社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2008－SLP－74（18）
2008 / 12 / 10

音声対話用音声認識システム

谷口　徹† 藤江 真也†† 小林 哲則†

† 早稲田大学情報理工学科
〒 169–8555 東京都新宿区大久保 3–4–1
†† 早稲田大学高等研究所
〒 169-8050 東京都新宿区西早稲田 1–6–1
E-mail: †ttani@ieee.org

あらまし 会話ロボット等の音声対話システムへの適用を目指し，音声認識方法の検討を行っている．音声対話システムにおいては，開発者としては開発のし易さから文法ベースの認識器が望ましいが，文法外発話に対して誤認識を起こす，文法が大きくなると認識性能が低下するなどの問題がある．さらに，ユーザによる文中の発話休止や，音声区間検出の誤検出により，発話のフラグメント化が起こる．我々はこれらの問題に対し，音声認識システムにおいて，複数文法の並列認識，文法外発話の棄却機能，デコーダによる複数発話片の統合手法を実現することで対処した．
キーワード 音声対話システム，音声認識，マルチデコーディング，音声区間検出，フラグメント発話

Speech recognition system for spoken dialogue system

Toru TANIGUCHI†, Shinya FUJIE††, and Tetsunori KOBAYASHI†

† Department of Computer Science and Engineering, Waseda University
3–4–1 Okubo, Shinjuku-ku, Tokyo, 169-8555 Japan
†† Waseda Institute for Advanced Study, Waseda University
1–6–1 Nishi-waseda, Shinjuku-ku, Tokyo, 169–8050 Japan
E-mail: †ttani@ieee.org

Abstract We have developed speech recognition method for spoken dialogue systems such as communication robots. In the case of development of spoken dialogue systems, grammar-based speech recognizers are desirable rather than n-gram based ones because of the ease of dialogue design. However, there are problems of the recognizers that out-of-grammar utterances cause recognition errors and that the performance of the recognizer declines if the size of the grammar increases. Furthermore, fragmentation of a spoken sentence occurs because of unfilled pause in the sentence and errors by voice activity detection. To overcome these problems, multi-decoding using multiple grammars, the rejection method of out-of-grammar utterances, and the recognition method with integration of fragmentary spoken sentences has been realized.
Key words spoken dialogue system, speech recognition, multi-decoding, voice activity detection, fragmented utterance

1. はじめに

我々は，対話状況がタスク内容に応じて時々刻々変化し，自由な発話形式を許容した上で，騒音下などの実環境で動作する会話ロボットの構築を目指している．本稿では，そのために従来の音声認識システムを拡張した，新しい音声認識システムを提案する．

現在までに多くの音声認識システムが提案されており，その一部は音声ディクテーションソフト，カーナビゲーションや携帯電話などの商品に搭載され利用されている．また，Julius [1], Sphinx-4 [2] などの音声認識システムはオープンソースウェアとして公開され，音声を用いたアプリケーションの構築に利用できる．しかし，これら従来の音声認識システムを音声対話システムへ適用することを考えると未だ多くの機能が不足しており，そのままでは適用することが困難である．そこで我々は音声認識システムに対して以下の拡張を行う．

まず，音声認識システム内で異なる言語モデルを持つデコーダが同時に動作できるよう拡張する．音声対話システム構築の際には，まず対話の状態系列を設計し，次に各状態で受理する文を設定する．このとき，（1）対話システム全体で大きな言語モデルを 1 つ用意する，（2）状態毎に小さな言語モデルを個別に用意し，認識時には状態毎に切り替えて用いる，という方法

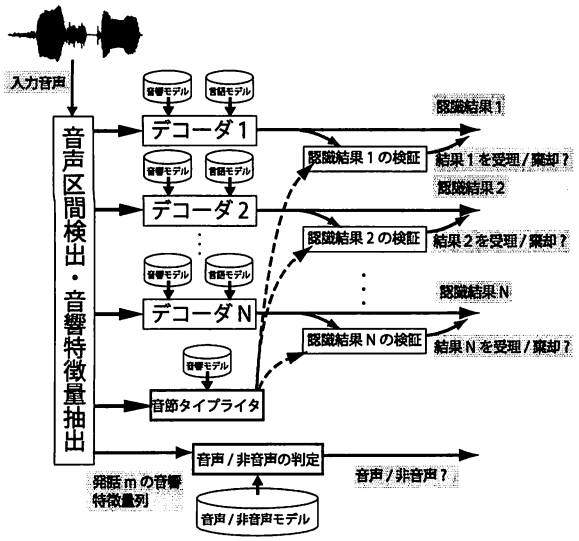

図1　音声対話のための音声認識システム

Fig. 1　Overview of the proposed speech recognition system for speech dialogue system.

