

シーケンスバリエーションの比較と 電子カルテの分析への応用

李玉清¹ Le Hieu Hanh² 松尾亮輔³
山崎友義⁴ 荒木賢二⁵ 横田治夫⁶

シーケンスバリエーション(SV)はシーケンス中に分岐を含む半順序集合の束で、シーケンス中の変化を扱うようなアプリケーションではSV間の比較は重要な課題である。比較の際に、共通なサブシーケンスと共通でない差異となるサブシーケンスを検出することが必要である。本論文では、SV間の比較に必要な定義を行い、比較アルゴリズムを提案する。まず、最長共通サブシーケンスに基づき最長共通サブシーケンスバリエーション(LCSV)と、複数のSVの共通・差異部分を示す併合シーケンスバリエーション(MSV)を新たに定義する。次に、LCSVとMSVを生成するためのアルゴリズムを示す。提案したアルゴリズムの効果を検証するために、実際の複数医療機関の電子カルテの医療シーケンスに適用し、医療機関間の医療指示シーケンスパターンの比較を実施する。比較結果は医療関係者に有益な情報であると評価された。

1 序論

シーケンスは順序性を持つ要素の集合やリストとして多くの場面で出現する。例えば、書籍の購入履歴や人間の日中に行動したイベントの履歴などはシーケンスとして表現され、複数シーケンスを比較することで、次商品の購入や次行動に合わせた推薦に利用される。複数シーケンスの比較のために、シーケンス間の最長共通部分列(Longest Common Sequence, 以下LCS)を求めて活用する手法が多く提案されてきた[20]。

しかし、LCSはシーケンス中の分岐を扱うアプリケーションに対応できていない。シーケンス中に分岐を含む半順序集合の束の場合には、単純な一列のシーケンスとして表現しただけでは不十分である。1つ例としては、検査オーダー、処方、手術などの医療指示が含まれている電子カルテの解析で抽出される典型的な医療指示の流れ「クリニカルパス」が挙げられる。このようなクリニカルパスに対して、患者の状態等で医療従事者が医療指示を変更する医療オプションをシーケンス中に含む場合がある[9, 12]。すなわち、クリニカルパスは医療指示シーケンス中に分岐を含ん

だ半順序集合の束となるため、普通のシーケンスで表現不可になり、比較するためにはLCSの概念を拡張する必要がある。

そのような分岐を含む医療指示シーケンスの比較が求められる背景としては、国として医療情報の二次利用の必要性が認識され、電子カルテなどの医療・健康に関する記録を全国規模で一元的に集める「千年カルテプロジェクト」[18]等が始動していることが挙げられる。「千年カルテプロジェクト」に参加する医療機関のデータがまとめて保存・管理されているため、複数の医療機関を跨いだ比較を行う研究が期待されている。医療記録のより効率的な利用の一環として医療機関が異なることによる複数クリニカルパスの比較によって治療パターンの多様性を知ること非常に有用であり、医療の質と効率の向上が期待される。

本論文では半順序集合の束をシーケンスバリエーション(Sequence Variant, 以下SV)と定義し、複数SVの共通・差異部分を導出するアルゴリズムを提案する。SVを有効グラフと考えた場合に、グラフ編集距離[5]に基づいたグラフ比較手法を適用することで、複数SV間の違いの定量化が可能ではあるが、共通・差異の部分まで示すことができない。そこで、複数SVの比較のために、本論文ではLCSの概念を拡張した最長共通サブシーケンスバリエーション(Longest Common Subsequence Variant, 以下LCSV)と併合シーケンスバリエーション(Merged Sequence Variant, 以下MSV)を定義し、それらを生成するアルゴリズムを提案する。具体的に、シーケンスデータベースからシーケンスパターンマイニング(Sequence Pattern Mining, 以下SPM)を適用し、出現頻度の高い頻出シーケンスを抽出し、頻出シーケンス間の共通部分をまとめることで、SVを生成する[9]。次に、2SVを入力とし、LCSの生成方法に基づいてLCSVを抽出する。最後に、LCSVと元々のSVと比較しながら、共通部分と差異部分を表すMSVを作っていく。

本研究の有効性を検証するために、2医療機関の電子カルテのデータを用い、同一疾病に対するクリニカルパスを抽出し、それらのパスの共通・差異部分を表すLCSVとMSVを生成する。最後に生成された実験結果に関する医療関係者の解析と評価を得る。

本論文は以下の通り構成される。2節では本研究に関連する概念を背景知識として説明する。3節では提案概念であるLCSVとMSVを求めるアルゴリズムについて述べる。4節では、3節の手法を用いて実際の医療指示データを解析することと治療パターンのMSVの作成を行う。5節では、提案手法に関する考察結果を報告する。最後に6節でまとめと今後の課題について述べる。

2 背景知識と関連研究

本節では、本研究に関連するシーケンシャルバリエーション(SV)とシーケンシャルパターンマイニング(SPM)について説明する。

2.1 背景知識

2.1.1 シーケンス

定義 1 (アイテムセット) アイテムセット I を以下のように定

¹ 学生会員 東京工業大学情報理工学院

li@de.cs.titech.ac.jp

² 正会員 東京工業大学情報理工学院

hanh1h@de.cs.titech.ac.jp

³ 非会員 一般社団法人ライフデータイニシアティブ

matsuo@ldi.or.jp

⁴ 非会員 東京工業大学情報理工学院

yamazaki.cp@gmail.com

⁵ 非会員 宮崎大学医学部附属病院医療情報部

taichan@med.miyazaki-u.ac.jp

⁶ 正会員 東京工業大学情報理工学院

yokota@cs.titech.ac.jp

義する.

$$I = \{i_1, i_2, \dots, i_n\}$$

$i_j \in I$ がアイテムとなる.

アイテムを解釈するとき, i をアイテム i の「name」とも呼ぶ.

