2008－SLP－73 (3)
2008／10／24

ＦＳＡ言語モデルの自動構築と動的代替パスサーチによる音声認識

森元 逞　　高橋 伸弥
福岡大学工学部電子情報工学科　〒814-0180 福岡市城南区七隈 8-19-1
Email: {morimoto, takahasi}@tl.fukuoka-u.ac.jp

あらまし　語彙規模が小または中程度（~1,000 語）の音声認識では，ＦＳＡ言語モデルが広く用いられている．しかしそのモデルを人手で定義するのはかなり大変である．我々はすでに学習コーパスから FSA 言語モデルを自動的に構築する方法を提案した．生成された言語モデルは，クローズド・データに対しては極めて高い音声認識性能を達成できる．しかしオープン・データに対しては満足できるような性能は達成できなかった．その主な理由は，生成されたＦＳＡモデルにおいて，必要なパスが途中で切れてしまうためである．そこでこの問題に対処するため，認識途中において，ある単語の認識に失敗したと判断される場合は代替パスを動的にサーチし，代替パスがみつかればそのパスで認識を継続するようなサーチ方法を提案する．また実験結果によりその有効性をしめす．

キーワード　FSA 言語モデル，言語モデル自動構築，動的代替パスサーチ

Automatic Construction of a FSA Language Model and Speech Recognition on it with Dynamic Alternative Path Search

Tsuyoshi MORIMOTO, Shin-ya TAKAHASHI

Department of Electronics Engineering and Computer Science
Fukuoka University, Fukuoka, Japan
Email: {morimoto, takahasi}@tl.fukuoka-u.ac.jp

Abstract　For a small- or middle-size (around 1,000 words) vocabulary speech recognition, a Finite State Automaton (FSA) language model is widely used. However, defining a FSA model with sufficient coverage and consistency requires much human effort. We already proposed a method to automatically construct a FSA language model from learning corpus by use of FSA DP matching algorithm. Experiment results show that this model attains quite high recognition correct rate for closed data, but only low rate for open data. This is mainly because an necessary path does not appear in a generated FSA. To cope with this problem, we propose a new search algorithm that allows to jump dynamically to an alternative path when speech recognition of some words seems to fail. Experiment results shows the effectiveness of the algorithm.

Keywords　FSA language model, automatic construction of a language model, dynamic path search

1. はじめに

語彙規模が小または中程度（~1,000 語）の音声認識では，ＦＳＡ言語モデルが広く用いられている．しかしそのモデルを人手で定義するのはかなり大変である．認識システムによっては，正規文法から自動的に変換するツールが用意されているものもある（例えば，HParse[7]）が，充分なカバレージを持ち，矛盾の無い文法を定義するのは容易ではない．一方，多量の学習テキストが入手できる場合は，バイグラムやトライグラムなどの統計言語モデルが利用できる．しかし学習データ量が少ない場合は，コーパスより得られた統計量自身の信頼性が低くなり，結果的に性能の良くな

い言語モデルとなってしまう，すなわち“スパースネスの問題”が起きる恐れがある．

学習データからＦＳＡモデルを自動的に構築する方法について，すでにいくつかの提案がなされている．ここで注意して欲しいのは，与えられたデータから非循環型(acyclic)のＦＳＡを作成するのは，比較的単純な問題であることである．すなわち，共通的な部分をプレフィックスとするようなトライ構造を構築し，次に等価な状態をマージすることによってＦＳＡを最小化すればよい．しかし，単純にこの方法を実現しようとすると，計算量が膨大となる．そこでＡＩ的なアプローチにより計算効率を改善する方法が提案されている[2][3]．ただし，これらの方法はまだ基礎研究の段階であり，言語モデルなどのような実際的な問題には適用されていない．一方，言語モデルを対象とし，別のアプローチにより効率の改善を行なおうとする研究もいくつか報告されている[4]-[6]．しかしこれらの方法はいずれも統計量を利用するものであるため，学習データ量が充分でないと，統計言語モデルと同様な“スパースネスの問題”が起きてしまう．