がある．しかし，(1) では類似した文の間で認識誤りが多くなり，(2) では，状態推定を誤った時に認識ができないという問題がある．そこで，1つの音声認識システム内に，個別の言語モデルを持ったデコーダを並列に動作させることにする．これによって，類似文間の認識誤りを削減し，対話の状態推定に依存せず認識を行うことが可能となる．

次に，音声認識システムで各デコーダの認識結果を検証するよう拡張する．上記のように音声認識システムを拡張すると，個々の言語モデルがカバーする文の種類は少なくなり，文法外発話が多くなることで誤認識が増える．そこで，認識結果から想定される音声と実際の音声が音響的に大きく異なる場合には，認識結果の棄却を行い，文法外発話が対話制御に用いられないようにする．

最後に，対話音声のフラグメント化に対応するために認識デコーダアルゴリズムの拡張を行う．音声対話では，話し言葉の使用，フィラー・休止の挿入など自由な発話形式が許容され，また，雑音下など音響的に不利な状況でも動作することが求められる．その際の音声認識システムに関する問題として，1つの文発話がいくつかのフラグメントに分割されてしまうことが挙げられる．発話が文中で休止したり，音声認識の前段に行われる音声区間検出（VAD）の誤りがその原因である．本研究では，連続する複数の発話フラグメントに渡って文仮説の探索を行うように音声認識デコーダを拡張する．

以下，2. では本稿で提案する音声認識システムの構成を述べ，3. では，提案する認識システムの中核となる音声認識デコーダのアルゴリズムについて述べる．4. で従来の音声認識技術との関連について述べた後，5. で結論を述べる．

2.　音声対話のための音声認識システム

我々が構築を目指しているロボット対話システムのように，対話場面や話題が時々刻々変化するような音声対話においては，場面や話題に応じて，複数の言語モデルを準備し，それぞれの言語モデルに基づく各ユーザ発話の音声認識結果を総合的に用いて対話を進行させることが望ましい．

他の方式として，対話場面によらずシステムが受理すべき全ての文を受理可能な言語モデルを用意する方法があるが，環境雑音や，対話のような自由な発話形式という厳しい条件下においては誤認識が多く，システムが望む認識結果が得られる可能性は大変低くなる．この問題に対処するため，例えば最尤の認識結果のみでなく，尤度の高い方から複数の文候補（Nベスト候補）を得て用いるという方法がある．しかし，Nベスト候補は多くの場合，最尤候補の単語を少しずつ変化させたものに過ぎないので，その対話場面で求めている結果が必ずしも得られないことがある．また，n-gram 言語モデルに対して，想定される対話場面や直前の対話履歴によって適応を行うという方法も考えられる．しかし，計算速度の問題から，逐次進行する対話中に適応を行うことは困難である．言語モデルは複数用意するが，そのうち1つだけを対話場面毎に選択して用いるという方法も考えられるが，常に対話システムが想定している対話場面が正しいとは限らない．従って，あらゆる対話場面の可能性を考慮して複数の言語モデルによってユーザ発話を待ち受けておいて，各モデルによる認識結果から事後的に結果を選択する方がより頑健な対話管理を行えると考えられる．

そこで，図1に示すような音声対話のための音声認識システムを提案する．提案する音声認識システムにおいては，入力音声に対して，複数の音声認識デコーダを並列に動作させることができる．各デコーダの認識結果は音響尤度を用いて検証され，結果の受理・棄却を決定し，認識結果と共に対話管理部に送られる．また，以上の処理とは別に，同じ音声に対して音声・非音声の判定を行い，その結果も対話管理部に送信される．

