論文 / 著書情報
Article / Book Information

論題(和文)	WFST音声認識デコーダの省メモリ化に関する検討
Title(English)	
著者(和文)	大西 翼, ディクソン ポール, 岩野 公司, 古井 貞煕
Authors(English)	Oonishi Tasuku, Paul Dixon, Koji Iwano, SADAOKI FURUI
出典(和文)	日本音響学会2008年春季講演論文集, Vol. , No. 1-10-3, pp. 7-10
Citation(English)	, Vol. , No. 1-10-3, pp. 7-10
発行日 / Pub. date	2008, 3

WFST音声認識デコーダの省メモリ化に関する検討*

◎大西 翼, ディクソン ポール, 岩野 公司, 古井 貞熙 (東工大)

1 はじめに

東京工業大学では，経済産業省「情報家電センサー・ヒューマンインターフェイスデバイス活用技術開発・音声認識基盤技術」プロジェクトの一環として，Weighted Finite State Transducer (WFST) に基づく高精度な音声認識デコーダの開発を行っている．

WFST に基づく音声認識では，探索に先だち，音響モデルや言語モデル，単語発音辞書などの構成要素を合成してまとめあげ，一つの巨大な WFST 形式のネットワークを構築する．認識はこのネットワークを探索することで進められる．従来の探索ネットワークを動的に生成して認識する手法に比べて，探索ネットワークを保持するためのメモリ量を多く必要とする反面，モデルの融合が事前に行われることから，探索時に動的なモデル融合を行う必要がなく，高速なデコーディングが可能となる．また，様々な形式のモデルや辞書を扱う必要が生じても，最終的に WFST の形式でネットワークに変換できれば，モデルに応じてデコーダ自体を変更する必要がないため，柔軟なデコーダが実現できる．

本稿では WFST 音声認識デコーダで問題になるメモリ消費量の増加に対して，省メモリ化の検討を行った結果について述べる．また，認識精度の向上のため，複数の音響モデルを並列に利用する手法についての検討を行う．複数モデルについても省メモリ化手法が有効であるかを検討する．

2 WFSTを利用した音声認識

WFST とは，入力記号列に対して状態遷移を繰り返し，それに対応した出力記号列と重みを出力する有限状態オートマトンの一種である．音声認識に用いる際には，利用される様々な情報 (音響モデル，発音辞書，N-gram など) をそれぞれ WFST で表現し，それらを合成することで一つの探索ネットワークを作成する．

通常，大語彙連続音声認識では以下の 4 つの WFST を構成する．

H HMM の状態から文脈依存音素への WFST

C 文脈依存音素から文脈非依存音素への WFST

L 文脈非依存音素から単語への WFST

G 単語から単語 N-gram への WFST

合成演算を $\circ$ と表現すると，これらを合成した探索ネットワークは以下の式で表現される．

$$H \circ C \circ L \circ G \tag{1}$$

デコーダは，この探索ネットワークを用いて最尤となる単語列を探索し音声認識を行う．探索ネットワークを構築する際には，WFST の基本操作である決定化 (determinization)，最小化 (minimization) 等を用いてネットワークの最適化を行う．

この最適化によって探索の効率化が計られ高速に音声認識を行うことができる．また探索ネットワークに統一的な枠組を用いることにより，デコーダの変更を伴わずに様々なモデルを柔軟に利用することができる．

3 認識時の省メモリ化

WFST による音声認識では，肥大化した探索ネットワークの読み込みに伴う，メモリ消費量の増大がしばしば問題となる．その対策として，1) 事前のネットワーク構築の段階ですべての WFST を合成せず，一部の WFST については，探索中に動的に合成するようにして，読み込む探索ネットワークの肥大化を防ぐ手法 (on-the-fly 合成 [1–3])，2) 認識時に探索ネットワーク全体をメモリ上に読み込むのではなく，ディスク上に展開しておき，必要分だけを随時メモリ領域に読み込んで利用する方法 (disk-based search [4]) の２つについて検討を行った．

3.1 On-the-fly 合成

どの部分の WFST を事前の探索ネットワークの合成に利用するかによって，様々な方式が考えられるが，本稿では，以下の２つの式で表される合成について検討を行った．式中の括弧で示される部分が事前に合成される WFST である．

$$(H \circ C \circ L_{uni}) \circ G_{tri/uni} \tag{2}$$

$$H \circ (C \circ L \circ G) \tag{3}$$

L_{uni} は単語辞書に unigram 確率を付与した WFST，$G_{tri/uni}$ は trigram 確率を unigram 確率で割った値を持つ WFST である．

