化してよりコンパクトなラティス形式 (confusion network) を抽出する手法が提案されている⁵⁾。このように、単語グラフ探索に基づく手法は一般に、1パスビーム探索による単語グラフ生成およびグラフの後処理の複数段階からなる。

3. トリートレリス探索におけるグラフ生成

トリートレリス探索⁸⁾⁹⁾において単語グラフを生成する手法を考える。最も簡便な方法は、最後まで探索を行いN-best 候補のリストを得てから同一箇所の同一単語をマージする方法であるが、十分なサイズのグラフを得るには大量の文候補を見つける必要があり計算効率が良くないため、探索過程において動的に単語仮説を確定し、単語グラフを生成することが望ましい。

トリートレリス探索法と単語グラフに基づく認識手法との比較を表1に示す。前者の前段パスのビーム探索は後者の1パス単語グラフ探索に、前者の後段パスであるスタックデコーディングは生成されたグラフ上でのリスコアリングに対応する。単語グラフ探索では、前段パスでグラフを生成して単語仮説空間を決定するため、最終的な認識精度は前段パスの認識精度に大きく依存する。一方、トリートレリス探索では可能な単語終端を全て残し、後段パスで最終的に確定する。このため、前段パスの誤りの影響が小さく、1-best 近似等を用いてより少ない計算量で計算できる。また後段パスでは再計算により正確なN-best 候補を求めるため、その探索過程から動的に単語グラフを生成することで、より精度の高い単語グラフが最終的に得られる可能性がある。ただし、後段パスの計算量はリスコアリングに比べて大きいと見込まれる。

3.1 スタックデコーディングにおける単語仮説の逐次確定

第2パスのスタックデコーディングにおいて探索中に単語グラフを動的に生成する方法を述べる。スタックデコーディングにおける探索空間は、始端から終端へ向かって一方向に進む、単語を単位とする木構造で表される。この探索木に含まれる単語仮説をスタック中の仮説とは別に逐次確定・保存していくことで、単語

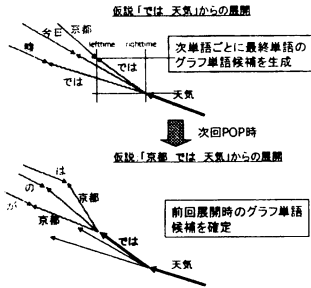


図1 グラフ単語候補の遅延確定
Fig.1 Delayed decision of a word candidate on graph.

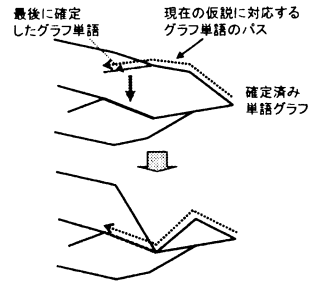


図2 スタックデコーディングにおける単語グラフの動的生成
Fig.2 Dynamic generation of word graph on stack decoding.

グラフを動的に生成する。以後、このグラフの一部として保存されていく単語候補をスタック上の仮説中の単語仮説と区別して「グラフ単語」と呼ぶ。各グラフ単語は、始末端フレーム、音響尤度、始端からの累積スコア、前後に接続しうるグラフ単語のリストを持つ。

単語グラフを動的生成するためには、各単語仮説の境界を探索中に確定する必要がある。ここでは、単語グラフ探索と同様に単語対近似を仮定する。すなわち、単語境界が直前・直後の単語のみに依存するという仮定を導入することで、探索中に単語境界を逐次確定する。実際のグラフ単語の確定の様子を図1に示す。単語展開時に、単語境界の後続単語依存性を考慮して後続単語それぞれに対して別々の候補を生成するが、この時点では後続単語の境界が未確定であるため登録は行わず、次に仮説がスタックから取り出されたときに初めてグラフ単語として確定される。

3.2 単語グラフの動的生成

逐次確定を行っていったとき、同一単語が同一区間に出現したとき、それぞれの枝を統合する。具体的には、ある部分文仮説から展開する際に、その時点で確定したグラフ単語と同じ単語が、それまでに確定したグラフ単語集合中の同一区間に既に存在する場合、それらをひとつにマージする。これによって木構造中の類似部分を動的に束ね、グラフを動的に構築する。このグラフ生成の様子を図2に示す。探索終了後、入力終端まで達していない枝を削除することで、単語グラフが完成する。

本手法では、探索中に部分文仮説の時点でマージを行うため、途中まで有望であったが最終的な N-best 文仮説候補には残らないような単語仮説も残すことができる。このため、同じ探索量でより多くの仮説をグラフに残すことができると予測される。

