論　文

階層的時系列データのための識別モデル

金子　悟士[†]　林　朗[†a)]　末松　伸朗[†]　岩田　一貴[†]

Discriminative Models for Time Series Data with Hierarchical Structure

Satoshi KANEKO[†], Akira HAYASHI[†a)], Nobuo SUEMATSU[†], and Kazunori IWATA[†]

あらまし　階層隠れマルコフモデル (HHMM) は隠れマルコフモデル (HMM) を階層化した階層構造をもつ生成モデルである．本研究では，HHMM に対応する新たな識別モデルとして SVM$^{\mathrm{HHMM}}$ を提案する．HHMM や階層隠れ条件付き確率場 (HHCRF) が確率モデルであるのに対して，SVM$^{\mathrm{HHMM}}$ は確率モデルではなく，確率的遷移や出力に関するパラメータを用いてマージン最大化に基づいて識別を行う．これらの階層モデルはモデルパラメータを推定した後に，観測系列から階層状態系列を推定することができる．人工データと実データを用いた実験により HHMM と HHCRF，SVM$^{\mathrm{HHMM}}$ の性能比較を行い，SVM$^{\mathrm{HHMM}}$ の有効性を示す．

キーワード　隠れマルコフモデル，構造化 SVM，階層的時系列データ

1. まえがき

自然言語処理や音声認識などの時系列データ解析には生成モデルである隠れマルコフモデル (HMM) が幅広く用いられてきた．近年，識別モデルである条件付き確率場 (CRF) が提案され [1]，様々な応用問題に対して HMM よりも性能が良いことが示されている．生成モデルと異なり識別モデルは，変数の確率的独立性を仮定する必要がなく，観測系列からラベル系列を推定する問題設定では観測のモデル化が必要ないという特長をもっており，多くの時系列ラベル付け問題でその有効性が示されてきた．

階層 HMM (HHMM) は HMM を階層化した生成モデルであり，階層的に時系列データの状態を表現することができる [2]．HHMM は多くの時系列がもつ多重時間スケール構造を自然に表現することができるため，大いに注目されている．Murphy らは HHMM がダイナミックベイジアンネットワーク (DBN) で表現可能であり，効率的な状態推定アルゴリズムが適用可能であることを示した [3]．

階層隠れ CRF (HHCRF) は HHMM に対応する識別モデルである [4], [5]．HHCRF は状態系列推定及びセグメンテーション実験に関して HHMM よりも優れていることが実験的に示されている．しかしながら，HHCRF はモデルが大きくなりすぎると計算コストが大幅に増えてしまうため，実験の時間的制限から訓練データサイズを制限することもあり，訓練データサイズが大きい場合により高い性能を示す [5] という識別モデルの利点を十分に生かしきることができなかった．

一方，Altun らは構造化 SVM の一つとして SVM$^{\mathrm{HMM}}$ を提案した [6]．SVM$^{\mathrm{HMM}}$ は HMM に対応する識別モデルであり，HMM と CRF に対する優位性や計算の効率性が示されている [7]．

本研究では，HHMM に対応する新たな識別モデルとして SVM$^{\mathrm{HHMM}}$ を提案する．SVM$^{\mathrm{HHMM}}$ はマージン最大化に基づいた時系列データの識別モデルであり，カーネルトリックが適用可能であるなど，同じ識別モデルである HHCRF にはなかった特長をもつ．更に SVM$^{\mathrm{HHMM}}$ は HHCRF より計算が効率的であることから，大規模な階層時系列データを扱うことが可能になり，その結果，高精度な識別が達成できると考えられる．

本研究の実験では階層的時系列データの生成モデルと識別モデルの性能比較を目的に，人工データと実データの両方を用いて HHMM と HHCRF，SVM$^{\mathrm{HHMM}}$ の時系列データの階層的ラベル付け性能の比較を行う．

本論文の構成は次のとおりである．まず，従来モデ

[†] 広島市立大学大学院情報科学研究科，広島市
Graduate School of Information Sciences, Hiroshima City University, 3–4–1 Ozuka-Higashi, Asa-minami-ku, Hiroshima-shi, 731–3194 Japan
a) E-mail: akira@hiroshima-cu.ac.jp

ルである HHMM と HHCRF を簡単に説明する．次に **3.** にて提案モデルである $\mathrm{SVM}^{\mathrm{HHMM}}$ を説明する．**4.** では三つのモデルに対して人工データと実データを用いた比較実験の結果を掲載する．**5.** で実験結果の考察を行い，**6.** にて本研究をまとめる．

2. 従来モデル

2.1 HHMM

階層隠れマルコフモデル (HHMM：Hierarchical HMM) は HMM を一般化した生成モデルであり，時系列データをモデル化する際，状態を階層的に表すことができる．HHMM は状態に短期的な依存関係と長期的な依存関係が同時に存在するモデル構造を表現することを目的として提案された．

HHMM は図 1 のように木構造で表される．HHMM は円で表される内部状態，台形で表される外部状態，内側に end と書かれた長方形で表される終了状態の 3 種類の状態をもつ．円や台形中の数字は状態番号である．また，各状態は，下層への縦遷移（破線の矢印），同層間での横遷移（実線の矢印），上層への強制遷移（点線の矢印）の 3 種類の状態遷移によって結ばれる．時系列データは HHMM から以下のようにして生成される．

（手順 1） 時刻 $t = 1$，根状態から状態遷移が開始される．

（手順 2） 縦遷移：現在の状態（内部状態若しくは根状態）から下層の状態へ遷移する．もし遷移先が内部状態であるならば，外部状態に達するまで更に下層への縦遷移を繰り返す．

（手順 3） 観測値の出力：外部状態から観測値 o_t を出力し，時刻を 1 進める．

（手順 4） 横遷移：同じ階層の状態への状態遷移を行う．もし遷移先の状態が内部状態ならば手順 2 に戻る．または，もし遷移先の状態が外部状態ならば手順 3 へ戻る．若しくは，終了状態へ遷移するならば手順 5 へ進む．

（手順 5） 強制遷移：現在の階層を呼び出した一つ上の層の状態へ遷移し，手順 4 に戻る．

HHMM を提案した Fine らは Inside-Outside アルゴリズムをもとにした状態推定アルゴリズムを導いた [2]．しかし，そのアルゴリズムはあまり効率的ではなく，時系列の長さを T とすると，観測系列から状態を推定するために $O(T^3)$ の計算コストがかかる．後に Murphy らは HHMM が DBN (Dynamic Bayesian Network) で表現できることを示した [3]．それにより，計算コストが線形時間 $O(T)$ の Forward-Backward アルゴリズムやビタビアルゴリズムが適用可能となった．

