概要

従来の単語音声認識においては, 主に音声の音韻的特徴が用いられてきた. しかし, 日本語では,「箸」,「橋」のような音韻的には同一だがアクセントの違いによって弁別できる単語が存在する. 過去の研究において, 同音異義語の音声認識の研究はあまり行われていない [6] .

そこで本研究では, 音素 HMM にアクセントの情報を加えたアクセントモデルを提案し, 有効性検証のために単語音声認識実験を行い, 同音異義語の認識率を調査した. また, 特徴パラメータに一般的に使用されているフォルマント成分は音韻的特徴しか含んでいないため, 韻律的情報を含む FBANK を用いた特徴パラメータを用いて認識精度を調査した.

その結果, アクセントモデルの同音異義語認識の有効性を確認した. また,FBANK を用いた特徴パラメータの有効性を確認した.

目 次

図 目 次

表 目 次

1　はじめに

従来の単語音声認識においては, 主に音声の音韻的特徴が用いられてきた. しかし, 日本語では,「箸」,「橋」のような音韻的には同一だがアクセントの違いによって弁別できる単語が存在する. 過去の研究において, 同音異義語の音声認識の研究はあまり行われていない [6] . 韻律的特徴を用いた研究としては, 高橋ら [5] の研究がある. 高橋らの研究は音声からピッチパターンを抽出し単語のアクセント型の 0 型,1 型,N 型 (0,1 型以外) を認識する研究であり, 認識においては音声の音韻とアクセントは別々に認識される.

本研究では音韻とアクセントを別々に認識する必要はないとし, 単語のアクセント型の情報と各モーラ位置でのアクセントの高低の情報を音素 HMM に付与し, 音声の音韻とアクセントを同時に認識するようにして同音異義語の認識精度を調査した. また, 特徴パラメータに一般的に使用されている MFCC は音韻的特徴しか含んでいないため, 韻律的情報を含む FBANK を用いた特徴パラメータを用いて認識精度を調査した.

実験の結果, 音素 HMM に単語のアクセント型の情報と各モーラ位置でのアクセントの高低の情報を加えることによって, 同音異義語が認識でき, 精度が高いことを確認した. また, アクセントの情報を用いたモデルは用いないモデルより単語音声認識精度が高いことを確認した。そして, アクセントの情報を用いた同音異義語認識において,FBANK を用いた特徴パラメータは,MFCC を用いた特徴パラメータより精度が高いことを確認した。

2　音声分析

2.1　音声の生成構造

音声信号は, 人間の調音器官により生成される音響信号である. 音声の生成は,「音源」により生成された音が「調音器官」により形成される音響的なフィルタを通過することでさまざまに変化し, 口または鼻から「放射」されるというのが基本的な構造になる. 調音フィルタは, ほとんどの場合に伝達関数が

$$H(z) = \frac{b_0}{1 - a_1 z^{-1} - a_2 z^{-2} - \cdots - a_p z^{-p}} \tag{1}$$

により伝えられる全極形のシステムであると仮定され, 音声認識の分野では広く妥当な仮定として受け入れられている [2] .

2.2　音声の特徴抽出

音声認識のための信号分析の目的は, 与えられた信号を生成した調音フィルタの性質を信号より推定することにあり, 信号の周波数領域における表現がその基礎を与える. 音声から連続する数十 ms 程度の時間長の信号区間を切り出し, 切り出された信号が定常確率過程に従うと仮定して, スペクトル解析を行う. すなわち, 与えられた信号 $s(n)$ に長さ N の分析窓を掛けることで以下のように信号系列 $s_w(m;l)$ を取り出す.

$$s_w(m;l) = \sum_{m=0}^{N-1} w(m)s(l+m)(l = 0, T, 2T, \cdots) \tag{2}$$

ここで, 添え字 l は, 信号の切出し位置に対応している. すなわち,l を一定間隔 Tっで増加されることで, 定常とみなされる長さ N の音声信号系列 $s_w(n)(n = 0, \cdots, N-1)$ が間隔 T で得られる. この処理はフレーム化処理と呼ばれ,N をフレーム長,T をフレーム間隔と呼ぶ. また, フレーム化処理を行う窓関数 $w(n)$ としては, ハミング窓やハニング窓がしばしば用いられる.

$$\text{ハミング窓} : w(n) = 0.54 - 0.46\cos(\frac{2n\pi}{N-1})(n = 0, \cdots, N-1) \tag{3}$$

$$\text{ハニング窓} : w(n) = 0.5 - 0.5\cos(\frac{2n\pi}{N-1})(n = 0, \cdots, N-1) \tag{4}$$

フレーム化処理によって得られた音声信号系列の短時間フーリエスペクトルは, 離散フーリエ変換 (DTFT) により以下で与えられる.

$$S(e^j\omega) = \sum_{n=0}^{N-1} s_w(n)e^{-j\omega n} \tag{5}$$

実際の信号処理過程では, 離散フーリエ変換 (DFT) をその高速算法である FFT を用いて実行し, 当該音声区間のスペクトル表現とすること t が一般的である. すなわち

$$S'(k) = S(e^{j\frac{2\pi}{N}k}) = \sum_{n=0}^{N-1} s_w(n)e^{-j\frac{2\pi}{N}kn}(k = 0, \cdots, N-1) \tag{6}$$

なる複素数系列 $S'(k)$ が音声のスペクトル表現として最も一般的に用いられる. 音声信号の音素的特徴は主として調音フィルタの振幅伝達特性に含まれている. したがって, 音声認識においては, 音声信号の振幅スペクトル, あるいはその 2 乗値であるパワースペクトルが注目すべきスペクトル表現である.

2.3 ケプストラム

音声のパワースペクトラムは, 声帯の振動や, 摩擦による乱流などの音源信号に調音フィルタが畳み込まれたものであり, 音素の音響的な特徴は, 調音フィルタの振幅伝達特性によって, 主として担われている. このため, 音声信号から音素の特徴を抽出するためには, 観測された音声のパワースペクトラムから, 音源信号のスペクトルと, 調音フィルタのスペクトルを分離し, 調音フィルタの特性にのみ関連する情報を抽出すれば良い. しかし音声信号から聴音フィルタを分離する問題は, 出力信号 $y(n) = x(n) * h(n)$ から, 入力信号 $x(n)$ とシステムの伝達関数 $h(n)$ を分離する問題である.

ケプストラム (cepstrum) $c(\tau)$ は, 波形の短時間振幅スペクトル $|S(e^{j\omega})|$ の対数の逆フーリエ変換として定義される. 音源信号のスペクトラムを $G(e^{j\omega})$, 調音フィルタの伝達特性を $H(e^{j\omega})$ とすると次の関係が得られる.

$$S(e^{j\omega}) = G(e^{j\omega})H(e^{j\omega}) \tag{7}$$

この対数を取ると,

$$log|S(e^{j\omega})| = log|G(e^{j\omega})|log|H(e^{j\omega})| \tag{8}$$

となる. 次にこれをフーリエ逆変換すると,

$$c(\tau) = \mathcal{F}^{-1}log|S(e^{j\omega})| = \mathcal{F}^{-1}log|G(e^{j\omega})| + \mathcal{F}^{-1}log|H(e^{j\omega})| \tag{9}$$

となり, これがケプストラムである. 離散フーリエ変換 (DFT) で求めると,

$$c_n = \frac{1}{N}\sum_{k=0}^{N-1} log|S(k)|e^{j2\pi kn/N}(0 \leq n \leq N-1) \tag{10}$$

となる.

ケプストラムという言葉は, スペクトルを逆変換するという意味から,spectrum をもじって作った造語であり, その変数は frequency をもじってケフレンシー (quefrency) と呼ばれる [2] .

従来の音声認識では, 特徴パラメータとしてケプストラムが使われてきた. ケプストラムは低次にフォルマント情報を高次にピッチ情報を含んでいる. しかしピッチ情報は正確なピッチ周波数の抽出が困難であるため, 音声認識ではフォルント情報しか用いられていない.