定義 2 (シーケンス) アイテムセット I に対し, シーケンス S を以下のように定義する.

$$S = (\{s_1, s_2, \dots, s_m\}, <_S)$$

$s_j = (id, i), i \in I$. id は S の中でユニークなインデックスとなる. $<_S$ は S 上の全順序関係である. $\forall s_i, s_j \in S, s_i <_S s_j \vee s_i <_S s_j$.

2.1.2 SV

定義 3 (シーケンスバリエント) アイテムセット I に対し, 束 SV をシーケンスバリエントと定義し, 以下のように表す.

$$SV = (\{sv_1, sv_2, \dots, sv_l\}, <_{SV})$$

$sv_j = (id, i), i \in I$. id は SV の中でユニークなインデックスとなる. $<_{SV}$ は SV 上の半順序関係である. また, SV が上限および下限を持っている.

例として, 医療指示列において分岐が発生した場合, シーケンスバリエントになる. シーケンス (入院→検体検査→手術→注射→退院) と (入院→看護タスク→手術→処方→退院) に対して, 注射と処方がバリエントと呼ばれ, この二つのシーケンスをまとめてシーケンスバリエントと考える. SV のバリエントに対する安全性と効率性を定量的に比較できるため, SV を抽出することは非常に有用である. また, バリエントが現れる要因を調査することで, 特定の患者に対する治療のような医療指示の改善支援が可能となる.

以下, A, B, C のような大文字をアイテムとして説明する場合, A, B, C は id 付きアイテムの「name」を指す. また, $A < B, B < C, B < D, C < E, D < E$ を, $< A, B, (C, D), E >$ と表す.

SV を可視化する方法として, シーケンスバリエントグラフを定義する.

定義 4 (シーケンスバリエントグラフ) シーケンスバリエント SV に対し, 有向非巡回グラフ G をシーケンスバリエントグラフと定義し, 以下のように表す.

$$G = (V, E)$$

頂点集合 V が SV のアイテム集合となる. また, $\forall e = (sv_{out}, sv_{in}) \in E, sv_{out} <_{SV} sv_{in}$.

シーケンスバリエントグラフを可視化することで, 簡単にシーケンスバリエントを示すことができる. 例として, A, B, C, D, E がアイテムで, $SV = < A, B, (C, D), E >$ のシーケンスバリエントグラフが図1に示す.

シーケンスバリエントとシーケンスバリエントグラフは互いに変換可能であるため, 以下両者を同じものと扱う.

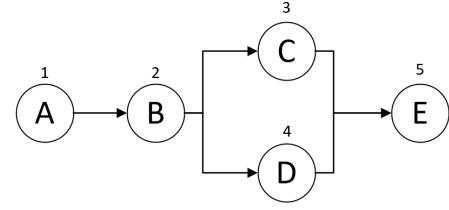


図1 SVのシーケンスバリエントグラフ

2.1.3 LCS

定義 5 (共通部分列(CS)) シーケンス S_α と S_β に対し, 共通部分列 CS を下記の条件を満たすシーケンスと定義する.

$$\forall cs_i, cs_j \in CS, cs_i, cs_j \in S_\alpha \cap S_\beta \text{ かつ } cs_i <_{CS} cs_j \iff cs_i <_\alpha cs_j \wedge cs_i <_\beta cs_j.$$

CS の部分列も CS となるので, CS は複数存在する. そのため, 最長共通部分列を定義する.

定義 6 (最長共通部分列(LCS)) シーケンス S_α と S_β に対し, 最長共通部分列 LCS を下記の条件を満たす共通部分列と定義する. S_α と S_β に対する任意の共通部分列 CS に対し, $|CS| \leq |LCS|$. $|CS|$ は CS のノード数を表す.

例として, シーケンス $< A, B, C, D >$ と $< A, C, B, A, D >$ を考える. 長さ2の共通部分列が5つ存在する: $< A, B >, < A, C >, < A, D >, < B, D >, < C, D >$. 長さ3の共通部分列が2つ存在する: $< A, B, D >, < A, C, D >$. それ以上長い共通部分列が存在しない. 従って, 最長共通部分列が $< A, B, D >$ と $< A, C, D >$ である.

入力列の個数が任意である一般の場合については, この問題はNP困難である [13]. 入力列の個数が一定のときには, この問題は動的計画法によって多項式時間で解く事ができる.

2.2 関連研究

2.2.1 SPM

Agrawalらによって提案されたSPMはシーケンシャルデータベース(以下, SDB)から頻出シーケンスを抽出する手法であり [2], 医療・Eコマース・インターネットなどの領域で注目されている [4]. アイテムの順列をシーケンスと呼び, SDBはあるシーケンス集合に属するシーケンスと, シーケンスの識別子であるシーケンスIDを組みとする要素からなる. SDBから出現頻度が設定された最小支持度 (minsup) より大きいシーケンスを抽出することとなる.

アプリアリに基づいたアルゴリズム [2]はよく知られているが, 抽出するとき, 冗長なパターンを大量に生成するところと大規模データセットの場合, データセットをスキャンする時間が大量に要求されるところが問題点であった. データセットのスキャンを減らすために, PrefixSpan [14]が提案されている. PrefixSpanは, 末端アイテムを除いた頻出パターンであるprefixへの射影を繰り返すことで, 探索対象のデータセットを徐々に縮小するため, 効率的な探索ができる. さらに効率的に抽出を行うために, 飽和オーダー列を抽出するCSPan [15]等も提案されている.