我々はすでに学習コーパスからFSAのDPマッチングによりFSA言語モデルを自動的に構築する方法を提案した[1]．生成された言語モデルは，クローズド・データに対しては極めて高い音声認識性能を達成できる．しかしオープン・データに対しては満足できるような性能は達成できなかった．その主な理由は，生成されたＦＳＡモデルにおいて，必要なパスが途中で切れてしまうためである．そこでこの問題に対処するため，認識途中において，ある単語の認識に失敗したと判断される場合は，代替パスを動的にサーチし，代替パスがみつかれば，そのパスで認識を継続するようなサーチ方法を導入した．本論文では，まずこれまでに報告したＦＳＡの自動構築方法を紹介し，次に今回新たに導入した動的パス・サーチの機能と，それを用いた音声認識の実験結果を報告する．

２．ＦＳＡ言語モデルの自動構築

2.1 FSA DPマッチングによるFSA言語モデルの構築

以下のような手順により構築する．

(1) 学習テキストを，文間の距離により，いくつかのグループ（クラスタ）に分ける．次にクラスタ内の文間でDPマッチングを行なう．クラスタに分けるのは，あまり似ていない文同士のDPマッチングを避けるためである．文間の距離尺度としては種々なものが考えられるが，簡単化のため以下のような方法を採用している．

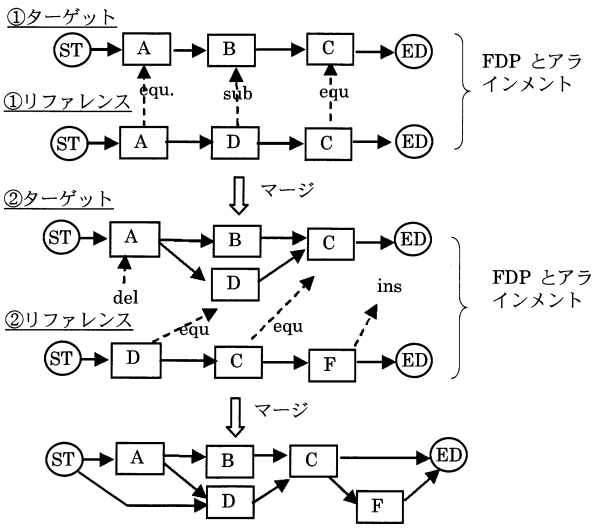
$$d(S_x, S_y) = \frac{Num(w \mid w \in S_x \cap S_y)}{Num(w \mid w \in S_x \cup S_y)}$$

ただし，

$d(S_x, S_y)$: 文 S_x と S_y の距離

$Num(w)$: 単語 w の数

(2) 各クラスタ内の DP マッチングは以下のように行なう．まずランダムに１文を取り出し，これを１パスのFSAに変換し，DPマッチングにおけるy軸に配置する（これをターゲットFSAと呼ぶ）．また別の１文を取り出し，同じように１パスのFSAに変換して，x軸に配置する（これをリファレンスFSAと呼ぶ）．次にこれら２つのFSA間のDPマッチングを行なって（以下ではこれをFDPと呼ぶ），両者のアラインメントをとる．この結果，ターゲットFSAのノードに対するレファレンスFSAのノードの関係，すなわち，「一致(equal)」，「置換(substitution)」，「削除(deletion)」，「挿入(insertion)」などの関係が得られるので，この関係に基づいてリファレンスFSAのノードをターゲットノードのFSAにマージする（図１参照）．



ST: start node　ED: end node

図１ FDP とマージ

以上の処理で得られたターゲットFSAが次のFDPのターゲットFSAとなるが，このFSAは1パスではなくなる．そこで，以降のFDPでは，ターゲットFSAの各ノードを，開始ノードからの距離によってトポロジカルソートを行い，これを次のFDPのy軸に配置する．またDPマッチングは，ターゲットのFSAに定義されたパスに基づいて実行する．例えば，実行パスは図2のようになる．

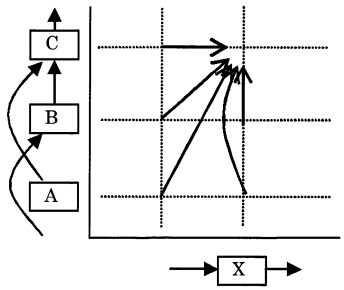


図2　FDPの実行パス

またFDPにおける距離計算は以下のように行なう．

$$gd(x,y)=\min\begin{pmatrix}gd(x-1,y)\\gd(x-1,\Delta y)\\gd(x,\Delta y)\end{pmatrix}+ld(x,y) \qquad (2)$$

