2C1-5

時系列パターン分析におけるアイテム間制約の効果

The Effect of Constraints among Items for an Sequential Pattern Analysis

櫻井 茂明*1 Shigeaki Sakurai　早川 ルミ*1 Rumi Hayakawa　岩崎 秀樹*1 Hideki Iwasaki

*1東芝ソリューション (株)IT 研究開発センター
Advanced IT Research Center, Toshiba Solutions Corporation

This paper verifies the effect of constraints among items in the discovery task of sequential patterns. The constraints are introduced in order to discover characteristic sequential patterns from sequential data. They can be described as the combinations of items given by analysts. Each item is composed of an attribute and its attribute value. The discovery method can pick up only the patterns satisfying the constraints. This paper applies the constraints to practical sequential data. It evaluates the time and the number of discovered patterns. Lastly, it shows that the constraints can dramatically reduce the number of the patterns through the comparison with the method without the constraints.

1. はじめに

コンピュータ環境及びネットワーク環境の発展に伴って、多量のデータが簡便に収集、蓄積されるようになった。このため、これらデータを分析する技術が活発に研究開発されている。このような大規模データは近年益々巨大化しており、ビックデータと呼ばれる流行語が、IT 分野では生み出されている。多くの企業では、ビックデータの分析に向けた研究開発に取り組んでいる。

ビックデータは多様なデータによって構成されているものの、Twitter や YouTube の隆盛、スマートグリッドやスマートコミュニティへの期待の高まりに見られるように、時系列的に収集されるデータが多数存在しており、この傾向は益々加速するものと考えられる。時系列的に収集されるデータを分析する技術は、当初、統計学や信号処理の分野で研究がなされており、時系列的に与えられる数値データを対象としていた。1990 年代の後半に入ると、頻出パターンの発見問題は、時系列的に拡張され、時系列パターンを発見する技術が研究開発されるようになった。本技術によって、離散的なアイテム及びアイテムの集合が時系列的に並べられたデータから、特徴的なアイテムやアイテム集合の並びを発見することができる。時系列パターンの発見技術は、商品の購買履歴の分析にその端を発するが、近年は、Twitter から時系列的に抽出されるキーワード列に基づいたトレンド分析や、定期的に検査されるヘルスケアデータを離散化した時系列データの分析などに、その適用範囲が広がっている。また、より分析者のニーズに合った時系列パターンを発見するために、時系列パターン発見研究の初期において対象としていた、頻出性以外の基準に基づいた、特徴的な時系列パターンを発見することも求められるようになっている。

このような背景の下、我々のグループでは離散的な時系列データから、特徴的な時系列パターン発見する技術の研究開発に取り組んでいる。本論文では、分析者の背景知識を利用することにより、特徴的な時系列パターンを発見する方法 [3][4] に着目する。本手法は、分析者が指定したふたつ以上のアイテム間の関係 (アイテム間制約) を満たす時系列パターンのみを抽出することにより、特徴的な時系列パターンを効率よく発見することを可能としている。このアイテム間制約の効果を定量的に評価するために、実データを対象とした評価実験を実施し、アイテム間制約を利用した場合と利用しない場合とで、実験結果を比較し、その効果を定量的に検証する。

連絡先: 〒 183-8512 東京都府中市片町 3-22
東芝ソリューション (株)IT 研究開発センターアドバンストソリューション開発グループ
Tel:042-340-6628　E-mail:Sakurai.Shigeaki@toshiba-sol.co.jp 櫻井 茂明

2. 時系列パターンと時系列データ

本論文で対象とする時系列データとは、複数のアイテム集合が時系列的に並べられたアイテム集合の系列のことである。このとき、各アイテム集合には、同一のアイテムはせいぜいひとつしか含まれないという前提がおかれている。本論文では、複数の時系列データに現れる特徴的な部分時系列を時系列パターンとして発見する。時系列パターン発見における初期の研究 [1][2] では、特徴的かどうかの判定には、式 (1) で定義される支持度が利用されており、頻出する時系列パターンを特徴的な時系列パターンとして発見することを試みている。すなわち、与えられた部分時系列の支持度が、指定した最小支持度以上となる部分時系列を、特徴的な時系列パターンとして発見している。

$$\text{支持度}\ (s) = \frac{s\ \text{を含む時系列データ数}}{\text{時系列データ数}} \tag{1}$$

ただし、s は時系列パターンを表すとする。

また、相関ルールの発見問題の場合と同様に、発見された時系列パターンを、前提となる部分時系列パターン s_p と、結論となる部分時系列パターン $s - s_p$ に分けることにより、時系列パターンのルール化を行うことができる。このとき、ルールの良し悪しを測る基準としては、式 (2) によって定義される信頼度が利用されている。

$$\text{信頼度}\ (s|s_p) = \frac{s\ \text{を含む時系列データ数}}{s_p\text{を含む時系列データ数}} \tag{2}$$

