電子カルテデータにおける インタラクティブな頻出パターン抽出手法の評価

青柳 結衣[1]　Le Hieu Hanh[1]　松尾 亮輔[1]　山崎 友義[1]　荒木 賢二[1]　横田 治夫[2]　小口 正人[1]

概要：データ活用が進む中，頻出パターンを抽出するシーケンシャルパターンマイニング (SPM) が注目されており，その中でも閾値を調整しながら解析を繰り返すインタラクティブな SPM が不可欠である．このような問題設定に対し，解析結果を知識ベース (KB) として保存することで，再計算を避け，実行時間の短縮を実現する代表的手法として KISP が提案されている．しかし，閾値が単調減少する場合には高速化が期待できる一方で，KB の容量が大きい場合や増減を繰り返す場合は，探索や管理に要するコストが増大し，実行時間がかかる．また，過去の解析で得られた頻出パターンが十分に考慮されていないため，本来生成されるべき候補シーケンスが生成されない場合がある．そこで，KISP の改善をした手法を提案する．本稿では提案手法の実装変更を行い、処理の最適化を行う．また，電子カルテに適用し，より実利用環境に即した評価を行う．

Evaluation of an Interactive Frequent Pattern Mining Method for Electronic Health Record Data

YUI AOYAGI[1]　HIEU HANH LE[1]　RYOSUKE MATSUO[1]　TOMOYOSHI YAMAZAKI[1]　KENJI ARAKI[1]
HARUO YOKOTA[2]　MASATO OGUCHI[1]

1. はじめに

1.1 研究背景

様々なビッグデータの蓄積に伴い，シーケンスを対象とする SPM(Sequential pattern mining) が注目されている．SPM とは，閾値 (minsup) を指定し，頻度の高いシーケンシャルパターンを抽出する解析手法である．購買行動分析 [11]，医療カルテ解析 [13]，ウェブページのクリックストリーム分析 [5] 等，アイテムの発生順序が重要となるデータベースに対して，頻出アイテムセット抽出手法 [1] より有益な情報を得ることができる．特に医療分野においては，電子カルテなどの時系列データから有用な知見を抽出することが期待されている．SPM のアルゴリズムは盛んに提案されており，有名なアルゴリズムとして，ID リストを用いた SPADE[12]，ID のリストとビットマップを組み合わせた SPAM[3]，プレフィックスとポストフィックスを用いた PrefixSpan[10] などが挙げられる．

SPM に使われる minsup は，データセットの最小出現回数である．シーケンスがデータセット内に出現する頻度をサポート値とし，サポート値が minsup より大きい場合に頻度が高いとみなされる．最適な minsup はデータセットに依存するため，指定する minsup が小さすぎると頻出シーケンシャルパターン数が増えて，minsup が大きすぎるとパターンが出力されない．そのため，minsup を調整しながら解析を行うのに長けているインタラクティブな SPM が不可欠である．インタラクティブな SPM の有名なアルゴリズムとして KISP[7] が挙げられる．

1.2 本研究の目的

インタラクティブな SPM の代表的手法として KISP が提案されており，過去の計算結果を知識ベース (Knowledge Base, KB) として保持することで，再計算を回避し高速化を実現している．しかし，KISP にはいくつかの課題が存在する．具体的には，minsup が増加する場合や増減を繰り

[1] お茶の水女子大学
Ochanomizu University
[2] 城西大学
Josai University

返す場合において実行時間が増加する点や，候補シーケンス生成に不備があり，本来抽出されるべきパターンが得られない可能性がある点が挙げられる．これらの課題に対し，先行研究では候補シーケンス生成式の拡張および KB 構造の改良を行うことで，正確性の向上と実行時間の改善を実現した．しかしながら，これらの手法は主にウェブページのクリックストリームデータを用いた評価にとどまっており，実際の応用環境における有効性は十分に検証されていない．

本研究では，既存のインタラクティブ SPM 手法およびその改良手法を対象として，電子カルテデータを用いた評価を行う．具体的には，提案手法を大規模データセットに適用可能となるよう実装を拡張し，医療電子データを用いた実験を通じて，実環境における有効性を明らかにすることを目的とする．

