音声言語情報処理 29－33
(1999. 12. 21)

パースフィルタリングを用いた連続音声の認識

花沢 健, 坂井 信輔

NEC C&C メディア研究所

〒216-8555 川崎市宮前区宮崎 4-1-1

E-mail: {hanazawa,sakai}@ccm.cl.nec.co.jp

あらまし

N-gram 言語モデルを利用して生成した単語列仮説を、文法に基づくパーシングにより検証、棄却する、パースフィルタリスコアリング法を提案する。本方式は、N-gram 言語モデルの与える単語連鎖の局所的な確からしさを利用しながら、かつ従来の N-gram 言語モデルではカバーしきれなかった文全体にわたる言語制約を適用できるという特長を有する。評価実験により、パースフィルタリスコアリング法を用いると trigram のみを用いる場合と比較してより高い文正解率が得られることを確認した。

キーワード　統計的言語モデル、文法知識、パーシング、リスコアリング

Continuous Speech Recognition with Parse Filtering

Ken Hanazawa, Shinsuke Sakai

C&C Media Research Laboratories, NEC Corporation

4-1-1 Miyazaki, Miyamae-ku, Kawasaki 216-8555, Japan

E-mail: {hanazawa,sakai}@ccm.cl.nec.co.jp

Abstract

We propose a new rescoring method called "parse-filtered rescoring". In this method, word sequence hypotheses generated using N-gram language models are verified by parsing based on a grammar, and rejected if they are unparsable. This approach has an advantage that sentence-level linguistic constraints can be applied for evaluating hypotheses, as well as utilizing the local measure of reliability of hypotheses provided by conventional N-gram language models. Experimental results show that this method gives a higher sentence accuracy than just using a trigram language model.

key words　statistical language models, grammatical knowledge, parsing, rescoring

1 はじめに

高精度な連続音声認識システムを構築するためには、正解候補の音響的な確からしさを求めるために用いられる音響モデルに加えて、言語的な確からしさを求めるために用いられる言語モデルが必要である。現在、言語モデルとしては、N-gram 言語モデルが主流になっている [1]。N-gram は、ある単語並びの次に出現する単語の確からしさを与える統計的なモデルである。N の値が大きいほどより詳細なモデルが構築できるが、大量の学習データを必要とするため、多くの場合 N=2 である bigram あるいは N=3 である trigram が利用されている。しかしながら、trigram でも高々単語の 3 連鎖しか見ていないため、文全体として妥当な解となるような制約を与えることは難しく、また学習データと異なるドメインに対しては、必ずしも効果を発揮しない。対象となるドメインの学習データを十分に集めることが理想ではあるが、膨大な量のデータをドメイン毎に収集するのは非常に困難である。

統計的言語モデル以外に、言語知識をトップダウンに用いた言語モデルとして文脈自由文法 (CFG) も従来用いられている [2]。小語彙のシステムであれば文脈自由文法を音声認識の探索時に用いても問題が少ない。しかし、大語彙連続音声認識の場合にはサーチ時の探索空間が増大する。探索空間の爆発を抑えるために探索時にプルーニング (絞り込み) を行うことが考えられるが、文脈自由文法を用いる場合にはプルーニングのための判定尺度として音響的なスコアしか用いることができないため、精度の点で問題がある。

そこで今回、N-gram 言語モデルを利用して生成した正解単語候補列を文法を利用して検証する、「パースフィルタを用いたリスコアリング法」を提案する。パースフィルタリスコアリングでは N-gram 言語モデルの与える単語連鎖の局所的な確からしさを利用しながら、かつ従来の N-gram 言語モデルではカバーしきれなかった文全体の言語制約や文法知識を適用できることが特長である。

2 パースフィルタを用いたリスコアリング

2.1 音声認識システムの構成

音響モデルと N-gram 言語モデルを用いて生成される単語列の仮説を、文法を用いて検証するパースフィルタリングの実現形態としては、マルチパス探索で一旦ワードグラフのような複数仮説のコンパクトな表現を生成してその上で行う方法と、スタックデコーダのような最良優先探索の仮定で逐次検証して行く方法が考えられるが、今回は前者の方法をとっている。音声認識システムとしては、図 1 に示すようなマルチパス探索方式の構成になっている。