2.2.2 SPMによる医療情報の解析

佐々木ら [16]は、実際の電子カルテデータに、タイムインターバルSPM [3] (以下 TI-SPM) を適用して、時間間隔を考慮した頻出シーケンシャルパターンの抽出を行った。Leら [10]は、効率的なSPMであるCSpan [15]を拡張して、タイムインターバルの統計情報を収集するT-CSpanを提案し、実電子カルテに適用した。山田ら [6]は、頻出シーケンスがどのシーケンスと対応関係にあるのかという情報を、シーケンスの識別子であるシーケンスIDを保持しながらマイニングすることによって、頻出シーケンス抽出と同時に取得し、頻出シーケンスごとの安全性や効率性の指標を算出し、SV評価が円滑かつ正確に行える可視化手法を提案し、実電子カルテを用いて推薦結果の評価を行った。坂本ら [8]は、分岐先頻度と実施時刻情報を用いたグループ化による適正分岐候補選択を行い、頻出シーケンス分岐先の適正候補選択による併合シーケンスの生成を行った。また、分岐要因である患者情報を拡充し、併合シーケンスの分岐への要因適用による医療指示推薦を提案した。Leら [11]は、電子カルテ解析におけるプライバシー保護に関して差分プライバシーの概念を適用する手法を提案している。

2.2.3 SVの解析

本田ら [9]は、実電子カルテの医療指示のSVを抽出し、グラフで表現して医療従事者への視認性の高い可視化ツールの提供を行った。さらに本田ら [7]は、SVに関する医療関係者の解析と評価を得るに対して、患者の性別や年齢といった静的情報と、血圧や検査結果といった動的情報に対する多変量解析を行うことでバリエーションの要因分析を行う手法を提案し、実電子カルテを用いて効果を示した。

3 提案手法

本節では、LCSVとMSVの定義とそれらを算出するアルゴリズムについて述べる。また、MSVの算出によるSV比較方法を提案する。前提として [6]で用いられたシーケンスIDを保持するSPMを行い、頻出シーケンスを抽出し、SVを生成しているものとする。

本節からアイテムをノードとも呼ぶ。ノードには四つの属性があり、それぞれ $name$ 、ノードの名称を指す; $nextList$ 、ノードの次に来るノードのリストを指す; id 、ノードの唯一の番号を指す; $label$ 、ノードの属しているSVを指す。具体的には、 $sv.nextList = \{sv' \in SV | (sv, sv') \in E\}$ 。

3.1 LCSV

SV間の差異を示すために、まずSV間の共通部分を抽出する。従来のLCSでは、SVに適用できないため、CSとLCSを拡張した概念とした共通サブシーケンスバリエーション(CSV)と最長共通サブシーケンスバリエーション(LCSV)を提案する。

定義 7 (共通サブシーケンスバリエーション(CSV)) シーケンスバリエーション SV_α と SV_β に対し、共通サブシーケンスバリエーションCSVを下記の条件を満たすシーケンスバリエーションと定義する。 $\forall csv_i, csv_j \in CSV, csv_i, csv_j \in SV_\alpha \cap SV_\beta$ かつ $csv_i <_{CSV} csv_j \iff csv_i <_\alpha csv_j \wedge csv_i <_\beta csv_j$ 。

Input: SV_1, SV_2

Output: $LCSV$

```

1 number  $SV_1$  from 1 to  $s$ ;
2 number  $SV_2$  from  $s + 1$  to  $t$ ;
3  $r_1 \leftarrow$  the number of routes of  $SV_1$ ;
4  $r_2 \leftarrow$  the number of routes of  $SV_2$ ;
5 for  $i = 1, 2, \dots, r_1$  do
6   for  $j = 1, 2, \dots, r_2$  do
7      $LCS(i, j) \leftarrow$  the LCS of route  $i$  of  $SV_1$  and route  $j$  of  $SV_2$ ;
8   end
9    $LCS(i) \leftarrow$  the longest one in
       $LCS(i, 1), LCS(i, 2), \dots, LCS(i, r_2)$ 
;
10 end
11  $LCSV \leftarrow$  combination of  $LCS(1), LCS(2), \dots, LCS(r_1)$ ;
12 re-number nodes in  $LCSV$  from  $t + 1$ , also re-number them
    in  $SV_1$  and  $SV_2$ ;
```

Algorithm 1: LCSVを導出するアルゴリズム

CSVのサブシーケンスバリエーションもCSVとなるので、CSVは複数存在する。そのため、最長共通サブシーケンスバリエーションを定義する。

定義 8 (最長共通サブシーケンスバリエーション(LCSV)) シーケンス SV_α と SV_β に対し、最長共通サブシーケンスバリエーションLCSVを下記の条件を満たす共通サブシーケンスバリエーションと定義する。 SV_α と SV_β に対する任意の共通サブシーケンスバリエーションCSVに対し、 $|CSV| \leq |LCSV|$ 。|CSV|はCSVのノード数を表す。

LCSVを導出するアルゴリズムと具体的な例を説明する。

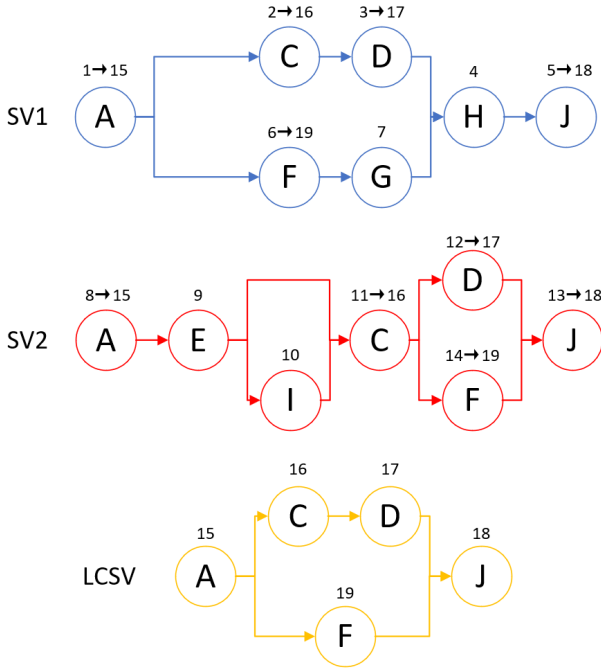
図2が示しているように、 SV_1 にパスが2つあり、 SV_2 にパスが4つある。LCSVはアルゴリズム 12により算出され、中のノードがリナンバーされた。