1.3 本稿の構成

本稿は以下の通りに構成される．2 節では本研究の関連研究について説明する．3 節では提案手法について述べる．4 節では，実験方法および評価結果を示す．最後に 5 節で本稿のまとめと今後の課題について述べる．

2. 関連研究

本節では，本研究に関連する SPM，インタラクティブな SPM，KISP と先行研究について説明する．

2.1 SPM

Agrawal らによって提案されたシーケンシャルパターンマイニング (SPM)[2] は，シーケンスデータベース (SDB) から頻出シーケンシャルパターンをすべて抽出するための手法である．SDB はシーケンスとシーケンス ID の組の集合で表される．SPM では入力された minsup よりも頻度が高いパターンを頻出パターンとする．minsup が 0.1 の場合，SDB 内の 10%以上のシーケンスに含まれているパターンが出力される．minsup を小さくすると多くのパターンが出力されるが，有益な情報が埋もれてしまうことがある．逆に minsup を大きくすると頻出パターンが出力されないため，適切な minsup を指定する必要がある．

2.2 インタラクティブな SPM

従来の SPM は，データベースが静的であると仮定している．実際，頻出シーケンシャルパターンを取得するためにデータベースに一回のみ適用するよう設計されている [6]．その後，データベースが更新されると，アルゴリズムを再び最初から実行する必要がある．データベースの変更が小規模であり，再び完全に探索する必要がない場合があるため，この手法は非効率である．

この問題を解決するために，いくつかの増分的な SPM アルゴリズムが設計されている [4], [8], [9]．増分的なアルゴリズムには，ユーザが minsup などのパラメータを変更することを考慮して，インタラクティブにマイニングするよう設計されたものもある．インタラクティブな SPM の一つである KISP は，GSP アルゴリズムを拡張したものである．

2.3 KISP

KISP は，KB を使用するインタラクティブな SPM であり，minsup を変更し，繰り返し実行するのに適している．指定された minsup が KB.base より大きい場合は，minsup を満たすパターンを KB から単純に取得するため, パフォーマンスが大幅に向上する．

以下に KISP のアルゴリズムを説明する．まず, 指定された minsup と KB.base の比較を行う．minsup が KB.base 以上である場合は，頻出シーケンシャルパターンを KB から取得する．minsup を満たす頻出パターンは全て KB に保存されているため，データセットにアクセスせず，頻出パターンを得ることができる．minsup が KB.base より小さい場合は新たにパターンを探索する．

探索をする際，最初に新しい候補シーケンスを生成する．ここでは今までのマイニングで候補シーケンスとされなかったシーケンスのみを生成する．KISP は既存のインタラクティブな SPM より候補シーケンス数が少ないため，高速である．最後に候補シーケンスのサポート値を算出し，KB に保存する．全ての候補シーケンスについて算出後，minsup の条件を満たす頻出シーケンシャルパターンを取得できる．

2.3.1 候補シーケンスの生成

KISP は新しい候補シーケンスを生成する際，全ての候補を生成したのち，カウント済みのシーケンスを取り除くのではなく，新しい候補を直接生成する．具体的には以下の式で長さ k の候補 X_k を求める．

$$X_k = (S_{k-1}[\text{KB.base}] \otimes N_{k-1}[\text{minsup}]) \cup (N_{k-1}[\text{minsup}] \otimes N_{k-1}[\text{minsup}]) \quad (1)$$

$S_k[\text{minsup}]$ は KB 内に存在する長さ k の頻出シーケンスの集合を表し，$\otimes$ は結合操作を意味する．x と y の結合を試みた際，x の先頭のアイテムを削除して得られる部分シーケンスと，y の最後のアイテムを削除して得られる部分シーケンスが一致した時は結合することができる．結合の結果，得られる候補は x に y の最後のアイテムを付加したシーケンスとなる．$N_k[\text{minsup}]$ を，KB にすでに存在する長さ k の頻出シーケンスとは対照的に，minsup の変更によって新たに頻出となった長さ k のシーケンスの集合と定義する．