ただし

$gd(x,y)$：座標(x,y)におけるグローバル距離

$ld(x,y)$：単語w_xと単語w_yのローカル距離であり，以下のように計算する

$$ld(x,y)=\begin{cases}0.0 & (w_x=w_y)\\1.0 & (w_x\neq w_y)\end{cases}$$

Δy：1つ前のy軸の座標

以上の処理をクラスタ内の全文に対して行い，各クラスタごとに1つのFSAを生成する．

(3) 以上の処理により得られたクラスタごとのFSAに，全体で1個の共通の開始ノード（ST）と終了ノード（ED）を付加し，1つのFSAとする．

2.2 生成されたFSAの性能

以上のように生成されたFSAの性能を評価するため，音声認識実験を行なった．FSAの作成に用いたコーパスは，市販されている旅行会話日英例文集4冊から，その日本語部分を集めたものである．ただしあまりに口語的な表現，断片的な表現は排除した．またこのようにして集めた例文の数がやや少なかった（約950文）ため，集めた例文の表現を多少変えて約50文ほどを学習データに追加し，計1,000文からなるコーパスを用意した．表1にコーパスの諸元をしめす．

表1　コーパスの諸元

語彙数	1,254 語
文数	1,000 文
1文あたりの平均単語数	8.87 語

また，表2にコーパスに現われた文の例をしめす．

表2　コーパス内の例文

1	美術館めぐりのツアーはありませんか
2	贈物用に包装してもらえますか
3	ここでタバコを吸ってもいいですか

音声認識実験では，デコーダとしてHVite[7]を用い，またHMMとしてJuliusに付属した不特定話者のトライフォンモデル[8]を使用した．音声認識のテストデータは，男性話者3名が各々異なる20文ずつ発話した計60発話を用いた．

(1) クローズド・データに対する認識実験

コーパス内の1,000文全てを学習データとしてFSAを作成し，テストデータの音声認識実験を行なった．結果を表3にしめす．クラスタ数としては，30，50，70の3つの条件で実験した．また表3には，参考までにバイグラムによる認識結果をあわせてしめしている．

表から分かるように，クラスタ数が異なっても，ほとんど性能はかわらない．一方，バイグラムに比べると，認識率が大幅に向上している．特に文認識率は20ポイント以上も良い．

表3 音声認識実験結果（クローズド・データ）

スラスタ数	平均分岐数	単語 %correct	文 %correct
30	1.45	98.7	90.0
50	1.42	98.7	90.0
70	1.40	98.9	91.7
bi-gram	5.88 (perplexity)	93.0	70.0

(2) オープン・データに対する認識実験

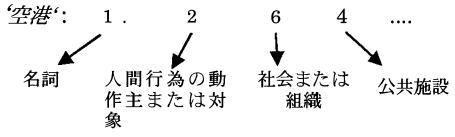
学習データから，音声認識実験に用いる 60 文を取り除き，残りの 940 文で FSA を作成する．ただし元の学習データの文数が少ないので，単純に取り除くと必要な単語も学習データから削除される恐れがある．そこで，940 文に出てきた名詞を一旦意味素性コードに変換し，意味素性コードが混在した文を作成する．これらの文について，上と同様な手続きで，意味素性コードをノード名として含むような FSA を生成する．ただし，FDP におけるノードのローカル距離の計算を以下のように変更する．

$$ld(x,y)=\begin{cases}0.0\ (w_x = w_y)\\ 0.5\ (w_x \neq w_y, sem_x = sem_y)\\ 1.0\ ((w_x \neq w_y, (sem_x \neq sem_y\ or\ w_x \neq noun))\end{cases} \quad (3)$$

sem_x：単語 x の意味素性

FDP の結果として FSA を得た後，その中の意味素性コードのノードを，その意味素性に含まれる全名詞のノードに展開し，これを最終的な FSA とする．

意味素性コードは国立国語研究所の「分類語彙票[9]」における意味コードのうち，小数点以下2桁までを使用した．



このように作成した FSA を用いた音声認識実験の結果を表4にしめす．

表4 音声認識実験結果（オープン・データ）

クラスタ数	平均分岐数	単語 %correct	文 %correct
30	1.70	78.8	26.7
50	1.67	79.9	21.7
70	1.68	79.4	28.3
bi-gram[2)]	6.36 (perplexity)	58.4	3.3

