3C4-5

オンライン処理による多次元時系列データのモチーフ長を考慮したモチーフ発見

Online Multi-Dimensional Motif Discovery with Automatic Detection of Motif Length

鷹取 留亞子　　上原 邦昭
Ruako Takatori　　Kuniaki Uehara

神戸大学大学院システム情報学研究科計算科学専攻
Graduate School of System Informatics, Kobe University

Time series motif is previously unknown, frequently appearing pattern in a time series data. Motif can be used for various data mining tasks. Method of the motif discovery can be devided into two types; Batch approach and Online approach. Batch approach stores all streams in databases and detects motifs from entire data. Online apprach detects motifs from most recent data and discards the oldest stream history. More useful and realistic method is online approach, because many time series datas come one after another and continue forever. In this paper, we propose an online motif discovery algorithm to extract a motif from multi-dimensional time series data. Additionally, our system can decide the motif length automatically in online method. We show the result of our work and examine about some problems.

1. はじめに

近年，様々な時系列データからデータマイニングが行われている．時系列データの特徴を捉えるデータマイニング手法の一つとして，モチーフの発見がある．モチーフとは，時系列データの中に繰り返し出現する特徴的な部分時系列である．モチーフを発見することにより，様々な異常を検知したり，特徴を捉えることができると言われている．

モチーフ発見には様々な課題が存在する．例として，多次元データの扱い方，パラメータの決定方法等が挙げられる．また，データの発生に合わせてモチーフを得る場合はオンライン処理と呼ばれるが，使用メモリ量や計算時間の制約が存在するため，さらにモチーフ発見は困難となる．本研究では，多次元時系列データを対象とし，オンライン処理でモチーフ発見を行う手法，およびモチーフ長を自動的に決定する手法を提案する．

2. 背景と関連研究

2.1 モチーフの概要

モチーフとは，時系列データの中で繰り返し出現する特徴的な部分時系列である．時系列データ $T = x_1, x_2, \ldots, x_n$ が存在するとき，時刻 p から始まる長さ m の部分時系列を $C_p = x_p, x_{p+1}, \ldots, x_{p+m-1} (m \leq n, 1 \leq p \leq n-m+1)$ と表す．部分時系列 C_c と類似した部分時系列の集合は，$M_c = \{C_p | dist(C_c, C_p) < R\}$ と表される．$dist(C_p, C_q)$ は 2 つの部分時系列 C_p, C_q 間の距離を表し，R はモチーフ半径と呼ばれる．一般に，最も要素数の多い M_c がモチーフ集合として発見される．モチーフ集合の例を図 1 の (a) に示す．

2.2 バッチ処理・オンライン処理

モチーフの発見手法には，主にバッチ処理とオンライン処理がある．バッチ処理では，データを全て取得した後にモチーフ発見を行う．したがって，豊富な計算リソースの使用が前提となり，計算の精度向上や，巨大なデータを対象とする場合の高速化等を目的とした研究が多く行われている．

連絡先: 鷹取 留亞子，神戸大学システム情報学研究科計算科学専攻，takatori@ai.cs.kobe-u.ac.jp

一方，オンライン処理では，データの発生と同時にモチーフ発見を行う．したがって，1 つのデータが発生してから次のデータが発生するまでの限られた時間で処理を行う必要がある．また，データは無限に続くという前提を持つ．Mueen と Keogh [Mueen 10] はこの前提に基づいて，オンライン処理におけるモチーフ発見を以下のように再定義している．

まず，時系列データから長さ w の最新の部分時系列を Sliding Window とし，Sliding Window に含まれるデータのみをモチーフ探索の対象としている．Sliding Window はデータの生成に合わせて時系列上をスライドし，最新のデータを取り込み最も古いデータを破棄する．また，最も $dist(C_p, C_q)$ が小さい対 C_p, C_q がモチーフ対となる．すなわち，対象とするモチーフを集合ではなく対であるとしている．オンライン処理におけるモチーフ対の例を図 1 の (b) に示す．Lam ら [Lam 11] は Mueen らの手法のデータ構造を改良し，さらに空間計算量の削減と高速化を行う *nMotif* アルゴリズム等を提案している．