3. 時系列パターンの発見法

前節で紹介した時系列パターンは、時系列パターンを構成するアイテムの個数が増えるにしたがって、支持度が単調に減

少するアプリオリ性を利用することにより、指定した最小支持度以上のすべての時系列パターンを効率よく発見することを可能としている。このアプリオリ性に基づいた時系列パターンの発見法としては、複数の方法 [1][2] が提案されているが、本論文では、発見済みの時系列パターンから、より大きな候補を生成することにより、すべての時系列パターンを発見する方法 (候補に基づいた方法)[1] により、特徴的な時系列パターンを発見する。以下においては、候補に基づいた方法を簡単に紹介する。

候補に基づいた方法では、時系列データを構成する各アイテムに対して、当該アイテムを含む時系列データの数を算出して支持度を計算し、その支持度が最小支持度以上となるすべてのアイテムの発見を行う。この発見されたアイテムが 1 次頻出アイテム集合となる。次に、発見された 1 次アイテム集合を組み合わせることにより、ふたつのアイテムで構成された候補を生成し、その支持度が最小支持度以上となる候補を 2 次頻出アイテム集合として発見する。同様に、2 次頻出アイテム集合を組み合わせることにより、3 つのアイテムで構成された候補を生成し、その支持度が最小支持度以上となる候補を 3 次頻出アイテム集合として発見する。このような、アイテム集合の成長を順次実施していくことにより、すべての頻出アイテム集合の発見を行う。ここまでの頻出アイテム集合の発見は、時系列データを単位に頻度を計算することを除いては、頻出パターンの発見法と同一の方法となっている。

候補に基づいた方法では、この頻出アイテム集合の中からふたつのアイテム集合を取り出して組み合わせることにより、ふたつのアイテム集合が時系列的に並んだ時系列パターンの候補 (2 次候補時系列パターン) を生成する。このとき、取り出したアイテム集合の並べ方を変えることにより、2 種類の候補が生成されることに注意する必要がある。候補に基づいた方法では、生成された候補を含む時系列データの支持度を算出し、その支持度が最小支持度以上となる場合に、当該候補を 2 次頻出時系列パターンとして判定する。同様に、2 次頻出時系列パターンを組み合わせることにより、3 つのアイテム集合が時系列的に並んだ 3 次候補時系列パターンを生成する。また、生成された候補の支持度を算出し、3 次頻出時系列パターンになるかどうかの判定を行う。以上のような系列の成長を繰り返していくことにより、すべての時系列パターンの発見を行う。

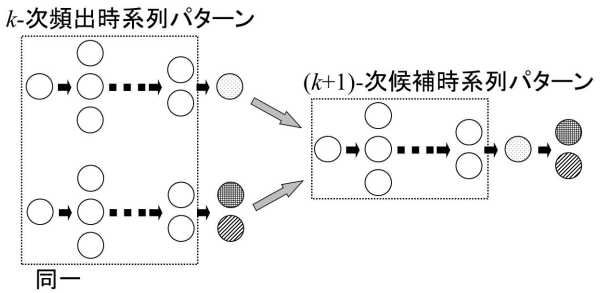


図 1: 候補時系列パターンの生成

図 1 は、k 次頻出時系列パターンから $(k+1)$ 次候補時系列パターンを生成する様子を示している。図においては、各丸がひとつのアイテムを表しており、同一の時刻に発生したとみなされるアイテムが矢印によって区切られている。また、同じ模様の丸は同一のアイテムであることを示している。図から分かるように、前方にある $k-1$ 個のアイテム集合が同一であり、最後尾のアイテム集合が異なるふたつの k 次頻出時系列パターンを組み合わせることにより、候補の生成が行われていることに注意する必要がある。ただし、1 次頻出時系列パターンから 2 次候補時系列パターンを生成する際には、前方のアイテム集合が存在しないため、任意の 1 次頻出時系列パターンを組み合わせることができる。また、図の例では、上部にある k 次頻出時系列パターンの最後尾のアイテム集合を共通する部分の次に配置することにより、候補を生成しているが、下部にある k 次頻出時系列パターンの最後尾のアイテム集合を共通する部分の次に配置する候補も生成できることに注意する必要がある。