2.1.1 HHMM の DBN 表現

BN (Bayesian Network) とは，確率変数間の条件付き独立関係を表す有向グラフである．その BN を時間とともに変化する確率変数（確率過程）へ拡張したものが DBN である．

図 2 に 3 階層の HHMM の DBN 表現を示す．図中の確率変数 o_t は時刻 t ($t = 1, \ldots, T$) における出力（観測値）を表す．本論文では，簡単のため，最下層の状態が観測値を出力する外部状態であり，出力は離散値であると仮定する．q_t^d は状態変数と呼ばれ，時刻 t における階層 d ($d = 1, \ldots, D$) の状態を離散値で表す．ただし，d は第 d 層を表す変数，D は HHMM の

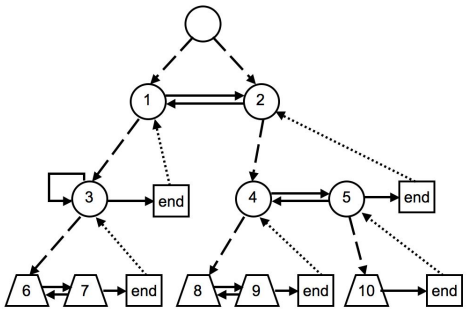


図 1 3 階層をもつ HHMM の状態遷移図の例
Fig. 1 Example of an HHMM with a three-level hierarchy.

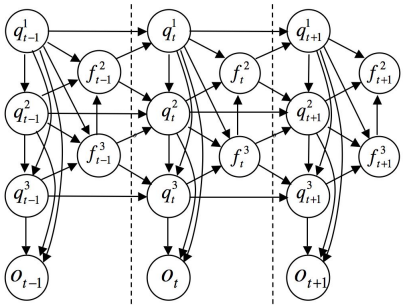


図 2 3 階層の HHMM の DBN 表現．時刻 $t-1$ から $t+1$ までの区間のみを表す．
Fig. 2 DBN representation of a three-level HHMM, which draws state-transitions from time $t-1$ to $t+1$.

全階層数であり，$d=1$ が最上層，$d=D$ が最下層を指す．

f_t^d は終了指標変数と呼ばれる 2 値変数であり，終了状態への遷移の有無を示す．時刻 t において階層 d の状態 q_t^d が終了状態へ遷移する場合に 1，それ以外で 0 をとる．終了指標変数は HHMM を DBN で表現するために重要な確率変数である．HHMM では下層の状態が終了状態へ遷移することによって，その上層における状態遷移が起こる．つまり，$f_t^d=1$ は一つ上の第 $d-1$ 層への遷移操作の切り替えを意味する．また，このとき次時刻 $t+1$ の階層 d における状態変数 q_{t+1}^d の値は縦遷移によって決定される．また，$f_t^d=0$ ならば $d'<d$ を満たす全ての d' に対して，$f_t^{d'}=0$ という関係が成り立つ．すなわち，終了指標変数の値が 0 となる階層の総数は遷移操作が現在どの階層で行われているかを表す．

HHMM の状態遷移確率と出力確率の集合でモデルを完全に定義できる．そのため，全状態遷移確率と全出力確率の集合が HHMM のモデルパラメータとなる．

以下に，HHMM の状態遷移確率と出力確率を示す．

$$p(q_t^d=j'|q_{t-1}^d=j, f_{t-1}^d=f, f_{t-1}^{d+1}=b, q_t^{1:d-1}=\vec{i}) = \begin{cases} \delta(j,j') & \text{if } b=0 \\ A_{\vec{i}}^d(j,j') & \text{if } b=1 \text{ and } f=0 \\ \pi_{\vec{i}}^d(j') & \text{if } b=1 \text{ and } f=1 \end{cases} \quad (1)$$

$$p(f_t^d=1|q_t^d=j, q_t^{1:d-1}=\vec{i}, f_t^{d+1}=b) = \begin{cases} 0 & \text{if } b=0 \\ Ae_{\vec{i}}^d(j) & \text{if } b=1 \text{ and } d\neq 1 \end{cases} \quad (2)$$

$$p(o_t=s_l|q_t^{1:D}=\vec{i})=B(l,\vec{i}) \quad (3)$$

ただし，任意の d に対して，$q_t^{1:d}=\{q_t^1,\ldots,q_t^d\}$ は時刻 t における階層 1 から階層 d までの状態を格納したベクトルであり，ベクトルを整数 $\vec{i}$ に対応させて表す．なお，階層 d の（終了状態を除いた）状態数を N^d とすれば，$1\leq q_t^d\leq N^d$ である．また，記述を簡潔にするために根状態 $q_t^0=0$ を仮定している．更に，はじめの時刻における状態遷移を表すために $f_0^1=1$ を，最下層における状態遷移を表すために $f_t^{D+1}=1$ をそれぞれ仮定している．

式 (1) 中の $\delta(j,j')$ は $j=j'$ のときに 1，そうでないとき 0 をとる．また，$A_{\vec{i}}^d(j,j')$ は $q_t^{1:d-1}=\vec{i}$ のときに階層 d において状態 j から状態 j' に横遷移する確率であり，$\forall j,\forall j', 1\geq A_{\vec{i}}^d(j,j')\geq 0$，及び $\forall j, \sum_{j'=1}^{N^d} A_{\vec{i}}^d(j,j')=1$ を満たす．$\pi_{\vec{i}}^d(j')$ は $q_t^{1:d-1}=\vec{i}$ のときに階層 $d-1$ から階層 d の状態 j' へ縦遷移する確率（初期分布確率）であり，$\forall j', 1\geq \pi_{\vec{i}}^d(j')\geq 0$，及び $\sum_{j'=1}^{N^d}\pi_{\vec{i}}^d(j')=1$ を満たす．式 (2) 中の $Ae_{\vec{i}}^d(j)$ は $q_t^{1:d-1}=\vec{i}$ のときに階層 d において状態 j が終了状態へ遷移する確率であり，$\forall j, 1\geq Ae_{\vec{i}}^d(j)\geq 0$ を満たす．式 (3) の $B(l,\vec{i})$ は観測値の集合を $\{s_l|1\leq l\leq L\}$ としたとき，$q_t^{1:D}=\vec{i}$ のときに，l 番目の観測値 s_l が出力される確率であり，$\forall l, B(l,\vec{i})\geq 0$，及び $\sum_{l=1}^{L}B(l,\vec{i})=1$ を満たす．

2.2 HHCRF

生成モデルである HMM に対し，CRF は識別モデルと呼ばれる．生成モデル HHMM が観測系列と階層状態系列の同時確率をモデル化するのに対し，識別モデル HHCRF は観測系列を条件とした階層状態系列の条件付き確率をモデル化する．