3.3 重複仮説の探索中斷

さらに、部分文仮説の履歴がマージされた時点で探索を中斷する方法を提案する。探索の制約として単語 3-gram を用いる場合、言語制約のコンテキストとして有効なのは 2 単語のみである。また単語対近似を導入すれば、Viterbi パスの履歴依存距離は 2 単語より

遠くへは依存しないと仮定できる。よって、ある部分文仮説において、その末端の 2 単語と同じ単語系列が他の部分文仮説において既に同一区間に存在し探索された後であれば、その仮説から以降に展開される仮説系列は以前に展開したものと類似する可能性が高い。

前節で述べたグラフ単語のマージが行われた時点で、さらにその後続単語と同一の単語がマージされたグラフ単語中に既に存在する場合、すでにそのコンテキストからの展開がなされている重複仮説であるとみなして、そこから単語を展開せずにそのパスの探索を打ちきる。これにより、それ以降の探索においてすでに探索されたものと似た仮説が探索中に繰り返し展開されることを防ぎ、より少ない探索数で効率よくグラフが生成できると考えられる。

ただし、スコア上有望な仮説であってもそれまでにグラフに登録してあれば仮説の探索は中斷されるため、最尤の文仮説が最終的に得られる保障はなくなる。

3.4 具体的アルゴリズム

これまでに述べた単語グラフ生成の具体的なアルゴリズムを以下に示す。

- (I) グラフ単語集合を空に初期化する。
- (II) 初期仮説を生成してスタックに入れる。
- (III) 以下の (1)~(7) を文候補が目的数得られるまで繰り返す。
 - (1) スタックから最もスコアの高い部分文仮説 $S = (w_1, \dots, w_n)$ を取り出す。入力端に達していれば文候補として出力し、以下の処理をスキップする。
 - (2) 取り出された仮説の保持する未登録のグラフ単語候補 $WC(S)$ を参照する。これは仮説中の単語 w_{n-1} に対応する。
 - (3) S の前向きトレリスを更新して $WC(S)$ の境界時間を確定する。
 - (4) (既に確定された) グラフ単語集合の中に $WC(S)$ と同じ単語が同一区間に存在する場合、それらをマージする。両者の単語接続情報を結合した新たなグラフ単語候補を生成して $WC(S)$ に

格納する。元のグラフ単語は破棄する。

- (5) $WC(S)$ をグラフ単語集合に追加する。
- (6) 重複仮説判定を行う。もし前ステップでグラフ単語がマージされ、かつその後続単語 (w_n) が $WC(S)$ の直左コンテキストとして既に存在していれば、この仮説を破棄して以下の処理をスキップする。
- (7) 次に接続しうる次単語集合を単語トレリスから抽出し、それぞれに対して下記の (a) から (d) の手順で仮説展開を行う。
 - (a) 現仮説に次単語 w_{n+1} を接続した展開仮説 $S' = (w_1, \dots, w_n, w_{n+1})$ を生成する。
 - (b) スコアが最も高くなる w_{n+1} と w_n の境界フレームを単語トレリスより求め、仮説スコアを更新する。
 - (c) w_n の始終端フレームとスコア情報を新たな未確定単語グラフ候補 $WC(S')$ として生成し、 S' に格納する。
 - (d) S' をスタックに入れる。

4. 単語信頼度に基づく枝刈り

単語信頼度を利用して、より効率よく単語グラフを生成する手法を考える。信頼度の高い単語を優先的に認識結果として残すことで、より効率よく精度の高い単語グラフを求めることができると考えられる。

探索における単語信頼度の利用方法として、単語グラフ上で単語信頼度を仮説のスコアに組み入れて探索する手法が提案されている⁶⁾⁷⁾¹⁰⁾。しかし、探索性能が単語信頼度の精度に大きく依存ようになるため、精度の高い単語信頼度を得るために、大きなサイズのグラフを得る必要がある。また、単語事後確率に基づく単語信頼度の付与はグラフ全体の尤度が必要なため、単語グラフ全体が得られた後の処理となり、計算効率の面で不利である。

一方、探索中にオンラインで単語信頼度を求める方法が提案されている¹¹⁾。これは A*探索に基づくスタックデコーディング探索において、単語展開時にその同一時間フレーム上に存在する単語仮説候補集合の (ヒューリスティックを含む) 仮説評価値の分布から単語事後確率を求める手法である。グラフ全体のスコアを用いないため、探索中に単語仮説を展開する時点で単語信頼度を得られる。