4. アイテム間制約

時系列パターンの発見問題の適用範囲が広がるにつれて、表構造で構成されたデータがその適用対象に含まれるようになってきた。このような表構造データの場合、各アイテムは属性と属性値で構成されており、同じ属性を持つアイテム同士は、異なる属性を持つアイテム同士よりも、意味的に深い関係を持つと考えられる。この意味的な関係を制約として利用することにより、分析者にとって特徴的な時系列パターンを効率的に発見することが期待できる。以下においては、属性と属性値から構成されるアイテムを考え、このアイテム間の関係を制約として指定するアイテム間制約を紹介する。

アイテム間制約は、一般に式 (3) のように記述することができる。

$$C_1 \to C_2 \to \cdots \to C_n \tag{3}$$

ただし、$n(>0)$ はアイテム間制約によって指定される系列の長さを表すとし、C_i は、時系列的に i 番目に出現するアイテム集合を表すとする。C_i は一般に、$m_i(>0)$ 個のアイテムから構成されており、各アイテムは属性と属性値によって構成されている。従って、式 (4) によって C_i を記述することができる。

$$C_i = \{A_{i1} : a_{i1}, A_{i2} : a_{i2}, \cdots, A_{im_i} : a_{im_i}\} \tag{4}$$

式 (4) においては、A_{ij} 及び a_{ij} が属性及び属性値に対応しており、$A_{ij} : a_{ij}$ によってアイテムが表現されている。アイテム間制約においては、属性及び属性値の両方に対して具体的な値を指定することも可能であるが、属性のみを指定したり、属性値のみを指定したりすることもできる。時系列パターンの発見法は、頻出する時系列パターンに対して、指定されたアイテム間制約を適用することにより、制約を満たす頻出する時系列パターンを特徴的な時系列パターンとして発見する。このとき、アイテム間制約は同時に複数指定することも可能であり、複数指定されている場合には、いずれかのアイテム間制約を満たす頻出する時系列パターンが、特徴的な時系列パターンとして発見されることになる。

例として、表 1 に示す 5 つの属性とその属性値からなる表構造データが日単位に収集されており、このような時系列データから、アイテム間制約を利用して特徴的な時系列パターンを発見する場合を考えてみることにする。ただし、表 1 においては、気温と湿度の場合における属性値「普通」と、人手と交通量の場合における「普通」とを区別するために、属性値「普通」に対して添え字が付与されている。このとき、式 (5) に示すアイテム間制約が指定されているとすれば、ある日において、天気が晴れであり、気温が高い、普通 $_1$、低いのいずれかの場合であった場合に、その後の日において、人手あるいは交通量が多くなるような頻出する時系列パターンを、特徴的な時系列パターンとして発見することになる。

$$\{\text{天気} : \text{晴れ}, \text{気温} : *\} \to \{* : \text{高い}\} \tag{5}$$

表 1: 表構造データ

属性	属性値
天気	晴れ、雨、曇り、雪
気温	高い、普通 $_1$、低い
湿度	高い、普通 $_1$、低い
人手	多い、普通 $_2$、少ない
交通量	多い、普通 $_2$、少ない

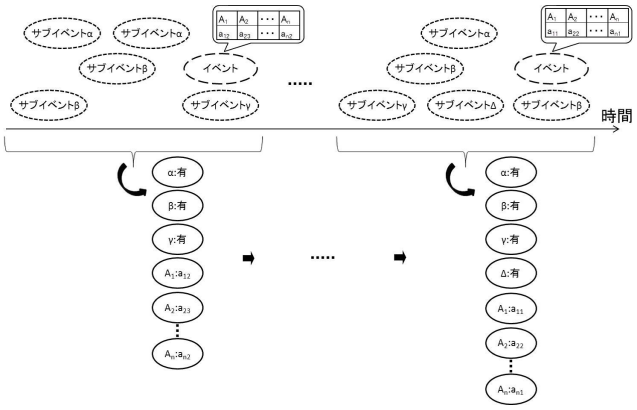


図 2: 時系列データ生成のイメージ

5. 数値実験

5.1 実験データ

社内に蓄積されている時間情報の付随したデータから、時系列パターンマイニング用の時系列データを生成する。本データは、特定のイベントの発生を起点として、その起点までに発生したサブイベントをまとめて、時系列データを構成するひとつのアイテム集合を生成している。特定のイベントには、イベントの結果として与えられる表構造データが紐付いているのに対して、サブイベントはその起点までに発生した回数に関わらず、そのサブイベントが発生した場合にアイテムとして識別される。また、サブイベントには、複数の種類が存在しており、サブイベントごとにアイテムとして識別が行われている。このため、イベントに付随する表構造データから生成されるアイテムは、属性と属性値から構成されるものの、サブイベントから生成されるアイテムは、時系列パターンマイニングの初期から扱われている通常のアイテムとなっている。図 2 は、蓄積されている時間情報の付随したデータから、時系列データが生成されるイメージを示しており、図の上部に記載されているデータから、下部に記載されている時系列データが生成されている。