3.2 MSV

複数のSVを比較するために、MSVを導入する。

定義 9 (併合シーケンスバリエーション(MSV)) シーケンスバリエーション SV_α, SV_β とそのLCSVに対し、併合シーケンスバリエーションMSVを下記の条件を満たすシーケンスバリエーションと定義する。 $\forall sv_i, sv_j \in SV_\alpha \cup SV_\beta, sv_i, sv_j \in MSV$ かつ $sv_i <_{MSV} sv_j \iff sv_i <_\alpha sv_j \wedge sv_i <_\beta sv_j$ 。 $\forall msv_i, msv_j \in MSV, msv_i <_{MSV} msv_j \iff (msv_i, msv_j \in SV_\alpha \wedge msv_i <_\alpha msv_j) \vee (msv_i, msv_j \in SV_\beta \wedge msv_i <_\beta msv_j)$ 。また、LCSVに出現したノードにlabel $\{\alpha, \beta\}$ を与え、出現していないノードに属しているSVにより、label α か β を与える。

SV間のMSVを導出するためのアルゴリズムを以下に述べる。

図2 SV₁とSV₂のLCSVを計算する**Input:** SV₁, SV₂**Output:** MSV

- 1 $LCSV \leftarrow LCSV(SV_1, SV_2)$;
- 2 $map \leftarrow NodeMappingGeneration(SV_1, SV_2, LCSV)$;
- 3 restore the MSV from map (map contains label and order information);

Algorithm 2: MSVを導出するアルゴリズム

map はkey-value型のデータ構造で、keyがノードのidで、valueがノードとなる。最後のステップで、 map_1 と map_2 を併合するとき、同時に両者に存在するノードのnextListを併合してから map の中に入れる。他のノードはそのまま入れる。

図2の例に対してMSVを求めるアルゴリズムを適用したものを図3に示す。図では、ノードのlabelを表すために色を用い、SV₁を青で、SV₂を赤で、共通部分であるLCSVを黄で示している。

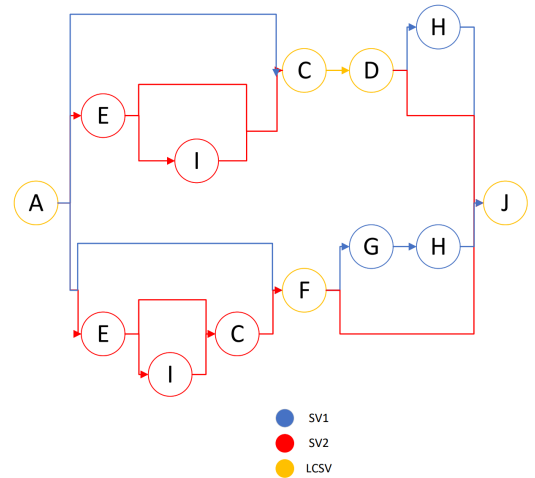
MSVの作成において、同じ名前を持つノードがSV中に複数出現する場合、それを区別するためにノードに対するidの番号によって区別する。

さらに、それぞれのSVにおいて、共通の名前のノード間の順序関係が変わる場合には、元のシーケンスの順序を保つため、MSV中に新たなノードを作成する。例として、図2のノードCとノードFの順序関係が2パターンある。このため、図3のMSVには、ラベル1+2のノードCとラベル2のノードCが同時に出現した。この点を正確に示すことは非常に重要である。

このため、アルゴリズム20でサーチをするとき、一度出現したLCSVに属しているノードがまた見つかったら、新しいノードを作り、新しいidを与える。この操作により、 map に異なるラベル

Input: SV₁, SV₂, LCSV**Output:** a mapping map

- 1 $map_1 \leftarrow \emptyset$;
- 2 $map_2 \leftarrow \emptyset$;
- 3 $list \leftarrow node(LCSV)$;
- 4 $n \leftarrow$ the maximum number of nodes in $LCSV$;
- 5 **while** $list \neq \emptyset$ **do**
- 6 $CN_c \leftarrow$ the node with smallest number in $list$;
- 7 $CN_1 \leftarrow$ the node in SV_1 with the same number as CN_c ;
- 8 $CN_2 \leftarrow$ the node in SV_2 with the same number as CN_c ;
- 9 $newCN_c \leftarrow CN_c$;
- 10 $newCN_c.next \leftarrow CN_1.next \cup CN_2.next$;
- 11 put $\{newCN_c.id : newCN_c\}$ into map_1 and map_2 ;
- 12 $nextList \leftarrow CN_c.next$;
- 13 $map_1 \leftarrow SearchNext(CN_1, nextList, CN_c, n, map_1)$;
- 14 $nextList \leftarrow CN_c.next$;
- 15 $map_2 \leftarrow SearchNext(CN_2, nextList, CN_c, n, map_2)$;
- 16 $list.remove(CN_c)$;
- 17 **end**
- 18 combine map_1 and map_2 as map ;

Algorithm 3: NodeMappingGeneration 関数図3 SV₁とSV₂のMSV

を持つノードを入れることが可能になった。

4 実験

本節では、医療機関Aと医療機関Bから提供された実際の電子カルテデータに対して提案手法を適用し、有効性を確認する。

4.1 実験方法と環境

有効なLCSVとMSVを得るためには、それぞれの最小支持度(minsup)をどのように設定するかが問題で、そのためにまずはminsupを変化させたSV同士のLCSVとMSVを求めてみることから始める。「調剤料計算」、「食事療養」など治療行為とは関