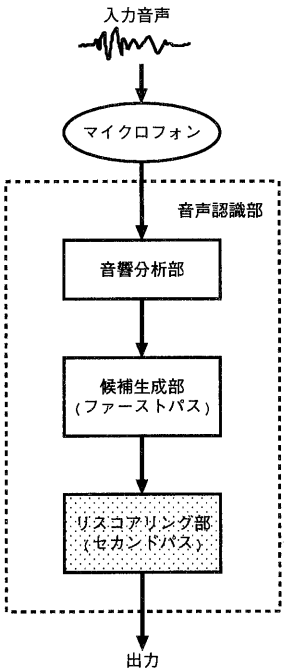


図 1: 音声認識システムの構成

マイクロフォンから入力された音声を分析し、特徴量の時系列に変換する。この特徴量時系列を入力として、音響モデルと粗い言語モデルである bigram を用いて最尤の単語列候補群を求めるための探索を行い、正解候補群であるワードグラフを生成する (ファーストパス)。ワードグラフを対象に、より詳細な言語モデルを用いてリスコアリングを行い、認識結果の一文を出力する (セカンドパス)。パースフィルタリングは、このセカンドパスで行われる。

2.2 パースフィルタリングの概念

従来の N-gram のみを用いた認識方式においては、第一位の認識結果が誤りである場合に

- 認識結果は局所的にみると隣接して出現しやすい単語の連鎖とはなっているが、文としての形

を成していない

- ワードグラフ中には正解となるようなパスが存在している

という事例が見受けられる。このような場合は、CFG等のより高度な言語制約による文法適合性判定により第一位候補を棄却し、次に確からしい候補を調べていけば、下位に埋もれている正解候補を見つけられる可能性がある。以上がパースフィルタリングの原理である。

例えば、図2に示すようなケースでは、'Breakfast WAS included.' という発声に対して、'Breakfast WITH included.' という、一部を音響的に近い単語に誤った候補が第一位候補となっているが、これを文法知識を用いて文として不適切であると判定し棄却すると、文として適切な第二位候補が採用されることになる。同様に、第一位候補に呼吸音などによる湧き出し誤りが存在する場合も、その湧き出した単語を含まない候補が二位以下に存在すれば、それを採用して誤りを避けられる可能性がある。

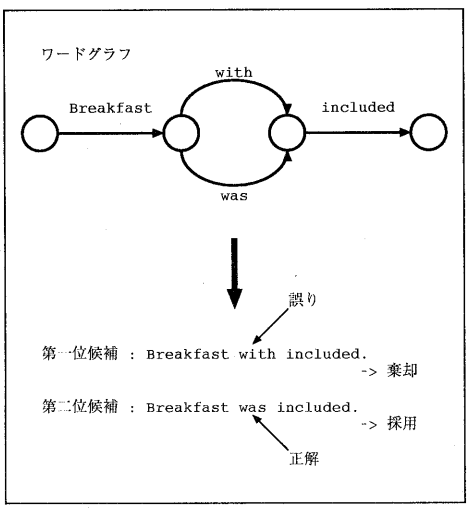


図 2: パースフィルタリングの例

パースフィルタリングによる文法適合性判定を実現するために、提案手法では文法記述の枠組としてCFGを用いている。

2.3 パースフィルタリスコアリング

パースフィルタリスコアリング部では、図3のようにファーストパスの結果であるワードグラフを入力とし、認識結果の一文を出力する。

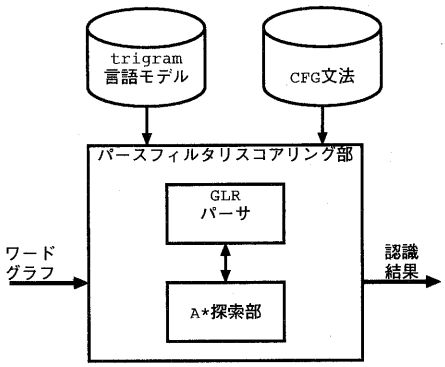


図 3: パースフィルタリスコアリング部の構成

2.3.1 最良スコア仮説の探索方式

ワードグラフ上でCFGの制約を満たしながら音響モデルスコアと言語モデルスコアの和が最良である仮説を高速に求めるために、探索方式としてA* 探索を採用している [3]。A* 探索は、ある評価尺度において最良のものから順番に一位、二位、三位と候補を選んでいくのに適している。よって今回のパースフィルタリスコアリングにおいて、文法適合性判定によって棄却された候補の次に確からしい候補を順に判定していくためにA* 探索を用いることにした。

探索しているワードグラフ中のあるノードにおいて、その探索の基準に用いるスコアを「そこまでの累積スコアとそこから終端までに得られると予測されるスコアとの和」として表現する。この予測スコアはヒューリスティクスコアと呼ばれる。探索中の各探索パスのうち、スコアが高いものから順にパスを伸ばして行き、最初に終端に到達したものを採用することで最良優先探索を実現している。