表3の結果と比較すると，認識率がかなり低下していること，特に文認識率が大幅に低下していることが分かる．ただし，それでもバイグラムに比べると，かなり良い値となっている．この文認識率が大幅に低下した原因を分析するため，音声認識が誤った発話を取り出し，元の文の単語列を FSA で受理可能かどうか，もし受理できなかった場合は，単語が FSA 内に全く存在しなくなったのか（テストデータを学習データから除いたため），それとも単語は存在するがパスが存在しなくなったのか（パスが途中で切れている）を調べた．クラスタ 70 において誤認識となった 17 文におけるこの内訳を表5にしめす．この表から分かるように，認識誤りの原因の 70%以上は，パスが存在しなくなったためである．

表5 文認識誤りとなった原因の内訳

原因	文数
単語が存在しない	4 (23.5%)
パスが存在しない	13 (76.5%)
計	17 (100%)

3．動的代替パスサーチ

3．1 動的代替パスサーチ機能

前章で述べたように，オープン・データに対し

2) バイグラムでは，テスト文の 60 文を削除せず，Witten-Bell によるバックオフを用いた

て文認識率が大幅に低下したのは，必要なパスが存在しなかったためでる．そこで以下のような機能を新たに実現し，デコーダに組み込んだ．

(1) デコーダにおいて各単語の認識が終った時点で，その単語の認識スコアを調べ，単語の認識が成功したか失敗したかどうかを判断する．

(2) もし失敗したようであれば，全く違う単語の認識を行なった恐れが大きい．そこで，1つ前の単語（以下，前単語）と同じ単語がFSA内の他のパスに存在しないかどうかを調べる．

(3)そのような単語がみつかれば（以下，同一単語と呼ぶ），前単語までの認識情報（認識スコアやパス情報）を，同一単語にコピーし，同一単語の次から認識を始める．

(4) 単語認識の成功／失敗の判断が誤っている恐れもあるため，現在のパスでの認識もそのまま継続する．

ここで問題となるのは，上記(1)において，単語認識の成功/失敗をどのような方法で判断するかであるが，今回の実験では，以下のような比較的簡単な方法を用いている．

・各パスの認識において，1フレーム当たりのスコア(score per frame: spf)を計算しておく．なお，スコアとしてはHViteで用いられているlog-likelihoodをそのまま用いているので，値(マイナス値）が小さいほど信頼性が低いことになる．
・ある単語の認識が終わったら，その単語のspfを計算する．
・単語のspfがパスのspfのα倍以下であれば，その単語の認識は誤ったと判断する．ただしαは実験的に決める（今回は1.44とした）．

3．2　実験結果

動的代替パスサーチ機能の有効性を確認するため音声認識実験を行なった．実験の条件は表3のオープン・データに対する実験条件と同じである．ただし，クラスタ数は70のみとした．結果を表6にしめす．表3の結果と比較すると，以下のことが言える．

・　文%correctが，大幅に向上している（11.7ポイントの向上）
・　単語%correctも，多少向上している（3.2ポイントの向上）

表6．　音声認識実験結果
（動的代替パスサーチ）

クラスタ数	単語 %correct	文 %correct
70	82.6	40.0

動的代替パスサーチ機能を組み込んだことにより文%correctが大幅に向上した理由として，日本語会話文においては，共通的な文末表現が多く使用されるためであろうと思われる．実際に，成功した発話例を表7にしめす．

・発話1では，「ませ ん か」が「ます か」と正しく認識できるようになった．
・発話2では，「が 上がっ て いただけ ます か」の部分が「を 預っ て いただけ ます か」と正しく認識できるようになった．
・発話3で「です」が「と いい ます」と正しく認識できるようになった．しかし，この文における元々の誤りの原因は，DPマッチングによるアラインメント方法にあると思われる．すなわち，コ