Input: $CN, nextList, CN_c, n, map$ **Output:** map

```

1 if nextList =  $\emptyset$  or  $CN.next = \emptyset$  then
2   return
3 end
4 for node in  $CN.next$  do
5   if map.keySet().contains(node.id) and
6      $CN_c.next.contains(node)$  then
7     newNode  $\leftarrow$  node;
8     newNode.id  $\leftarrow$  n + 1;
9     n ++;
10    put {newNode.id : newNode} into map;
11  else
12    put {node.id : node} into map;
13  end
14  if node  $\in$  nextList then
15    map.get(node.id).label  $\leftarrow$  {1, 2};
16    nextList.remove(node);
17  else
18    SearchNext(node, nextList,  $CN_c, n, map$ );
19  end
20 end

```

Algorithm 4: SearchNext 関数

表1 実験環境

CPU	AMD Ryzen Threadripper 3960X 24-Core Processor
Memory	128 GB
OS	Ubuntu 18.04.1 LTS
Java.ver	Java 11.0.11
R.ver	R 3.6.3

係が薄い医療指示を除外し、各病院のデータベースにSPMを行い、頻出シーケンスを求める。得られた頻出シーケンスをSVに併合する。次にLCSVのアルゴリズムを適用し、各医療機関のSVのLCSVを算出する。さらに、MSVを作成し、グラフとして可視化する。治療に直接関係する医療指示を最大限に含んでいるため、この実験は最長の頻出シーケンスのみを考慮している。

実験の環境を表1に示す。

4.2 データセット

本研究では宮崎大学医学部の臨床研究情報基盤に蓄積されている両医療機関で2015年4月から2020年3月までに記録された、実際に使用されているクリニカルパスを元に行った医療指示データを対象とする。この医療指示データは個人情報保護の観点より患者を一意に特定できるような情報を含まない。なお、本研究で医療機関Aならびに医療機関Bの電子カルテデータを、多施設共同研究として医療従事者支援に用いることは宮崎大学のHP [19]に記載されており、宮崎大学の倫理審査委員会及び東京工業大学の人を対象とする研究倫理審査委員会の承認を得ている。

表2 実験に関する情報

	医療機関A	医療機関B
延患者数	438	4,265
最大医療指示数	816	1,100
平均医療指示数	85.61	69.51
最長在院日数	48	50
平均在院日数	11.52	6.87

先行研究で使っていた医療機関Aの電子カルテデータは時刻情報が入っており、時刻を使っていた。本研究もそれに準じて時刻を基に解析を予定していたが、複数病院の比較をするために、臨床研究情報基盤の電子カルテデータを使うことにした。臨床研究情報基盤の電子カルテデータに処置の日付情報はあがるが、詳しい時刻情報はない。千年カルテプロジェクトのデータにも同じく、時刻情報がない。データベースに入っている順番を使うことにしたが、一日内の処置の順番は必ずしも正確ではない。今後日付のみに基づく実験を行う。

本実験ではカテールを用いた狭心症・筋梗塞の治療法である経皮的冠動脈インターベンション(Percutaneous Coronary Intervention, PCI)を受けた患者の入院期間中に行われた医療指示履歴を対象とする。両医療機関で共通しているため、PCIを選んだ。実験に関する情報を表2に示す。

4.3 実験結果

4.3.1 シーケンスバリエーションの抽出

両医療機関のデータセットが異なるため、同じ処置でも用語が異なる場合がある。それを統一するために、レセプト電算コードを使う。同じレセプト電算コードを持つ処置を同じ処置と見なす。処置の名称とレセプト電算コードの対照を表3に示す。表には、処置についての簡単な説明も加えた。特に、PCIは様々な種類があるため、対応するレセプト電算コードも多いである。処置毎に費用に関わる診療報酬点数(以下、点数と表記)が付いている。PCIは対応する点数が多いため、平均値24,802を使う。さらに詳しい情報を知りたい場合は、[17]で処置の名称あるいはレセプト電算コードで区分番号(例えば、A000-00)を探して、区分番号を用いて[1]で処置の詳細内容が得られる。

前に説明した通り、minsupをいくつか試して、実際に医療機関Aでは0.15と0.2、医療機関Bでは0.25と0.3を採用した。抽出されたシーケンスバリエーションを図4、図5、図6と図7に示す。

病院毎に総点数を算出する。医療機関Aのminsup=0.15の場合、パスが4つある。それぞれのパスにおける点数の合計は25,411, 25,400, 25,261と25,250で、平均値は25,330.5となる。医療機関Aのminsup=0.2の場合、パスが2つある。それぞれのパスの合計の点数は25,213と25,099で、平均値は25,156となる。医療機関Bのminsup=0.25の場合、総点数は25,524となる。医療機関Bのminsup=0.3の場合、総点数は25,388となる。今回の場合は、シーケンスバリエーション間および医療機関間の差は0.5%以下で、比較的小さいである。