表 2 は生成した時系列データの特徴を示しており、本表から、今回対象とする時系列データが、多くのアイテムが同時に出現する一方で、時系列としては比較的短いものになっていることが分かる。また、各アイテムは平均して、10%の割合で時系列データに出現している。

5.2 抽出対象時系列パターン

今回の実験では、指定した特定のアイテムの原因となるアイテムを分析することを想定して、アイテム間制約を導入する。表構造データから生成したアイテムは、属性の種類に応じて同種の属性値が与えられている。そこで、指定した属性値の後に特定のアイテムが出現するといった状況を、アイテム間制約として与えることにする。一方、サブイベントから生成したアイテムは、属性と属性値からは構成されてはいない。各アイテムの後に、特定のアイテムが出現するといった制約を、個々にアイテム間制約として記述することも可能ではあるが、多くの制約を記載することが必要となる。そこで、アイテムそのものを属性とみなし、そのアイテムが出現することを属性値とみなすことにより、本アイテムに関するアイテム間制約を一括して指定する。ただし、アイテムが出現しないことに対応するアイテムは別途生成しないことに注意する必要がある。表 3 は、このようにして設定したアイテム間制約の特徴を示している。

今回の実験では、この他の制約として、抽出する時系列パターンを構成する各アイテム集合は、ひとつのアイテムだけから構成されているとし、系列の長さは最大 2 であるとする。

表 2: 実験データの特徴

時系列データ数	28,398
アイテム種類数	314
平均系列長	1.42
系列内平均アイテム数	43.10
系列内特定アイテム平均出現率	0.105

表 3: アイテム間制約の特徴

制約数	42
制約の長さ	2
結論	特定アイテム

5.3 実験方法

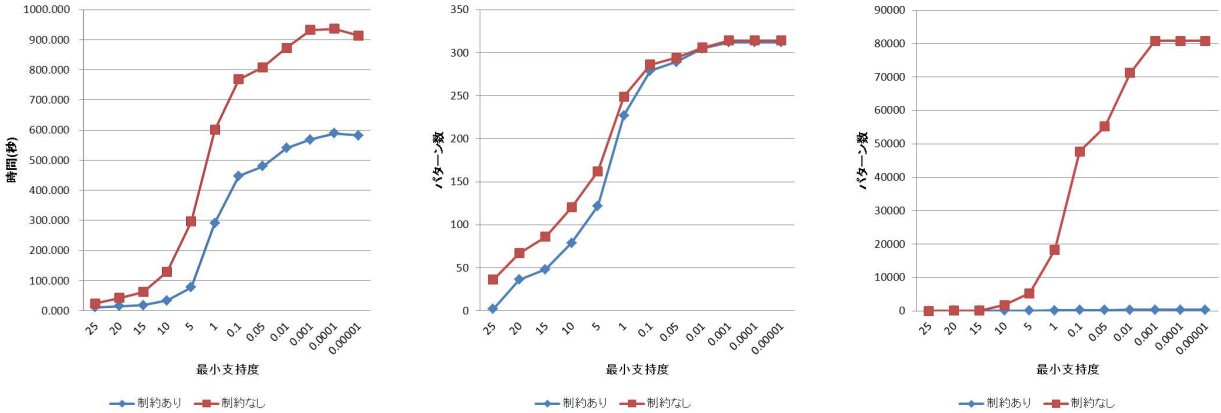
アイテム間制約を適用した場合に発見される特徴的な時系列パターン数と、適用しない場合に発見される頻出時系列パターン数を比較する。また、各場合における時系列パターンの発見時間を比較する。比較実験においては、頻出時系列パターンとして判定するしきい値である最小支持度を、25%、20%、15%、10%、5%、1%、0.1%、0.05%、0.01%、0.001%、0.0001%、0.00001%と変化させる。

実験システムは、C 言語によって実装されており、MinGw-5.1.6 の gcc によってコンパイルされている。また、コンピュータ環境としては、Dell vostro 220s(Intel^R Core^{TM} Duo CPU E8400@3.00GHz 2.99GHz、2.96GB RAM)、Microsoft Windows XP Professional Vesrsion 2002 Service Pack 3 を利用している。

5.4 実験結果

実験結果として、時系列パターンの発見に要する時間と、発見される時系列パターンの数を示す。各図においては、x 軸が最小支持度を表しており、y 軸が発見までに要した時間 (秒) あるいは発見されたパターンの数 (個数) を表している。また、制約ありとラベル付けされた青のラインがアイテム間制約を利用した場合の結果を示しており、制約なしとラベル付けされた赤のラインがアイテム間制約を利用しなかった場合の結果を示している。

