

# 卒業論文

## HMMを用いたリスナー の印象変化の推定

5年情報工学科

東海林研究室

出席番号 27号

エドワード パック エン フィ

# 【目次】

1. 目的	- - - - -	3
2. 隠れマルコフモデルの紹介		
2.1. 隠れマルコフモデルについて	- - - - -	4
2.2. 隠れマルコフモデルの記号	- - - - -	7
3. ビタビーアルゴリズムの紹介		
3.1. ビタビーアルゴリズムについて	- - - - -	8
3.2. 復号化問題の解	- - - - -	8
3.3. ビタビーアルゴリズムのソースプログラム	- - - - -	10
4. コード進行と印象変化のモデル化		
4.1. コード進行に関する時系列データ	- - - - -	13
4.2. 印象変化に関する時系列データ	- - - - -	13
4.3. コード進行と印象変化の同時確率	- - - - -	14
4.4. HMM によるモデル化	- - - - -	14
5. 楽曲推薦システムへの応用	- - - - -	15
5.1. しきい値の選択	- - - - -	15
6. 実験	- - - - -	16
6.1. 実験 1	- - - - -	16
6.1.1. 実験の手順	- - - - -	17
6.1.2. 結果と考察	- - - - -	17
6.2. 実験 2	- - - - -	20
6.2.1. 実験の手順	- - - - -	20
6.2.2. 結果と考察	- - - - -	20
6. まとめ	- - - - -	23
7. 参考文献	- - - - -	24

# 1. 目的

最近、音楽のオンライン配信サービスなどという音楽活動が盛んになってきている。2005年の夏には、Apple社のiTunes Music Storeが登場し、ここで用意された楽曲の数は100万曲以上であり、デジタル音楽に関する話題を呼ぶ。さらに、2006年のはじめに、iTunes Music Storeは10億曲のダウンロードを発表した。この様に、オンラインエンタテインメントコンテンツを提供するビジネスはますます盛んになると予想される。

最新のアーティストの情報や曲が常に更新されているため、それらをフォローするのが困難である。楽曲のジャンルやアーティスト名など各楽曲が持つ固定的な情報だけでなく、各リスナーの楽曲に対する個人的な印象に基づいて楽曲を検索する楽曲推薦システムの構築が盛んに行われている。②

従来の研究では、テンポやビートなど、楽曲の特徴を抽出する楽曲推薦システムや、協調フィルタリングによる楽曲推薦などがある。⑤⑥

そこで、本研究の目的は楽曲の進行につれて変化するリスナーの楽曲に対する印象変化を時系列データとみなして隠れマルコフモデル (Hidden Markov Model:HMM) を用いて推定し、楽曲推薦システムへの応用を提案する。

## 2. 隠れマルコフモデルの紹介

### 2.1. 隠れマルコフモデル(HMM)について

隠れマルコフモデル (Hidden Markov Model) は、不確定な時系列のデータをモデル化するための有効な統計的手法であり、確率的な状態遷移と確率的な記号出力を備えたオートマトンである。

観測された出力記号系列から、その出力記号系列を生成した状態遷移系列を一意的に復元することができないため (言い換えれば、状態遷移系列がモデル内部に隠れていて外からは見えないため)、「隠れ (hidden)」という名前がつけられている。

英語の品詞付けを例として、隠れマルコフモデルの基本的な考え方について説明する。①”Charls talks like a parrot”には、複数の解釈が存在する。これは、各単語が複数の品詞および意味を持っているためである。最も一般的な解釈では「チャールズは話し好きな人である」という文を表しているが、「チャールズの話はオウムを好む」という奇妙な解釈もある。2番目の解釈は、“talks”を名詞(話し)、“like”を動詞(好む)としたために起こったものである。