医療機関BのSVにはパスが1つしかない。患者数が多いため、頻出パターンの抽出が困難であることが原因として考えられる。

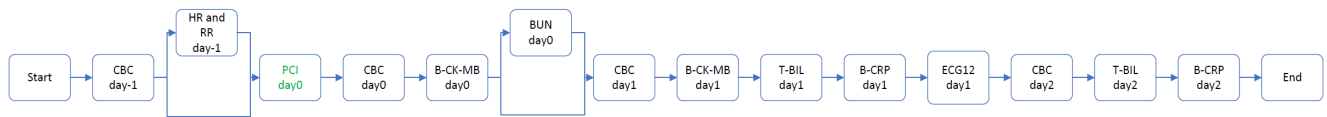


図4 医療機関Aのminsup=0.15のSV

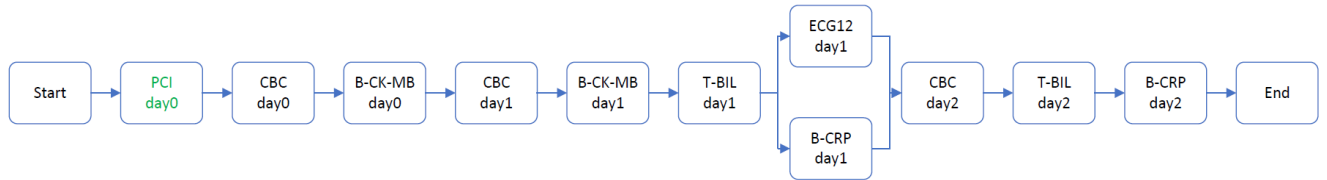


図5 医療機関Aのminsup=0.2のSV



図6 医療機関Bのminsup=0.25のSV

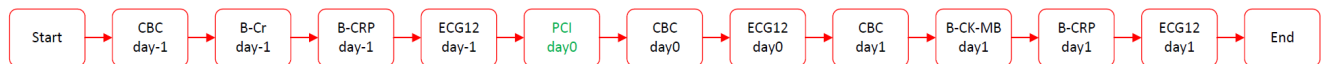


図7 医療機関Bのminsup=0.3のSV

表3 処置の詳細

ノード名	レセプト番号	点数	医療機関A ¹	医療機関B ²	簡単な説明
CBC	160008010	21	末梢血液一般	末梢血液一般	血液形態・機能検査
HR and RR	160102510	150	呼吸心拍監視(3時間超 7日以内)	-	呼吸心拍監視
B-CK-MB	160114710	90	CK-MB	B-CK-MB	血液化学検査
BUN	160019010	11	BUN	-	血液化学検査
T-BIL	160017010	11	BIL/総	-	血液化学検査
B-Cr	160019210	11	-	B-クレアチニン	血液化学検査
B-CRP	160054710	16	C反応性蛋白(CRP)	B-CRP	血漿蛋白免疫学的検査
B-BNP	160162350	136	-	B-BNP	内分泌学的検査
ECG12	160068410	130	心電図(四肢単極・胸部誘導含む12誘導)	ECG12	心電図
PCI	150374910	36,000	-	-	-
	150375010	22,000	-	-	-
	150375110	19,300	-	-	-
	150260350	28,280	-	-	-
	150284310	24,720	-	-	-
	150359310	24,720	-	-	-
	150375210	34,380	-	-	-
	150375310	24,380	-	-	-
	150375410	21,680	-	-	-
	160107550	17,720	-	-	-
	150318310	19,640	-	-	-



図8 医療機関A(0.15)と医療機関B(0.25&0.3)のLCSV



図9 医療機関A(0.2)と医療機関B(0.25&0.3)のLCSV

の上部に示す。

図から簡単に処置の情報が得られる。黄色の部分は両医療機関で共通のサブシーケンスで、赤と青の部分から両医療機関それぞれがどのように医療指示を行うかも分かる。赤と青の部分と比較することで、両医療機関の治療パターンの違いを知ることができる。例として、「T-BIL(総ビリルビン)」は医療機関Aでは何回か出てくるが、医療機関Bでは出なかった。また、「ECG12(12誘導心電図)」は医療機関Bでは毎日実施されるが、医療機関Aでは一回しか実施されなかった。

得られた情報に関して、医療関係者に確認したところ、このような示し方は有益であるという評価であった。具体的な違いに関しては、「ECG12」の違いは発生しうる違いであり、術後は患者モニターで3点誘導の心電図情報を絶えず取っているのが一般的で、医療機関Aでは3点に加えて12点で取るのは1回で十分と判断している可能性がある。一方、医療機関Bでは、3点では不十分で毎日12点を取る必要があると判断している可能性がある。「T-BIL」の違いに関しては、オリジナルデータベースを確認した結果、医療機関Bではたまに実施されるが、頻度が低いいため抽出されなかった。医療機関Aの場合に、パッケージとして含まれている可能性が考えられる。