各デコーダでは，それぞれ異なる内容の言語モデルを持つことを想定している．各デコーダは時間フレーム同期で動作し，音響モデルを共有していれば，デコーダの計算で多くを占めている音素 HMM の音響尤度計算の結果をキャッシュしてデコーダ間で共有することで，計算量を大幅に減らすことができる．各デコーダとして，n-gram デコーダ [4]，次の 3. で述べる FST デコーダ（主に記述文法に対応）といった各種類のデコーダを選択して用いることができる．

音声認識結果の検証には音節タイプライタによる方法 [5] を用いる．音節タイプライタによる対数音響尤度を S_p，デコーダによる対数音響尤度を S_d としたとき，下式のように，それらの差により検証を行う．

$$\frac{|S_p - S_d|}{N} \leqq S_T \qquad \text{(受理)} \qquad (1)$$

$$\frac{|S_p - S_d|}{N} > S_T \qquad \text{(棄却)} \qquad (2)$$

なお，N はフラグメント中の時間フレーム数，S_T は予め定め

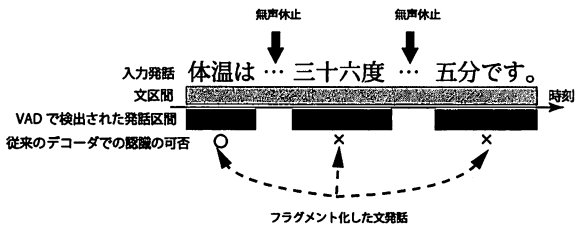

図2 フラグメント化した発話の例.
Fig. 2 Example of fragmented utterances.

た閾値である．音節タイプライタには，3.1 で説明する FST デコーダに対して，あらゆる音節の並びを許すよう記述した文法を与えたものを用いる．従って，音節タイプライタは通常の FST デコーダなので，他のデコーダと並列して動作させることができ，先に述べたキャッシュの効果により計算量は比較的少なくすることができる．

音声・非音声の判定には，音声，非音声の音響信号からそれぞれ抽出した音響特徴量を混合正規分布モデル（GMM）でモデル化したものを用いる [6]．入力発話に対して算出した各モデルによる尤度を比較し，尤度が大きい方のカテゴリとして判定する．

以上のデコーダ，認識結果検証部と音声・非音声の判定部を含む音声認識システムの構成や，用いる音響モデル・言語モデル，各デコーダの結果の出力方式などは XML 形式の設定ファイルで記述するよう実装されており，容易にシステムの構成や動作を変更することが可能である．

3. 音声認識デコーダの拡張

フラグメント化した発話に対応するため行った，FST デコーダの文仮説探索アルゴリズムの拡張について述べる．

対話システムのユーザ発話が単語ではなく文やフレーズのように比較的長い単位の場合，言い淀みによる文中の無声休止によって，1 文が複数のフラグメントに分割されてしまうことがある（図 2）．対話音声において発話休止は，文の終了を判断する重要な手がかりであり，しばしばシステム発話のトリガーとして用いられる．しかし，文中の無声休止の存在を考慮すると [7]，それらはユーザ発話終了の決定的な証拠とは言えない．また，環境雑音が多く入力音声の SNR が低い場合，VAD の誤りによっても発話のフラグメント化が起こりやすい．そのため，発話休止を文終了の手がかりとする対話システムは，ユーザ発話中など，誤ったタイミングで応答してしまうことがある．

発話のフラグメント化は，音声認識結果そのものにも大きな影響を与える．多くのデコーダでは文の仮説探索は VAD で検出された発話単位で行われてしまうため，フラグメント化が起こると言語モデルによる言語的制約が利用できなくなる．特に言語モデルとして記述文法を利用している場合には，認識結果に致命的な影響を与える．そもそも発話においては，音響信号に基づく VAD により文区間を検出することは本質的に困難であり，しばしば各フラグメントに分割される形で 1 文の発話が検出されることとなる．従って，音声認識デコーダにおいて言語モデルによる言語的制約を用いて，トップダウンに文区切りを決定しながら音声認識を行うのが適切だと考えられる．

