隠れマルコフモデルによるデータストリームのモニタリング手法

藤原靖宏† 櫻井保志‡ 喜連川優†
東京大学生産技術研究所† ＮＴＴコミュニケーション科学基礎研究所‡

1. はじめに

本研究では以下の問題に取り組んだ．

Given: 隠れマルコフモデルの集合，
データストリームの長さ n のサブシーケンス
$X=(x_1,\cdots,x_n)$

Find: X に対して最も尤度の高いモデル
なおここで x_n は最も新しいシーケンスの値とする．

本問題の応用例としては交通量の監視[1]やコンピュータの不正検出が挙げられる [2]．
提案手法には以下のような特長がある
・高速：大規模なデータに対して高速に探索
・正確：最も尤度の高いモデルを正確に探索

2. 前準備

HMM は以下の要素で構成される．表１に主な記号とその定義を示す．

初期状態確率：$\alpha=\{\alpha_i\}$，時刻 1 において状態が u_i である確率 $(i=1,\cdots,m)$．

状態遷移確率：$\beta=\{\beta_{ij}\}$，時刻が 1 つ進んだときに状態 u_i から状態 u_j への遷移する確率．

シンボル出力確率：$\gamma=\{\gamma_i(v_j)\}$，状態 u_i においてシンボル v_j を出力する確率 $(j=1,\cdots,s)$．

シーケンス X の尤度 Φ は以下のように Viterbi アルゴリズムにより計算される．

$$\Phi=\max_{1\le i\le m}(\phi_{in})\quad\begin{cases}\phi_{it}=\max(\phi_{j(t-1)}\cdot\beta_{ji})\cdot\gamma_i(x_t),(2\le t\le n)\\ \phi_{i1}=\alpha_i\cdot\gamma_i(x_1),(t=1)\end{cases}$$

Viterbi アルゴリズムでは確率を図 1 のように状態を縦軸に，時間を横軸に並べたときに構成されるトレリス構造から計算する．Viterbi アルゴリズムの計算量 $O(nm^2)$ となる．

3. 提案手法

提案手法は以下の３つのアイデアによって構成される．なお紙幅の関係からすべての証明は省略したが，手法の詳細等は文献[3]に述べられている．

3.1 トレリス構造の縮退

トレリス構造が大きい場合 Viterbi アルゴリズムは高い計算コストを必要とする．そのため状態をクラスタリングし，結合することによって状態の数を削減する．この状態数の削減により，トレリス構造は縮退され，高速に近似尤度を計算することが可能になる．トレリス構造を縮退させるために粒度 g が与えられたとき，m 個の状態は m/g 個に削減される．この結果，尤度の計算量は

x_t	シーケンス X における時刻 t の値
u_i	隠れマルコフモデルにおける i 番目の値
n	シーケンス X の長さ
m	状態の個数
$\alpha=\{\alpha_i\}$	状態 u_i の初期状態確率
$\beta=\{\beta_{ij}\}$	状態 u_i から u_j への状態遷移確率
$\gamma=\{\gamma_i\}$	状態 u_i におけるシンボル v_j のシンボル出力確率
Φ	正確な尤度
Ψ	近似尤度

表 1. 主な記号とその定義

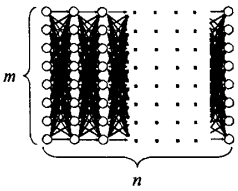

図 1. トレリス構造

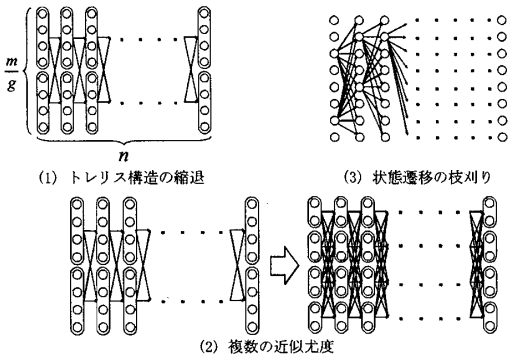

図 2. 提案手法の概要

$O(nm^2/g^2)$ に低減化される．

状態 u_i を以下の特徴量 F_i を用いて k-means 法によりクラスタリングする．

$$F_i=(\alpha_i,\beta_{i1},\cdots,\beta_{im},\beta_{1i},\cdots,\beta_{mi},\gamma_i(v_1),\cdots,\gamma_i(v_s))$$