英語には、1つの単語に対し異なった複数の品詞があるという多品詞語が多く、正しい品詞としてどれを選ぶかという処理が形態素解析においては大切である。

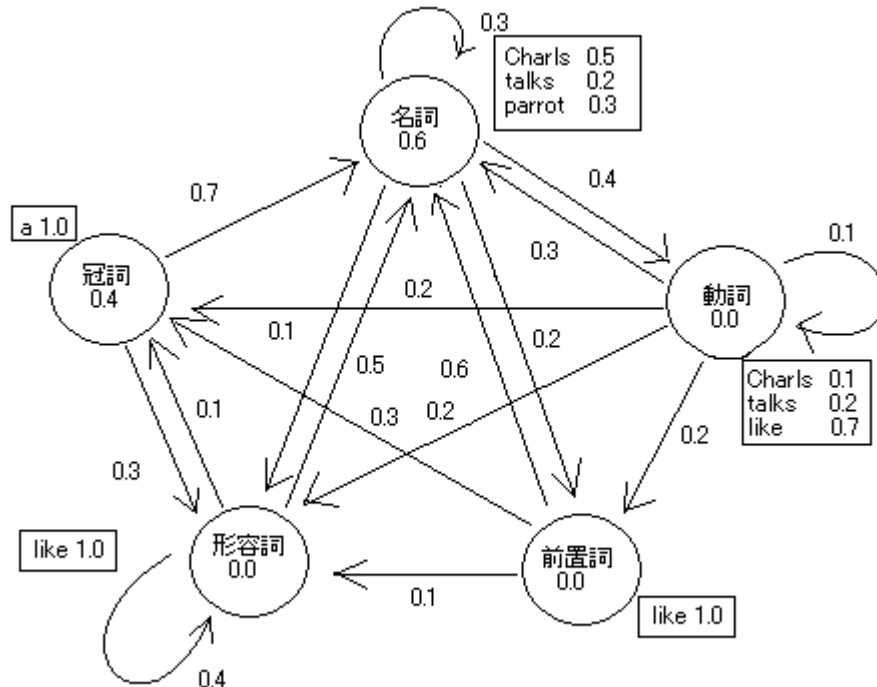


図 1

図1は、品詞を状態、単語を出力記号とする隠れマルコフモデルの例を表している。矢印は状態遷移を表しており、各状態遷移にはその遷移をたどる確率が記されている。状態内の数字はその状態が初期状態である確率を示しており、状態の横にはその状態から出力される単語と出力確率が記されている。たとえば、初期状態が「名詞」である確率は 0.6 であり、この状態からは3つの単語“Charls”, ”talks”, “parrot” が、それぞれ 0.5, 0.2, 0.3 の確率で出力されることを意味している。

ここで、名詞が初期状態である確率を  $P_i(\text{名詞})$ 、名詞の状態から”Charls”が出力される確率を  $P_o(\text{Charls}|\text{名詞})$ 、名詞の状態から動詞の状態への遷移確率を  $P_t(\text{動詞}|\text{名詞})$  等で表すことにする。このとき、「チャールズは話し好きな人である」に対応する系列「Charls/名詞, talks/動詞, like/前置詞, a/冠詞, parrot/名詞」が得られる確率  $P_1$  は、次のようにして計算することができる。

$$\begin{aligned}
P_1 &= P_i(\text{名詞})P_o(\text{Charls} \mid \text{名詞})P_t(\text{動詞} \mid \text{名詞})P_o(\text{talks} \mid \text{動詞}) \\
&\quad P_t(\text{前置詞} \mid \text{動詞})P_o(\text{like} \mid \text{前置詞})P_t(\text{冠詞} \mid \text{前置詞})P_o(\text{a} \mid \text{冠詞}) \\
&\quad P_t(\text{名詞} \mid \text{冠詞})P_o(\text{parrot} \mid \text{名詞}) \\
&= 0.6 \times 0.5 \times 0.4 \times 0.2 \times 0.2 \times 1.0 \times 0.3 \times 1.0 \times 0.7 \times 0.3 \\
&= 0.0003024
\end{aligned}$$

同様にして、「チャールズの話しはオウムを好む」に対応する系列「Charls/名詞, talks/名詞, like/動詞, a/冠詞, parrot/名詞」が得られる確率  $P_2$  は、次のように求めることができる。

$$\begin{aligned}
P_2 &= P_i(\text{名詞})P_o(\text{Charls} \mid \text{名詞})P_t(\text{名詞} \mid \text{名詞})P_o(\text{talks} \mid \text{名詞}) \\
&\quad P_t(\text{動詞} \mid \text{名詞})P_o(\text{like} \mid \text{動詞})P_t(\text{冠詞} \mid \text{動詞})P_o(\text{a} \mid \text{冠詞}) \\
&\quad P_t(\text{名詞} \mid \text{冠詞})P_o(\text{parrot} \mid \text{名詞}) \\
&= 0.6 \times 0.5 \times 0.3 \times 0.2 \times 0.4 \times 0.7 \times 0.2 \times 1.0 \times 0.7 \times 0.3 \\
&= 0.0002117
\end{aligned}$$