4.3.2 LCSVの導出

いくつかのminsupの組み合わせを試し、LCSVを導出する。結果は図8と9に示す。

処置の名称と相対的日付を全体として扱う。例えば、「B-CK-MB day0」と「B-CK-MB day1」は異なる医療指示である。

実際に、医療機関Bのminsup=0.25とminsup=0.3のSVには細かい違いしかないので、導出したLCSVは同じである。

4.3.3 MSVの導出

各minsupの組み合わせのMSVは図10、図11、図12と図13に示す。差異部分の処置の点数を、それぞれのSVに対応するノード

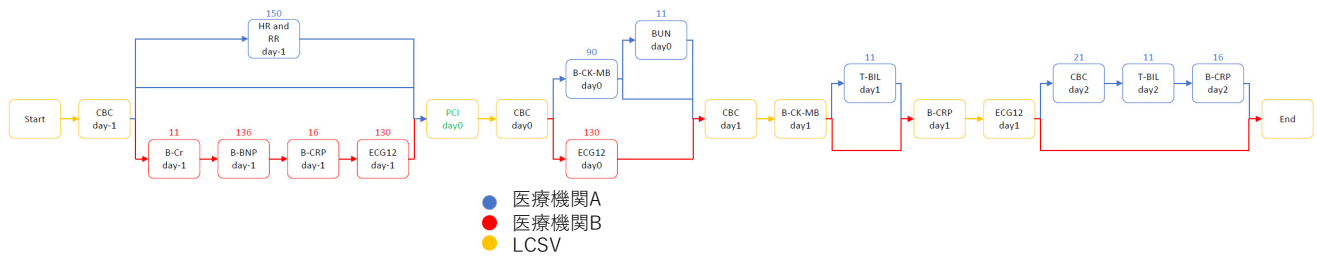


図10 医療機関A(0.15)と医療機関B(0.25)のMSV

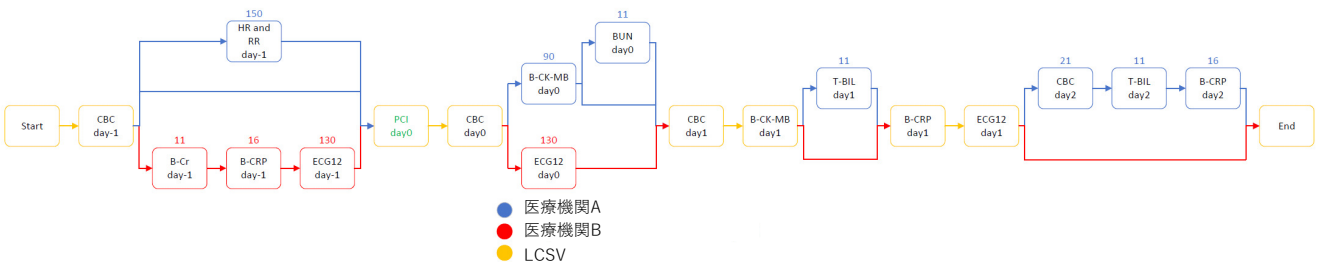


図11 医療機関A(0.15)と医療機関B(0.3)のMSV

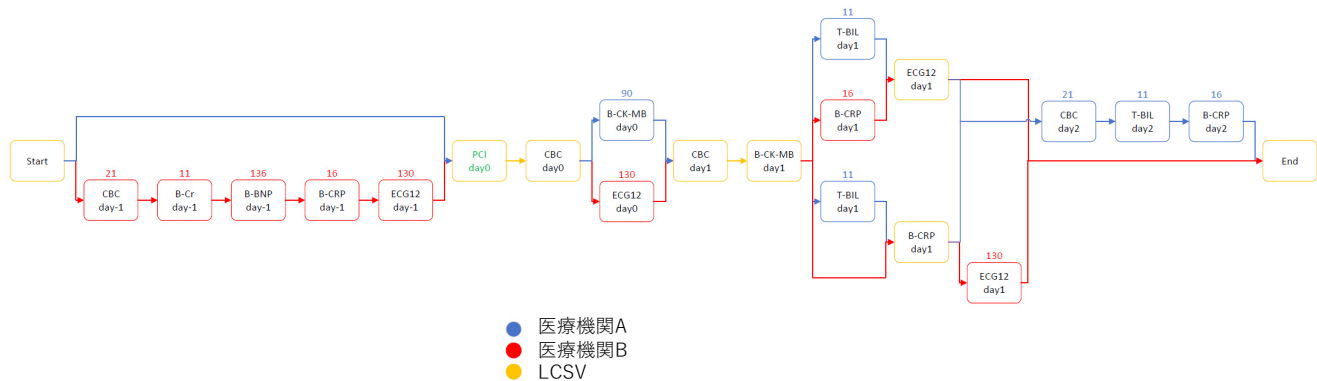


図12 医療機関A(0.2)と医療機関B(0.25)のMSV

5 考察

1つの医療機関のデータセットから頻出パターンを抽出する場合、minsupは抽出された情報の数と頻出パターンの読みやすさで決めることができる。また、minsupをいくつか試し、すべての結果を示すのも選択肢である。しかし、複数の医療機関のパターンを比較する場合、どのような比較は医学的に意味があるか、どのようにそれぞれのminsupを設定するかが問題になる。例として、minsupを実験で用いた値より小さくすると、SVの長さが100以上になる。実際に本研究は任意の2つのSVを比較できる方法を提案したため、minsupの組み合わせをいくつか試し、それを全部比較することができる。どの組み合わせが最も医学的に意味があるかを決めるのは困難だが、全部の組み合わせを医療従事者に示し、有用な組み合わせを選んでもらうことができる。

6 結論

6.1 まとめ

本論文はシーケンス中に分岐を含む半順序集合の束であるシーケンスバリエント (SV) を比較するためのアルゴリズムを提案した。SV比較のために、LCSに基づき最長共通サブシーケンスバリエント (LCSV) を抽出し、LCSVと元SVを比較しながら2SVの共通・差異を示す併合シーケンスバリエント (MSV) を生成し、可視化する。本提案の有効性を検証するために、実際に、2種類のminsupの設定を用いて、2医療機関の電子カルテのデータに提案手法を適用し、LCSVとMSVを求め、両病院間の差異を可視化した。その結果を医療関係者に提示した結果、医療機関間の治療方針の違いと思われる差異が分かり有益である評価が得られた。

6.2 今後の課題

今後の課題として、まず3医療機関以上の扱いに関しても検討

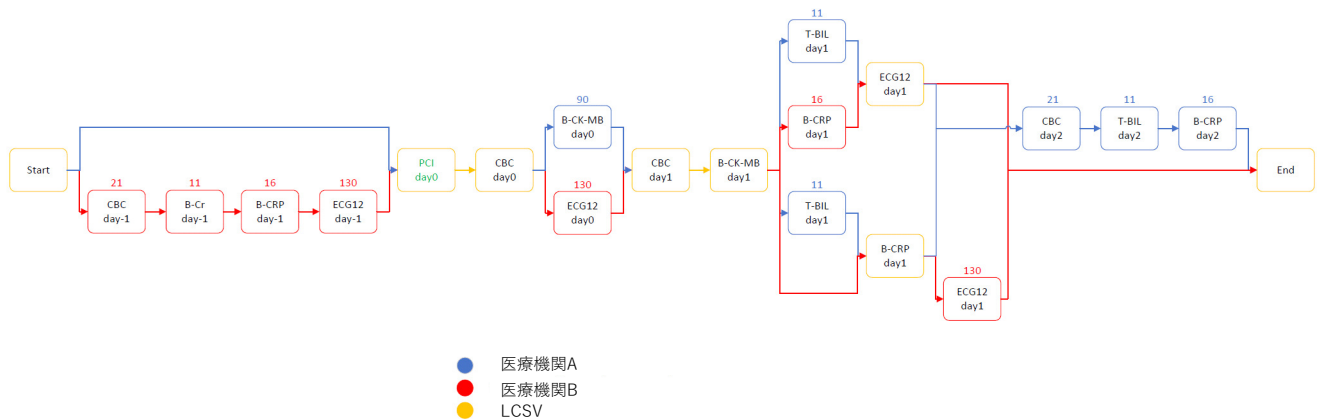


図13 医療機関A(0.2)と医療機関B(0.3)のMSV

し、適切なminsupを選択する指針に関して検討する。更に、電子カルテシステムの医療指示の情報には時刻情報が含まれない場合に、日単位での医療指示のシーケンスとして扱う必要があり、その場合のLCSVやMSVについても検討を行う。また、提案手法は様々な施術を対象にすることができるとは、他の施術に対する適用も考える。最後に、シーケンスバリエーションの差分をより分かりやすくするために、MSVを表示する可視化ツールを開発する予定である。