長さ k の新しい候補シーケンスを求める場合は，まず KB 内のサポート値が kb.base 以上の長さ k-1 のパターン

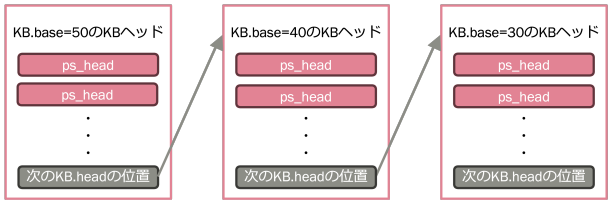

図 1 KB ヘッドの例

と，指定した minsup で新しく頻出になった長さ k-1 のパターンを結合する．さらに，新しく頻出になった長さ k-1 のパターン同士を結合し，それら全てのパターンを候補とする．

2.3.2 KB 構造

KISP の KB は最小の KB.base と複数の KB ヘッドで構成されている．新しい情報が追加されるたびに KB ヘッドを作成する．KB ヘッドは，KB ヘッドが生成された時の minsup，パターンサポートテーブルを要約したパターンサポートヘッドとその総数，次の KB ヘッドへのリンクの 4 つの要素で構成されている．パターンサポートテーブルは同じサイズのパターンをグループ化して保存しており，特定サイズのパターンを迅速に見つけることが可能である．候補シーケンス生成時にパターンの長さが重要であるため，グループ化して保存している．

KB ヘッドの構造の例を図 1 に示す．最初に minsup=50 を与えると KB.base=50 の KB ヘッドが生成される．次に minsup=40 を与えると KB.base=40 の KB ヘッドを生成する．このように，KB に存在しない新しいパターンのサポート値を計算するたびに KB ヘッドが増加する．KB.base=40 の KB ヘッドには minsup=50 のシーケンスを求めるときに必要な情報と minsup=40 を求めるときに必要な情報の差分のみが挿入されている．

2.4 先行研究における KISP の改良

先行研究では，上記の課題に対して以下の改良が行われている．まず，候補シーケンス生成において，従来の結合条件を拡張することで，取りこぼしのない網羅的な候補生成を実現している．これにより，パターン抽出の正確性が向上している．次に，KB の構造を見直し，minsup 変更時に必要な情報へ高速にアクセス可能なデータ構造を導入することで，実行時間の短縮を図っている．特に，minsup が増減する場合においても効率的に再利用が可能となるよう設計されている．これらの改良により，従来手法と比較して正確性および効率性の向上が確認されている．

しかしながら，実運用を想定した場合，いくつかの課題が残されている．まず，大規模データへの適用に関する検討が不十分である．先行研究では小規模なデータセットを対象として評価が行われており，データサイズの増加に伴うメモリ使用量や計算コストの増大については十分に議論されていない．そのため，大規模データに対しては処理が困難となる可能性がある．

次に，実データへの適用に関する課題が挙げられる．特に医療電子データのような実データは，単純なシーケンシャルデータとは異なる特性を持つ．しかしながら，先行研究ではこのような実データ特有の問題への対応について十分に検討されていない．

そこで本研究では，これらの課題を解決するために，先行研究の改良手法を基盤とし，大規模データおよび実データへの適用を考慮した拡張を行う．これにより，実環境における適用可能性を明らかにすることを目的とする．

3. 提案手法

これらの改良手法の問題を解決するため，本研究ではサポート値計算の効率化を目的とした手法を提案する．本手法は，ハッシュツリー構造の改良および射影データベースの導入の 2 つの要素から構成される．また，これらの手法を本論文では Prefix-based counting (pbc) と呼称する．

3.1 ハッシュツリー構造の改良

従来手法では，候補シーケンスは個別に管理され，各ノードごとに独立してサポート値計算が行われていた．これに対し，本研究では最後のアイテムを除いた部分シーケンスが同一なシーケンスをまとめて管理する構造を導入する．これにより，冗長な探索処理を削減し，効率的な支持度計算を実現する．この構造により，従来は個別に処理されていた複数の候補シーケンスをまとめて処理できるため，計算回数の削減が期待される．