発話休止の他に文終了を決定する手がかりとしては，音声の基本周波数の変化といった韻律情報も有力であるが [8]，本稿ではデコーダのアルゴリズムを改良することで，言語モデルによる言語的な制約を利用し，フラグメント化した発話から音声認識を行うことを試みる．

3.1 従来の FST デコーダ

藤江らは，音声対話システムに相槌・復唱機能を実現するために有限状態トランスデューサ（FST; finite state tranceducer）を用いた音声認識デコーダを作成している [9]．我々はこの音声認識デコーダを基にフラグメント化した発話を認識可能なデコーダを作成する．

基となるデコーダにおいては，単語辞書，記述文法をそれぞれ FST で記述し，合成することで文仮説ネットワークを事前に作成している．デコード時には，それと音素 HMM を組み合わせて文仮説の探索を行う．この FST 作成の際，合成した FST の最小化・決定化を認識に先だって行っておくことによりネットワークを最適化し，音声認識の探索を高速化することができる．これらの FST に対する処理は AT&T FSM Library [10] を用いて行っている．こうして作成した FST の入力には音素（列），出力には認識結果に対応する単語（列），相槌・復唱を表す記号，意図理解のためのキーワードなどを設定する．これによって，デコード時には，発話内容を文として認識すると共に，音声認識の後段の対話管理のヒントとなる情報を得ることができる．

仮説探索は音響特徴量のフレーム同期でビームサーチを行っており，入力音声に対して遅延が少なく高速な音声認識を実現している．そのため，高速な応答が必要な対話システムに適した方式のデコーダとなっている．

3.2 フラグメント化した発話のための FST デコーダ

前節で説明した，従来の FST デコーダアルゴリズムでは，VAD で切り出された音声区間毎に文仮説の探索を行い，次の発話区間の開始時点で仮説を初期化する．そこで，発話区間終了時にデコーダ内で保持している仮説のうち，文末に達していないものを保存しておき，次の発話区間でも保存した仮説について引き続き探索を行うようにすることで，フラグメントにまたがった文の認識を行うことができる．

3.2.1 文仮説スコアの算出

仮説探索に用いるスコアは通常，1 フラグメント毎に計算される．ここで提案するデコーダのアルゴリズムでは，1 フラグメント毎ではなく，複数のフラグメントに渡って計算する．それらのスコアはフラグメント間での文の接続を考慮して，以下のように計算することにした．まず，m 番目のフラグメントに対応する部分文仮説を s_m，入力音声を x_m とした時，事後確率 $p(s_m|x_m)$ は

$$p(s_m|x_m) \propto p(x_m|s_m)p(s_m) \tag{3}$$

のように計算できる．上式中の $p(x_m|s_m)$，$p(s_m)$ はそれぞれ，音響モデル，言語モデルに基づいて計算する．そして，連続する

2つのフラグメント $m-1$, m の事後確率 $p(s_{m-1}, s_m|x_{m-1}, x_m)$ は以下のように計算する.

$$p(s_{m-1}, s_m|x_{m-1}, x_m) \propto p(x_{m-1}|s_{m-1}) \cdot p(x_m|s_m) \cdot p(s_{m-1}) \cdot p(s_m|s_{m-1}) \quad (4)$$

つまり，前後のフラグメントの音響尤度，先行フラグメントの言語尤度，2フラグメント間の部分文仮説の連鎖確率 $p(s_m|s_{m-1})$ の積によって計算する.

ここで，連鎖確率 $p(s_m|s_{m-1})$ を，先行する文仮説 s_{m-1} が文末まで達しているか否かによって場合分けして考える．先行フラグメントの文仮説が文末まで達していれば，新しい文仮説が文頭から開始される確率は，そうでない場合と比較して非常に高くなる（1と考えてよいだろう）はずである．一方，仮説が文の途中で中断している場合は，次のフラグメントで発話が文中から再開される場合と，文頭に戻って発話される場合の両者を考慮する必要がある．前者は文中の無声休止，後者は言い直し，言い換えに相当する．具体的には，先行する文仮説 s_{m-1} が文末に達している場合，文中で中断している場合をそれぞれ，$-\bullet$, $-\circ$ と表し，後続する文仮説 s_m が文頭から開始される場合，文途中から開始される場合をそれぞれ，$\bullet-$, $\circ-$ と表すことにするとそれぞれ，

$$p(s_m|s_{m-1}) = p(w_1^m|s_{m-1}) \cdot p(w_2^m, w_3^m \cdots |w_1^m) \cdot \begin{cases} p(\bullet-|-\bullet) & = 1 \\ p(\circ-|-\bullet) & = 0 \\ p(\circ-|-\circ) & = 1-\alpha \\ p(\bullet-|-\circ) & = \alpha \end{cases} \quad (5)$$

と考えることができる．ここで，w_n^m は部分文仮説 s_m の n 番目の単語であり，α は $[0, 1]$ の実数値である.