ここで、上記のスタックデコーディング中に得られる単語信頼度に基づく枝刈りを導入する。単語展開時に信頼度がしきい値以下の単語仮説を除外することで、信頼度の低い単語の展開を抑えてより少ない探索ステップ数で単語グラフを生成する。従来手法は認識処理により単語グラフを生成した後に単語信頼度を求めるが、本手法では単語信頼度を求めながら探索を行い、より効率の良い単語グラフを生成する。

関連研究として、スタックデコーディング中に単語

の信頼度を推定しながら言語モデルの重みを適応的に変化させる手法が提案されている¹²⁾。この研究では単語信頼度を単語の正解・不正解時の尤度分布から決定木を用いて算出しているが、本研究では単語事後確率に基づく信頼度を用いており、さらに単語信頼度は枝刈り条件のみに用いている点異なる。また、本手法はローカルスタックの設定に近い効果を持つと考えられるが、しきい値がスコア比や数でなく尤度分布によって定められる点異なる。

5. 評価実験

認識実験によって提案手法の評価を行った。N-best 候補を生成後グラフ形式に圧縮する従来手法と、探索中に仮説をマージしてグラフを動的生成する手法、さらに重複仮説の探索を中断する手法のそれぞれを比較する。また、単語信頼度に基づく枝刈りを、これらの各探索手法において適用した性能を評価する。これらの手法をすべて認識エンジン Julius¹³⁾ rev. 3.4.2 に実装し、同一条件のもとで認識実験を行った。

評価タスクとして、生駒市の市民ホールに常設された音声情報案内エージェントに対する質問タスクを用いる¹⁴⁾。エージェントに対して自由に発話されたユーザ発話のうち、成人の比較的クリアな発声を 500 文抽出してテストセットとした。平均発話長は 2.1 秒、平均単語数は 6.4 単語である。言語モデルは 3-gram を使用し、語彙数は 41248 語、テストセットパープレキシティは 11.4 である。音響モデルは展示会雑音を重畳して学習した不特定話者 phonetic tied-mixture トライフォンモデル (3000 状態、129 コードブック、64 混合) を用いる。

十分量の仮説がグラフ候補として残るよう、前段パスのビーム幅を広く設定した (-b 1500)。この条件での 1 位文候補の単語誤り率は 18.26%、前段パスで生成される単語トレリスの深さ (フレームごとのトレリス単語数の平均) は 64.58 であった。認識処理時間は Linux (CPU: Pentium Xeon 2.4GHz) 上で測定した。なお前段パスのみを実行した場合の平均実行時間は約 1.85 秒である。単語信頼度の算出で用いる信頼度計算のスケーリング係数 α は 0.05 に固定した。

なお、ほぼ同じ区間に同じ単語候補が大量に出現するのを防ぐため、探索終了時に始終端フレームがそれぞれ 5 フレーム以内で重なっている同一単語をマージして評価を行った。

5.1 単語グラフ生成法の比較

まず、求める文仮説数 N を 1 から 300 まで指定した場合の提案手法の単語グラフ生成法 (動的グラフ生成 + 重複仮説中断) の結果を表 2 に示す。単語信頼度による枝刈りはここでは行っていない。表中の GER はグラフ単語誤り率であり、グラフ上で許される文候補 (単語系列) のうち誤りが最小となる単語系列の単語誤り率を表す。WGD はグラフの深さを表し、グラ

表 2 トリートレリス探索における単語グラフ生成結果
Table 2 Result of word graph generation on tree trellis search

N	GER[%]	WGD	gene	# of hypo. push	pop	time(sec)
1	17.03	1.53	6194	1047	62	2.042
2	12.82	2.64	13480	1435	162	2.204
3	11.36	3.48	18837	1669	239	2.332
5	10.06	4.78	26172	1946	345	2.480
10	8.98	7.12	36998	2359	509	2.732
50	7.49	17.10	65841	3925	1151	3.478
100	7.53	21.08	72147	4655	1499	3.832
300	7.53	21.57	72878	4771	1586	3.902

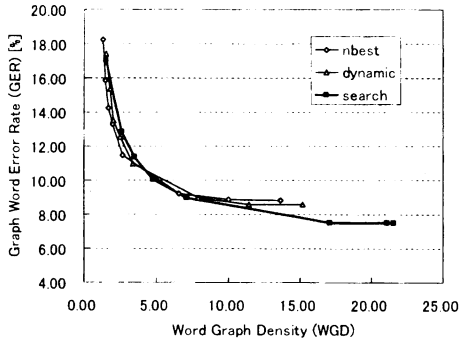


図 3 グラフの深さに対するグラフ認識率

Fig. 3 Comparison of GER for each resulting size of word graph.