3.2 射影データベースの導入

本研究では，各葉ノードに対応する射影データベースを構築し，それを参照することで支持度計算を行う．具体的には，同一 prefix を持つ候補シーケンス群に対して，共通の射影データベースを生成する．

例として，候補シーケンス (A, B, C)，(A, B, D)，(A, B, E) が同一の葉ノードに属する場合，prefix である (A, B) に基づいて射影データベースを構築する．このとき，元のデータベースから (A, B) を含むシーケンスを抽出し，その後続要素のみから構成される射影データベースを生成する．そして，この射影データベース内において，各要素が出現するシーケンス数をそれぞれのサポート値として計算する．

このように，候補シーケンスごとに個別のデータベース探索を行うのではなく，prefix 単位で構築された射影データベースを共有することで，サポート値計算を一括して実行することが可能となる．その結果，元データベースに対するスキャン回数を削減することができ，計算効率の向上が期待される．

表 1　実験環境

サーバ	Dell PowerEdge R740
CPU	Intel Xeon Platinum 8260L 2.4GHz (24 cores/48 threads)
OS	Ubuntu 24.04.1 LTS
メモリ	384GB
python	3.12.3

表 2　データセットの概要

シーケンス数	888
平均要素数	17.94

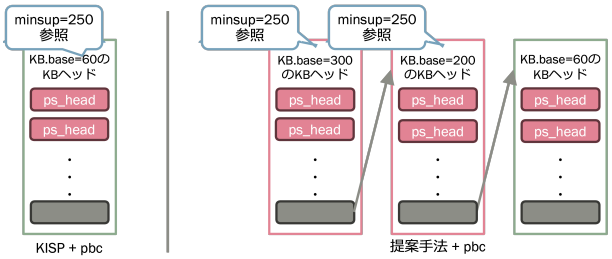

図 2　作成する KB ヘッドと参照先の比較

4.　実験

本実験では電子カルテデータを用いた実行時間の検証を目的とする．

4.1　実験環境

表 1 に実験環境を示した．

本実験では，27 の医療機関から 2015 年から 2024 年までに収集された匿名加工済み電子カルテデータセットを使用した．本実験で使用するデータセットの利用にあたっては，次世代医療基盤法認定作成事業者である一般社団法人ライフデータイニシアティブの倫理委員会 (No.2024_MIL_0004_A001) の承認を受けた．今回は，子宮頸・体部の悪性腫瘍患者のデータセットのみ抽出して使用しており，詳細な内容は表 2 に示す．

4.2　実験内容

候補シーケンスのサポート値計算手法の変更による有効性を示すため，2 種類の実験を行った．

一つ目は KISP と先行研究における提案手法の双方で変更前後の比較を行う．minsup が増加した後に減少する場合の挙動を確認するため，60, 200, 300, 250 という順で与える．この minsup 列を 1 セットとして通しで実行し，各 minsup における実行時間の平均を算出した．

二つ目は最大 KB.base 更新時，KB ヘッドを追加作成による比較である．比較を行う手法の違いを図 2 に示す．pbc を用いた手法でも，KB ヘッドの追加作成が有効か確認する目的で行う．minsup の与え方は一つ目と同様である．

minsup	60	200	300	250	合計
KISP	1,969,788.38	200.35	179.28	175.28	1,970,343.34
KISP + pbc	1,893,118.24	201.88	180.82	176.60	1,893,677.57

図 3　KISP における pbc 有無による実行時間の比較 (ms)

minsup	60	200	300	250	合計
先行研究	2,021,144.90	192.20	0.22	0.14	2,021,337.48
先行研究 + pbc	1,885,912.18	202.24	0.22	0.14	1,886,114.81

図 4　先行研究手法における pbc 有無による実行時間の比較 (ms)

minsup	60	200	300	250	合計
KISP + pbc	1,893,118.24	201.88	180.82	176.60	1,893,677.57
先行研究 + pbc	1,885,912.18	202.24	0.22	0.14	1,886,114.81

図 5　KB ヘッド追加作成による実行時間の比較 (ms)