図1の隠れマルコフモデルからは、ほかにも“like”を形容詞とする系列など、全部で6個の状態遷移系列(単語/品詞系列)が生成される。したがって、文 “Charls talks like a parrot”の生成確率は、これら6個の系列の確率の和として求めることができる。また、これらの系列の中で最も高い確率を与えるものが、この文に対する最適な品詞付けであると解釈することができる。

## 2.2. 隠れマルコフモデルの記号

マルコフ過程の各状態において、確率的な記号の出力を考えたモデルを隠れマルコフモデル (hidden Markov model: HMM) と名前がつけられる。したがって、隠れマルコフモデルは、5項組  $M=(Q, \Sigma, A, B, \pi)$  により定義される。⑦⑧

(1)  $Q = \{q_1, \dots, q_N\}$  : 状態の有限集合

(2)  $\Sigma = \{o_1, \dots, o_M\}$  : 出力記号の有限集合

(3)  $A = \{a_{ij}\}$  : 状態遷移確率分布

$a_{ij}$  は状態  $q_i$  から状態  $q_j$  への遷移確率であり、 $\sum_j a_{ij} = 1$  を満たす

(4)  $B = \{b_i(o_t)\}$  : 記号出力確率分布

$b_i(o_t)$  は状態  $q_i$  で記号  $o_t$  を出力する確率であり、 $\sum_t b_i(o_t) = 1$  を満たす

(5)  $\pi = \{\pi_j\}$  : 初期状態確率分布

$\pi_i$  は状態  $q_i$  が初期状態である確率  $P(X_1=q_i)$  である。

上の定義では、状態から記号が出力されるが、状態遷移の際に記号が出力されるという定義を用いることもある。この場合には、記号出力分布は  $b_{ij}(o_t)$  のようになる。

## 3. Viterbi アルゴリズムの紹介

### 3.1. Viterbi アルゴリズムについて

Viterbi アルゴリズムは、隠れマルコフモデル(HMM)において与えられた出力記号系列を出力した可能性(尤度)が最も高い状態遷移系列(最尤状態列)を計算するアルゴリズムであり、動的計画法の一種である。このアルゴリズムは必ずしも正しい最尤状態遷移列を返すものではなく、ある時刻  $t$  での最尤状態遷移列は  $t$  までに観測された出力記号系列と、 $t-1$  までで最も確からしい最尤状態遷移列だけに依存する。⑨⑩⑪

### 3.2. 復号化問題の解

復号化問題は、記号系列  $\mathbf{O}$  を生成したモデル  $\mathbf{M}$  の最適な状態遷移系列を求める問題である。ここで、記号系列  $\mathbf{o}_1^T = o_1 \dots o_T$  に対する最適な状態遷移系列

とは、 $P(\mathbf{o}_1^T, \mathbf{q}_1^T | \mathbf{M})$  を最大化するような系列  $\mathbf{q}_1^T = q_1 \dots q_T$  である。

記号系列  $\mathbf{o}_1^T = o_1 \dots o_T$  を生成して、時刻  $t$  で状態  $q_i$  に到達する状態遷移系列は一般的に複数あるが、このうち最大の確率値を与えるものだけを記憶していけば、最終的に最適な状態遷移系列を求めることができる。いま、時刻  $t$  で状態  $q_i$  に到達する状態遷移系列に関して、最大の確率値を  $\delta_t(i)$  で表すことにする。

$$\delta_t(i) = \max_{\mathbf{q}_1^{t-1}} P(\mathbf{X}_t = q_i, | \mathbf{M})$$



$\delta_t(\mathbf{i})$ は、以下のように再帰的に計算することができる。

$$\delta_{t+1}(\mathbf{j}) = \max_i [\delta_t(\mathbf{i})a_{ij}]b_j(o_{t+1})$$

そして、状態遷移系列を復元するためには、最大確率値  $\delta_{t+1}(\mathbf{j})$  を与える直前の状態  $\mathbf{i}$  も同時に記憶しておく必要がある。下のアルゴリズムでは、これを  $\phi(\cdot)$  に記憶している。