$Q^{1:D}=\{q_1^{1:D},\ldots,q_T^{1:D}\}$ を状態変数系列と呼ぶ．$Q^{1:D}$ は各時刻における階層 1 から D までの状態変数の値の集合である．

$F^{2:D}=\{f_1^{2:D},\ldots,f_T^{2:D}\}$ を終了指標変数系列と呼ぶ．$F^{2:D}$ は各時刻における階層 2 から D までの終了指標変数の値の集合である．

$O=\{o_1,\ldots,o_t\}$ を観測変数系列（観測系列）と呼ぶ．O は各時刻における観測値の値の集合である．

HHCRF は以下の条件付き確率をモデル化する．

$$p(Q^{1:D},F^{2:D}|O;\Lambda) = \frac{1}{Z(O;\Lambda)}\exp\left(\sum_{k=1}^{K}\lambda_k\Phi_k(Q^{1:D},F^{2:D},O)\right). \quad (4)$$

ただし，$\Phi_k(Q^{1:D},F^{2:D},O)$ $(k=1,\ldots,K)$ はフィーチャであり，λ_k はフィーチャ Φ_k の重みを表す．モデルに含まれる全てのフィーチャの重みの集合 $\Lambda=\{\lambda_1,\ldots,\lambda_K\}$ が HHCRF のモデルパラメータとなる．フィーチャの詳細は玉田らの文献 [5] または後の **4.** を参照されたい．$Z(O;\Lambda)$ は $p(Q^{1:D},F^{2:D}|O;\Lambda)$ の正規化項である．

3. 提案モデル：SVM$^{\text{HHMM}}$

SVM$^{\text{HHMM}}$ (Hierarchical Hidden Markov Model Support Vector Machine) は観測系列から階層状態系列を出力することができる識別モデルである．

$\mathrm{SVM}^{\mathrm{HHMM}}$ は HHCRF と同じく HHMM に対応する識別モデルである．モデル学習に関して，HHMM と HHCRF は確率の最大化を行うのに対し，$\mathrm{SVM}^{\mathrm{HHMM}}$ はマージンの最大化を行う．

3.1 学習問題の設定

$\mathrm{SVM}^{\mathrm{HHMM}}$ を Murphy らが提案した HHMM の DBN 表現 [3] に従いモデル化する．

$\mathcal{Q}$ を全ての階層状態系列からなる集合とする．

$\mathcal{O}$ を全ての観測変数系列からなる集合とする．

新たに予測関数 $\mathrm{f}:\mathcal{O}\rightarrow\mathcal{Q}$ を定義する．予測関数 f は観測系列 O が与えられたとき，階層状態系列 $(Q^{1:D},F^{2:D})$ を予測するために用いられる．

ここでは，識別関数 $\mathcal{F}:\mathcal{Q}\times\mathcal{O}\rightarrow\mathbb{R}$ を学習するアプローチを採用する．予測関数 f は，与えられた観測系列 O に対して，識別関数 $\mathcal{F}$ を最大化する状態系列 $(Q^{1:D},F^{2:D})$ を見つけることにより導出される．つまり，

$$\mathrm{f}(O;\mathbf{w}) = \mathrm{argmax}_{(Q^{1:D},F^{2:D})\in\mathcal{Q}}\mathcal{F}(Q^{1:D},F^{2:D},O;\mathbf{w}). \tag{5}$$

ここで，$\mathbf{w}$ はパラメータベクトルである．識別関数 $\mathcal{F}$ は観測系列と階層状態系列を組み合わせたフィーチャベクトル $\mathbf{\Phi}(Q^{1:D},F^{2:D},O)$ に関して線形であると仮定し，

$$\mathcal{F}(Q^{1:D},F^{2:D},O;\mathbf{w}) = \langle\mathbf{w},\mathbf{\Phi}(Q^{1:D},F^{2:D},O)\rangle, \tag{6}$$

と定義する．ただし $\langle\cdot,\cdot\rangle$ は内積を表現する．

3.2 フィーチャベクトル

フィーチャベクトル $\mathbf{\Phi}(Q^{1:D},F^{2:D},O)$ は，観測系列 O とそれに対応する階層状態系列 $(Q^{1:D},F^{2:D})$ とのマッチングを表現する．フィーチャベクトル $\mathbf{\Phi}$ は K 個のフィーチャ$\{\Phi_k|1\leq k\leq K\}$ から構成され，各フィーチャ Φ_k は素性関数 $\{\phi_k\}$ を用いて計算される．なお，素性関数の英訳は feature function であり本来フィーチャと同じであるが，ここでは，フィーチャと素性関数を別の意味で使い分ける．素性関数は任意に構成することができ，状態推定に有利な素性関数を選択できる．ただし，素性関数の形は推論コストと密接に関係するので，注意が必要である．ここでは，フィーチャベクトルを階層状態系列，観測系列に対し，

$$\mathbf{\Phi}(Q^{1:D},F^{2:D},O)=\begin{pmatrix}\Phi_1(Q^{1:D},F^{2:D},O)\\ \vdots\\ \Phi_K(Q^{1:D},F^{2:D},O)\end{pmatrix}, \tag{7}$$

と表す．HHMM では，ある時刻 t における状態 $q_t^{1:D}$ が与えられたとき，その時刻における観測値 o_t は他の時刻の観測値とは独立であることを仮定する．一方，$\mathrm{SVM}^{\mathrm{HHMM}}$ は確率モデルではないため独立性の仮定が不要である．そのため，観測系列を表すための有効な特徴を任意に付け加えることができる．したがって，例えば時刻 t の状態 $q_t^{1:D}$ に対して同時刻の観測値 o_t だけでなく，時刻 $t+1$ の観測値 o_{t+1} との依存関係をもたせることも可能である．

しかし，本研究では HHMM と等価なモデル構造をもつような素性関数を用いる．具体的には，まず，式 (1)，(2)，(3) で示した HHMM の横遷移確率 $A_i^d(j,j')$，縦遷移確率 $\pi_i^d(j')$，終了確率 $Ae_i^d(j)$，そして観測確率 $B(l,\vec{t})$ に対応する 4 種類の素性関数，$\{\phi^{(\mathrm{Hor})}_{d,\vec{i},j,j'}(q_{t-1}^{1:D},\ q_t^{1:D},\ f_{t-1}^{2:D},\ f_t^{2:D},\ o_t)\}$，$\{\phi^{(\mathrm{Ver})}_{d,\vec{i},j'}(q_{t-1}^{1:D},\ q_t^{1:D},\ f_{t-1}^{2:D},\ f_t^{2:D},\ o_t)\}$，$\{\phi^{(\mathrm{End})}_{d,\vec{i},j}(q_{t-1}^{1:D},\ q_t^{1:D},\ f_{t-1}^{2:D},\ f_t^{2:D},\ o_t)\}$，$\{\phi^{(\mathrm{Occ})}_{\vec{i},l}(q_{t-1}^{1:D},q_t^{1:D},f_{t-1}^{2:D},f_t^{2:D},o_t)\}$ を以下のように定義する．