提案手法では、探索時の仮説のスコアとしてファーストパスで得られた音響スコアとtrigramによる言語スコアの和を用いる。A* 探索のヒューリスティクスコアとしては、ワードグラフ中の現在のノード以降の音響スコアとtrigram言語スコアの和の最良のものを用いる。

2.3.2 パースフィルタリング

パースフィルタリスコアリングでは、上記で述べたワードグラフを対象とした探索において単語を一語展開するたびに CFG による部分パースを行い、部分パースで棄却された場合にはそのパスは枝刈りされる。ワードグラフの終端に達したパスが、パースの結果受理された場合にはその結果を出力し、棄却された場合にはそのパスは枝刈りされる。なおパーシング方式としては、部分パースが可能であるという性質から一般化 LR 法を採用している。

ワードグラフから生成し得るすべての仮説がパースフィルタリングで棄却された場合は、音響モデルと trigram 言語モデルによるスコアの最も高い結果を出力することにした。

2.3.3 フィルタリングの効率化

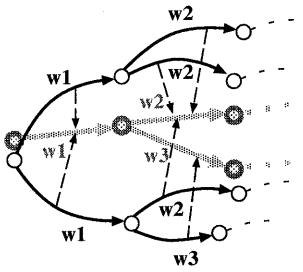
音響モデルと bigram を用いた探索により生成されたワードグラフには、発話始端からの部分単語列としては全く同一でありながら、単語アークの始終端時刻のみが少しずつ異なっているような部分経路が多く存在することがある。これらのそれぞれを独立にパースすると、同一の処理を何度も行うという処理時間の浪費と、同じ情報をもつパース状態 (スタック) を多数保持することによるメモリ使用量増大の問題が生じる。そこで、今回は、ワードグラフと別に、発話始端からの単語列 prefix 木をワードグラフの探索の途中で動的に生成するようにし、パース状態は、この単語列 prefix 木のアークに付随するようにしている (図 4)。

探索の過程でワードグラフのアークをひとつ進める際には、そのアークまでの部分経路と同じ単語系列がすでに単語列 prefix 木に存在しない場合のみ、単語列 prefix 木でアークを伸ばし、パースを行う。すでに対応する部分単語列が存在する場合は、単語列 prefix 木における対応するアークへの対応付けをおこない、すでに計算済みの部分パース結果を利用することにより重複した計算を避けて効率化をはかっている。

2.4 文法

2.4.1 文法の枠組

文法記述の枠組として、素性つきの CFG を使用している。非終端記号は、任意個の多値の素性をとることが可能である。素性つき CFG における書き換え規則の例を図 5に示す。



黒線はワードグラフ、灰色線は単語 prefix 木をあらわす。

図 4: 単語列 prefix 木

```
(a)  ST(X,T,N,P) -> NP(N,P,C=SBJ,G=NO)
                    VP(F=Y,T,N,P,X)
(b)  NP(N,P,C,W,G=NO) -> NP(N,P,C,W,G=NO)
                         PP(X=NO)
(c)  NC(N=SG,G=NO) -> accessory
(d)  V(T=PRES,N=SG,P=3RD,V=I) -> arrives
```

((a) 文の基本構造, (b) 前置詞句による修飾, (c) 名詞の語彙規則, (d) 動詞の語彙規則)

図 5: 素性付き書き換え規則の例

この枠組においては、以下のような規則に従いパーシングが行われる。

- 素性名が書き換え規則の複数の位置に出現するならば、パース途中の部分構造のこれらに対応する部分は値が同一である必要がある。例えば、図 5(a) の右辺において、素性 N が、右辺第 1 要素の NP に対応するパース途中の部分構造においてある特定の値をもつならば、第 2 要素の VP においても同じ値をもつ必要がある。また、左辺の N は、reduce 後の部分構造 (非終端記号 ST) に素性が伝播して、右辺と同じ値となることを示す。 この枠組を用いて、英語における数の一致等の制約を記述することが可能である。
- また、書き換え規則中である素性が値まで指定されている場合は、パース途中の部分構造の対応する部分は、これと素性値が矛盾しない必要がある。たとえば、図 5(a) の右辺第一要素に対

応づけられる部分構造は、属性Cが、SBJ以外の値をもっていてはならない。

- パース途中の部分構造が素性値をもっていても、書き換え規則の右辺にその素性が記述されていなければ、その素性値は無視される。

- reduce の結果の部分構造がもつ素性は、書き換え規則の左辺に記述されている素性のみである。例えば、図 5 (a) において、素性Cは、reduce 結果には付与されない。