## Viterbi アルゴリズム

1. 各状態  $\mathbf{i} = 1, \dots, N$  に対して、変数の初期化を行う。

$$\delta_1(\mathbf{i}) = \pi_i b_i(o_1)$$

$$\phi_1(\mathbf{i}) = 0$$

2. 各時刻  $t = 1, \dots, T-1$ , 各状態  $\mathbf{j} = 1, \dots, N$  について、再帰計算を実行。

$$\delta_{t+1}(\mathbf{j}) = \max_i [\delta_t(\mathbf{i})a_{ij}]b_j(o_{t+1})$$

$$\phi_{t+1}(\mathbf{j}) = \operatorname{argmax}_i [\delta_t(\mathbf{i})a_{ij}]$$

3. 再帰計算の終了。

$$\hat{P} = \max_i \delta_T(i)$$
$$\hat{q}_T = \operatorname{argmax}_i \delta_T(i)$$

4. バックトラックによる最適状態遷移系列の復元。  $t = T - 1, \dots, 1$  に対して、次を実行する。

$$\hat{q}_t = \phi_{t+1}(\hat{q}_{t+1})$$

### 3.3. Viterbi アルゴリズムのソースプログラム

今回の実験で用いたビタビーアルゴリズムのソースプログラムは以下の通りである。

```
// viterbi アルゴリズム
//
void viterbi(const double *pi, // 初期状態の確率
            double **a,      // 遷移確率
            double **b,      // 条件付確率
            const int N,     // 状態数
            const int *o, int *q, const int T // 時間
)
{
    double delta[T][N];
    double k, max=0, qt=0;
    int phi[T][N];
```

```

int i, j, t, maxi=0, argmax;

//delta と phi の初期化
for (i = 0; i < N; i++) {

    delta[0][i] = pi[i]*b[o[0]][i];
    phi[0][i] = 0;

}

//max の計算
for (t = 0; t < T - 1; t++) {
    for (j = 0; j < N; j++) {

        // max[delta]
        max = 0;
        for (i = 0; i < N ; i++)
        {
            k = delta[t][i]*a[i][j];
            if (k > max) { max = k; maxi = i; }

        }
        delta[t+1][j]=max*b[o[t+1]][j];
        phi[t+1][j]=maxi;
    }
}

//qT の計算

for (i=0; i<N; i++)
{
    if(delta[T-1][i]>delta[T-1][maxi])
        maxi = i;
}
q[T-1]=maxi;

```

```

//qt の計算
for (t=T-2; t>=0; t--) {

    q[t]=phi [t+1][q[t+1]];

}

for (i = 0; i <= T - 1; i++) {

    printf("t=%3d ¥n", i);

    for (j = 0; j <= N - 1; j++) {

        printf("%d ", phi[i][j]);
    }
    printf("¥n");
}
}

```

## 4. コード進行と印象変化の HMM による モデル化<sup>⑫</sup>

### 4.1 コード進行に関する時系列データ

楽曲に含まれるコードを先頭から2つずつ分割して取り出し、それらの集合を時系列データと考えると楽曲のコード進行データとする。たとえば、コードが先頭から C → F → G → C → F → D の順で現れるとき、 $\{\{C, F\}, \{G, C\}, \{F, D\}\}$  がその楽曲のコード進行データとする。

次にこのコード進行データを数値化する。本研究ではメジャーコードだけを考慮する。さらに調の違いも考慮する必要がある。そこでトニック、つまり調の基礎となるスケールの出発点にあたる音の根音と同じ根音を持つコードすべてに状態番号 0 を割り当て、トニックの根音から  $k$  だけ半音上の音を根音とするコードすべてに状態番号  $k$  が割り振られる。<sup>④</sup>たとえば上で挙げたコード進行の場合は、C メジャースケールのもとでは $\{\{0, 5\}, \{7, 0\}, \{5, 2\}\}$ となる。根音とはコードの基礎となる音である。<sup>③</sup>

最後に、時刻  $t$  ( $1 \leq t \leq T$ ) におけるコード進行を  $c_t$  とおいてコード進行データを  $CT = c_1c_2\dots c_T$  という時系列データの形式で表すことにする。たとえば上で挙げたコード進行の場合は  $C3 = c_1c_2c_3$ ,  $c_1 = \{0, 5\}$ ,  $c_2 = \{7, 0\}$ ,  $c_3 = \{5, 2\}$  となる。

### 4.2 印象変化に関する時系列データ

状態 (印象) 数として  $M$  個を選んだとき、各状態に 0 から  $M-1$  番までの数値を割り当てる。本研究では「好き」(状態番号 1)、「普通」(状態番号 0) の  $M=2$  状態とする。