式 (2) の方式は Dolfing らの研究 [1] に基づいている．この方式では，unigram の確率を付与した WFST を，事前に構築した静的な探索ネットワーク中に組み込むことで，早期に単語の統計量を探索に利用することができ，先読みの効果を付与することができる．探索は，到達した状態において，次に遷移する状態や入力記号列などを合成演算に基づいて動的に生成することで行われる．この方式ではメモリ消費量を少なく抑えられる反面，合成演算を状態ごとに行うため，その計算に伴うオーバヘッドが問題になる可能性がある．

式 (3) では，HMM のトポロジーが反映されている H の部分を探索時に動的に融合する．ただし，本デコーダでこの合成方式を用いる場合に

*Memory reduction techniques for WFST-based ASR decoder by Tasuku Oonishi, Paul R. Dixon, Kouji Iwano, Sadaoki Furui (Tokyo Institute of Technology)

は，HMM としてスキップなしの left-to-right 型を扱うこととした．このような制限を与えると，H の WFST は各文脈依存音素が，一本のパスで表現されるような非常に簡単な構造となり，合成は「H 上の該当する音素のパスを事前に合成されたネットワークに当てはめる」という操作のみで行うことが可能であり，いわゆる合成演算を必要としない．この様な操作は一般的にデコーダで利用されている「ネットワーク状態遷移の動的な展開」と同じ処理を行うことにより実現できる．このため，式 (2) の説明中にあるような計算に伴うオーバヘッドを削減する効果が期待できる．

3.2 Disk-based search

Willett らは，WFST を利用した音声認識における省メモリ化の対策として，探索ネットワーク全体をディスク上に展開し，探索で到達した状態ごとに，必要となる情報のみをメモリ上に読み出す手法の提案を行っている [4]．我々も同様の処理の実装を行った．これでは，ある状態で仮説の展開を行う場合，その状態から遷移して到達する全ての状態の状態番号及び遷移時の入力記号，出力記号，重みの情報をメモリ上にコピーする．全ての仮説の展開が終了した際に，コピーされたデータをメモリ上から解放する．これらの処理により，探索ネットワークの保持に使用される占有メモリ領域の増加を抑えている．またディスクへのデータアクセスを素早くするため，予め「状態番号」と「遷移先の情報が格納されているディスク上の位置」の関連を表すテーブルをメモリ上に確保し，それを利用している．

3.3 状態・状態遷移の圧縮

ある状態とある状態を結ぶパス中に複数の状態が存在する状況を考えたときに，1) そのパス中の全ての状態が，そのパス以外の状態との遷移経路を持たず，しかも，2) パスから出力される記号が唯一である，という条件を満した場合に，該当パスを一つの状態遷移として新たに定義する．これにより探索ネットワークの状態・状態遷移の圧縮を行うことができる．文献 [5] では factoring の一つとして，この圧縮操作が提案されており，本研究でも，この手法を用いて省メモリ化の評価実験を行う．

4 省メモリ化手法の評価実験

実験には，日本語話し言葉コーパス (Corpus of Spontaneous Japanese) [6] を用いた．学習用データとして，音響モデルには 967 学会講演，言語モデルには学会講演と模擬講演の計 2,682 講演を用いた．音響特徴量には，フレームシフト 10ms, 分析窓幅 25ms の MFCC 12 次元+Δ MFCC 12 次元+Δ Δ MFCC 12 次元+Δ 対数パワー+Δ Δ 対数パワーの計 38 次元を用いた．音響モデルには 3,000 状態 32 混合の triphone HMM を用い，言語モデルには語彙サイズ 55,000 単語の trigram を用いた．

評価データには，テストセット 1 の 10 講演 (perplexity = 65.9, 未知語率= 0.52%) を用いた．実験には Intel Core2 2.4GHz 2GB メモリ，OS は Fedora 6 64 bit の計算機を使用した．

Table 1 on-the-fly 合成によるメモリ消費量削減の効果

Network type	Memory usage (MB)
$H \circ C \circ L \circ G$	580
$H \circ C \circ L_{uni} \circ G_{tri/uni}$	310
$H \circ C \circ L \circ G$	260
$fac(H \circ C \circ L \circ G)$	260
$fac(H \circ C \circ L_{uni}) \circ G_{tri/uni}$	260
$H \circ fac(C \circ L \circ G)$	240

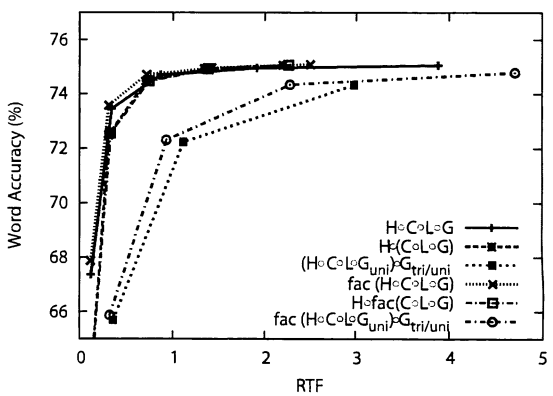

Fig. 1 on-the-fly 合成を行った場合の認識率と認識時間との関係 (CSJ テストセット 1)