謝辞

本研究の一部は、日本学術振興会科学研究費補助金(#20H04192, #21K17746)の助成により行われた。本研究で用いた電子カルテデータの利用に関しては、宮崎大学のHP [19]に記載されており、宮崎大学の倫理審査委員会及び東京工業大学の人を対象とする研究倫理審査委員会の承認を得ている。関係者各位の協力に感謝する。

参考文献

- [1] 医科診療報酬点数表. <https://shirobon.net/medicalfee/latest/>. 最終アクセス: 2022年11月9日.
- [2] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In *Proceeding of the 20th International Conference on Very Large Data Bases (VLDB 1994)*, pages 487–499, 1994.
- [3] Yen-Liang Chen, Mei-Ching Chiang, and Ming-Tat Ko. Discovering time-interval sequential patterns in sequence databases. *Expert Systems with Applications*, 25:343–354, 2003.
- [4] 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*, 1(1):54–77, 2017.
- [5] Xinbo Gao, Bing Xiao, Dacheng Tao, and Xuelong Li. A survey of graph edit distance. *Pattern Analysis and Application*, 13:113–129, 2010.
- [6] 山田達大, 本田祐一, 菅原正彬, Le Hieu Hanh, 串間宗夫, 小川泰石, 松尾亮輔, 山崎友義, 荒木賢二, 横田治夫. SIDを保持するシーケンシャルパターンマイニングによるクリニカルパスバリエーション分析. In *DEIM Forum 2019, D1-1*, 2019.
- [7] 本田祐一, 山田達大, 菅原正彬, Le Hieu Hanh, 串間宗夫, 小川泰石, 松尾亮輔, 山崎友義, 荒木賢二, 横田治夫. 患者の固有情報及び動的状況を考慮したクリニカルパス分岐要因推定. In *DEIM Forum 2019, D1-5*, 2019.
- [8] 坂本任駿, 小林莉華, Le Hieu Hanh, 松尾亮輔, 山崎友義, 荒木賢二, 横田治夫. 頻度と実施時刻によるグループ化を採り入れたシーケンス解析に基づく医療指示推薦. In *DEIM Forum 2021, C25-1*, 2021.
- [9] Yuichi Honda, Muneo Kushima, Tomoyoshi Yamazaki, Kenji Araki, and Haruo Yokota. Detection and visualization of variants in typical medical treatment sequences. In *Proceeding of the third VLDB workshop on data management and analytics for medicine and healthcare (DMAH 2017)*, pages 88–101. Springer, 2017.
- [10] Hieu Hanh Le, Henrik Edman, Yuichi Honda, Muneo Kushima, Tomoyoshi Yamazaki, Kenji Araki, and Haruo Yokota. Fast generation of clinical pathways including time intervals in sequential pattern mining on electronic medical record systems. In *Proceeding of the fourth International Conference on Computational Science and Computational Intelligence (CSCI 2017)*, pages 1726–1731, 2017.
- [11] Hieu Hanh Le, Muneo Kushima, Kenji Araki, and Haruo Yokota. Differentially private sequential pattern mining considering time interval for electronic medical record systems. In *Proceedings of the 23rd International Database Engineering and Applications Symposium (IDEAS 2019)*, page 95–103, 2019.
- [12] Hieu Hanh Le, Tatsuhiro Yamada, Yuichi Honda, Masaaki Kaya-hara, Muneo Kushima, Kenji Araki, and Haruo Yokota. Analyzing sequence pattern variants in sequential pattern mining and its application to electronic medical record systems. In *Proceedings of the 30th International Conference on Database and Expert Systems Application (DEXA 2019) Part 2*, pages 393–408, 2019.
- [13] David Maier. The complexity of some problems on subsequences and supersequences. *Journal of the ACM*, 25(2):322–336, 1978.
- [14] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Mei-Chun Hsu. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceeding of 2001 International Conference on Data Engineering (ICDE 2001)*, pages 215–224, 2001.
- [15] Raju V Purushothama and Varma GP Saradhi. Mining closed sequential patterns in large sequence databases. *International Journal of Database Management Systems*, 7(1):29–39, 2015.
- [16] 佐々木夢, 荒堀喜貴, 串間宗夫, 荒木賢二, 横田治夫. 電子カルテシステムのオーダログデータ解析による医療行為の支援. In *DEIM Forum 2015, G5-1*, 2015.
- [17] 厚生労働省保健局. 医科診療行為告示・通知情報一覧. <https://shinryohoshu.mhlw.go.jp/shinryohoshu/paMenu/doPaListSp>. 最終アクセス: 2022年11月9日.
- [18] 吉原博幸. 千年カルテプロジェクト: 本格的日本版chrと医療データの2次利用に向けて. 情報管理, 60(11):767–778, 2018.
- [19] 宮崎大学医学部附属病院臨床研究支援センター. 公開しなければならぬ臨床研究(公示). <http://www.med.miyazaki-u.ac.jp/home/crsc/patient/notice/>. 最終アクセス: 2022年11月9日.
- [20] 山元理絵, 小林大, 吉原朋宏, 小林隆志, 横田治夫. アクセスログに基づくWebページ推薦におけるLCSの利用とその解析. 情報処理学会論文誌データベース, 48(SIG11 (TOD34)):38–48, 2007.