2.4 FBANK

FBANK は音声波形をフーリエ変換して得られたパワースペクトラムの周波数を使用する. パワースペクトラムを少ない次数で効率的に表現するために, メル分割されたフィルタバンクの対数パワーを使用する. またパワーケプストラムの全域に, 人間の聴覚の特性にあわせて低周波部分は細かく, 高周波部分は大まかに調べるためメルスケールに沿って等間隔に配置された三角関数のフィルタをかける. この三角関数の個数がフィルタバンクのチャンネルのチャンネル数 (特徴パラメータにおける次数) を表している. 周波数メル分割の式は

$$Mel(f) = 2592 \log_{10}(1 + \frac{f}{700}) \tag{11}$$

となる. そして, フィルタバンクの出力に log 対数をとったものを FBANK として使用する.

2.5 MFCC

ケプストラムパラメータには, 多様な計算方法がある. その中には MFCC(Mel-Frequency Cepstrum Coefficient) がある.MFCC の計算では, スペクトラル分析は周波数軸上に三角窓を配置し, フィルタバンク分析により行う. すなわち, 窓の幅に対応する周波数帯域の信号のパワーを, 単一スペクトルチャンネルの振幅スペクトルの重みづけ和で求める. さらに, 窓はメル周波数軸上に等間隔に配置される.

最終的に, フィルタバンク分析により得られた帯域におけるパワーを離散コサイン変換することで,MFCC が求められる.

$$c_i = \sqrt{\frac{2}{N}} \sum_{j=1}^{N} m_j \cos(\frac{\pi i}{N}(j - 0.5)) \tag{12}$$

N はフィルタバンクチャンネの数を表し,m_j は対数フィルタバンクの振幅を表す.

2.6 本研究で使用する特徴パラメータ

従来の音声認識の特徴パラメータに用いられている MFCC には韻律情報が含まれていないため, 同音異義語を認識する実験において認識精度が低いと予想される. そのため, 本研究では, 韻律情報が含まれている FBANK を特徴パラメータとして使用する. なお,FBANK を用いると MFCC より単語音声認識精度が向上することが知られている [9] .

3 HMMを用いた音声認識

3.1 HMMを用いた音声認識の理論

音声認識は, パターン認識の一分野である. 音声波形から認識に有効な特徴パラメータが抽出された後は、通常のパターン認識の技術と本質的に変わりはない. 通常のパターン認識との違いは, 音声パターンが時系列パターンであることと言語情報の制約を受けることである. パターン認識には構造的・構文的パターン認識法と統計的・確率的パターン認識法が存在する。最近になって, 音声パターンの時系列パターンに対しての統計的・確率的パターン認識法が HMM(Hidden Markov Model; 隠れマルコフモデル) による手法である [1] .

HMM は, 出力シンボルによって一意に状態遷移先が決まらないという意味での非決定状態オートマトンとして定義される. このモデルでは, 状態と出力シンボルの 2 課程を考え, 状態が確率的に遷移するときに対応して確率的にシンボルを出力する. このとき観測できるのはシンボル系列だけであることから Hidden(隠れ) マルコフモデルとよばれている.

HMM による音声認識では, 各カテゴリの HMM に対して入力パターンの特徴パラメータ時系列に対する尤度を求め, それを最大にするモデルに対応するカテゴリを認識結果とするのが基本手法である.

HMM は以下の組から定義される.

- 状態の有限集合;$S = \{s_i\}$
- 出力シンボルの集合;$O = \{o_i\}$
- 状態遷移確率の集合;$A = \{a_{ij}\}$; a_{ij} は状態 s_i から 状態 s_j への遷移確率, ここで $\sum_j a_i j = 1$.
- 出力確率の集合;$B = \{b_{ij}(k)\}$; $b_i(k)$ は状態 s_i から においてシンボル k を出力する確率.
- 初期状態確率の集合;$\pi = \{\pi_i\}$;π_i は初期状態が s_i である確率,$\sum_j \pi_j = 1$.
- 最終状態の集合;F

出力シンボルを連続値として表す場合と, 有限個のシンボルの組合せで表現する場合があり, 以下のように分類される [3] .

3.1.1 連続 HMM

出現するスペクトルパターンを連続値として表す分布モデルである. 出現確率を表す方法としては単一ガウス分布や混合ガウス分布が用いられる. パラメータの自由度を減らすために無相関ガウス分布を用いることが多い.

出現確率 $b_ij(o_t)$ が混合ガウス分布に従う場合は,

- M_{ij}...状態 i から状態 j の遷移における混合数
- C_{ijm}...状態 i から状態 j の遷移における混合数のときの重み
- $\mathcal{N}(;\mu,\Sigma)$...平均ベクトルμ, 共分散行列Σをもつ混合ガウス分布

とすると, 以下のように計算される.

$$b_{ij}(o_t) = \sum_{m=1}^{M_{ij}} C_{ijm}\mathcal{N}(o_t;\mu_{ijm},\Sigma_{ijm}) \tag{13}$$

$\mathcal{N}(;\mu,\Sigma)$ は

- n...観測行列の次元数
- $(O-\mu)^t$...$(O-\mu)$ の天地行列
- $\mid \Sigma \mid$ ...Σの固有値
- Σ^{-1}...Σの逆行列

とすると, 以下の式で表現される.

$$\mathcal{N}(O;\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^n \mid \Sigma \mid}} \exp(-\frac{1}{2}(O-\mu)^t\Sigma^{-1}(O-\mu)) \tag{14}$$

3.1.2 離散 HMM

出現するスペクトルパターンを有限個のシンボルの組合せで表す分布モデルである. スペクトルパターンのベクトル量子化によって, 符号ベクトルを生成し, 各符号ベクトルの出現確率の組合せによって出現確率を表す.

3.1.3 半連続型 HMM

半連続型 HMM は離散 HMM の出力確率値に分布を与えた HMM である. 半連続分布は, 離散 HMM の符号張の 1 つずつのベクトルに分布を与えたもので, 連続密度符号張 (continuous density codebook) とも呼ばれている. そこでは, 出力確率を連続密度符号張の分布の混合で表す. 符号張のなかの分布数を M とすると,

$$b_{ij}(x) = \sum_{m=1}^{M} \lambda_{ijm} b_{ijm}(x) \tag{15}$$

と混合正規分布で表す. ただし,

$$\sum_{m=1}^{M} \lambda_{ijm} = 1 \tag{16}$$

である. 平均値と共分散はすべての出力確率で同一であり, 遷移 $s_i \to s_j$ での分布の重み λ_{ijm} のみが変わる [4].

3.2 HMM の例

音声認識に用いられる HMM は,left-to-right モデルと呼ばれるものである.left-to-right モデルの例を図 1 に示す.