クラスタリングしてできた新しい状態の確率は各要素のうち最大の確率を用い定義する．例えば状態 u_i と u_j が同じクラスタになった場合，新しい初期状態確率は最大の確率値から $\alpha'=\max(\alpha_i,\alpha_j)$ となり，状態遷移確率は $\beta'_{ii}=\max(\beta_{ii},\beta_{ij},\beta_{ji},\beta_{jj})$，$\beta'_{ik}=\max(\beta_{ik},\beta_{jk})$，$\beta'_{ki}=\max(\beta_{ki},\beta_{kj})$，（ただここで $k\neq i,j$ とする）となり，同様にシンボル出力確率は $\gamma'=\max(\gamma_i(v_k),\gamma_j(v_k))$ となる．

近似尤度 Ψ は結合後の状態数 $m'(=m/g)$ と確率 α'，β'，γ' を用いての以下のように計算する．

$$\Psi=\max_{1\le i\le m'}(\varphi_{in})\quad\begin{cases}\varphi_{it}=\max(\varphi_{j(t-1)}\cdot\beta'_{ji})\cdot\gamma'_i(x_t),(2\le t\le n)\\ \varphi_{i1}=\alpha'_i\cdot\gamma'_i(x_1),(t=1)\end{cases}$$

近似尤度 Ψ は正確な尤度 Φ の上限値となる．

[定理 1] *状態を結合後のモデルとシーケンスが与えられたとき，$\Psi\ge\Phi$ の関係が成立する．*

3.2 複数の近似尤度

トレリス構造を縮退することで計算される近似尤度により高速な探索が可能になるが，近似計算には近似精度と計算時間のトレードオフが存在する．すなわちトレリス構造を過度に縮退させると近似尤度の計算コストは小さ

Efficient data stream monitoring for HMM.
†Yasuhiro Fujiwara (contact author), NTT Cyber Space Lab.
‡Yasushi Sakurai, NTT Communication Science Labs.
*Masaru Kitsuregawa, University of Tokyo
Email: fujiwara.yasuhiro@lab.ntt.co.jp

くなるが，近似尤度の上限値は大きくなってしまう．
そのため探索処理において徐々にトレリス構造のサイズを大きくしていき，近似尤度の精度を上げていく．
提案手法では$h+1$個の粒度を用いる．レベル$i(0 \le i \le h)$の近似においては粒度$g_i = 2^i$を用いて，m個の状態を$\lfloor m/g_i \rfloor$個に結合する．最も粗い粒度はg_hであり，g_0はオリジナルのモデルである．レベルが上がるにつれてg_iは指数的に小さくなり，近似尤度の精度は上がっていく．

3.3 状態遷移の枝刈り

トレリス構造を縮退することにより高速な探索が可能になるが，正確な探索結果は近似尤度からは求めることができない．そのため近似尤度で求めた候補に対して厳密な尤度を求める．しかし厳密な尤度を計算するには大きなコストが必要なため，提案手法ではトレリス構造において不要な計算を省いて尤度計算を高速に打ち切る．
状態u_iにおける時刻tの尤度計算において遷移を枝刈りするための推定値e_{it}を以下のように導入する．

$$e_{it} = \begin{cases} \phi_{it} \cdot (\beta_{\max})^{n-t} \cdot \prod_{j=t+1}^{n} \gamma_{\max}(x_j), (1 \le t \le n-1) \\ \phi_{in}, (t = n) \end{cases}$$

ここで$\beta_{\max}$と$\gamma_{\max}$は以下のように状態遷移確率とシンボル出力確率の最大値である．

$$\beta_{\max} = \max_{ij}(\beta_{ij}), \quad \gamma_{\max} = \max_{t}(\gamma_i(v))$$