ここでコード進行データと同様に、時刻  $t$ , ( $1 \leq t \leq T$ ) における印象を  $s_t$  とおいて印象変化データを  $ST = s_1s_2\dots s_T$  という時系列データの形式であらわすことにする。たとえば、リスナーがある楽曲を聴いたときに、先頭から普通 → 普通 → 好き → 普通 という印象が現れたとすると、その印象変化データは  $S4 = s_1s_2s_3s_4$ ,  $s_1=0$ ,  $s_2=0$ ,  $s_3=1$ ,  $s_4=0$  となる

### 4.3 コード進行と印象変化の同時確率

実際にリスナーの印象変化をサンプリングするときは、コード進行データ  $CT$  の状態  $c_t$  が切り替わる時点とサンプリングのタイミングを一致させる。したがって、ひとつの楽曲からコード進行データと印象変化データの時系列の集合  $\{CT, ST\}$  が得られる。 $\{CT, ST\}$  の同時確率を  $P(CT, ST)$  とする。 $ST$  が未知、 $CT$  が既知であるとき、この同時確率を最大にする  $\hat{ST}$  が  $ST$  の最優推定量となる。

### 4.4 HMMによるモデル化

$P(CT, ST)$  を最大化する  $\hat{ST}$  を求めるためにHMMを適用する。本研究では  $ST$  をHMMの未観測である内部状態、 $CT$  を実際に観測されたシンボル出力として、図1のようなHMMを構成した。ここで  $a_{ij}$  は内部状態  $i$  から  $j$  に遷移する確率、 $b_i(k)$  は内部状態  $i$  のときにシンボル  $k$  が現れる確率である。なお、時刻  $t+1$  におけるコード進行  $c_{t+1}$  の出現確率は、実際には  $s_{t+1}$  だけではなく  $c_t$  の影響も受けるが、条件数が増大することから今回は  $c_t$  と  $c_{t+1}$  は独立であるとした。

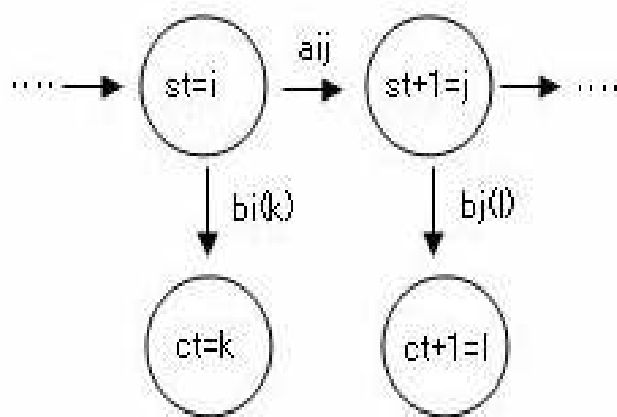


図2 提案するHMM

## 5. 楽曲推薦システムへの応用<sup>⑬⑭</sup>

ある楽曲から求めた印象変化の推定値  $\hat{S}_T$  の時間毎の平均

$$E(\hat{s}_t) = \frac{1}{T} \sum_{t=1}^T \hat{s}_t$$

を考えると、あるしきい値よりも大きい値の楽曲は好きであるコード進行が多いということを意味している。そのしきい値よりも大きい値の楽曲をユーザに推薦することによって楽曲推薦システムを構築できる。

### 5.1. しきい値の選択

今回の実験でのしきい値は学習データを使用して決める。全曲の正解率が最大となる閾値を選択する。しきい値を0から0.1ずつ増やして、正解率の一番高いものにする。

## 6. 実験

### 6.1 実験 1

今回の実験では、HMMの内部状態の推定アルゴリズムとしてviterbiアルゴリズムを用いた。リスナーがいくつかの楽曲を聴いたときの印象変化をサンプリングし、印象変化の遷移確率 $a_{ij}$ と、ある状態 (印象)  $i$  におけるのシンボル (コード進行)  $k$  の出現確率 $b_i(k)$ を推定する。

つぎに確率の推定に用いなかった楽曲のコード進行データ $CT$ から印象変化の推定値 $\hat{ST}$ を図2のHMMをもとに求めて実際の $ST$ と比較する。楽曲としてメジャースケールの童謡40曲のデータを使用した。

評価基準としては2乗誤差基準

$$r(ST, \hat{ST}) = \frac{1}{T} \sum_{t=1}^T (st - \hat{st})^2 \quad (0 \leq r \leq 1)$$