言語モデルに n-gram を用いている場合，式 (5) の1行目右辺は言語モデルによる算出される単語連鎖の確率となる．言語モデルに 2-gram を用いている場合，$p(w_1^m|s_{m-1})$ は先行する文仮説の最後の単語と w_1^m の 2-gram から計算する．ただし，先行する文仮説が単語途中で中断している場合は $p(w_1^m|s_{m-1}) = 1$ とする．式 (5) の2行目で，$\alpha > 0$ とすると，元々n-gram に記述されていない，文中から文頭への接続を許すことになる．元々話し言葉において学習された n-gram なら $\alpha = 0$ でよいはずだが，例えば，新聞記事のような書き言葉のコーパスで学習された n-gram を利用している時には，$\alpha > 0$ と設定すると，言い直しに対応できる.

言語モデルに記述文法を用いている場合，式 (5) の1行目右辺は計算することができないので，常に1と設定する．従って，式 (5) の2行目が重要になる．α の値が大きいほど，フラグメント間を横断した文が認識されやすくなる．我々の現在の実装では，α は適当な大きさの定数としている．しかし，実際は固定値ではなく，フラグメント間の無音声区間の長さや先行フラグメントにおける基本周波数変化などの韻律情報，FST ネットワークの複雑さなどの言語的制約にも影響を受ける確率と考えられる．α の値については今後さらなる検討を行う予定である.

Algorithm 1 拡張した FST デコーダのフラグメント終了・開始間のアルゴリズム

1: （先行するフラグメントでの探索を終了）
2: **if** 文末に達した最尤仮説のスコア ≧ それ以外の最尤仮説のスコア × α **then**
3: $L \leftarrow$ 文末に達した最尤仮説のスコア
4: 文末に達した最尤仮説の文を出力.
5: **else**
6: $L \leftarrow$ それ以外の最尤仮説のスコア × α
7: それ以外の最尤仮説の文を出力.
8: **end if**
9: 文末に達している，または，音素 HMM の最終状態に達していない仮説を次のフラグメントで探索を行う探索仮説群から取り除く.
10: 文頭から開始する仮説を探索仮説群に追加．そのスコアを L に設定.
11: （後続するフラグメントでの探索を開始する）

3.2.2 仮説探索・認識結果出力のアルゴリズム

上に述べた方法により，フラグメントを横断した文仮説のスコアを計算することができる．そして，実際のデコード時には，先行フラグメントの探索終了時から後続フラグメントの探索開始時の間に Algorithm 1 の処理を行うことで，FST デコーダの拡張を行うことができる．これにより，式 (4) を最大化する規準で，連続するフラグメント中で最尤の文仮説を決定・出力することができる.

Algorithm 1 の 2〜8 行目では，後続するフラグメントにおいて新規に文が開始される仮説に対して，式 (5) を考慮して，後続フラグメント開始時点で取り得る最尤の仮説の選択を行っている．同時に，その際に選択された仮説の文を出力している．しかし，後続フラグメントの探索結果次第では，この時選択されなかった他の仮説が後のフラグメント終了時には最尤となることもあり得る．そのため，本デコーダの認識結果を受け取る対話管理部においては，現在のフラグメントでの認識結果のみでなく，後続フラグメントの認識結果も踏まえて総合的に行動決定をする必要がある．例えば，後続するフラグメントの結果を一定時間待って，次の行動を選択するなどの戦略を取る必要がある．また，後のフラグメントの認識結果が先のフラグメントの認識結果を否定していることが判断できるように，音声認識システムの認識結果には発話開始・終了時刻情報を付与しておく必要がある.