$$\begin{aligned}&\phi^{(\mathrm{Hor})}_{d,\vec{i},j,j'}(q_{t-1}^{1:D},q_t^{1:D},f_{t-1}^{2:D},f_t^{2:D},o_t)\\ &\quad=\delta(q_t^d=j')\delta(q_{t-1}^d=j)\delta(f_{t-1}^{d+1}=1)\\ &\qquad\times\delta(f_{t-1}^d=0)\delta(q_t^{1:d-1}=\vec{i}),\ \ \forall d,\forall\vec{i},\forall j,\forall j',\end{aligned} \tag{8}$$

$$\begin{aligned}&\phi^{(\mathrm{Ver})}_{d,\vec{i},j'}(q_{t-1}^{1:D},q_t^{1:D},f_{t-1}^{2:D},f_t^{2:D},o_t)\\ &\quad=\delta(q_t^d=j')\delta(f_{t-1}^{d+1}=1)\\ &\qquad\times\delta(f_{t-1}^d=1)\delta(q_t^{1:d-1}=\vec{i}),\quad\forall d,\forall\vec{i},\forall j',\end{aligned} \tag{9}$$

$$\begin{aligned}&\phi^{(\mathrm{End})}_{d,\vec{i},j}(q_{t-1}^{1:D},q_t^{1:D},f_{t-1}^{2:D},f_t^{2:D},o_t)\\ &\quad=\delta(f_t^d=1)\delta(q_t^d=j)\\ &\qquad\times\delta(q_t^{1:d-1}=\vec{i})\delta(f_t^{d+1}=1),\ \ \forall d\geq 2,\forall\vec{i},\forall j,\end{aligned} \tag{10}$$

$$\begin{aligned}&\phi^{(\mathrm{Occ})}_{\vec{i},l}(q_{t-1}^{1:D},q_t^{1:D},f_{t-1}^{2:D},f_t^{2:D},o_t)\\ &\quad=\delta(q_t^{1:D}=\vec{i})\delta(o_t=s_l),\quad\forall\vec{i},\forall l,\end{aligned} \tag{11}$$

次に，式 (8) から (11) で定義された全ての素性関数（全部で K 個あるとする）をあらためて $\phi_1,\phi_2,\ldots,\phi_K$ としたうえで，k 番目 $(1\leq k\leq K)$ のフィーチャを

$$\begin{aligned}&\Phi_k(Q^{1:D},F^{2:D},O)\\ &\quad=\sum_{t=1}^{T}\phi_k(q_{t-1}^{1:D},q_t^{1:D},f_{t-1}^{2:D},f_t^{2:D},o_t),\end{aligned} \tag{12}$$

と定義し，これら K 個のフィーチャを縦に並べたものをフィーチャベクトル $\mathbf{\Phi}(Q^{1:D}, F^{2:D}, O)$ とする．

また，SVM$^{\mathrm{HHMM}}$ のモデルパラメータはフィーチャの重み $\mathbf{w}$ である．

$$\mathbf{w} = (w_1, w_2, \ldots, w_K)^T. \tag{13}$$

3.3 予測関数

入力観測系列に対する階層状態系列は識別関数 $\mathcal{F}$ を最大にする階層状態系列として求められる．

$$\begin{aligned}(\hat{Q}^{1:D}, \hat{F}^{2:D}) &= \mathrm{argmax}_{Q^{1:D}, F^{2:D}} \mathcal{F}(Q^{1:D}, F^{2:D}, O; \mathbf{w}), \\ &= \mathrm{argmax}_{Q^{1:D}, F^{2:D}} \langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D}, F^{2:D}, O) \rangle.\end{aligned} \tag{14}$$

式 (14) に関して，以下の一般化ビタビアルゴリズムを適用することで階層状態系列を得ることができる．

与えられた観測系列 $O = (o_1, o_2, \ldots, o_T)$ に対する識別関数 $\mathcal{F}$ を最大にする一本の階層状態系列 $\hat{Q}^{1:D} = (\hat{q}_1^{1:D}, \hat{q}_2^{1:D}, \ldots, \hat{q}_T^{1:D})$, $\hat{F}^{2:D} = (\hat{f}_1^{2:D}, \hat{f}_2^{2:D}, \ldots, \hat{f}_T^{2:D})$ を見つけるために $\delta_t(\vec{q}, \vec{f})$ を定義する．$\delta_t(\vec{q}, \vec{f})$ は初期状態からスタートして，$o_1, o_2, \ldots, o_t$ を観測しながら時刻 t で階層状態 $(\vec{q}, \vec{f})$ で終わっている識別関数の部分状態系列に対するベストスコアであり，以下の式で定義される．

$$\begin{aligned}&\delta_t(\vec{q}, \vec{f}) \\ &= \max_{q_{1:t-1}^{1:D}, f_{1:t-1}^{2:D}} \left(\sum_{t'=1}^{t-1} \langle \mathbf{w}, \vec{\phi}(q_{t'-1}^{1:D}, q_{t'}^{1:D}, f_{t'-1}^{2:D}, f_{t'}^{2:D}, o_{t'}) \rangle \right. \\ &\quad \left. + \langle \mathbf{w}, \vec{\phi}(q_{t-1}^{1:D}, q_t^{1:D}, f_{t-1}^{2:D}, f_t^{2:D}, o_t) \rangle |_{q_t^{1:D} = \vec{q}, f_t^{2:D} = \vec{f}} \right).\end{aligned}$$

$$\forall \vec{q}, \forall \vec{f}, 1 \le t \le T, \tag{15}$$

ここで，

$$\begin{aligned}&\vec{\phi}(q_{t'-1}^{1:D}, q_{t'}^{1:D}, f_{t'-1}^{2:D}, f_{t'}^{2:D}, o_{t'}) \\ &= \begin{pmatrix} \phi_1(q_{t'-1}^{1:D}, q_{t'}^{1:D}, f_{t'-1}^{2:D}, f_{t'}^{2:D}, o_{t'}) \\ \vdots \\ \phi_K(q_{t'-1}^{1:D}, q_{t'}^{1:D}, f_{t'-1}^{2:D}, f_{t'}^{2:D}, o_{t'}) \end{pmatrix},\end{aligned} \tag{16}$$

である．

Viterbi アルゴリズム [8], [9] のように $\delta_t(\vec{q}, \vec{f})$ を前向きに計算していくことで最適階層状態系列 $(\hat{Q}^{1:D}, \hat{F}^{2:D})$ を得る．