を使用する。

今回は状態数として好き (1)、普通 (0) の2通りのみを使用しているので2乗誤差基準は誤り率 ( $0 \leq r \leq 1$ ) を表している。実験に用いた楽曲のうち一曲を取り除き、他の楽曲を用いて遷移確率やコードの出現確率を推定し、取り除いた楽曲の $\hat{ST}$ を求めて2乗誤差基準を計算した。

$r = 0$ であれば、ある一曲の推定した印象変化の正解率が100%である。それに対して、 $r = 1$ であれば、ある一曲の推定した印象変化の正解率が0%である。



## 6.1.1 実験手順

まず、童謡の40曲のアンケートを6人の被験者に対して行った。採ったアンケートを.csvの拡張子を持つ入力データファイルにして、HMMによる推定を行った。推定された値は実際の最優推定量と比較した。評価基準としては誤答率を使用した。

## 6.1.2. 結果と考察

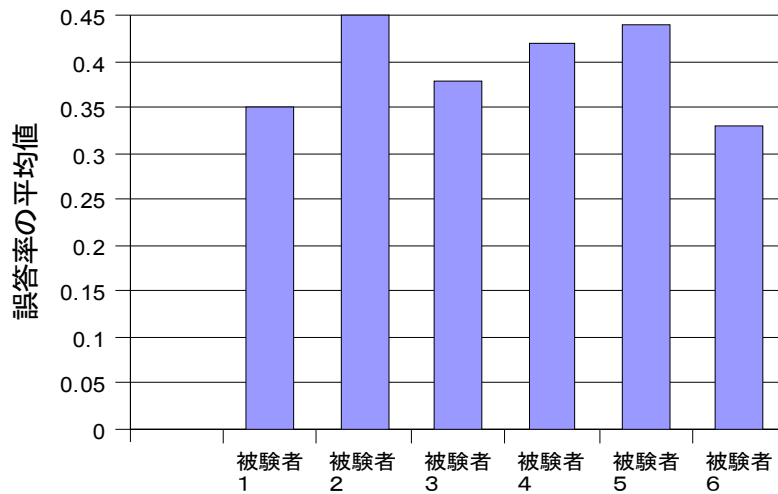
各リスナーの曲毎に誤答率の平均と標準偏差を出した結果を表1に示す。さらに信頼度95%の信頼区間を求めた。

曲の番号	被験者 1	被験者 2	被験者 3	被験者 4	被験者 5	被験者 6
0	0.3	0.5	0.5	0.1	1	0.1
1	0.5	0.58	0.25	0.17	0.5	0.25
2	0.62	0.46	0.31	0.38	0.54	0.23
3	0.2	0.6	0.5	0.1	0.3	0.3
4	0.43	0.43	0.29	0.38	0.29	0.14
6	0.25	0.5	0.63	0.3	0.63	0.5
7	0.5	0.3	0.1	0.33	0.36	0
8	0.67	0.67	0.5	0.33	0.17	0.17
9	0.5	0.42	0.5	0.67	0.58	0.25
10	0.33	0.44	0.22	0.67	0.11	0
11	0.45	0.55	0.36	0.45	0.27	0
12	0.17	0.67	0.67	0.29	0.5	0.43
13	0	0.5	0	0.25	0.5	0
14	0.8	0.2	0.4	0.6	0.4	0.6
15	0.5	0.13	0	0.63	0	0
16	0.25	0.25	0.4	0.88	0.75	0.38
17	0	0.29	0	0.29	0.57	0.29

19	0.38	0.38	0.25	0.25	0.5	0.63
20	0.7	0.5	0.14	0.3	0.6	0.2
21	0.38	0.38	0.38	0.38	0.13	0.75
23	0.43	0.43	0.57	0.57	0	0.29
24	0	0.38	0	0	0.38	0.5
25	0.33	0.67	0.5	0.5	0.5	0.5
26	0.18	0.55	0.73	0.73	0.27	0.18
28	0.7	0.3	0.5	0.5	0.8	0.3
29	0.33	0.78	0.66	0.67	0.56	0.44
30	0.33	0.17	0.33	0.33	0.5	0.67
31	0.13	0.38	0.25	0.25	0.25	0.5
32	0.22	0.44	0.33	0.33	0.78	0.11
33	0.25	0.63	0.25	0.25	0.75	0.63
34	0.55	0.55	0.82	0.82	0.27	0.09
35	0.4	0.4	0.4	0.4	0.6	0.4
36	0.25	0.5	0.5	0.5	0.63	0.5
37	0	0.5	0.67	0.67	0.33	0.5
39	0.29	0.29	0.57	0.57	0.14	0.86
誤答率の平均	0.35	0.45	0.38	0.42	0.44	0.33
標準偏差	0.21	0.15	0.22	0.21	0.24	0.23
信頼区間の幅	0.07	0.05	0.07	0.07	0.08	0.08
信頼区間の下限	0.28	0.4	0.31	0.35	0.36	0.25
信頼区間の下限	0.42	0.5	0.46	0.5	0.52	0.41