推定値e_{it}はトレリス構造においてパスが時刻tで状態u_iを通るとしたときの尤度の上限値となる．

[定理 2] *時刻tで状態u_iを通過するパスに対しては，時刻nで状態u_jの尤度に対して$e_{it} \ge \phi_{jn}$の関係が成り立つ．*

3.4 探索アルゴリズム

データストリームは時々刻々と変化していくが，一つの値が新たに加わってもその変化の割合はそれほど多くはないことに着目した探索を行う．
まずに前時刻の解を候補として正確な尤度を計算する．そして前時刻に計算した粒度より粒度を1段階荒くし近似尤度を計算する．もしモデルが枝刈りできなければ1段階粒度を細かくして近似尤度を計算する．この処理を繰り返して最も正確な尤度が高いモデルを解として出力する．

4. 理論的解析

提案手法によって得られる解について定理 1 及び定理 2 から以下のことが成り立つ．

[定理 3] *提案手法はモデルを探索するときに解の厳密性を保証する．*

また Viterbi アルゴリズム においては以下の定理が成り立つ．

[定理 4] *Viterbi アルゴリズムは$O(m^2 + ms)$のメモリ量と$O(nm^2)$の計算時間を要する．*

また提案手法は複数の近似尤度を用いるが，それらの粒度は指数的に減少するため以下の定理が成り立つ．

[定理 5] *提案手法は$O(m^2 + ms)$のメモリ量を要し，少なくとも$O(n)$の，多くとも$O(nm^2)$の計算時間を要する．*

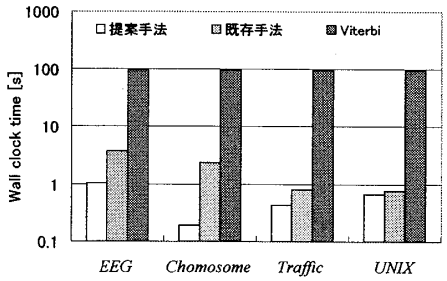

図3. データストリームのモニタリング処理時間

5. 評価実験

実験は CPU が Intel Quad Core の 3.33GHz，メインメモリが 32GB のマシンで行った．実験では以下の標準的な4つのデータセットを用いた．

・EEG：このデータセットは EEG とアルコール依存症の関係を調べるために行われた大規模な実験で得られたものであり，UCI のウェブサイト[4]からダウンロードすることができる．

・Chromosome：人間の第 2, 18, 21, 22 番目の染色体のデータであり，NCBI のウェブサイト[5]から得ることができる．

・Traffic：このデータセットは UCI のウェブサイト[4]から得たフリーウェイ交通量の測定値である．

・UNIX：このデータセットは UNIX の操作履歴であり，UCI のウェブサイト[4]から得ることができる．

本実験では提案手法と既存研究[6]において述べられている手法，および Viterbi アルゴリズムと比較を行った．それぞれのアルゴリズムの実験結果を「提案手法」,「既存手法」,「Viterbi」として図 3 に示す．実験において状態数を 100 とし，モデルの数を 10,000 とした．

本研究における提案手法はその他の手法に対して優位であり，特に Viterbi アルゴリズムに対しては４９０倍高速であることが確認された．

6. まとめ

本研究ではデータストリームを隠れマルコフモデルによって効率的にモニタリングする手法について取り組んだ．複数の実データを用いて検証したところ既存の手法より最大 490 倍まで高速化が可能であることを確認した．

7. 参考文献

[1] P. Bickel et al., Traffic flow on a freeway network, In Workshop on Nonlinear Estimation and Classification.
[2] T. Lane, Hidden Markov Models for human/ computer interface modeling, IJCAI-99 Workshop.
[3] Y. Fujiwara et al, Fast likelihood search for Hidden Markov Models, ACM Transactions on Knowledge Discovery from Data (TKDD).
[4] http://archive.ics.uci.edu/
[5] http://www.ncbi.nlm.nih.gov
[6] Y. Fujiwara et al, SPIRAL: Efficient and exact model identification for Hidden Markov Models, In Proc. of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2008).