3.4 パラメータ推定

SVM$^{\mathrm{HHMM}}$ のモデルパラメータであるパラメータベクトル $\mathbf{w}$ の推定は訓練集合 $D = \{Q^{1:D^{(n)}}, F^{2:D^{(n)}}, O^{(n)}\}_{n=1}^N$ が与えられたとき次の最小化問題を解くことによって達成される：

$$\mathrm{Minimize}_{\mathbf{w}, \xi_1, \xi_2, \ldots, \xi_N} \quad \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{N}\sum_{n=1}^{N} \xi_n, \tag{17}$$

Subject to

$$\begin{aligned}&\forall n, \forall \{Q^{1:D}, F^{2:D}\} \\ &\quad \langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D^{(n)}}, F^{2:D^{(n)}}, O^{(n)}) \rangle \\ &\qquad \ge \langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D}, F^{2:D}, O^{(n)}) \rangle \\ &\qquad + \Delta\big((Q^{1:D^{(n)}}, F^{2:D^{(n)}}), (Q^{1:D}, F^{2:D})\big) - \xi_n, \\ &\forall n,\ \xi_n \ge 0,\end{aligned}$$

ここで，C はチューニングパラメータであり，目的関数のマージン最大化項 $\|\mathbf{w}\|^2/2$ と訓練データの誤り最小化項 $\sum_{n=1}^{N} \xi_n/N$ のトレードオフを調整する．制約条件式では正解系列に対する内積値 $\langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D^{(n)}}, F^{2:D^{(n)}}, O^{(n)}) \rangle$ と非正解系列に対する内積値 $\langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D}, F^{2:D}, O^{(n)}) \rangle$ 間のマージンを調整するために損失関数 Δ が右辺に加えられている．損失関数 Δ は正解系列 $\{Q^{1:D^{(n)}}, F^{2:D^{(n)}}\}$ に対する，$\{Q^{1:D}, F^{2:D}\}$ の損失を表現する．すなわち，正解系列に似ている $\{Q^{1:D}, F^{2:D}\}$ に対するマージンを狭め，似ていないものに対するマージンを広げることができる．この方法はマージン再スケーリングと呼ばれる [10]．各訓練データ $\{Q^{1:D^{(n)}}, F^{1:D^{(n)}}, O^{(n)}\}$ に対して，“非最適な” ラベル $(Q^{1:D}, F^{2:D})$ ごとに制約条件が存在している．そのため，SVM の学習を定義するためにはばくだいな数の制約が存在している．そこで，切除断面アルゴリズム (Cutting Plane Algorithm) を用いることで，制約条件数が多いにもかかわらず効率的に SVM$^{\mathrm{HHMM}}$ の最小化問題を解く．切除断面アルゴリズムでは，もとの制約条件を許容誤差 ϵ で近似した問題を解くまで制約条件を繰り返し増やしていく [10]．このアルゴリズムは制約条件がない状態から始まり，各訓練データに対して最も大きく制約条件を破る階層状態系列 $(\acute{Q}^{1:D}, \acute{F}^{2:D})$ を見つけていく．もしそれらに対応する制約条件における右辺と左辺の誤差が ϵ 以上であるならば，$(\acute{Q}^{1:D}, \acute{F}^{2:D})$ を制約条件集合へ加える．その更新された制約条件集合を用いて改めて最小化問題を解く．

このアルゴリズムでは，入力は訓練集合 $D =$

Input : $D = \{Q^{1:D^{(n)}}, F^{1:D^{(n)}}, O^{(n)}\}_{n=1}^{N}$、$C$、$\epsilon$
$S_n \leftarrow \emptyset$、$\xi_n \leftarrow 0$ for all $n = 1, ..., N$、$\mathbf{w} \leftarrow \mathbf{0}$
Repeat
  for $n = 1, ..., N$
    $H(Q^{1:D}, F^{2:D}; \mathbf{w}) = \Delta\big((Q^{1:D^{(n)}}, F^{2:D^{(n)}}), (Q^{1:D}, F^{2:D})\big)$
      $+\langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D}, F^{2:D}, O^{(n)})\rangle$
      $-\langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D^{(n)}}, F^{2:D^{(n)}}, O^{(n)})\rangle$
    $(\acute{Q}^{1:D}, \acute{F}^{2:D}) = \mathrm{argmax}_{(Q^{1:D}, F^{2:D}) \in \mathcal{Q}} H(Q^{1:D}, F^{2:D}; \mathbf{w})$
    $\xi_n = \max\big(0, \max_{(Q^{1:D}, F^{2:D}) \in S_n} H(Q^{1:D}, F^{2:D}; \mathbf{w})\big)$
    if $H(\acute{Q}^{1:D}, \acute{F}^{2:D}; \mathbf{w}) > \xi_n + \epsilon$ **then**
      $S_n \leftarrow S_n \cup \{(\acute{Q}^{1:D}, \acute{F}^{2:D})\}$
      $\mathbf{w} \leftarrow$ Optimize (17) over $S = \cup_n S_n$
    endif
  endfor
Until S_n has not changed during iteration

図 3　許容誤差 ϵ で最小化問題 (17) を解く切断平面アルゴリズム．
Fig. 3　Cutting Plane Algorithm which solves the minimization problem with an admissible error ϵ.

$\{Q^{1:D^{(n)}}, F^{1:D^{(n)}}, O^{(n)}\}_{n=1}^{N}$ とチューニングパラメータ C と ϵ である．

学習アルゴリズムは図 3 のように記述できる．ただし，$\big(\mathbf{w} \leftarrow$ Optimize (17) over $S = \cup_n S_n\big)$ は，これまでに得られた “許容誤差を超えて制約条件を破る階層状態系列” の集合により与えられる制約条件のみを用いて式 (17) の最小化問題を解くことを意味する．

このアルゴリズムは有限回の繰り返し後終了することが保証されている [10]．

また，最も大きく拘束条件を破る階層状態系列 $(\acute{Q}^{1:D}, \acute{F}^{2:D})$ は訓練データ $(Q^{1:D^{(n)}}, F^{2:D^{(n)}}, O^{(n)})$ に対して以下のように定義されている．

$$
\begin{aligned}
&(\acute{Q}^{1:D}, \acute{F}^{2:D}) \\
&= \mathrm{argmax}_{Q^{1:D}, F^{2:D}} \{\Delta\big((Q^{1:D^{(n)}}, F^{2:D^{(n)}}), \\
&\quad (Q^{1:D}, F^{2:D})\big) + \langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D}, F^{2:D}, O^{(n)})\rangle \\
&\quad - \langle \mathbf{w}, \mathbf{\Phi}(Q^{1:D^{(n)}}, F^{2:D^{(n)}}, O^{(n)})\rangle\} \qquad (18)
\end{aligned}
$$