図 3(a) は、最小支持度を変えた場合における、特徴的な時系列パターンの発見までに要する時間の推移を示しており、図 3(b) 及び (c) は、最小支持度を変えた場合における、時系列パターンの数を示している。図 3(b) の 1 次時系列パターンは、これだけでは今回指定したアイテム間制約のいずれをも満たし



(a) 発見時間　(a) 1 次時系列パターン数　(b) 2 次時系列パターン数

図 3: 実験結果

てはいないため、特徴的な時系列パターンとみなすことはできない。しかしながら、アイテム間制約を考えたことにより、2 次時系列パターンを生成するための元となる 1 次時系列パターンが減ることを確認するために図示している。アイテム間制約を導入することにより、事前にアイテム間制約に合致しない時系列データを削除することが可能であり、この影響で、1 次時系列パターンの数が減っていることに注意する必要がある。

5.5 考察

発見時間: 制約のありなしに関わらず、最小支持度が小さくなるにしたがって、急速にパターンの発見時間が増加している。特徴的な時系列パターンとして発見されるだけでなく、その前段として作成する候補の数が増えたことがその原因としてあげられる。しかしながら、アイテム間制約を導入した場合には、その上昇は抑えられており、アイテム間制約によって、パターン発見時間を抑える効果を確認することができる。

一方、最小支持度が極端に小さい場合に、一部発見時間が逆転する現象を確認することができる。このような場合においては、ほとんどすべての時系列パターンが発見されており、発見時間の差はほとんどないと考えられる。このため、この現象は、計測誤差の影響によるものと考えられる。

パターン数: 指定した 42 個のアイテム間制約のいずれをも満たさない時系列データを事前に削除しているため、頻出する 1 次時系列パターンの数が、アイテム間制約を導入した場合に、減少していることを確認することができる。しかしながら、ほとんどのアイテムをカバーした制約を導入しているため、その差はそれ程大きなものにはなってはいない。特に、最小支持度を小さくした場合には、ほとんどのアイテムが抽出されることになるため、その差はより小さなものとなっている。

一方、特徴的な 2 次時系列パターンの数を比較してみると、その差が顕著に表れている。制約を利用した場合には、最小支持度をかなり小さくしたとしても、300 個程度の時系列パターンしか発見されていないものの、制約を利用しない場合には 80,000 個を超える時系列パターンが抽出されている。制約を導入しない場合における、このような規模の時系列パターンは、分析者が逐次確認して、その正しさを確認することができない。このため、制約を利用しない場合には、高い最小支持度を指定しなければならず、特徴的な時系列パターンが事前にフィルタリングされてしまう危険性が高くなるといえる。これに対して、制約を利用した場合には、かなり小さな最小支持度を指定することができるため、このような危険性を回避することができる。

以上の考察に基づいて、提案しているアイテム間制約は、単なる頻出性では発見できない特徴的な時系列パターンの発見に寄与することができると考えられる。

6. まとめと今後の課題

本論文では、特徴的な時系列パターンを発見するために導入したアイテム間制約の効果を、実データに基づいて検証した結果を報告した。アイテム間制約を導入することにより、劇的に時系列パターンを絞り込むことができ、分析者にとって特徴的な時系列パターンを容易に発見することができる。

現在、我々のグループでは、時系列パターンマイニングに基づいた因果関係の分析に取り組んでおり、アイテム間制約などの制約を利用して、より意味のある因果関係の発見に取り組んでいく予定である。また、これら成果を、ビックデータ時代における、ヘルスケア分野及びスマートコミュニティ分野における因果関係の分析に利用していく予定である。

参考文献

[1] R. Agrawal and R. Srikant: Mining Sequential Patterns, *in Proc. of the 11th International Conference Data Engineering*, pp. 3-14 (1995)

[2] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, M. Hsu: PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth, *in Proc. of the 2001 International Conference Data Engineering*, pp. 215-224 (2001)

[3] S. Sakurai, Y. Kitahara, and R. Orihara: Discovery of Sequential Patterns based on Constraint Patterns, *International Journal of Computational Intelligence*, Vol. 4, No. 4, pp. 275-281 (2008)

[4] S. Sakurai, Y. Kitahara, R. Orihara, K. Iwata, N. Honda, and T. Hayashi: Discovery of Sequential Patterns Coinciding with Analysts' Interests, *Journal of Computers*, Vol. 3, No. 7, pp. 1-8 (2008)