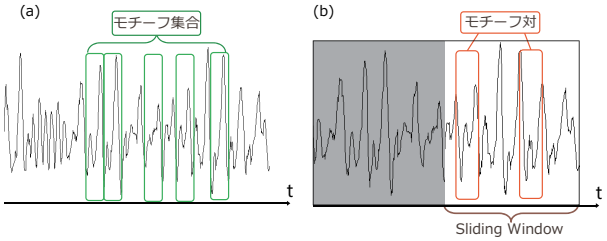

図 1: (a) モチーフ集合 (b) モチーフ対

2.3 多次元データ

多次元データからモチーフを発見する手法として，様々な方法がある．Lin ら [Lin 02] や Mueen ら [Mueen 10] は，各次元から独立してモチーフを発見し，データの特性や実験目的に合わせた処理を行うべきであるとしている．Minnen ら [Minnen 07a] や Vahdatpour ら [Vahdatpour 09] は，各次元からそれぞれモチーフの抽出を行った後に，別の次元のモチーフ同士を組み合わせる方法を提案している．Tanaka ら [Tanaka 05] は，主成分分析によって次元を統合し，得られた主成分からモチーフを発見する方法を提案している．Kurasawa ら [Kurasawa 12]

は，オンライン処理で同じモチーフを異なる次元から発見する方法を提案している．このように，様々なアプローチが存在するが，データの特性やモチーフ発見の目的に合わせて手法を選択する必要がある．

2.4 パラメータの決定

モチーフ発見で決定すべきパラメータとして，探索する部分時系列の長さ m，バッチ処理におけるモチーフ半径 R，Sliding Window の幅 w 等が挙げられる．これらのパラメータは，通常，実験を繰り返すことによって決定される．しかし，パラメータは複数存在し，組み合わせが膨大なものとなるため，最適なモチーフを得るためには非常に多くの実験を行わなければならない．したがって，パラメータを自動的に決定する手法が必要であると言われている．

バッチ処理については，パラメータ決定の自動化について，様々な研究が行われている．Tanaka ら [Tanaka 05] は最小記述長原理（Minimum Description Length Principle;MDLP）を用いて部分時系列長の自動的な決定を可能としている．また，そのアルゴリズムを応用し，異なる長さの類似するサブシーケンスを 1 組のモチーフ集合として発見する手法を提案している．Minnen ら [Minnen 07b] は，randam projection という符号化手法を用い，最適なモチーフ半径を決定する手法を提案している．しかしながら，オンライン処理については，パラメータを自動的な決定方法についての研究は存在しない．Mueen らは，部分時系列長の長さ m や Sliding Window の幅 w と計算量との関係について考察し，必要なメモリ量や計算速度を考慮して，パラメータの決定をすべきであるとのみ言及している．

3. 提案手法

本研究では．オンライン処理によって多次元時系列データの主成分分析を行い，最小記述長原理を用いてモチーフ長を決定する，モチーフ発見アルゴリズム *OnMDLMotif* を提案する．なお，データ構造の更新には Lam らの *nMotif* アルゴリズムを用い，新しいデータが発生する度に 1 回の更新を行う．

3.1 オンライン主成分分析

主成分分析は，多変量で表されるデータから，特徴を表す指標である主成分を発見する手法である．長さ n の d 次元データ $T = \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ に主成分分析を適用する場合，まず T の各次元 j について平均 $\bar{x_j}$ をもとめ，平均ベクトルを $\bar{\boldsymbol{x}} = (\bar{x_1}, \bar{x_2}, \ldots, \bar{x_d})$ とする．また，T から平均ベクトルを引いたデータを $\hat{T} = \hat{\boldsymbol{x}_1}, \hat{\boldsymbol{x}_2}, \ldots, \hat{\boldsymbol{x}_n}$ とする．次に，$\hat{T}$ の分散共分散行列 $\Sigma_{\hat{T}}$ をもとめる．この共分散行列から求められる固有値を $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d$ とする．また，固有値 λ_i に対する固有ベクトルを $[e_{1\lambda_i} e_{w\lambda_i} \ldots e_{m\lambda_i}]$ とする．これらの固有値と固有ベクトルを用い，時刻 t における第 i 主成分は，$p_{t,\lambda_i} = e_{1,\lambda_i}\hat{x_1} + e_{2,\lambda_i}\hat{x_2} + \ldots + e_{d,\lambda_i}\hat{x_d}$ として得られる．