ここで，損失関数 Δ は

$$
\begin{aligned}
&\Delta\big((Q^{1:D^{(n)}}, F^{2:D^{(n)}}), (Q^{1:D}, F^{2:D})\big) \\
&\quad = \sum_{t=1}^{T} \Delta\big((q_t^{1:D^{(n)}}, f_t^{2:D^{(n)}}), (q_t^{1:D}, f_t^{2:D})\big), \qquad (19)
\end{aligned}
$$

すなわち，時系列全体の損失は各時刻ごとの損失の和で表されるものとする．このとき，**3.3** で説明した一般化ビタビアルゴリズムとほぼ同じアルゴリズムで拘束条件を最も大きく破る階層状態系列を求めることができる．

4. 実　　験

人工データと実データを用い実験を行い，HHMM と HHCRF，SVM$^{\mathrm{HHMM}}$ の状態系列推定性能の比較を行う．実データ実験には Skounakis らがテキストからの情報抽出のために用いた文章データを用いる [11]．

なお，HHMM と HHCRF については MATLAB を用いて，SVM$^{\mathrm{HHMM}}$ については C 言語を用いて実装を行った．

4.1　SVM$^{\mathrm{HHMM}}$ の設計

本実験では次の損失関数を定義する．

- ある時刻における全ての階層状態が正しい場合，その時刻での損失が 0，一つでも違う場合 1 とする．
- 各時刻での損失の和を系列の長さ T で割ったものを損失関数の値とする．

また，本実験のパラメータ推定には 1-スラック変数の最小化問題を用いる [12]．

4.2　人工データ実験

人工データを用いて HHMM，HHCRF，SVM$^{\mathrm{HHMM}}$ の状態系列推定性能の比較を行う．

4.2.1　実験条件

生成モデルから訓練集合と階層状態系列推定で用いるテスト集合を生成する．データ生成モデルは 2 階層であり，第 1 層の状態数は 2，第 2 層の状態数は 3 であり，一次マルコフモデルと二次マルコフモデルの混合次数モデルとする．ここで，二次マルコフモデルとは，その状態遷移が二次マルコフ過程に従うものである．つまり，次時刻の状態は現在時刻の状態だけではなく，前時刻の状態にも依存する．二次マルコフモデルにおける出力も現在時刻の状態だけではなく，前時刻の状態にも依存する．一次マルコフモデルと二次マルコフモデルの混合比率を $(1-\alpha) : \alpha$ とする $(0 \leq \alpha \leq 1)$．すなわち，$\alpha = 0$ で一次マルコフモデル，$\alpha = 1$ で二次マルコフモデルとなる．本実験では，性能を公平に評価するためにデータ生成モデルの各パラメータ（縦・横遷移確率，初期分布確率，出力確率）をそれぞれランダムに設定する．

データ生成モデルから生成されるデータ集合をそれぞれ 2 階層の HHMM と HHCRF，SVM$^{\mathrm{HHMM}}$ でモ

デル化する．いずれも第 1 層の状態数が 2，第 2 層の状態数が 3 のモデルである．また，三つのモデルの全状態は非隠れ状態であるとする．

訓練データ集合とテストデータ集合を以下のように定める．

訓練データ集合：それぞれが長さ $T = 20$ の N 本の時系列データ集合からなる．今回は訓練集合数に関して 2 通りの実験を行うため，$N \in \{5, 100\}$ とする．

テストデータ集合：それぞれが長さ $T = 20$ の 50 本の時系列データ集合からなる．

HHCRF のモデルパラメータの初期値には学習後の HHMM のパラメータを用いる．その理由は，HHCRF の訓練時間を短くするためである．なお，モデルに関し隠れ状態はないとしているため HHCRF の対数ゆう度関数は凸関数になる．したがって，初期値にかかわらず大域的最適解が得られることが保証されている．また，SVM$^{\mathrm{HHMM}}$ のマージン最大化と訓練誤差最小化のトレードオフを調整するチューニングパラメータである C を $C \in \{100, 1000\}$ とする．拘束条件に関する許容誤差を調整するチューニングパラメータである ϵ は $\epsilon = 0.5$ とした．

上記で示した実験を 10 回行う．

4.2.2 性能評価

訓練データ集合を用いて HHMM と HHCRF，SVM$^{\mathrm{HHMM}}$ を訓練した後，テストデータ集合に対して階層状態系列の推定を行う．推定アルゴリズムには一般化ビタビアルゴリズムを使用する．推定した階層状態系列 $(\hat{Q}^{1:D}, \hat{F}^{2:D})$ の評価には階層状態系列の正解率を用いる．正解の階層状態系列を $(\tilde{q}_{1:T}^{1:D}, \tilde{f}_{1:T}^{2:D})$ とすると，その正解率は以下の式から計算される．

$$\begin{aligned} \text{正解率} &= \frac{\text{階層状態が正しく推定されている時刻数}}{\text{テストデータの長さ}}, \\ &= \left[\frac{1}{T} \sum_{t=1}^{T} \delta(\tilde{q}_t^{1:D} = \hat{q}_t^{1:D}) \delta(\tilde{f}_t^{2:D} = \hat{f}_t^{2:D}) \right], \end{aligned} \tag{20}$$

ここで，$\delta(\tilde{q}_t^{1:D} = \hat{q}_t^{1:D})$ は，$\tilde{q}_t^{1:D} = \hat{q}_t^{1:D}$ のとき 1，それ以外で 0 となる．また，訓練の計算に要した時間を求める．

4.2.3 実験結果

実験結果を表 1，表 2，表 3 に示す．訓練集合サイズが $N = 5$ と小さい場合，混合率が $\alpha = 0$ のときに HHMM の正解率が一番高い．しかし，混合率が $\alpha = 0.5$ または 1 のとき，全ての識別モデルの正解率は HHMM の正解率を上回っている．また，$C = 100$ の SVM$^{\mathrm{HHMM}}$ は正解率が一番高い．

訓練集合サイズが $N = 100$ と大きい場合，混合率に関係なく $C = 1000$ の SVM$^{\mathrm{HHMM}}$ の正解率が一番高い．ただし，混合率が $\alpha = 0.5$ 及び 1 のときの $C = 100$ の SVM$^{\mathrm{HHMM}}$ の正解率は低くなってしまっている．このことから SVM$^{\mathrm{HHMM}}$ の正解率は C に依存することが分かる．

表 3 に示したように，SVM$^{\mathrm{HHMM}}$ の計算時間は HHCRF の計算時間よりも少ないが，HHMM と HHCRF は MATLAB にて，SVM$^{\mathrm{HHMM}}$ は C 言語にて実装しているため，表 3 の値の差を手法の差としてそのまま評価することはできない．あくまで参考値である．

表 1 人工データに対する正解率 (%)．$N = 5$ であり，SVM$^{\mathrm{HHMM}}$*1 は $C = 100$，SVM$^{\mathrm{HHMM}}$*2 は $C = 1000$ である．

Table 1 Accuracy rate for artificial data (%). $N = 5$, C for SVM$^{\mathrm{HHMM}}$*1 is 100 and C for SVM$^{\mathrm{HHMM}}$*2 is 1000.