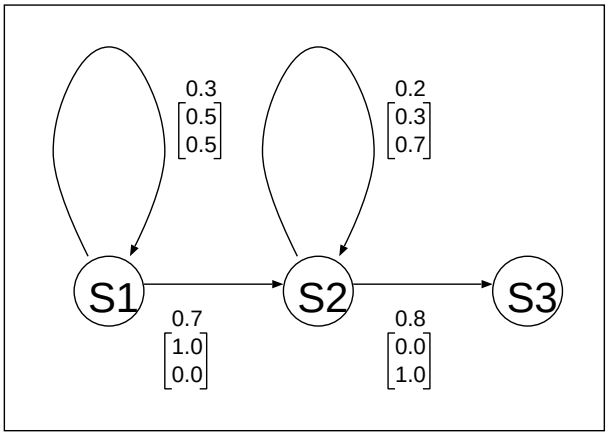

図 1: left-to-right モデルの例

例の HMM は 3 状態で構成され, 出力は有限個のシンボル a と b の 2 種類である. 最終状態を s_3 とし, 初期状態確率の集合 π を以下とする.

$$\pi = \begin{pmatrix} 1.0 & 0 & 0 \end{pmatrix} \tag{17}$$

状態遷移確率の集合 A は以下であり, 図では [] 上部の数字で示される.

$$A = \begin{pmatrix} 0.3 & 0.7 & 0.0 \\ 0.0 & 0.2 & 0.8 \\ 0.0 & 0.0 & 0.0 \end{pmatrix} \tag{18}$$

シンボル a の出力確率の集合 B_a は以下であり, 図では [] 内の上段の数字で示される.

$$B_a = \begin{pmatrix} 0.5 & 1.0 & 0.0 \\ 0.0 & 0.3 & 0.0 \\ 0.0 & 0.0 & 0.0 \end{pmatrix} \tag{19}$$

シンボル b の出力確率の集合 B_b は以下であり, 図では [] 内の下段の数字で示される.

$$B_b = \begin{pmatrix} 0.5 & 0.0 & 0.0 \\ 0.0 & 0.7 & 1.0 \\ 0.0 & 0.0 & 0.0 \end{pmatrix} \tag{20}$$

状態 s_1 を例にとれば, 状態 s_1 から s_2 の遷移は 0.7 の確率で行われ, 遷移の際に a を出力する確率は 1.0 であり,b を出力する確率は 0.0 である.

例の HMM の出力シンボルが”aab” である場合, 可能な状態遷移系列は $s_1s_1s_2s_3$ と $s_1s_2s_2s_3$ の 2 つで, それぞれの確率は以下のようにして求めることができる.

$$0.3 * 0.5 * 0.7 * 1.0 * 0.8 * 1.0 = 0.084 \tag{21}$$

$$0.7 * 1.0 * 0.2 * 0.3 * 0.8 * 1.0 = 0.0336 \tag{22}$$

よって, この HMM が”aab” を出力する確率は以下のようになる.

$$0.084 + 0.0336 = 0.1176 \tag{23}$$

3.3 認識アルゴリズム

一般に $P(y|M)$ の値は, 以下の trellis アルゴ リズムで求められる. 符号ベクトル y_t を出力して状態 s_i にある確率を $\alpha(i,t)$ とする.

$$\alpha(i,t) = \begin{cases} \log \pi_i & (t=0) \\ \sum_j \alpha(j,t-1) a_{ji} b_{ji}(y_t) & (t=1,2,\cdots,T) \end{cases} \tag{24}$$

これを計算して, 最後に

$$P(y|M) = \sum_{i,s_i \in F} \alpha(i,T) \tag{25}$$

を求めれば良い.

$P(y|M)$ を厳密に求めないで, モデル M が符号ベクトル系列 y を出力するときの最も可能性の高い状態系列上での出現確率を用いる Viterbi アルゴリズムと呼ばれる方法もある. 尤度は, 各遷移での確率値を対数変換しておくことで高速に求めることができる. このアルゴリズムを以下に示す.$i = 1, 2, \cdots, S$ において

$$f'(i,t) = \begin{cases} \log \pi_i & (t = 0) \\ \max_j \{f'(i, t-1) + \log a_{ji} b_{ji}(y_t)\} & (t = 1, 2, \cdots, T) \end{cases} \tag{26}$$

を計算し, 対数尤度

$$L = \max_{i,s_i \in F} f'(i,t) \tag{27}$$

を求める.

この Viterbi アルゴリズムによる利点は trellis 法に比べて以下のようなものである.

- 計算値のダイナミックレンジが小さく, アンダーフロー問題を解消できる.
- 計算量が少ない.
- 音声認識性能がほとんど変わらない.
- DP による効率のよい連続単語音声認識アルゴリズムに用意に適用できる.

このため Viterbi アルゴリズムは広く用いられている.

3.4 離散 HMM のパラメータ推定

学習用音声として,N 個の観測符号ベクトル系列 $\{y_1^{T(n)} = y_1, y_2, \cdots, y_{T(n)}\}_{n=1}^N$ が与えられたとき,

$$\prod_{n=1}^{N} P(y_1^T(n)|\pi_i, a_{ij}, b_{ij}(k)) \tag{28}$$

を最大化するパラメータセット $\{\hat{\pi}_i, a_{ij}, b_{ij}(k)\}$ は,Baum-Welch アルゴリズムによって, 次のように推定できる.

まず以下のような変数 $\beta(i,t)$,$\gamma(i,j,t)$ を定義する.

$\beta(i,t)$: 時刻 t に状態 s_i にあって, 以後符号ベクトル y_{t+1}^T を出力する確率

$\gamma(i,j,t)$: モデル M が y_1^T を出力する場合において, 時刻 t に状態 s_i から状態 s_j へ遷移し符号ベクトル y_t を出力する確率

このとき, 以下の関係が得られる.

$$\beta(i,T) = \begin{cases} 1 & s_i \in F \\ 0 & s_i \notin F \end{cases} \tag{29}$$

$$\beta(i,t) = \sum_j a_{ij} b_{ij}(y_t) \beta(j,t+1) \quad (t = T, T-1, \cdots, 1; i = 1, 2, \cdots, S) \tag{30}$$

$$\gamma(i,j,t) = \frac{\alpha(i,t-1) a_{ij} b_{ij}(y_t) \beta(j,t)}{P(y_1^t|M)} \tag{31}$$

以上を用いて, パラメータ pi_i, a_ij, $b_ij(k)$ を, 以下の再推定によって求める.

$$\hat{\pi_{ij}} = \frac{\sum_j \gamma(i,j,1)}{\sum_i \sum_j \gamma(i,j,1)} \tag{32}$$

$$\hat{a_{ij}} = \frac{\sum_{t=1}^T \alpha(i,t-1) a_{ij} b_{ij}(y_t) \beta(j,t)}{\sum_t \alpha(i,t) \beta(j,t)} = \frac{\sum_t \gamma(i,j,1)}{\sum_t \sum_j \gamma(i,j,t)} \tag{33}$$

$$\hat{b_{ij}} = \frac{\sum_{t,y_t=k} \gamma(i,j,t)}{\sum_t \gamma(i,j,t)} \tag{34}$$

実際は, すべての学習サンプルに対してこの計算を行ってから 1 回パラメータを更新するというサイクルを, 値が収束するまで繰り返す.

3.5 連続 HMM のパラメータ推定法

連続 HMM のパラメータ推定においては、初期確率 π_i と遷移確率 a_ij の推定式は離散 HMM の場合と同じである。

3.5.1 出現確率が単一 (多次元) ガウス分布で表される場合

出現確率のガウス分布 $N(\mu_{ij}, \Sigma_{ij})$ は次式のように最尤推定できる.

$$\hat{\mu_{ij}} = \frac{\sum_{t=1}^T \gamma(i,j,t) y_t}{\sum_{t=1}^T \gamma(i,j,t)} \tag{35}$$

$$\hat{\Sigma_{ij}} = \frac{\sum_{t=1}^{T} \gamma(i,j,t)(y_t - \mu_{ij})(y_t - \mu_{ij})^t}{\sum_{t=1}^{T} \gamma(i,j,t)} \tag{36}$$

離散 HMM の場合と同様に, この推定を値が収束するまで繰り返す.