本研究では，オンライン処理で主成分分析行うため，一度の更新毎に，新しく取得したデータに対して主成分に変換する．なお，主成分分析に必要な平均や分散共分散行列は，本来データ全体から集計されるものであるが，オンライン処理ではデータ全体というものが存在しないため，本研究では計算時点より前のデータから求めた平均や共分散等を累計することにより対応している．また，主成分分析を行うと入力されたデータの次元数と同数の主成分が得られるが，主成分は入力データの特徴をよく表す順に並んでいるため，本研究では第 1 主成分のみを用いる．

3.2 最小記述長原理

最小記述長原理 (Minimum Description Length Principle;MDLP) は確率モデルの最適化原理である．最小記述長原理は「与えられたデータを，モデル自身の記述長も含めて最も短く符号化できるような確率モデルが最良のモデルである」という考えに基づく原理である．

Tanaka ら [Tanaka 05] は，最小記述長原理によって最もデータの特徴を表すモデルを見つけられることに着目し，バッチ処理で最小記述長原理を用いてモチーフを発見する手法を提案している．最小記述長原理を用いることにより，探索するモチーフの長さを指定することなく，最良のモチーフを発見することが可能となる．また，最小記述長原理を適用するためには，データを符号列として表現する必要があるため，Tanaka らは SAX による符号化を行っている．本研究では，Tanaka らの手法に基づいて SAX による符号化を行い，記述長を最小化するモチーフの探索する．SAX（Symbolic Aggregate approXimation）[Lin 03] は符号化手法の 1 つであり，距離関係を維持した状態で実数値データを符号に変換できる．SAX の概要図を図 2 に示す．

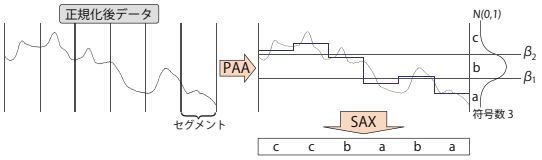

図 2: SAX による符号化の概要

まず，時系列データの正規化を行った後，セグメントに区切り平均を取る処理（Peacewise Aggregate Approximation;PAA）を行う．正規化された時系列データは平均 0，分散 1 の正規分布に従うことに基づいて，全ての符号の出現確率が等しくなるように分割点 β を定め，PAA を行ったデータによって符号に変換する処理（SAX）を行う．なお，オンライン処理では，一回の更新毎に符号化を行う．また，正規化を行う時点で平均と分散が必要となるため，主成分分析の場合と同様に，計算時点より前のデータから求めた平均や分散等を累計することにより対応している．

次に，得られた符号列に対し，最小記述長原理を適用する．部分符号列のパターン SC に対して，SC の長さを n_p，SC に使用される符号種数を s_p とすると，パターン SC の記述長 $DL(SC)$ は以下のように表される．

$$DL(SC) = \log_2 n_p + n_p \log_2 s_p \tag{1}$$

さらに，符号列 $\tilde{C}$ の中にパターン SC が q 回出現するとき，出現する SC を全て 1 つの符号に変換する．変換後の符号列の長さを n_a，符号列に使用される符号種数を s_a とすると，変換後の $\tilde{C}$ の記述長 $DL(\tilde{C}|SC)$ は以下のように表される．

$$DL(\tilde{C}|SC) = \log_2 n_a + n_a \log_2 (s_a + q) \tag{2}$$

記述長関数 $MDL(\tilde{C}|SC)$ は $DL(\tilde{C}|SC)$ と $DL(SC)$ の和となる．これを最小化するパターン SC が最小記述長原理を満たす最良のモデルとなり，最良のモチーフとなる．オンライン処理では，新しく得られた符号を含む部分符号列 SC_l を，長さ徐々に l を変化させながら生成する．生成した SC_l に対して MDL の計算を行う．