$N = 5$	HHMM	HHCRF	SVM$^{\mathrm{HHMM}}$*1	SVM$^{\mathrm{HHMM}}$*2
$\alpha = 0$	**56.5**	54.4	55.3	48.2
$\alpha = 0.5$	38.6	39.6	**39.8**	39.5
$\alpha = 1$	26.7	28.1	**29.0**	28.8

表 2 人工データに対する正解率 (%)，$N = 100$ であり，SVM$^{\mathrm{HHMM}}$*1 は $C = 100$，SVM$^{\mathrm{HHMM}}$*2 は $C = 1000$ である．

Table 2 Accuracy rate for artificial data (%). $N = 100$, C for SVM$^{\mathrm{HHMM}}$*1 is 100 and C for SVM$^{\mathrm{HHMM}}$*2 is 1000.

$N = 100$	HHMM	HHCRF	SVM$^{\mathrm{HHMM}}$*1	SVM$^{\mathrm{HHMM}}$*2
$\alpha = 0$	61.4	61.9	61.9	**63.2**
$\alpha = 0.5$	32.4	33.0	30.3	**34.3**
$\alpha = 1$	26.7	27.2	26.7	**28.8**

表 3 人工データ ($\alpha = 1$) に対する訓練の計算に要した時間（秒）．SVM$^{\mathrm{HHMM}}$*1 は $C = 100$，SVM$^{\mathrm{HHMM}}$*2 は $C = 1000$ である．

Table 3 Training time for artificial data (sec). C for SVM$^{\mathrm{HHMM}}$*1 is 100 and C for SVM$^{\mathrm{HHMM}}$*2 is 1000.

	HHMM	HHCRF	SVM$^{\mathrm{HHMM}}$*1	SVM$^{\mathrm{HHMM}}$*2
$N = 5$	0.13	1.52	0.20	0.52
$N = 100$	1.18	25.77	1.87	4.39

4.3 実データ実験

実データを用いて HHMM，HHCRF，SVM$^{\mathrm{HHMM}}$ の状態系列推定性能の比較を行う．

4.3.1 実験データ

実験には Skounakis らがテキストからの情報抽出の

ために用いたデータを用いる [11]．情報抽出は文章から特定のキーワードを自動的に取得するタスクである．データセットは 3 種類あり，それぞれ『あるタンパク質とその細胞内の場所』『遺伝子とそれに関連する疾患』『あるタンパク質間の相互作用』について書かれている文章を含んでおり，いずれも生物医学文献である．それぞれのデータセットを Protein-Location データセット，Gene-Disorder データセット，Protein-Protein データセットと呼ぶ．

4.3.2 実験条件

各データセットから訓練データ集合と状態系列推定で用いるテストデータ集合を構成する．それぞれ，Protein-Location データセットは 730，Gene-Disorder データセットは 825，Protein-Protein データセットは 5270 文のテキストデータを含んでいる．HHMM と $\mathrm{SVM}^{\mathrm{HHMM}}$ の比較実験についてはそれぞれのデータ全てを用いる．HHCRF の実験については，Protein-Location データセット，Gene-Disorder データセットについては全てのデータを使うが，Protein-Protein データセットについては，HHCRF の MATLAB 実装による計算時間を適当な長さに抑えるために，1500 文を用いる．

本実験には 5 分割交差検定を用いる．また，$\mathrm{SVM}^{\mathrm{HHMM}}$ の損失関数には実験 1 にて定義したものを用いた．

4.3.3 性能評価

訓練集合を用いて HHMM と HHCRF，$\mathrm{SVM}^{\mathrm{HHMM}}$ を訓練した後，テスト集合に対して階層状態系列の推定を行う．推定アルゴリズムには一般化ビタビアルゴリズムを使用する．評価の仕方は **4.2.2** と同様とした．また，$C = 1000$，$\epsilon = 0.5$ とする．

4.3.4 実験結果

実験結果を表 4 と表 5 に示す．

表 4 より，HHCRF は全てのデータセットにおいて HHMM よりも高い精度でラベル付けを行っていることが分かる．一方，$\mathrm{SVM}^{\mathrm{HHMM}}$ も全てのデータセットにおいて HHMM よりも高い精度でラベル付けを行っていることが分かる．また，HHCRF と $\mathrm{SVM}^{\mathrm{HHMM}}$ を比べると $\mathrm{SVM}^{\mathrm{HHMM}}$ の方がラベル付け精度が高いことが分かる．

また，表 5 より HHMM は他の二つのモデルよりも訓練の計算時間が短いことが分かる．識別モデル同士を比較すると，$\mathrm{SVM}^{\mathrm{HHMM}}$ は HHCRF よりも訓練にかかる計算時間が短いが，実装言語が異なるため，これはあくまで参考値である．

表 4 実データに対する正解率 (%)．Protein-Protein*1 の N は 1500．Protein-Protein*2 の N は 5270 である．

Table 4 Accuracy rate for real data (%). N for Protein-Protein*1 is 1500 and N for Protein-Protein*2 is 5270.

	HHMM	HHCRF	$\mathrm{SVM}^{\mathrm{HHMM}}$
Protein-Location	69.4	72.4	**75.9**
Gene-Disorder	74.9	78.9	**80.5**
Protein-Protein*1	67.3	70.2	**71.8**
Protein-Protein*2	71.2	no result	**73.5**

表 5 実データに対する訓練の計算に要した時間（秒）．Protein-Protein*1 の N は 1500 である．

Table 5 Training time for real data (sec). N for Protein-Protein*1 is 1500.