3.5.2 出現確率が混合ガウス分布で表される場合

混合ガウス分布の場の出現確率は, 次のように表される (ガウス分布の数を M とする).

$$b_{ij}(y) = \sum_{m=1}^{M} \lambda_{ijm} b_{ijm}(y) \tag{37}$$

ここで

$$\sum_{m=1}^{M} \lambda_{ijm} = 1 \tag{38}$$

$$\int b_{ijm}(y)dy = 1 \tag{39}$$

である. 混合ガウス分布の出現確率は, 単一ガウス分布の場合と同様に次式で表せる.

$$\hat{\lambda_{ijm}} = \frac{\sum_{t=1}^{T} \gamma(i,j,t,m)}{\sum_{t=1}^{T} \gamma(i,j,t)} \tag{40}$$

$$\hat{\mu_{ijm}} = \frac{\sum_{t=1}^{T} \gamma(i,j,t,m) y_t}{\sum_{t=1}^{T} \gamma(i,j,t,m)} \tag{41}$$

$$\hat{\Sigma_{ijm}} = \frac{\sum_{t=1}^{T} \gamma(i,j,t,m)(y_t - \mu_{ijm})(y_t - \mu_{ijm})^t}{\sum_{t=1}^{T} \gamma(i,j,t,m)} \tag{42}$$

ただし,

$$\gamma(i,j,m,t) = \alpha(i,t-1) a_i j \lambda_{ijm} b_{ijm}(y_t) \beta(j,t) \tag{43}$$

で,m 番目の分布関数の遷移 $q_i \to q_j$ の確率 (遷移回数) を表している. これらの推定も値が収束するまで繰り返す.

3.5.3 半連続 HMM の場合

符号張の中の分布数を M として, 出現確率は次のようになる.

$$b_{ij}(y) = \sum_{m=1}^{M} \lambda_{ijm} b_{ijm}(y) \tag{44}$$

ここで

$$\sum_{m=1}^{M} \lambda_{ijm} = 1 \tag{45}$$

である. この混合分布のパラメータの内, 分布の重み λ_{ijm} は, 遷移状態 $(s_i \rightarrow s_j)$ ごとに推定する. 平均値 μ_m および 共分散 Σ_m は, すべての出現分布で共通化してあるので, これらの推定式は,

$$\hat{\lambda_{ijm}} = \frac{\sum_{t=1}^{T} \gamma(i,j,t,m)}{\sum_{t=1}^{T} \gamma(i,j,t)} \tag{46}$$

$$\hat{\mu_{ijm}} = \frac{\sum_{all(s_i \rightarrow s_j)} \sum_{t=1}^{T} \gamma(i,j,t,m) y_t}{\sum_{all(s_i \rightarrow s_j)} \sum_{t=1}^{T} \gamma(i,j,t,m)} \tag{47}$$

$$\hat{\Sigma_{ijm}} = \frac{\sum_{all(s_i \rightarrow s_j)} \sum_{t=1}^{T} \gamma(i,j,t,m)(y_t - \mu_{ijm})(y_t - \mu_{ijm})^t}{\sum_{all(s_i \rightarrow s_j)} \sum_{t=1}^{T} \gamma(i,j,t,m)} \tag{48}$$

となる.

3.6 連結学習

音声認識においては, 通常, 音響モデルとして音素のようなサブワードを単位とするモデルが用いられる. サブワードモデルを学習するためには, 大量の音声データが必要とされる. 音声データ中のサブワードの境界を人手でラベル付けすることはできるが, 人手で行う方法では得られるデータの量はとても限られている. このため学習において連結学習という方法が用いられる. 連結学習ではラベル付けされていない大規模なデータベースを扱うことができる. しかし, 各音声データの発話のシンボルが記述されたテキストが必要とされる. まず, 各サブワードモデルを音声データの発話のシンボルが記述されたテキストを基に連結する. このとき, 前のモデルの最終状態が次のモデルの初期状態になる. 次に,Baum-Welch アルゴリズムによって, 音声データから連結されたモデルのパラメータの推定を行う. 連結学習では, 初期モデルが重要であり, 通常は, ラベル付けされた音声データを用いて初期モデルを作成する.

連結学習の例を図 2 に示す. 音声データの音素表記 “pau a i pau” を元にして各音素 HMM を連結し, 連結した HMM のパラメータを音声データから推定する.

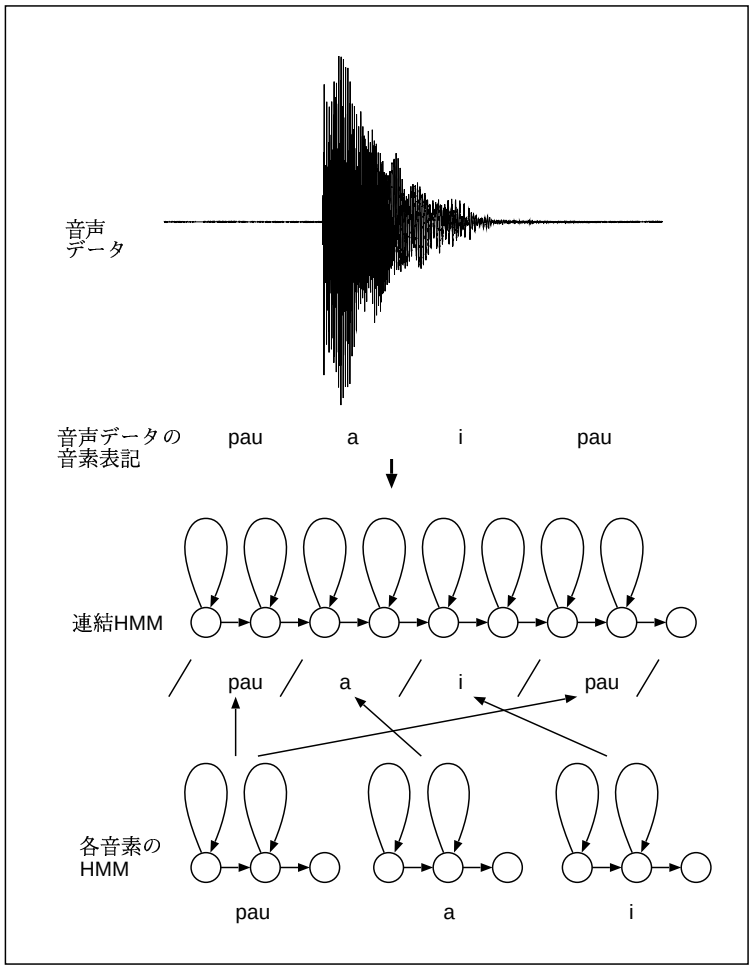

図 2: 連結学習の例

4　アクセントとモーラ情報

4.1　アクセント

アクセントは他の単語との区別を明確にするのに用いられ, 英語においては強弱で, 日本語においては高さで表現される. 日本語の単語のアクセントは, 日本語での仮名文字単位に相当するモーラごとに高低の 2 レベルが与えられる. そして、アクセントのあるモーラの直後にレベルが高から低に移る. これをアクセント核と呼ぶ.k モーラ目に核が存在するアクセント型を k 型と呼び, 核のないものを 0 型と呼ぶ. アクセント型の例を図 3 に示す.

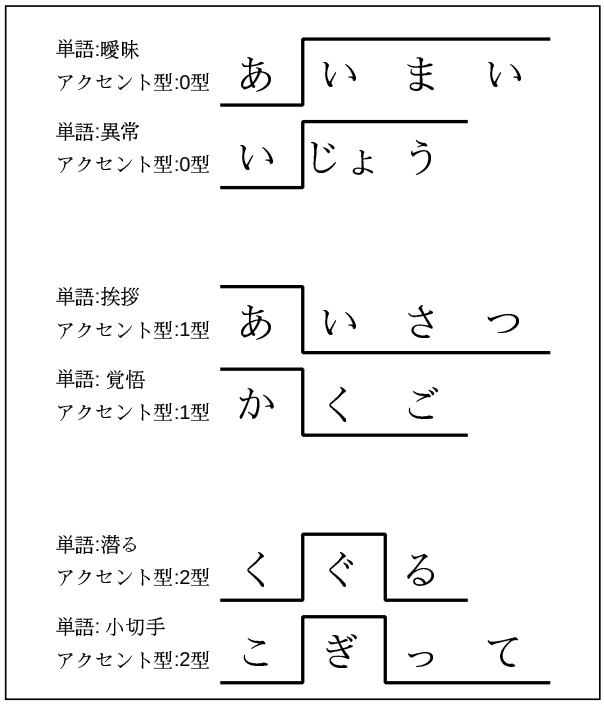

図 3: アクセント型の例

4.2 モーラ情報

モーラとは, 日本語での仮名文字単位に相当し, 和歌においての 5,7,5 の数え方をするときの単位である. 伸ばす音の長母音「ー」, 詰まる音の促音「ッ」, 跳ねる音の撥音「ン」なども 1 モーラに当たる. モーラ数は単語のモーラの総数を示し, モーラ位置は単語でのモーラの位置を示す. 本研究では, モーラ数とモーラ位置をあわせたものをモーラ情報と定義する. 単語におけるモーラ情報の例を表 1,2,3 に示す.