SC_l の生成と MDL の適用例を図 3 に示す．図中では，新しく得られた符号である “A” を含むように部分符号列

$SC_3, SC_4, \ldots$ を生成する．例として SC_3 =“ABA” に着目すると，符号種数は 2 であり，符号列全体の中に 4 回出現する等，各変数がもとめられるため，MDL を計算することができる．Sliding Window 内の部分符号列について，MDL を最小とするようにデータ構造を更新し，最も小さい MDL を持つ SC_l がモチーフとなる．

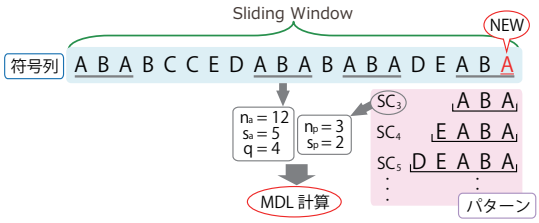

図 3: SC_l の生成と MDL の適用例

バッチ処理では，全ての考えうる部分時系列について，それと類似する部分時系列の集合をもとめるため，モチーフ集合を得ることが可能である．しかしオンライン処理では，データが取得に合わせて次々と変わっていくという前提と計算量の制約により，全ての考えうる部分時系列の集合をもとめることは困難である．そのため，オンライン処理の先行研究 [Mueen 10, Lam 11] では，部分時系列同士のユークリッド距離が最も近い対のみを，モチーフ対として発見している．

提案手法では，MDL の計算を行うために符号化を行い，部分符号列同士の一致によって類似度を計算している．そのため，先行研究で用いられたユークリッド距離と比べて，類似度の計算にかかる時間は大幅に少ない．また，MDL の計算過程で，図 3 のようにパターンを符号列全体から探索している．この時，探索されたパターン全ての位置を記録することにより，ある MDL の値に対応した部分符号列の集合を扱うことができる．したがって，先行研究では不可能であったオンライン処理におけるモチーフ集合の発見が可能となる．

3.3 OnMDLMotif の拡張

OnMDLMotif の問題点として，完全に一致しないパターンは発見できないことが挙げられる．例えば，Sliding Window 内に完全に一致するパターンが存在しない場合，その更新ではモチーフが得られない．この問題点の解決のため，*OnMDLMotif* を改良した *OnMDLNearMotif* アルゴリズムを提案する．

OnMDLNearMotif アルゴリズムでは，まず，a と b，b と c 等の隣り合った符号を類似符号とする．また，パターン SC の符号を類似符号でそれぞれ置き換えた部分符号列のうち，SC とのハミング距離 d_{ham} がパターン長の半分より小さいものを類似パターンとする．

次に，MDL の計算過程で，パターン SC と同時に類似パターンも探索する．類似パターンを発見した場合は，SC とのハミング距離 d_{ham} の合計もとめて h とする．類似パターンの記述長を考慮するため，MDL の計算は式 1 を以下のように変形して行っている．

$$DL(SC) = \log_2 (n_p + h) + (n_p + h) \log_2 s_p \qquad (3)$$

4. 実験

本実験では，まず，提案手法によって多次元データから自動的にパラメータを決定したモチーフ発見が可能であることを示す．次に，提案手法である *OnMDLMotif*，*OnMDLNearMotif* について比較する．最後に，既存手法と提案手法の計算時間について比較する．本実験のため，*OnMDLMotif*，*OnMDLNearMotif* を実装した．また，比較手法として，固定長のモチーフ発見を行う Lam ら [Lam 11] の *nMotif* アルゴリズムを使用する．実験には，3 次元の動作に関する時系列データを用いた．サンプリング周波数 100Hz で取得されたセンサデータであり，周期的な波形と周期的でない波形やノイズが混在している．