Algorithm 1 の 9 行目で，音素 HMM の最終状態に達していない仮説を捨てているのは，VAD や無声休止でフラグメント化する場合，その分断は音素内ではなく音素間で起こると想定しているからである．また，これにより後続フラグメント開始時の仮説数を減らし，計算量も少なくなることも期待している.

3.2.3 単語モデル FST の拡張

本デコーダでは，音素から文を出力する FST を構築し，言語モデルを表現している．記述文法からそのまま FST を作成した場合は，特に文中での無声休止は考慮しないことが普通である．しかし，フラグメント化した発話では，無声休止や VAD の影響

でフラグメントの冒頭と末尾の音声に短い無音区間（ショートポーズ）が生じる．そのため，従来の方法で作成したFSTでは，フラグメントの冒頭と末尾で，ショートポーズと音素HMMのミスマッチにより音響尤度の低下が起こる可能性がある．

この問題に対応するため，記述文法からFSTを構築する際に，各単語の末尾にショートポーズに対応する音素を必ず挿入するようにした．そのショートポーズはスキップも許すようにFSTのネットワークを記述することで，フラグメントが起こらない場合にも対応する．

3.2.4 認識結果検証法の拡張

本デコーダでは，複数のフラグメントを一つの文として認識し，出力する．一方，認識結果の検証時には，音節タイプライタとデコーダにおける音響尤度を比較することで，認識結果の受理・棄却を行う．そのため，音節タイプライタにおいては，認識結果に対応する複数のフラグメントの音響尤度を全て積算してデコーダの音響尤度と比較する必要がある．このために，各フラグメントにおける音節タイプライタの音響尤度を保存しておき，必要に応じて取り出せるようにした．また，過去全ての音響尤度を保存しておくのは記憶領域の無駄なので，音声認識システム内の全てのデコーダが保持する仮説のうち最も古い仮説に対応する音響尤度までを残すようにしている．

ところで先に述べたように，フラグメント化した発話に対応したデコーダは，認識結果が後の時刻の認識結果によって上書きされることがある．上書きされるような認識結果は，しばしばこの結果検証によって棄却することができる．従って，本デコーダの拡張の際には，この認識結果の検証の役割がより重要となると考えられる．

4. 関連する音声認識技術

本章では，提案した音声認識システムと関連する技術について述べることで，本システムの得失について議論する．

Stolckeらは対話の場面に基づいて言語モデルの重みを変更することで，わずかだが音声認識精度を改善できることを示した[11]．この手法は，人間同士の対話に関してオフラインで音声認識を行っており，ユーザとシステム間の対話をオンラインで認識を行う本研究の目的とは異なっている．一方，安田らは，大語彙の言語モデルと，ある場面で想定されるユーザ発話に文法を限定した言語モデルによる認識結果を事後的に選択することで，音声認識精度を改善できることを示した[3]．本研究は，安田らの方法を，2つ以上の言語モデルを用いるように一般化する試みと捉えることができる．

李は音声認識システムJulius[1]において，複数のn-gram，または記述文法デコーダを並列動作させる機能を実現している．それに対して本研究でも同様の機能をn-gramデコーダとFSTデコーダに関して実現している．FSTデコーダを用いることで，3.2で述べたような言語ネットワークの拡張，デコードアルゴリズムの拡張など，対話システムの構築に必要な拡張を柔軟に行うことができた．

河原らは，講演音声における話し言葉音声認識のためにデコーダのいくつかの改善を行っている[12]．講演音声においては，対話音声と同様に，文中に頻繁にショートポーズが発生する．そこで，2パスデコーダの1パス目でショートポーズを検知する度に，それまでの2パス目の認識を行うことで，文仮説の探索範囲を小さくし，さらに認識精度を向上させている．しかし，ショートポーズを越えて探索中の文仮説を引き継がないため，本研究が想定しているように記述文法を用いている場合には，フラグメント化した発話を認識することができないと考えられる．N-gramの場合でも，ショートポーズ前の多くの仮説を捨てることで，認識精度が低下する可能性もあり，今後の検討が必要である．