表 1: 単語「参加」におけるモーラ情報

単語	さ	ん	か
音素表記	sa	N	ka
モーラ数		3	
モーラ位置	1	2	3

表 2: 単語「国会」におけるモーラ情報

単語	こ	っ	か	い
音素表記	ko	q	ka	i
モーラ数		4		
モーラ位置	1	2	3	4

表 3: 単語「ジュース」におけるモーラ情報

単語	ジュ	ー	ス
音素表記	ju	u	su
モーラ数		3	
モーラ位置	1	2	3

5　アクセントを用いた単語音声認識

本研究では同音異義を認識するために, 音素 HMM に単語のアクセント型と各モーラ位置のアクセントの高低の情報を加えたモデル (以下, アクセントモデル) を提案する. そしてアクセントモデルの有効性検証のために, 単語音声認識実験を行い, 同音異義語の認識精度を調査する. また, 従来の音声認識で用いられている特徴パラメータの MFCC は音韻情報しか含んでいないため, 同音異義語認識において精度が低いことが予想されるので, 韻律情報を含む特徴パラメータの FBANK を用いて, 同音異義語の認識精度を調査する. なお, 通常の音素ラベルを用いて学習した音素 HMM を基本モデルとする.

5.1　アクセントを用いた音素ラベルの分類

ラベルファイルには母音, 撥音, 促音にモーラ数およびモーラ位置およびアクセント位置の情報を付け加える. 具体的には母音, 撥音, 促音の音素の後ろの 2 桁の数字で単語のモーラ数を表す. さらにその後ろ 2 桁の数字でモーラ位置を表す. そしてその後ろ 2 桁の数字で単語のアクセントの型を表す. 最後に付け加えられている数字 1 桁はそのモーラ位置でのアクセントの高低を示し,0 か 1 である.0 は低,1 は高であることを表す. ラベル表記の例を表 4 に示す. 例における 1 モーラ目の音素表記の数字の説明を表 5 に示す.

表 4: 音素ラベルの分類例
単語:間 (音素列 aida アクセント辞書表記 011)

基本モデル	a	i	d	a
アクセントモデル	a0301000	i0302001	d	a0303001

表 5: アクセントモデル音素表記

a	03	01	00	1
	単語のモーラ数	単語のモーラ位置	アクセント型	アクセントの高低

5.2 音素 HMM の作成

HMM は初期モデルが重要であるため, アクセントモデルの初期モデルは基本モデルから作成する. また, パラメータを同一にして実験を行うために, 半連続型 HMM を使用する [10].

実験手順を図 4 に示す. 初めに基本モデルの初期モデルとして連続型 HMM を作成する (図中 a). 次に半連続型の基本モデルを作成する (図中 b). 次に作成された基本モデルの HMM を複製してアクセントモデルの初期モデルとする (図中 c). 最後に連結学習を行いアクセントモデルの HMM を作成する (図中 d).

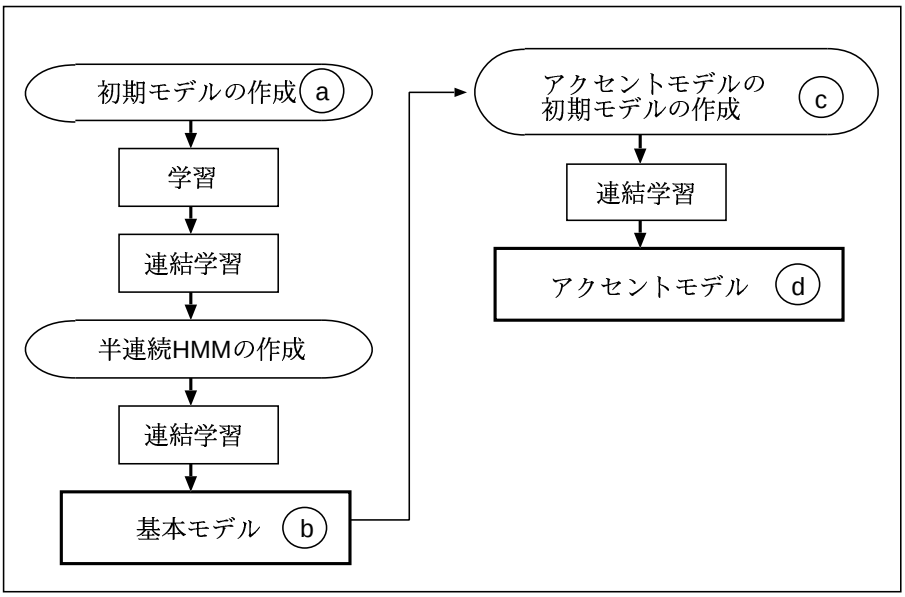

図 4: 音素 HMM の作成手順

5.3 学習データと評価データ

本研究でアクセントモデルの有効性検証のために使用する音声データのデータベースには,ATR 単語発話データベース Aset を用いる. なお, データベースには, 男性話者 10 名と女性話者 10 名が発話した単語の音声データが収録されていて, 各話者毎に 5240 単語の音声データが含まれている. また, 各音声データには, 人手によって付与された音素境界位置情報が与えられる.

本研究の実験は, 男性話者 3 名と女性話者 3 名で行う. なお, データベースの奇数番を学

習データに, 偶数番を評価データに用いる. 評価データ中には 31 組の同音異義語が存在する. 評価データ中の同音異義語を表 7 に示す. 表中のデータ番号はデータベースにおいて付けられているデータの番号を示す. また, 表 7 , 8 の括弧内の数字の 0 はアクセントの低,1 は高を意味する. なお, アクセントが異なる同音異義語は 11 組である. 実験で用いられる同音異義語を表 8 に示す. 評価データ 2620 単語を音声認識し, その中の同音異義語に注目する. 実験に用いたデータベースの詳細を表 6 に示す.

表 6: 実験に用いたデータベース

データベース	ATR 単語発話データベース Aset　5240 単語/話者
話者	6 話者 (男性 3 話者 (mau,mmy,mnm), 女性 3 話者 (faf,ftk,fms))
学習データ	奇数番号 2620 単語/話者
評価データ	奇数番号 2620 単語/話者
	(11 組のアクセントの異なる同音異義語が存在)

なお, 表 7 中の実験に用いる 6 話者分の単語のアクセントを人手で聴取したが, データ番号 10762 と 14882 の単語でアクセント辞典から決定したアクセントと異なることを確認した. 聴取結果を表 9 に示す. 聴取結果より他のデータにもアクセントの誤りがあると考えられるが, 数が多いためにアクセントの訂正は行っていない. 一方, 実験に用いる表 8 中の単語のアクセントは人手による聴取結果と一致することを確認した.