表 1

被験者6人の誤答率の平均



グラフ 1

表1から分かるように、印象変化がある程度うまく推定されていることが分かる。最悪の場合でも、一曲の中で**55%** のコード進行が正しく推定された。

もっと分かりやすくするために、被験者6人の誤答率の平均のグラフを描いた。特に被験者6の場合は、平均一曲毎に10個のコード進行があるとすれば、**6.7個**のコード進行がうまく推定されたようである。ここで、6人の誤答率の平均を採ってみると、誤答率が**39.5%**になった。すなわち、平均的に一曲の半分以上のコード進行の推定が満足的な結果が得られた。

しかし、**39.5%**といっても、かなり高い誤答率と思われるかもしれない。そこで、誤答率をさらに減らす方法を考えなければならない。たとえば、実験者5の曲番号0の誤答率を見れば、1になっていることが分かった。それを避ける方法として、長い楽曲をしようすればいいと思われる。楽曲が長いほど、誤答率が1になる確率が低くなるからである。

## 6.2. 実験2

今回の実験では、ある楽曲から求めた印象変化の推定値の時間毎の平均があるしきい値よりも大きい値の楽曲をユーザに推薦する楽曲推薦システムを構築する。

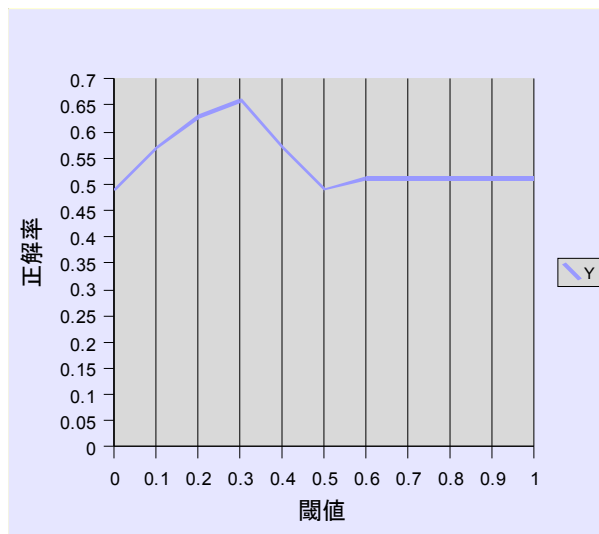
### 6.2.1. 実験の手順

まず、ある一人の被験者に対してしきい値と正解率の実験を行った。全曲の正解率が最大となる閾値を選択する。しきい値を0から0.1ずつ増やして、正解率の一番高いものにする。その結果について考察する。

### 6.2.2. 結果と考察

しきい値	正解率
0	0.49
0.1	0.57
0.2	0.63
0.3	0.66
0.4	0.57
0.5	0.49
0.6	0.51
0.7	0.51
0.8	0.51
0.9	0.51
1	0.51

表 2



グラフ 2

表2とグラフ2から分かるように、しきい値が0.3になったときに、正解率が一番高く（約0.66）なるので、今実験のしきい値を0.3にセットした。

一人の被験者に対して、しきい値を0.3にセットした実験結果は以下の通りである

曲番号	推薦値	推定	正解	当たり/外れ
0	0	X	O	外れ
1	0	X	X	当たり
2	0.15	X	O	外れ
3	0.2	X	O	外れ
4	0.29	X	X	当たり
6	0.38	O	O	当たり
7	0	X	X	当たり
8	0.17	X	O	外れ
9	0.58	O	X	外れ
10	0	X	X	当たり
11	0.09	X	X	当たり
12	0.33	O	O	当たり
13	1	O	O	当たり
14	0	X	X	当たり