	HHMM	HHCRF	$\mathrm{SVM}^{\mathrm{HHMM}}$
Protein-Location	3.029×10^1	3.409×10^4	2.300×10^2
Gene-Disorder	3.420×10^1	3.957×10^4	3.454×10^2
Protein-Protein*1	6.962×10^2	3.333×10^5	1.002×10^3

5. 考　　察

人工データを用いた実験 1 及び実データを用いた実験 2 では階層時系列モデルによるラベル付け実験を行った．実験 1 の訓練集合サイズが小さく一次マルコフ性をもつデータである場合のみ生成モデルである HHMM が最も高いラベル付け精度を示したが，その他では $\mathrm{SVM}^{\mathrm{HHMM}}$ は他のモデルよりも高いラベリング精度を示した．また，実験 2 では全てのデータセットに対し $\mathrm{SVM}^{\mathrm{HHMM}}$ が最も高いラベル付け精度を示した．これは学習においてマージンの最大化を行う $\mathrm{SVM}^{\mathrm{HHMM}}$ と確率の最大化を行う HHMM，HHCRF の差が現れた結果であると考えられる．HHCRF のラベル付け精度も同様に，実験 1 の訓練集合サイズが小さく一次マルコフ性をもつデータである場合以外では HHMM よりも高い．これは生成モデルと識別モデルの差が現れた結果であると考えられる．一方，実験 1 の結果から，$\mathrm{SVM}^{\mathrm{HHMM}}$ はチューニングパラメータ C の値によってラベル付け正解率が大きく変化してしまうことが示された．$\mathrm{SVM}^{\mathrm{HHMM}}$ の性能をより発揮させるためにはチューニングパラメータの適切な設定が重要である．

本研究では，上記 2 種類の実験において，訓練集合サイズが小さく一次マルコフ性をもつデータである場合を除き，$\mathrm{SVM}^{\mathrm{HHMM}}$ の優位性及び生成モデルに対する識別モデルの優位性を示すことができた．玉田ら

の文献[5]では，生成モデルであるHHMMと識別モデルであるHHCRFの比較を行っており，訓練集合サイズが大きい場合，ラベル付けに関して識別モデルHHCRFの方がより高い精度をもつという同じ結論を得ている．

6. むすび

本研究では階層隠れマルコフモデル(HHMM)に対応する識別モデルである$\mathrm{SVM}^{\mathrm{HHMM}}$を提案した．$\mathrm{SVM}^{\mathrm{HHMM}}$に関する学習問題の設定を行い，フィーチャベクトルと予測関数を設計し，マージン最大化に基づくパラメータ学習法を階層モデルへ導入することで$\mathrm{SVM}^{\mathrm{HHMM}}$を実装した．$\mathrm{SVM}^{\mathrm{HHMM}}$の特長は，学習においてマージンの最大化を行うこと，階層モデルであること，そして識別モデルであることの三つである．

人工データと実データを用いたラベリング実験では，訓練サイズが小さくかつ訓練データが一次マルコフ性をもつデータである場合を除き，$\mathrm{SVM}^{\mathrm{HHMM}}$はHHMM，HHCRFよりも高い精度でラベル付けできることが確認できた．

玉田らの文献[5]では隠れ状態がある階層時系列モデルを扱っているが，本研究では隠れ状態がない階層的時系列モデルを扱った．現在の$\mathrm{SVM}^{\mathrm{HHMM}}$では隠れ状態を扱うことができない．一方，Yuらは構造化SVMに隠れ状態を取り入れたモデルを提案した[13]．今後の$\mathrm{SVM}^{\mathrm{HHMM}}$に関する課題として，階層モデルにおける隠れ状態表現の導入と非線形カーネルの導入が挙げられる．

文　献

[1] J.D. Lafferty, A. McCallum, and F.C.N. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” Proc. 18th International Conference on Machine Learning, pp.633–723, 2001.

[2] S. Fine, Y. Singer, and N. Tishby, “The hierarchical hidden Markov model: Analysis and applications,” Machine Learning, vol.32, no.1, pp.41–62, 1998.

[3] K. Murphy and M. Paskin, “Linear time inference in hierarchical HMMs,” Advances in Neural Information Processing Systems, vol.14, pp.833–840, 2001.

[4] T. Sugiura, N. Gotou, and A. Hayashi, “A discriminative model corresponding to hierarchical HMMs,” Proc. 14th International Conference on Intelligent Data Engineering and Automated Learning, pp.375–384, 2007.

[5] 玉田寛尚，林　朗，末松伸朗，岩田一貴，“階層隠れCRF，”信学論（D），vol.J93-D，no.12，pp.2610–2619，Dec. 2010.

[6] Y. Altun, I. Tsochantaridis, and T. Hofmann, “Hidden Markov support vector machines,” Proc. 20th International Conference on Machine Learning, pp.3–10, 2003.

[7] N. Nguyen and Y. Guo, “Comparisons of sequence labeling algorithms and extentions,” Proc. 24th International Conference on Machine Learning, pp.681–688, 2007.

[8] A.J. Viterbi, “Error bounds for convolutional codes and an asymptotically optimal decoding algorithm,” IEEE Trans. Inf. Theory, vol.13, no.2, pp.260–269, 1967.

[9] L. Rabiner and B.H. Juang, 音声認識の基礎（下）6章隠れマルコフモデルの理論と実現法，NTTアドバンステクノロジー，1995.

[10] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, “Large margin methods for structured and interdependent output variables,” J. Machine Learning Research, vol.6, pp.1453–1484, 2005.

[11] M. Skounakis, M. Craven, and S. Ray, “Hierarchical hidden Markov models for information extraction,” Proc. 18th International Joint Conference on Artificial Intelligence, pp.427–433, 2003.

[12] T. Joachims, T. Finley, and C. Yu, “Cutting-plane training of structual SVMs,” Machine Learning, vol.77, no.1, pp.17–59, 2009.

[13] C. Yu and T. Joachims, “Learning structual SVMs with latent variables,” Proc. 26th Annual International Conference on Machine Learning, no.147, pp.1169–1176, 2009.

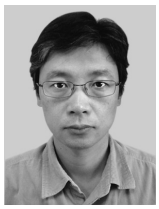
（平成24年3月14日受付，9月5日再受付）

金子　悟士　（学生員）

平22広島市立大・情報科学卒．平24同大大学院情報科学研究科博士前期課程了．

林　　朗　（正員）

昭49京大・理・数学卒，同年，日本IBM（株）入社．昭63ブラウン大学計算機科学科修士課程了．平3テキサス大学オースティン校計算機科学科博士課程了．同年，九州工業大学情報工学部客員助教授．平6広島市立大学情報科学部教授．現在は，同大大学院情報科学研究科教授．人工知能学会，情報処理学会，AAAI，ACM，IEEE各会員．



末松　伸朗　（正員）

昭 63 九大・理・物理卒．平 2 同大大学院修士課程了，同年，（株）富士通研究所入社．平 6 より広島市立大学情報科学部助手．現在は，同大大学院情報科学研究科准教授．博士（工学）．人工知能学会，情報処理学会，日本認知科学会各会員．

岩田　一貴　（正員）

平 12 名工大・工・知能情報システム卒．平 14 同大大学院工学研究科博士前期課程了．平 17 京大大学院情報学研究科博士後期課程了．平 14〜17 日本学術振興会特別研究員 DC1．平 17 広島市立大学情報科学部助手．現在は，同大大学院情報科学研究科講師．平 17 IEEE 関西支部 Student Paper Award 受賞．情報理論とその応用学会，IEEE 各会員．