表 7: 認識データ中の同音異義語の対

	データ番号		データ番号	
1.	10150	ある (10)	10152	有る (10)
2.	10192	息 (10)	10194	意気 (10)
3.	10322	居る (01)	10324	射る (10)
4.	10558	置く (01)	10560	億 (10)
5.	10666	折る (10)	10668	織る (10)
6.	10734	代える (011)	10736	返る (100)
7.	10760	書く (10)	10762	角 (10)
8.	10788	欠ける (011)	10790	駆ける (010)
9.	11042	器械 (010)	11044	機械 (010)
10.	11056	利く (01)	11058	菊 (01)
11.	11062	起源 (100)	11064	機嫌 (011)
12.	11520	公演 (0111)	11522	講演 (0111)
13.	11524	公開 (0111)	11526	航海 (1000)
14.	11564	公正 (0111)	11566	構成 (0111)
15.	11830	咲く (01)	11832	柵 (01)
16.	12118	氏名 (100)	12120	指名 (011)
17.	12616	住む (10)	12618	澄む (10)
18.	12642	背 (10)	12644	性 (10)
19.	12732	千 (10)	12734	線 (10)
20.	13020	度 (01)	13022	足袋 (10)
21.	13270	付ける (010)	13272	漬ける (011)
22.	13486	解く (10)	13488	徳 (01)
23.	13858	刃 (1)	13860	歯 (1)
24.	13890	吐く (10)	13892	掃く (10)
25.	13960	放す (010)	13962	離す (010)
26.	14216	拭く (01)	14218	服 (01)
27.	14520	巻く (01)	14522	幕 (01)
28.	14880	焼く (01)	14882	約 (01)
29.	15070	因る (01)	15072	夜 (10)
30.	15142	礼 (10)	15144	零 (10)
31.	15210	沸く (01)	15212	枠 (01)

5.4 実験条件

評価実験は, 男性話者 3 名と女性話者 3 名で行う. 実験には単語音声認識ツールの HTK [11] を使用する.HMM の共分散行列には Diagonal-covariance(以下, 省略形 Diagonal) と Full-covariance(以下, 省略形 Full) の 2 種類を使用する. その他の実験条件は表 10 に示す.stream 数は 3 に設定し ,MFCC を用いた実験では MFCC, Δ MFCC, 対数パワーと Δ 対数パワーを,FBANK を用いた実験では FBANK, Δ FBANK, 対数パワーと Δ 対数パワー

表 8: 認識データ中のアクセントの異なる同音異義語の対

1.	居る (01)	射る (10)
2.	代える (011)	返る (100)
3.	欠ける (011)	駆ける (010)
4.	機嫌 (011)	起源 (100)
5.	公開 (0111)	航海 (1000)
6.	置く (01)	億 (10)
7.	指名 (011)	氏名 (100)
8.	度 (01)	足袋 (10)
9.	徳 (01)	解く (10)
10.	付ける (010)	漬ける (011)
11.	因る (01)	夜 (10)

表 9: アクセントの聴取による評価
○:アクセント辞典と聴取結果が同一と判断
△:判断がつかないと判断
×:アクセント辞典と聴取結果が異なると判断

番号	mau	mmy	mnm	faf	fms	ftk
10762	△	×	×	×	×	△
14882	×	△	×	×	×	×
その他	○	○	○	○	○	○

をそれぞれ別の多次元ガウス分布で表現する. 実験条件は MFCC と FBANK で同一になるように混合分布数を決定している. なお, 特徴パラメータの次数は同一にするのが困難であるので同じではない.

Full-covariance の実験でのパラメータの再推定において, データ不足により作成できない音素 HMM が存在する場合, 混合分布数が MFCC 4, Δ MFCC 4, 対数パワー, Δ 対数パワー 2 で作成できない音素 HMM は MFCC 2, Δ MFCC 2, 対数パワー, Δ 対数パワー 2 にする. 混合分布数が MFCC 2, Δ MFCC 2, 対数パワー, Δ 対数パワー 2 で作成できない音素 HMM は MFCC 1, Δ MFCC 1, 対数パワー, Δ 対数パワー 1 にする. 混合数を MFCC 1, Δ MFCC 1, 対数パワー, Δ 対数パワー 1 にしても作成できない音素 HMM は実験には用いない.FBANK も同様にして作成できない音素の混合分布数を減らしていく.

表 10: 実験条件

基本周波数	16kHz
分析窓	Hamming 窓
分析窓長	25ms
フレーム周期	10ms
音響モデル	3 ループ 4 状態 半連続分布型
stream 数	3
MFCC	
特徴パラメータ	MFCC(12 次)+Δ MFCC(12 次) +対数パワー+Δ 対数パワー (計 26 次)
連続型 HMM の初期モデルの混合分布数	(母音・撥音・無音)MFCC 4, Δ MFCC 4 , 対数パワー, Δ 対数パワー 2 (その他の音素)MFCC 2, Δ MFCC 2, 対数パワー, Δ 対数パワー 2
半連続型 HMM の混合分布数	MFCC 128, Δ MFCC 128, 対数パワー, Δ 対数パワー 16
FBANK	
特徴パラメータ	FBANK(24 次)+Δ FBANK(24 次) +対数パワー+Δ 対数パワー (計 50 次)
連続型 HMM の初期モデルの混合分布数	(母音・撥音・無音)FBANK 4, Δ FBANK 4 , 対数パワー, Δ 対数パワー 2 (その他の音素)FBANK 2, Δ FBANK 2, 対数パワー, Δ 対数パワー 2
半連続型 HMM の混合分布数	FBANK 128, Δ FBANK 128, 対数パワー, Δ 対数パワー 16

6　実験結果

6.1　同音異義語の認識精度

Diagonal-covariance で行った同音異義語認識の実験結果を表 11 に,Full-covariance で行った実験結果を表 12 に示す. 表中の括弧内の分母は評価データ中の同音異義語の数である. そして, 分子の数字は誤認識した同音異義語を示す.

表 11: Diagonal を用いた同音異義語の誤り率

話者	MFCC	FBANK
mau	18%(4/22)	5%(1/22)
mmy	14%(3/22)	18%(4/22)
mnm	14%(3/22)	9%(2/22)
faf	0%(0/22)	0%(0/22)
fms	14%(3/22)	18%(4/22)
ftk	5%(1/22)	5%(1/22)
平均	11%(14/132)	9%(12/132)

表 12: Full を用いた同音異義語の誤り率

話者	MFCC	FBANK
mau	9%(2/22)	5%(1/22)
mmy	9%(2/22)	0%(0/22)
mnm	5%(1/22)	9%(2/22)
faf	0%(0/22)	0%(0/22)
fms	14%(3/22)	5%(1/22)
ftk	9%(2/22)	0%(0/22)
平均	8% (10/132)	3%(4/132)

実験の結果,Diagonal-covariance,Full-covariance 共通して FBANK を用いた特徴パラメータの方が MFCC よりアクセントの認識精度が高かった. また比較的多くの同音異義語を誤認識していた話者 mmy,fms の認識率が FBANK の Full-covariance では,Diagonal-covariance と比較して改善されていた. 最も同音異義語を認識できた実験では平均 97%の精度が得られた.

6.2 単語音声認識精度

基本モデルとアクセントモデルの単語音声認識の実験結果を表 13 〜 16 に示す. 表中の括弧内の分母は 6 話者の評価データ数である. なお,Full-covariance の FBANK の実験結果では, 学習データの不足で作成されなかった音素が存在するので分母が異なっている. 括弧内の分子の数字は誤認識した単語数を示す. なお, アクセントモデルにおいて同音異義語に誤認識している認識結果は正解として集計している. また, 付録に話者 mau,faf の各実験条件での誤認識の出力結果を示す.