4.3　実験結果

4.3.1　pbc 有無による実行時間の比較

3 回測定した平均実行時間の結果を図 3 と図 4 に示す．どちらの手法においても，合計実行時間の短縮が見られる．特に，最初の minsup である minsup = 60 における実行時間が短縮さた．今回の minsup 列において，サポート値計算を行うのは最初の minsup のみであり，pbc の有効性が確認できた．

4.3.2　KB ヘッド追加作成による実行時間の比較

3 回測定した平均実行時間の結果を図 5 に示す．minsup = 300, 250 の場合，実行時間の短縮が見られる．これはより小さい KB.base を持つ KB ヘッドを参照することができるためだとわかる．KB.base が小さい KB ヘッドほど挿入されているシーケンス数が少ないので，探索にかかる時間が短くなる．

5.　まとめと今後の課題

5.1　まとめ

本研究では，インタラクティブなシーケンシャルパターンマイニング手法である KISP に対し，サポート値計算の効率化を目的とした手法を提案した．具体的には，prefix 単位で候補シーケンスをまとめて管理するハッシュツリー構造の改良と，共通の射影データベースを用いたサポート値計算を導入した．電子カルテデータを用いた実験の結果，提案手法は従来手法と比較して実行時間を短縮できることを確認した．特に初回実行時において効果が顕著であり，データベース走査の冗長性削減が有効であることが示唆された．KB 構造の変更による性能向上も確認できた．今回以外の minsup 列でも同様の結果が得られるかは，今後の課題とする．

5.2　今後の課題

今後の課題としては，大規模データセットにおける性能評価を行い，計算時間およびメモリ使用量の観点から提案手法の有効性を検証する必要がある．また，prefix 単位で

の処理の粒度や，射影データベースの構築方法を最適化することで，さらなる高速化が可能であると考えられる．

謝辞

本研究の一部は日本学術振興会科学研究費 (#24K02943) の助成からの支援によって行われた．

参考文献

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB'94)*, pp. 487–499, 1994.

[2] R. Agrawal and R. Srikant. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering*, pp. 3–14, 1995.

[3] J. Ayres, J. Gehrke, , and T. Yiu J. Flannick. Sequential pattern mining using a bitmap representation. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'02)*, pp. 429–435, 2002.

[4] H. Cheng, X. Yan, and J. Han. IncSpan: incremental mining of sequential patterns in large database. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 527–532, 2004.

[5] P. Fournier-Viger, T. Gueniche, and V. S. Tseng. Using partially-ordered sequential rules to generate more accurate sequence prediction. In *International Conference on Advanced Data Mining and Applications*, pp. 431–442, 2012.

[6] Philippe Fournier-Viger, Jerry Chun-Wei Lin, Rage-Uday Kiran, Yun-Sing Koh, and Rincy Thomas. A survey of sequential pattern mining. *Data Science and Pattern Recognition*, Vol. 1, No. 1, pp. 54–77, 2017.

[7] M. Y. Lin and S. Y. Lee. Improving the efficiency of interactive sequential pattern mining by incremental pattern discovery. In *International Conference on System Sciences*, pp. 68–76, 2002.

[8] F. Masseglia, P. Poncelet, and M. Teisseire. Incremental mining of sequential patterns in large databases. *Data and Knowledge Engineering*, Vol. 46, pp. 97–121, 2003.

[9] S. N. Nguyen, X. Sun, and M. E. Orlowska. Improvements of incspan: Incremental mining of sequential patterns in large database. In *Advances in Knowledge Discovery and Data Mining*, pp. 442–451, 2005.

[10] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. PrefixSpan : Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceeding of 2001 international conference on data engineering*, pp. 215–224, 2001.

[11] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In *International Conference on Extending Database Technology*, pp. 1–17, 1996.

[12] Mohammed J. Zaki. Spade: An efficient algorithm for mining frequent sequences. *Machine Learning*, Vol. 42, No. 1-2, pp. 31–60, 2001.

[13] 横田治夫. 電子カルテデータ解析-医療支援のためのエビデンス・ベースド・アプローチ-. 共立出版, 2022.