4.1 On-the-fly 合成の効果

Table 1 に使用したネットワーク形態ごとの平均メモリ消費量を，Fig 1 に認識時間と単語正解精度との関係を示す．認識時間は Real Time Factor (RTF) で示す．Table 1 の上段は，ネットワークの圧縮を行わなかった場合のメモリ消費量を示しており，下段は行った場合 (fac(X) は WFST X に圧縮操作を行ったことを表す) を表している．メモリ消費量は Linux 上の top コマンドを利用して測定した．

ネットワークの圧縮を行わない場合では，式 (2), (3) に基づく on-the-fly 合成のいずれの場合でも 50% 程度のメモリ消費量が削減されており，さらにネットワークの圧縮操作を行うことでメモリ消費量の削減がされていることが分かる．

Fig. 1 の認識性能の結果を見ると，式 (2) に基づく on-the-fly 合成を行った場合には，行わない場合に比べ，多くの認識時間が消費されており，RTF=1 付近でも認識精度の収束が見られないことが分かる．これは，動的な合成に伴う合成演算のオーバヘッドの影響が大きいためであると考えられる．一方，式 (3) に基づいた on-the-fly 合成では，認識時間の増加はほとんど見られず，収束時の認識性能の違いもほとんど見られなかった．これは H の構造を制限して合成時の計算を簡略化した効果により，オーバーヘッドに伴う計算時間の増加を抑えて，WFST の合成を行うことができたためであると考えられる．

Table 2　Disk-based search によるメモリ消費量削減の効果

Netowork type	Memory usage(MB)
$fac(H \circ C \circ L \circ G)$	260
$fac(H \circ C \circ L \circ G) + disk$	130

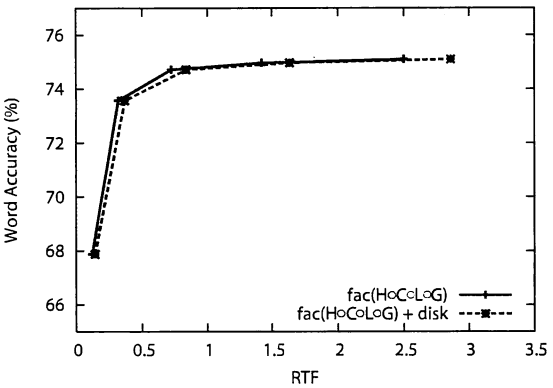

Fig. 2　Disk-based search を行った場合の認識率と認識時間の関係 (CSJ テストセット 1)

4.2　Disk-based search による効果

Table 2 に圧縮処理されたネットワーク ($fac(H \circ C \circ L \circ G)$) に対して disk-based search を行った場合のメモリ消費量を，Fig. 2 に認識時間と認識精度との関係を示す．Table 2 から disk-based search を用いることで 50 % 程度のメモリ消費量が削減可能であることが分かる．

Fig. 2 を見ると，disk-based search を行った場合と，行わない場合で認識時間の増加はわずかであり，大きな差異は見られない．このことからディスクアクセスに伴うオーバーヘッドが全体の認識時間と比較して少ないことが分かる．

5　複数モデルの利用による高精度化

実環境で音声認識システムの利用を想定した場合，システムには様々にな音声が入力されることが予想される．そのため認識システムには入力音声の変化に対する頑健性が求められる．そのため，種々の特徴を持った複数のモデルを利用することで，多様な音声に対して頑健な音声認識を実現する試みが行われている [7]．本研究でも同様の視点から，探索ネットワークに複数のモデルの情報を組み込むことによるモデルの高精度化についての検討を行う．

本研究で用いた探索ネットワークの構成手順について以下に示す．

1. 種々の特徴を持ったモデルから個別に WFST のネットワークを構築する
2. 構築したそれぞれの WFST ネットワークを統合操作により一つに統合し，探索ネットワークを構築する

認識時には，このように構築された探索ネットワークの最尤探索を行う．これにより，発話毎に種々のモデルで評価された仮説の中で最も尤度の高い仮説を選択した場合と同様の探索結果を得ることができるため，複数のモデルを利用して並列に認識し，それらを統合する処理を行う必要がない．このため様々なモデルの利用をデコーダを変更せずに容易に実現することが可能となる．

一方，この様な探索ネットワークを用いた場合にはモデル数に依存してネットワークのサイズが肥大化し，メモリ消費量が増加する可能性がある．以下では複数のモデルを利用した場合の認識性能と省メモリ化の効果について検討する．