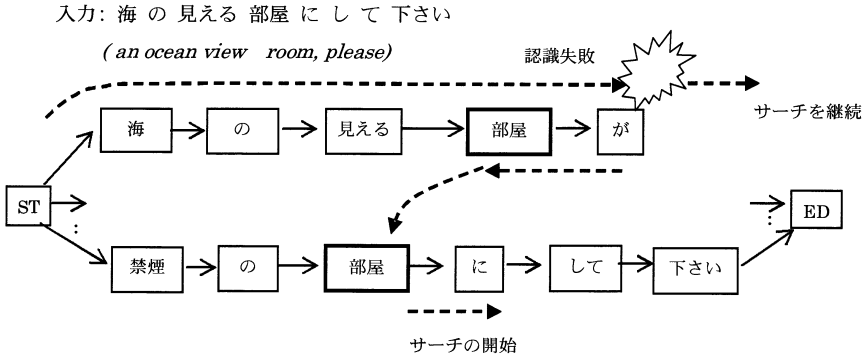


図3 動的代替パスサーチ

ーパスには，「名前は 小川 春子 です」という文と「名札 が つい て い て 名前 は 小川 春子 と いい ます」という文が存在した．しかし後者では，「名札 が つい て い て」のようにかなり長めの句が挿入されているため，両者での「名前 は 小川 春子」という部分が一致しているとは判断されなかった．そのため，両者は全く２つのパスとして定義されてしまい，「名前 は 小川 春子 と いい ます」というパスが作成されなかった．今回は動的代替パスサーチ機能により正しく認識されるようにはなったが，このように長い挿入句があってもアラインメントが取れるよう，アラインメント方法を改善するのが良いと思われる．

表７　動的代替パスサーチにより，新たに認識に成功した発話例

1	誤	明日 の 朝 まで に これ を クリーニング し て もらえ ませ ん か
	正	明日 の 朝 まで に これ を クリーニング し て もらえ ます か
2	誤	荷物 が 上がっ て いただけ ます か
	正	荷物 を 預っ て いただけ ます か
3	誤	名前 は 小川 春子 です
	正	名前 は 小川 春子 と いい ます

４．まとめ

コーパス内の例文間の DP マッチングを行なってアラインメントをとり，その結果を用いて FSA 言語モデルを自動的に構築する方法を示した．ただしそのままでは，必要なパスが途中で切れ，正しく認識できない場合が多い．この問題を解決するため，認識処理において各単語の認識が終った時点でその単語の認識スコアを調べ，もし単語認識に失敗した恐れがある場合は，前単語と同じ単語が FSA 内の他のパスに存在すれば，そこから認識を継続する，という動的代替パスサーチ機能を提案した．また実験によりその有効性をしめした．

FSA 言語モデルは，中規模程度の語彙サイズの音声認識には極めて有用なモデルである．しかし，定義されたモデルから少しでも外れると，認識できないという大きな問題がある．本手法は，このような問題を解決するための，FSA 言語モデルにおける一種のバックオフ機能と考えることができる．このようなバックオフという観点から考えれば，他にも以下のような方法が考えられる．

・前向きの認識に失敗したら，後ろ向きに認識を行う．前向き，後ろ向きの２つの部分認識結果を得て，両者が接続できそうであれば，接続したものを認識結果とする．

・発話途中のポース以降で認識誤りが発生したら，ポースに接続可能な全てのパスに対する認識を試みる．

今後はこのような手法の可能性についても検討していく予定である．

参考文献

[1] T. Morimoto and S. Takahashi, "Automatic Construction of FSA Language Model for Speech Recognition by FSA DP-Matching", Trends in Intelligent Systems and Computer Engineering (Edtd. by O. Castillo et al.), Springer, 515-524, 2008

[2] K. J. Lang, B. A. Pearlmutter, and R. Price, "Results of the Abbadingo One DFA Learning Competition and a New Evidence Driven State Merging Algorithm", Proc. of Int. Colloquium Grammatical Inference, 1-12, 1998

[3] S. M. Lucas and T. J. Reynolds, "Learning Deterministic Finite Automata with a Smart State Labeling Evolutionary Algorithm", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.27, No.7, July 2006

[4] C. Kermorvant, C. de la Hinguera, and P. Dupont, "Learning Typed Automata from Automatically Labeled Data", Journal Artificielle, Vol. 6, No.45, 2004

[5] J. Hu, W. Turin, and M. K. Brown, "Language Modeling with Stochastic Automata", Proc. of ICSLP-1996, 1996

[6] G. Riccardi, R. Pieraccini, and E. Boccieri, "Stochastic Automata for Language Modeling", Computer Speech and Language, Vol.10, No. 4, 265-293, 1996

[7] S. Young et al., "The HTK Book (for Ver. 3.0)",1999 (http://htk.eng.cam.ac.uk/)

[8] T. Kawahara, A. Lee, K. Takeda, K. Itou, and K. Shikano, "Recent Progress of Open-Source LVCSR Engine Julius and Japanese Model Repository -- Software of Continuous Speech Recognition Consortium -- ", Proc. of ICSLP-2004, 2004 (http://julius.sourceforge.jp/en/julius.html)

[9] National Language Research Institute, "Bunrui-Goi-Hyo (Word List by Semantic Principles)", Syuei-Shuppan, 1994 (in Japanese)