フの総単語数を正解単語数で割った値の平均である。gene はスタックデコーディングにおいて仮説を生成した数、push は仮説がスタックに格納された数、pop はスタックから仮説が取り出された回数、それぞれ 1 入力あたりの平均を表す。N を大きくするに従って多くの仮説が生成され、GER が上昇している。N=100 で探索空間は飽和したが、その際の WGD は 21.08 に抑えられた。ただし、N の増大に伴い探索時間が長くなる傾向が見られた。

グラフの深さごとの誤り率を図 3 に示す。図中の“nbest”は探索終了後に N-best リストを圧縮して単語グラフを生成する従来手法、“dynamic”は後段パスで動的に単語グラフを生成する方法、“search”はさらに重複仮説の探索を中断する最終的な提案手法である。なお search は表 2 に対応する。dynamic は nbest に比べて同一の探索量でより多くの有望な単語仮説を残せており、誤り率の改善が見られる。search は類似仮説の展開を中断することで、通常は展開されない下位の文候補が展開されるようになるため、グラフの深さに対する誤り率では nbest、dynamic に比べてやや劣化する傾向が見られた。ただし、N-best や dynamic では N を大きくした場合でも平均で約 13.86 の深さのグラフしか生成できなかったのに対して、search では類似仮説の生成を抑制するため 21.57 の深さまで候

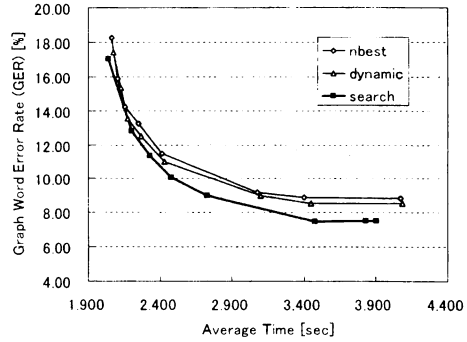


図 4 平均処理時間に対するグラフ認識率

Fig. 4 Comparison of GER for each resulting processing time.

補を生成でき、結果的に GER も 7.49 を示した。

実測計算量と誤り率の関係を図 4 に示す。図の横軸は前段パスを含む全ての認識処理時間である。dynamic の探索にかかる処理量は nbest と同一であるが、単語グラフの生成とマージの処理を行うため計算量が若干増大するが、途中で有望であった単語をも認識結果に残せるため、認識精度では勝っている。さらに search では、重複仮説の探索中断により、少ない処理量で単語グラフを出力できることが示された。

本実験では、探索の終了条件について改善の余地が見られた。本実験では得られる文仮説数 N で探索を打ち切ったが、search では仮説を途中で打ち切るため、同じ N に対して探索量が大きくなる傾向にあった。

5.2 信頼度による枝刈りの評価

次に、探索中の信頼度基準による枝刈りの評価実験を行った。N = 10 に固定して枝刈りを導入した場合の後段パスの 1 文あたりの平均仮説格納数とグラフ誤り率を図 5 に、平均処理時間とグラフ誤り率の変化を図 6 にそれぞれ示す。しきい値 thres は 0.001, 0.01, 0.03, 0.05, 0.1, 0.3, 0.5 とした。

図 5 より、信頼度の枝刈りしきい値を設定することで、仮説展開数を抑えられていることが各手法で確認できた。dynamic では 0.71% の認識精度劣化と引きかえに、平均仮説格納数を 1806 から 665 に抑えることができた。一方で、図 6 のように、実際の認識処理時間への寄与は小さく、若干の改善が見られるのみであった。これは仮説する展開数を抑えても実際に POP される仮説数に影響が小さく、探索終了までのステップ数が変化しなかったためである。また、グラフサイズあたりのグラフ誤り率もほとんど変化しなかった。またしきい値を大きくすると急激に探索が不安定になり解が得られない例が頻発した。調べたところ、探索中に得られる信頼度がヒューリスティックを含む値から計算されているため、特に短い単語において信頼度が著しく低くなる例が見られた。また、手法ごとの比

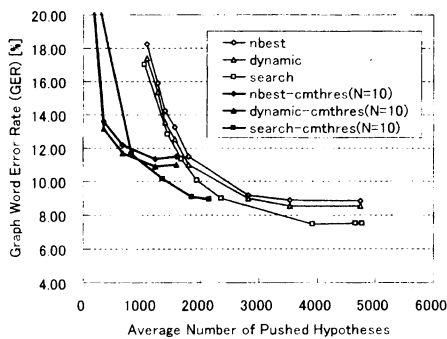


図5 信頼度基準枝刈りの結果：仮説格納数ごとの比較
Fig. 5 Result of confidence pruning by pushed hypotheses.