Table 3　メモリ消費量削減の効果

Netowork type	Memory usage(MB)
SM	260
MM	700
$MM + disk$	320

6　複数モデルの利用による効果

本実験では複数の音響モデルを利用した場合の認識性能への効果について検討する．ここでは性別が特定されない様々な話者の音声が入力される場合を想定し，音響モデルとして男性・女性・性別非依存の 3 つのモデルを利用した．従って，評価データには，男性・女性の話者を含む CSJ のテストセット 2 (10 講演 perplexity=68, 未知語率=0.3 %) を用いた．その他の実験条件は 4 節と同様である．

Fig. 3 に認識精度と認識時間との関係を示す．Fig. 3 の SM は，音響モデルとして性別非依存モデルのみを用いた場合，MM は音響モデルとして，男性・女性・性別非依存モデルの複数のモデルを用いた場合，$MM + disk$ は，MM に disk-based search を行った場合を表す．ネットワークには状態の圧縮操作を行っており，on-the-fly 合成は利用していない．

Fig. 3 より，複数のモデルを表現した探索ネットワークを用いた場合，同程度の認識時間で，より高い認識精度が得られていることが分かる．認識率の収束点を比較すると 1.5% 程度の認識精度の改善が見られる．

また disk-based search を用いることによる認識時間の大幅な増加は見られないことが分かる．

Table3 より，disk-based search を行うことで，行わない場合と比較し 50% 程度のメモリ消費量が削減されており，一つの音響モデルのみを用いた場合とほぼ同程度のメモリ消費量であることが分かる．

以上より，複数モデルを用いて disk-based search を行うことで，メモリ消費量及び認識時間の大幅な増加なく，より高精度な探索が実現可能であることが確認された．

7　まとめ

本稿では WFST を利用した音声認識デコーダの省メモリ化についての検討を行った．それにより，ネットワークの状態圧縮，on-the-fly 合成，

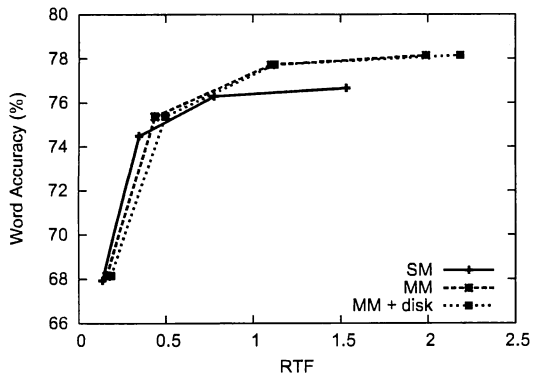

Fig. 3　複数モデルによる認識率と認識時間の関係 (CSJ テストセット 2)

disk-based search を行うことで 50% 程度のメモリ消費量が削減可能であることが確認された．さらに disk-based search を行った場合，認識速度の低下が少ないことが確認された．

また複数のモデルを用いることで，認識速度の低下なく認識精度の向上が可能であること，disk-based search を併用することにより，メモリ消費量の増加が抑制可能であることが確認された．

今後は言語モデルの高精度化と音響モデルと組み合わせた場合の効果やその省メモリ化について検討したい．

謝辞　本研究は 21 世紀 COE プログラム「大規模知識資源の体系化と活用基盤構築」及び経産省「情報家電センサー・ヒューマンインターフェースデバイス活用技術開発・音声認識基盤技術」プロジェクトの支援により行った．

参考文献

[1] H. J. G. A. Dolfing and I. L. Hetherington. Incremental language models for speech recognition using finite-state transducers. *Proc. ASRU*, 2001.

[2] D. A. Caseiro et al. A specialized on-the-fly algorithm for lexicon and language model composition. *IEEE Transactions on Audio, Speech, and Language Processing*, vol.14, no.4, pp. 1281–1291, 2006.

[3] T. Hori et al. Generalized fast on-the-fly composition algorithm for wfst-based speech recognition. *Proc. Interspeech*, pp. 847–850, 2005.

[4] D. Willett et al. Time and memory efficient viterbi decoding for lvcsr using a precompiled search network. *Proc. Eurospeech*, pp.847–850, 2001.

[5] M. Mohri et al. Weighted finite-state transducers in speech recognition. *Computer Speech and Language*, vol.16, no.1, pp.69–88, 2002.

[6] K. Maekawa et al. Corpus of spontaneous japanese: Its design and evaluation. *Proc. ISCA and IEEE Workshop on Spontaneous Speech Processing and Recognition*, pp.7–12, 2003.

[7] 篠崎他. 超並列デコーダを用いた話し言葉音声認識. 春季音響論, 2004.