4.1 実験結果

nMotif，*OnMDLMotif*，*OnMDLNearMotif* によって発見されるモチーフの一例を図 4 に示す．Sliding Window の幅を 400 として探索を行い，主成分から発見されたモチーフを太線で表示している．*nMotif* ではモチーフ長 m を 40 と設定した．*OnMDLMotif*，*OnMDLNearMotif* では，実験の結果それぞれ $m = 55$，45 となった．

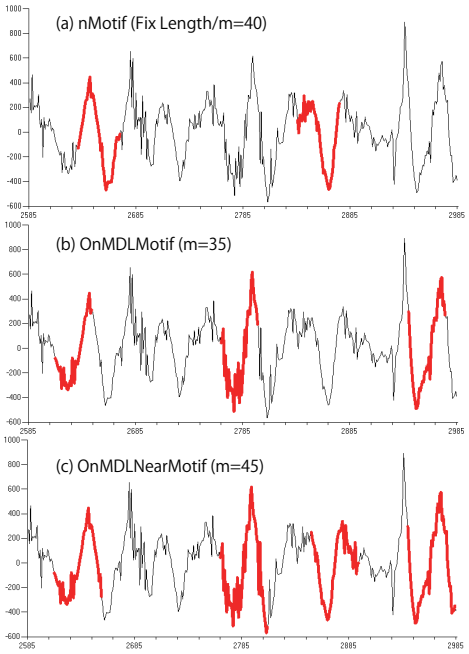

図 4: 実験結果

次に，各次元のデータに対して提案手法を用いた場合に得られるモチーフの長さを調べたところ，*OnMDLMotif* では 30 から 50，*OnMDLNearMotif* では 30 から 65 の様々な長さのモチーフを発見できた．最後に，各次元のデータに対して提案手法を用いた場合に得られたモチーフ集合の要素数を調べたところ，*OnMDLMotif* では 2 から 4，*OnMDLNearMotif* では 2 から 7 の，様々な要素数のモチーフ集合が発見できた．

4.2 *OnMDLMotif* と *OnMDLNearMotif* の比較

OnMDLNearMotif では探索対象とするパターンの種類が多くなるため，*OnMDLMotif* ではモチーフが発見されない部分においても，モチーフを発見できることが考えられる．*OnMDLMotif* ではモチーフが発見されず，*OnMDLNearMotif* では $m = 30$ のモチーフが発見された例を図 5 に示す．

4.3 計算時間の比較

まず，1 度の更新毎の平均計算時間について，図 6 の（a）に示す．*OnMDLNearMotif* と *OnMDLNearMotif* は，符号化の関係から，5 つのデータが取得される度に更新している．したがって，1 つのデータに必要な時間は更新時間の 1/5 である．このことを考慮した比較を行うため，提案手法の平均計算時間を 1/5 に換算したグラフを図 6 の（b）に示す．この結

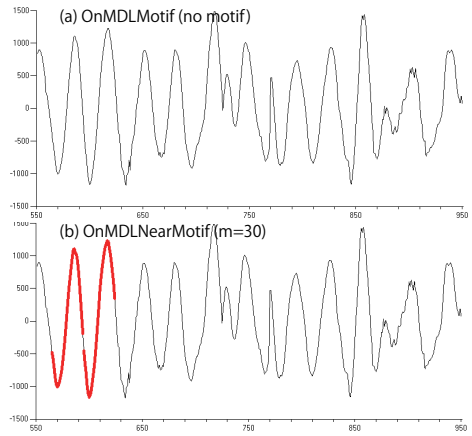

図 5: 提案手法の比較

果から，幅 1,000 までの Sliding Window では，提案手法が既存手法より高速に計算を行うことができることが分かる．また，今回使用したデータのサンプリング周波数は 100Hz であるため，データは 0.01s 毎に発生する．グラフより，提案手法はデータの発生に対し，余裕を持ってモチーフ発見を行えることが分かる．