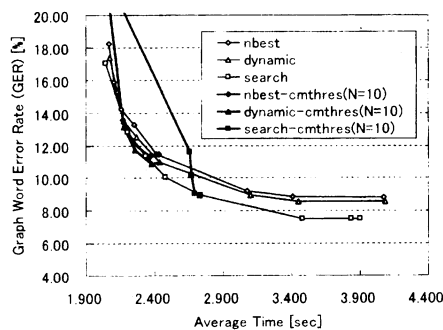


図6 信頼度基準枝刈りの結果：処理時間ごとの比較
Fig. 6 Result of confidence pruning by processing time.

較では、重複仮説の探索を中断する search で特に探索が不安定になる傾向が見られた。

6. おわりに

トリートレリス探索において単語グラフを効率よく生成する手法について提案した。後段パスであるスタックデコーディングにおいて、仮説の単語パスからの単語グラフ構造の動的生成、およびパスの重複する仮説の探索を中断することで、単語グラフ生成を主眼としたデコーディングを行う方法について述べた。さらに、後段パスにおいてオンラインで得られる単語信頼度を元に単語の展開を抑制する手法を導入した。実験的評価を行った結果、単純に N-best 候補を圧縮する従来手法に比べて、より精度の良いグラフを効率よく生成できることを示した。信頼度の枝刈りについては一定の効果が確認できたが、探索中の信頼度を用いることによる信頼度値の不安定さの影響が大きいことが分かった。

多くの音声インタフェースでは、認識結果に対する意味解釈やキーワード抽出などの処理を行うが、この

場合 1 発話ごとに何百もの大量の文候補を処理することは難しく、有望な候補のみが得られることが望ましい。この場合、本手法は N-best リストに比べてよりコンパクトで精度の高い単語グラフ形式をより効率よく求めることができ、有用である。一方、リスコアリングによる仮説の再評価のために大規模な単語グラフが要求される場合は、単語グラフ探索法が有利であると推察される。

今後の課題としては、一般的な単語グラフ探索による生成法との比較、探索中の信頼度計算法の改善、探索終了条件の設定、他コーパスでの評価が挙げられる。

参考文献

- 1) S. Ortman, H. Ney and X. Aubert, "A word graph algorithm for large vocabulary continuous speech recognition," *Computer Speech and Language*, Vol.11, No.1, pp.43-72, 1997.
- 2) H. Ney, S. Ortman and I. Lindam, "Extensions to the word graph method for large vocabulary continuous speech recognition," in *Proc. IEEE ICASSP*, pp.1787-1790, 1997.
- 3) T. Kemp and T. Schaaf, "Estimating confidence using word lattices," in *Proc. EUROSPEECH*, pp.827-830, 1997.
- 4) A. Sixtus and S. Ortman, "High quality word graphs using forward-backward pruning," in *Proc. IEEE ICASSP*, pp.593-596, 1999.
- 5) L. Mangu, E. Brill and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech and Language*, Vol.14, No.4, pp.373-400, 2000.
- 6) F. Wessel, R. Shuliter and H. Ney, "Using posterior word probabilities for improved speech recognition," in *Proc. IEEE ICASSP*, pp.563-566, 2000.
- 7) G. Evermann and P. C. Woodland, "Large vocabulary decoding and confidence estimation using word posterior probabilities," in *Proc. IEEE ICASSP*, pp.2366-2369, 2000.
- 8) F. K. Soong and E.-F. Huang, "A tree-trellis based fast search for finding the N best sentence hypotheses in continuous speech recognition," in *Proc. IEEE ICASSP*, vol.1, pp.705-708, 1991.
- 9) 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-DII, No.1, pp.1-9, 1999.
- 10) 緒方淳, 有木康雄, "音声認識精度向上のための信頼度尺度の比較", 情報処理学会研究報告, 2000-SLP-34-20, 2000.
- 11) A. Lee, K. Shikano and T. Kawahara, "Real-time word confidence scoring using local posterior probabilities on tree trellis search," in *Proc. IEEE ICASSP*, Vol.1, pp.793-796, 2004.
- 12) C. V. Neti, S. Roukos and E. Eide, "Word-based confidence measures as a guide for stack search in speech recognition," in *Proc. IEEE ICASSP*, pp.883-886, 1997.
- 13) 大語彙連続音声認識エンジン Julius:
<http://julius.sourceforge.jp/>
- 14) 西村竜一, 西原洋平, 鶴身玲典, 李晃伸, 猿渡洋, 鹿野清宏, "実環境研究プラットフォームとしての音声情報案内システムの運用", 電子情報通信学会論文誌, Vol.J87-D-II, No.3, pp.789-798, 2004.