5. 結　　論

本研究では，音声対話システムにおける特有の問題を解決した次の機能を持つ音声認識システムを提案した．（1）音声対話の対話場面毎に設定したユーザ発話を受理可能な言語モデルを作成し，各言語モデルを用いて，n-gram，FSTの各種デコーダで並列に認識を行う．（2）各デコーダの認識結果を，音響尤度を用いて受理・棄却の検証を行う．（3）文中の無声休止やVAD誤りでフラグメント化した発話を認識可能である．以上より，文法外発話や対話状態の推定誤りに対応可能で，話し言葉に特有の発話現象や雑音に頑健に，従来の音声認識システムを拡張することができた．

提案したシステムは接話型マイクで利用した際には十分な性能を発揮したが，会話ロボット上に搭載してハンズフリーの音声認識を試みたところ，認識結果検証部の棄却に失敗することがあった．また，フラグメント化発話の認識に関しては，用いる文法が大きくなるほど認識が困難となった．現在のアルゴリズムでは言語的制約のみでフラグメント横断の認識を行っているが，韻律情報などから得られる「文中らしさ」・「文末らしさ」を利用して認識精度を向上させることは今後の課題である．同時に，実際の音声を用いた認識結果検証，フラグメント発話認識の客観評価を行うことも今後の課題としてあげておく．

謝辞　本研究の一部は，NEDO戦略的先端ロボット要素技術開発プロジェクト・高齢者コミュニケーションRTシステムの支援によるものである．

文　　献

[1] 李：“大語彙連続音声認識エンジンJulius ver.4”, 情報処理学会研究報告. SLP, 音声言語情報処理, 2007, 社団法人情報処理学会, pp. 307–312 (2007).

[2] W. Walker, P. Lamere, P. Kwok, B. Raj, R. Singh, E. Gouvea, P. Wolf and J. Woelfel: “Sphinx-4: A flexible open source framework for speech recognition”, Tech. Rep. TR-2004-139, Sun Microsystems Laboratories (2004).

[3] 安田, 堂坂, 相川：“2つの認識文法を用いた主導権混合型対話制御”, 情報処理学会研究報告. SLP, 音声言語情報処理, 2002, 社団法人情報処理学会, pp. 127–132 (2002).

[4] 柴田, 小林：“ワンパストライグラムデコーダにおける単語履歴の束ね処理に関する検討”, 日本音響学会秋期研究発表会講演論文集, No. 3-9-11, pp. 151–152 (2002).

[5] 渡辺, 塚田：“音節認識を用いたゆう度補正による未知発話のリジェクション”, 電子情報通信学会論文誌, **J75-DII**, 12, pp. 2002–2009 (1992).

[6] N. Binder, K. Markov, R. Gruhn, 中村：“GMMを用いた音声区間の検出”, 日本音響学会秋季研究発表会, I, pp. 153–154 (2001).

[7] 伊藤, 秋葉, 上條, 田中："休止を区切りとした対話処理", 情報処理学会研究報告. SLP, 音声言語情報処理, 95, 社団法人情報処理学会, pp. 135–138 (1995).
[8] 野村, 河原, 堂下："F0 パターンに基づく講義音声の文単位へのセグメンテーション", 電子情報通信学会技術研究報告. SP, 音声, 99, 社団法人電子情報通信学会, pp. 31–38 (1999).
[9] 藤江, 福島, 柴田, 小林："FST と韻律情報を用いた相槌・復唱機能を持った対話ロボット", 人工知能学会研究会資料, SIG-SLUD-A401-03, pp. 15–20 (2003).
[10] M. Mohri, F. C. N. Pereira and M. D. Riley: "AT&T FSM library - finite-state machine library". http://www.research.att.com/~fsmtools/fsm/.
[11] A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, R. Bates, D. Jurafsky, P. Taylor, R. Martin, C. Ess-Dykema and M. Meteer: "Dialogue act modeling for automatic tagging and recognition of conversational speech", Computational Linguistics, **26**, 3, pp. 339–373 (2000).
[12] 河原, 加藤, 南條, 李："話し言葉音声認識のための言語モデルとデコーダの改善", 情報処理学会研究報告. SLP, 音声言語情報処理, 2001, 社団法人情報処理学会, pp. 15–22 (2001).