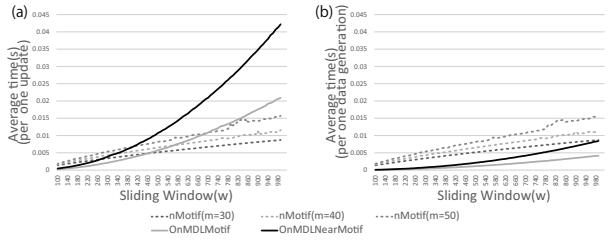

図 6: 計算時間 ((a) 1 回の更新毎 (b) 1 つのデータ毎)

5. まとめと考察

本研究では，多次元時系列データに対してオンライン処理によるモチーフ発見を行う手法として，主成分分析による次元縮約と最小記述長原理を用いたモチーフ長の最適化を提案した．結果より，2 つの問題点が考えられる．

まず，固定長のモチーフ発見を行う既存手法と，提案手法を比較する．図 4 の結果を人の目で見た場合，既存手法によるモチーフの方がより類似度が高いと判断できる．これは，提案手法では，データが符号化された状態で類似度の判定が行われることが原因であると思われる．また，提案手法によって得られるモチーフは，最小記述長原理上における最適なモチーフであり，実際に有用なモチーフであるかどうかは不明である．対策としては，提案手法によって提案された部分時系列長を基準値とし，最適な結果を得られるように部分時系列長の調整を行う手法の提案が考えられる．

次に，オンライン化の方法について検討する．本研究では，主成分分析と符号化についてオンライン化を行う際，データの平均等の累計を取るという方法を用いた．しかし，計算時点までの全てのデータに対しての累計を利用しているため，データの特性の変化に対する追従性に問題があると考えられる．対策としては，主成分分析や符号化に用いる累計値の範囲を Sliding Window 内とすることが考えられる．また，*OnMDLMotif* の計算時間の内訳を調べると，計算時間のほとんどはデータ構造の更新に必要とされていた．主成分分析や符号化に，より時間を割くことが可能なため，毎回の更新で Sliding Window 内の累計を取る手法の提案が考えられる．

参考文献

[Kurasawa 12] Kurasawa, H., Sato, H., Nakamura, M., and Matsumura, H.: Online Top-k Similar Time-Lagged Pattern Pair Search in Multiple Time Series, in *Proc. of 23rd International Conference on Database and Expert Systems Applications*, pp. 432–441 (2012)

[Lam 11] Lam, H. T., Calders, T., and Pham, N.: Online discovery of top-k similar motifs in time series data, in *Proc. of the SIAM International Conference on Data Mining*, pp. 1004–1015 (2011)

[Lin 02] Lin, J., Keogh, E., Lonardi, S., and Pranav, P.: Finding motifs in time series, in *Proc. of the 2nd Workshop On Temporal Data Mining*, pp. 53–68 (2002)

[Lin 03] Lin, J., Keogh, E., Lonardi, S., and Chiu, B.: A symbolic representation of time series, with implications for streaming algorithms, in *Proc. of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, pp. 2–11 (2003)

[Minnen 07a] Minnen, D., Isbell, C., Essa, I., and Starner, T.: Detecting subdimensional motifs: An efficient algorithm for generalized multivariate pattern discovery, in *Proc. of the 7th IEEE International Conference on Data Mining*, pp. 601–606 (2007)

[Minnen 07b] Minnen, D., Starner, T., Essa, I. A., and Isbell Jr, C. L.: Improving activity discovery with automatic neighborhood estimation, in *Proc. of the 20th International Joint Conference on Artificial Intelligence*, Vol. 7, pp. 2814–2819 (2007)

[Mueen 10] Mueen, A. and Keogh, E.: Online discovery and maintenance of time series motifs, in *Proc. of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1089–1098 (2010)

[Tanaka 05] Tanaka, Y., Iwamoto, K., and Uehara, K.: Discovery of time-series motif from multi-dimensional data based on MDL principle, *Machine Learning*, Vol. 58, No. 2-3, pp. 269–300 (2005)

[Vahdatpour 09] Vahdatpour, A., Amini, N., and Sarrafzadeh, M.: Toward unsupervised activity discovery using multi-dimensional motif detection in time series, in *Proc. of the 21st International Joint Conference on Artificial Intelligence*, Vol. 9, pp. 1261–1266 (2009)