15	0	X	O	外れ
16	0	X	O	外れ
17	0	X	X	当たり
19	0.13	X	X	当たり
20	0	X	X	当たり
21	0.13	X	X	当たり
23	0.43	O	O	当たり
24	0	X	X	当たり
25	0	X	O	外れ
26	0	X	O	外れ
28	0	X	O	外れ
29	0.11	X	X	当たり
30	0	X	X	当たり
31	0	X	X	当たり
32	0.44	O	O	当たり
33	0.38	O	O	当たり
34	0	X	O	外れ
35	0.4	O	O	当たり
36	0.25	X	X	当たり
37	1	O	X	外れ
39	0.14	X	X	当たり

表 3

表3より、推薦値がある決まった値を超えると、その楽曲をリスナーに推薦することになっており、「推定」のところに○がついている。アンケートで採ったリスナーの「真の印象」（表に「正解」と書いてある列）と比較すると、66%の正解率が得られた。すなわち、リスナーに勧めた楽曲は半分以上はリスナーの好む楽曲と考えてもいい。正解率を上げるにはデータ数を増やして、改めてしきい値を決めることが考えられる。

## 6. まとめ

本研究では楽曲のコード進行につれて変化するリスナーの楽曲に対する印象変化を HMM を用いて推定する手法を提案し、実験によりその有効性を実証した。また、推定した印象変化をもとにして楽曲推薦システムを構築することを提案した。実験の結果を見ると、半分以上の推定された印象変化は満足な結果になるが、全体的な正解率と最尤推定量を改善できれば、もっと信頼できるシステムが作れると考えられる。

また、被験者のフィーリングというのは個人差があるほか、気分と時間によって感性が変わるので、完璧に推測するのはなかなか困難であるため、学習機能が追加される楽曲推薦システムの開発が期待される。

今後の課題としてはメジャースケールだけではなく、マイナーコード、セブンスコードなどを追加して推測する。そして、誤答率を下げたり、最尤推定量をあげたりするためにリスナーと曲のデータの数を増やすことなどが考慮される。

## 7. 参考文献

- ① 北研二、確率的言語モデル、東京大学出版社、1999
- ② HMMによるナーの印象変化推定と楽曲推薦システムへの応用、東海林智也
- ③ ウィキペディア  
<http://www.wikipedia.com>
- ④ 音楽用語集  
<http://www.ongaku.tishiki.net/>
- ⑤ 印象語のグループ化を用いた楽曲推薦システム
  1. <http://www.jaist.ac.jp/jsai2006/program/pdf/100030.pdf>
- ⑥ 感性情報を用いた楽曲推薦システム  
<http://www.ieice.org/iss/de/DEWS/proc/2003/papers/8-P/8-P-06.pdf>
- ⑦ 確立的言語モデルによる自由発話認識に関する研究  
<http://unicorn.ike.tottori-u.ac.jp/murakami/doctor/node7.html>  
<http://unicorn.ike.tottori-u.ac.jp/murakami/doctor/node14.html>
- ⑧ 隠れマルコフモデル  
<http://www.eb.waseda.ac.jp/murata/masako.yoshimura/openhouse/method12.php>
- ⑨ 情報通信のメモ  
<http://www.yobology.info/text/viterbi/viterbi.htm>
- ⑩ 東京大学 工学部 計数工学科 / 物理工学科  
応用音響学：Viterbiアルゴリズム  
<http://ocw.u-tokyo.ac.jp/course-list/engineering/applied-acoustics-2005/lecture-notes/F2-ViterbiAlgorithm.pdf>



- ⑪ 失聴者のための音声認識技術を利用したマン・マン・インタフェースに関する研究  
<http://www.opencreation.org/~cans/research/paper/html/node23.html>
- ⑫ 田中勝人、現代時系列分析、岩波書店、2006
- ⑬ 状況と嗜好に関するアノテーションに基づくオンライン楽曲推薦システム  
<http://www.brl.ntt.co.jp/people/hirata/Papers/sigmus2004-Dec.pdf>
- ⑭ Robin Good'  
[http://www.masternewmedia.org/music\\_recommendation/music\\_recommendation\\_systems/FOAF\\_based\\_music\\_recommendation\\_system\\_uses\\_personal\\_profiles\\_20051001.htm](http://www.masternewmedia.org/music_recommendation/music_recommendation_systems/FOAF_based_music_recommendation_system_uses_personal_profiles_20051001.htm)