2.4.2 パーサの枠組

N-gram を言語モデルとして用いる連続音声認識方式において、文の始端から途中までの部分仮説において、文法の制約を満たさないことが分かればただちにその仮説を棄却できることが望ましい。そこでそのような性質をもつパーシング方式の一つである LR パーシング [4] を採用している。また、自然言語を記述する文法には曖昧性があるので、グラフ構造スタックを用いた一般化 LR パーシング [5] を行っている。

素性のチェック・伝播に関しては、LR パーシングテーブルを作成する際には考慮せず、パーシング実行時の reduce 操作を行う際に素性のチェックを行い、矛盾がある場合には不受理としている。素性に矛盾がなければ、書き換え規則にもとづいた素性の伝播が行われ、スタック中の非終端記号に、伝播した素性が付与、格納される。

2.4.3 文法の開発

英語テキストデータベースをもとに、人手で素性付きの文法を開発した。3000 文程度のテキストをもとに、文法記述に用いる素性、基本的な文法構造を記述するための書き換え規則および、この部分集合にあらわれる語彙をカバーする語彙的な規則を作成した。10 種類の素性を用いて、時制、数、人称、ギャップの有無、文法格、形容詞の級などの制約を記述している。

この段階の文法に、評価データの語彙をカバーする語彙的規則を追加し、次節で述べる予備的な評価実験を行った。

表 1: 実験条件

言語モデル	ファーストパス : bigram 言語モデル
	セカンドパス : trigram 言語モデル
辞書	語彙数 9k
文法	書き換え規則数 3890
	終端記号の種類 2325

表 2: 評価データ

話者	男性話者 20 名
発話数	3600 発話

3 評価実験

3.1 実験条件

前節で述べた文法を用いて、英語連続音声認識の予備的な評価実験を行った。実験条件、評価データを表 1、表 2にそれぞれ示す。表 2 に示した評価データの全文に対する今回作成した文法のパースカバー率 (パースが成功する文の割合) は 99.0% であった。

実験に用いたシステムでは、図 1 のファーストパスでは音響モデルと bigram 言語モデルを用い、セカンドパスでは音響モデル、trigram 言語モデルと素性つき CFG を用いてパースフィルタリスコアリングを行っている。

bigram, trigram 言語モデルは文法作成用に使用した 3000 文を含む 9 万文により学習した。

3.2 実験結果

結果を表 3 に示す。ファーストパスの性能は、単語正解率 87.3%、ワードグラフ中にどの程度正解が含まれているかを示すグラフ正解精度は 97.1% であった。また、文正解率は 64.3% であった。

表 3: 実験結果

	単語正解率 [%]	文正解率 [%]
セカンドパスなし	87.3	64.3
trigram のみ	90.4	72.0
パースフィルタ	91.7	77.6

まず、比較対象として行った trigram 言語モデルのみを用いたリスコアリングの場合は、単語正解率は 90.4%、文正解率は 72.0% であった。この trigram 言語モデルは、提案手法で用いているものと同じである。

一方提案手法によるリスコアリングでは、単語正解率は 91.7%、文正解率は 77.6% と高い文正解率を示している。誤りが、単語レベルで 14%、文レベルでは 20% それぞれ削減している。

以上より、語彙規則を追加しているため言語的にクローズな条件での評価実験ではあるが、本方式の基本的な効果を確認した。

4 まとめ

今回、高精度な音声認識結果を得るための手法として、パースフィルタリスコアリングを提案した。

評価実験の結果から、提案手法を用いると、従来の言語モデルとして trigram のみを用いるリスコアリングと比較して、より高精度な文正解率が得られることが期待できる。また、文法を対象ドメインに特化して作成することにより、より強い制約が得られる。

今後は言語的にオープンな評価実験を行う予定である。

謝辞

日頃御指導、御助言を頂く NEC C&C メディア研究所音声言語 TG、音声 HI 開発センターの関係者の皆様に感謝します。

参考文献

[1] F. Jelinek, "Self-Organized Language Modeling for Speech Recognition", Readings in Speech Recognition, Morgan Kaufmann, pp.450-506, 1990.

[2] 中川聖一, 甲斐充彦, "文脈自由文法制御による One Pass 型 HMM 連続音声認識法", 信学論 D-II, vol.7, pp.1337-1345, 1993.

[3] S. Russel, P. Norvig 著, 古川康一監訳, "エージェントアプローチ人工知能", 共立出版, 1997.

[4] A. Aho, R. Sethi, and J. Ullman, "Compilers: Principles, Techniques, and Tools", Addison-Wesley, 1986.

[5] M. Tomita, "An Efficient Augmented-Context-Free Parsing Algorithm", Computational Linguistics, Vol.13, No.1-2, 1987.