表 13: Diagonal の MFCC での実験の認識結果, 誤り率

話者	基本モデル	アクセントモデル
mau	6.76%(177/2620)	3.85%(101/2620)
mmy	7.21%(189/2620)	4.58%(120/2620)
mnm	8.13%(213/2620)	4.16%(109/2620)
faf	7.33%(192/2620)	3.78%(99/2620)
fms	7.06%(185/2620)	5.23%(137/2620)
ftk	6.82%(179/2620)	4.16%(109/2620)
平均	7.22%(1135/15720)	4.29%(675/15720)

表 14: Diagonal の FBANK での実験の認識結果, 誤り率

話者	基本モデル	アクセントモデル
mau	10.31%(270/2620)	7.02%(184/2620)
mmy	12.29%(322/2620)	7.10%(186/2620)
mnm	10.34%(271/2620)	7.33%(192/2620)
faf	8.70%(228/2620)	6.37%(167/2620)
fms	11.45%(300/2620)	8.21%(215/2620)
ftk	9.77%(256/2620)	7.18%(188/2620)
平均	10.48%(1647/15720)	7.20%(1132/15720)

表 15: Full の MFCC での実験の認識結果, 誤り率

話者	基本モデル	アクセントモデル
mau	4.69%(123/2620)	3.21%(84/2620)
mmy	6.18%(162/2620)	3.74%(98/2620)
mnm	5.46%(143/2620)	3.40%(89/2620)
faf	4.69%(123/2620)	3.05%(80/2620)
fms	5.50%(144/2620)	3.51%(92/2620)
ftk	4.85%(127/2620)	3.40%(89/2620)
平均	5.23%(822/15720)	3.38%(532/15720)

表 16: Full の FBANK での認識結果, 誤り率

話者	基本モデル	アクセントモデル
mau	5.21%(136/2611)	2.94%(77/2611)
mmy	6.09%(159/2611)	3.03%(79/2611)
mnm	5.48%(143/2611)	3.14%(82/2611)
faf	4.79%(125/2611)	3.33%(87/2611)
fms	5.52%(144/2611)	3.87%(101/2611)
ftk	5.78%(151/2611)	3.41%(89/2611)
平均	5.48%(858/15666)	3.29%(515/15666)

アクセントモデルの単語音声認識精度は, 基本モデルより高かった. 最も単語音声認識精度が高かったのは,FBANK の Full-covariance でのアクセントモデルの実験で 6 話者平均 96.71% の精度が得られた. 一方, 同条件の基本モデルは平均 94.52%の精度であった.

実験結果よりアクセントモデルは単語音声認識に対しても効果があることを確認した.

7　考察

7.1　同音異義語の認識結果に対する考察

同音異義語の誤認識結果を表 17 に示す. 表中のデータ番号はデータベースにおいて付けられているデータの番号を示す. そして, データ番号に*の付いている認識結果は単語を同音異義語ではない別の単語に誤認識していて,*の付いていない認識結果は単語を同音異義語に誤認識している. また, 括弧内の数字の 0 はアクセントの低,1 は高を意味する.

表 17: 同音異義語の誤認識結果

実験条件	話者	データ番号	認識結果	正解
Diagonal MFCC	mau	10558*	徳 (01)	置く (01)
		10560	置く (01)	億 (10)
		11064	起源 (100)	機嫌 (011)
		13272	付ける (010)	漬ける (011)
	mmy	10558*	徳 (01)	置く (01)
		10736*	堪える (010)	返る (100)
		15072	因る (01)	夜 (10)
	mnm	11064*	勤勉 (0111)	機嫌 (011)
		13272	付ける (010)	漬ける (011)
		15072*	酔う (10)	夜 (10)
	faf			
	fms	10560	置く (01)	億 (10)
		10788	駆ける (010)	欠ける (011)
		13272	付ける (010)	漬ける (011)
	ftk	10560*	解く (10)	億 (10)
Diagonal FBANK	mau	10560	置く (01)	億 (10)
	mmy	10324	居る (01)	射る (10)
		10734*	生える (010)	代える (011)
		11064*	悲劇 (100)	機嫌 (011)
		15072	因る (01)	夜 (10)
	mnm	11064	起源 (100)	機嫌 (011)
		13272	付ける (010)	漬ける (011)
	faf			
	fms	10560	置く (01)	億 (10)
		10734	返る (100)	代える (011)
		10790	欠ける (011)	駆ける (010)
		13272	付ける (010)	漬ける (011)
	ftk	13272	付ける (010)	漬ける (011)
Full MFCC	mau	10560	置く (01)	億 (10)
		13272	付ける (010)	漬ける (011)
	mmy	10324	居る (01)	射る (10)
		12120	氏名 (100)	指名 (011)
	mnm	13272	付ける (010)	漬ける (011)
	faf			
	fms	10560	置く (01)	億 (10)
		10788	駆ける (010)	欠ける (011)
		13272	付ける (010)	漬ける (011)
	ftk	10560	置く (01)	億 (10)
		13272	付ける (010)	漬ける (011)
Full FBANK	mau	10560	置く (01)	億 (10)
	mmy			
	mnm	10734	返る (100)	代える (011)
		13272	付ける (010)	漬ける (011)
	faf			
	fms	13272	付ける (010)	漬ける (011)
	ftk			

7.1.1 同音異義語の誤認識

全ての実験条件において, 同音異義語の誤認識としては, モーラ数 2 の高低のアクセントの同音異義語と, モーラ数 3 の低高高のアクセントの同音異義語を別の同音異義語に誤認識する例が多かった.

高低のアクセントの同音異義語の誤認識結果を表 18 に, 低高高のアクセントの同音異義語の誤認識結果を表 19 に示す. なお, 表 18 , 19 , 20 の括弧内の数字の 0 はアクセントの低,1 は高を意味する.

表 18: 高低のアクセントの同音異義語の誤認識結果

認識結果	正解
置く (01)	億 (10)
因る (01)	夜 (10)
酔う (10)	夜 (10)
解く (10)	億 (10)
居る (01)	射る (10)

表 19: 低高高のアクセントの同音異義語の誤認識結果

認識結果	正解
起源 (100)	機嫌 (011)
付ける (010)	漬ける (011)
勤勉 (0111)	機嫌 (011)
駆ける (010)	欠ける (011)
生える (010)	代える (011)
悲劇 (100)	機嫌 (011)
返る (100)	代える (011)
氏名 (100)	指名 (011)

7.1.2 単語の誤認識

単語を同音異義語ではない別の単語に誤認識した認識結果は Diagonal-covariance の MFCC を用いた実験で多く見られたが,Full-covariance の実験では見られなかった. なお, 誤認識結果を表 20 に示す.

表 20: 単語を同音異義語ではない単語とした誤認識結果

認識結果	正解
徳 (01)	置く (01)
堪える (010)	返る (100)
勤勉 (0111)	機嫌 (011)
酔う (10)	夜 (10)
解く (10)	億 (10)
生える (010)	代える (011)
悲劇 (100)	機嫌 (011)

7.1.3 まとめ

Full-covariance の FBANK を用いた実験では,mmy・faf・ftk の 3 話者に同音異義語を誤認識した結果は見られなかった. また, 同音異義語の認識率では 6 話者平均で 97%の精度が得られた. 同音異義語認識に対する高い精度の実験結果より, 同音異義語認識において本研究条件でのこれ以上の改良は困難であると考えられる.

7.2 単語音声認識に対する考察

アクセントモデルの認識結果は, 基本モデルの認識結果と比較すると, 特に連続母音が改善されている. アクセントモデルでの誤認識の改善はアクセントとモーラ情報によって連続母音を明確に別の音素として区別できるようになったためだと考えられる. 表 21 にアクセントモデルで基本モデルから改善された単語の例を示す. 表 21 , 表 22 の括弧内の数字の 0 はアクセントの低,1 は高を意味する.

アクセントモデルにおいて基本モデルから改善されなかった単語の認識結果を表 22 に示す. 改善されなかった認識結果には子音の間違いが多く見られる. 子音の誤りが改善されなかったのは, アクセントモデルにおける子音の分類は, 基本モデルの子音の分類と同じであるためだと考えられる.

表 21: 改善された単語例

基本モデルでの認識結果		正解	
単語	音素列	単語	音素列
木 (1)	k i	良い (10)	ii
解除 (100)	k a i j o	海上 (0111)	k a i j o u
根拠 (100)	k o N ky o	公共 (0111)	k o u ky o u
知れる (011)	sh i r e r u	仕入れる (0110)	sh i i r e r u
付属 (011)	f u z o k u	風俗 (1000)	f u u z o k u
割る (01)	w a r u	笑う (011)	w a r a u

表 22: 改善されなかった単語例

認識結果		正解	
単語	音素列	単語	音素列
自宅 (011)	j i t a k u	自覚 (011)	j i k a k u
スタート (0100)	s u t a a t o	スカート (0100)	s u k a a t o
伝える (0111)	ts u t a e r u	支える (0111)	ts u k a e r u
公共 (0111)	k o u ky o u	投票 (0111)	t o u hy o u
伝統 (0111)	d e N t o u	電報 (0111)	d e N p o u
掃く (10)	h a k u	百 (01)	hy a k u

8　おわりに

本研究では, 従来の単語音声認識において, あまり行われて来なかった同音異義語認識の研究を行った. 同音異義語を音声認識するために, 音素 HMM に単語のアクセント型の情報と各モーラ位置でのアクセントの高低の情報を加えたアクセントモデルを提案し, 単語音声認識実験を行った. そして, 評価データ中に含まれるアクセントの異なる同音異義語に注目した. また, 特徴パラメータに一般的に使われている MFCC は音韻的特徴しか含んでいないため, アクセントを用いた実験において認識精度が低いと予測され, 韻律的特徴を含む FBANK を用いて実験を行った.

実験の結果, 特徴パラメータに FBANK を用いた Full-covariance の実験において, 平均 97%の同音異義語を認識できことにより, アクセントモデルの同音異義語に対する音声認識の有効性を確認した. また, アクセントモデルの単語音声認識精度は, 通常の音素ラベルを用いて学習した音素 HMM を用いた基本モデルに比べて高いこにより, アクセントモデルの単語音声認識に対する有効性を確認した. そして,FBANK を用いた特徴パラメータでは MFCC に比べてアクセントモデルの同音異義語の認識精度が高いことにより,FBANK を用いた特徴パラメータの同音異義語認識に対する有効性を確認した.

本研究条件での同音異義語に対するこれ以上の改良は困難であると考えられるので, 今後の課題としては, 不特定話者認識や雑音環境におけるアクセントモデルの有効性について検証する必要があると考えられる.

謝辞

最後に, 一年間に渡って御指導, 御教授して頂きました鳥取大学工学部知能情報工学科計算機 C 研究室の池原教授と村上助教授に深くお礼申し上げます. また, 論文を執筆にあたり, 助言を頂いた徳久助手にお礼を申し上げます. 加えて, 本稿を執筆するにあたり参考にさせて頂いた論文, 本の著者の方々の皆様にもお礼申し上げます.

参考文献

[1] 中川 聖一:“確率モデルによる音声認識”, 社団法人 電子情報通信学会,(1988)

[2] 鹿野 清宏, 伊藤 克亘, 河原 達也, 武田 一哉, 山本 幹雄:“音声認識システム”, オーム社,(2001)

[3] 古井 貞熙:“音声情報処理”, 森北出版株式会社,(1998)

[4] 鹿野 清宏, 中村 哲, 伊勢 史朗:“音声・音情報のディジタル信号処理”, 株式会社 昭昇堂,(1997)

[5] 高橋 敏, 松永 昭一, 嵯峨山 茂樹:“ピッチパタン情報を用いた単語音声認識”, 日本音響学会講演論文集,1-3-20,pp.39-40(1990-3)

[6] 村上 仁一, 荒木 哲郎, 池原 悟:“音声におけるポーズ長およびアクセント位置の情報量の考察”, 日本音響学会講演論文集,3-3-11,pp.89-90(1988-3)

[7] “NHK 日本語発話アクセント辞典新版”,NHK 出版,ISBN4-14-011112-7(1998)

[8] 妹尾 貴宏, 村上 仁一, 池原 悟:“モーラ情報を用いた単語音声認識”, 日本音響学会講演論文集,1-4-8,pp.15-16(2003-3)

[9] 谷口 勝則, 村上 仁一, 池原 悟:“FBANK を用いた孤立単語音声認識”, 日本音響学会講演論文集,3-Q-3,pp.157-158(2003-3)

[10] X.D.Huang,Y.Ariki,M.A.Jack: “Hidden Markov Models for Speech Recognition”

[11] “HTK Ver3.2 reference manual”,2002 Cambridge Universit

付録

1. 実行シェルスクリプト
   (a) stage2(特徴ベクトルの出力)
   (b) stage2_5(基本モデルの初期モデルの作成)
   (c) stage3(基本モデルの初期学習)
   (d) stage5(基本モデルの再推定)
   (e) stage7(基本モデルの連結学習)
   (f) stage9(基本モデルの連続型 HMM の作成)
   (g) stage12(基本モデルの連結学習)
   (h) stage12_5(基本モデルの辞書, ネットワークグラマーの作成)
   (i) stage13c(基本モデルの認識)
   (j) stage9mac(基本モデルからアクセントモデルへの音素のコピー)
   (k) stage12mac(アクセントモデルの連結学習)
   (l) stage12_5mac(アクセントモデルの辞書, ネットワークグラマーの作成)
   (m) stage13mac (アクセントモデルの認識)
2. HTK コンフィグファイル
   (a) MFCC で用いた HTK コンフィグファイル-MFCC_conf
   (b) FBANK で用いた HTK コンフィグファイル-FBANK_conf
3. 実行シェルスクリプト中のプログラム
   (a) del_onso.rb(作成されなかった音素を除いた音素リストの作成)
   (b) del_telist.rb(作成されなかった音素が含まれるデータを除いた評価データリストの作成)
   (c) del_trainlist.rb(作成されなかった音素が含まれるデータを除いた学習データリストの作成)
   (d) make_MAC_lab.rb(アクセントモデルのラベル作成)

(e) make_Mbase.rb(基本モデルからアクセントモデルへの音素のコピー)

(f) make_bcplist.rb(音素リストの作成)

(g) make_dic.rb(辞書の作成)

(h) make_net.rb(ネットワークグラマーの作成)

(i) make_telist.rb(評価データリストの作成)

(j) make_trainlist.rb(認識データリストの作成)

(k) onso_error.rb(サブプログラム)

(l) onso_search.rb(サブプログラム)

4. 実験の誤認識の出力結果 (mau,faf)

(a) Diagonal,MFCC, 基本モデル,mau の出力結果-mau_dia_mfcc_normal

(b) Diagonal,MFCC, アクセントモデル,mau の出力結果-mau_dia_mfcc_mac

(c) Diagonal,MFCC, 基本モデル,faf の出力結果-faf_dia_mfcc_normal

(d) Diagonal,MFCC, アクセントモデル,faf の出力結果-faf_dia_mfcc_mac

(e) Diagonal,FBANK, 基本モデル,mau の出力結果-mau_dia_fb_normal

(f) Diagonal,FBANK, アクセントモデル,mau の出力結果-mau_dia_fb_mac

(g) Diagonal,FBANK, 基本モデル,faf の出力結果-faf_dia_fb_normal

(h) Diagonal,FBANK, アクセントモデル,faf の出力結果-faf_dia_mfcc_mac

(i) Full,MFCC, 基本モデル,mau の出力結果-mau_full_mfcc_normal

(j) Full,MFCC, アクセントモデル,mau の出力結果-mau_full_mfcc_mac

(k) Full,MFCC, 基本モデル,faf の出力結果-faf_full_mfcc_normal

(l) Full,MFCC, アクセントモデル,faf の出力結果-faf_full_mfcc_mac

(m) Full,FBANK, 基本モデル,mau の出力結果-mau_full_fb_normal

(n) Full,FBANK, アクセントモデル,mau の出力結果-mau_full_fb_mac

(o) Full,FBANK, 基本モデル,faf の出力結果-faf_full_fb_normal

(p) Full,FBANK, アクセントモデル,faf の出力結果-faf_full_fb_mac