DEIM Forum 2022 A21-4

シーケンシャルパターンマイニングに基づく多病院間の頻出治療パターンの比較

李　玉清† Le Hieu Hanh† 松尾　亮輔†† 山崎　友義††† 荒木　賢二†††
横田　治夫†

† 東京工業大学 情報理工学院　〒 152–8552 東京都目黒区大岡山 2-12-1
†† 一般社団法人ライフデータイニシアティブ　〒 606–0805 京都市左京区下鴨森本町 15
††† 宮崎大学 医学部附属病院 医療情報部　〒 889–1692 宮崎県宮崎市清武町木原 5200
E-mail: †{li,hanhlh}@de.cs.titech.ac.jp, †yokota@cs.titech.ac.jp,
††matsuo@ldi.or.jp,
†††{yama-cp,taichan}@med.miyazaki-u.ac.jp

あらまし　電子カルテの普及に伴い，蓄積された医療情報の二次利用による医療支援が期待されている．二次利用の 1 つの例として，これまで医療関係者が経験を基に作成していた疾病毎の典型的な医療指示の流れであるクリニカルパスの作成や改善支援のため，電子カルテデータから頻出の医療指示パターンを抽出する手法が提案されている．しかし，それらの対象は一つの医療機関に留まり，複数の医療機関の違いを示すものはなかった．多数の医療機関の電子カルテを集積する「千年カルテプロジェクト」が開始され，複数の医療機関の治療パターンの違いを提示することも求められるようになってきた．本研究では，異なる医療機関でのバリアントを含む頻出医療指示のシーケンスの違いを示すため，最長共通サブシーケンスバリアント（LCSV）と，それを用いた併合シーケンスバリアント（MSV）の概念とアルゴリズムを提案し，実際の 2 病院の電子カルテデータに適用し，違いを示す．

キーワード　電子カルテ，医療支援，医療機関間比較，シーケンシャルパターンマイニング，シーケンスバリアント

1 序　論

1.1 研究背景

近年，紙のカルテに代わり電子カルテの普及が進み，今後さらに日本全国の電子カルテの普及率が増加していくことが予想される．これに伴い，蓄積された医療情報の二次利用が期待されている [1]．二次利用の例として，疾病毎の患者に対して頻出医療指示シーケンスの抽出による，典型的な医療行為の流れ「クリニカルパス」の生成支援が挙げられる．従来，クリニカルパスの作成は医療関係者自身の医学的経験に基づいて行われており，人力でクリニカルパスを収集・分析して改善するのは容易ではなかった．そのため，データ工学における様々な手法を電子カルテに適用して解析することで，医療行為改善の支援を行う研究が始まった．

また，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」[2] が始まっている．「千年カルテプロジェクト」に参加する施設のデータがまとめに保存・管理されているため，複数の病院を跨いだ研究が期待されている．医療記録のより効率的な利用と医療機関が異なることによる治療パターンの多様性を知ることに非常に有用であり，医療の質と効率の向上が見込まれる．

1.2 本研究の目的

本研究は，複数の医療機関の電子カルテを解析し，医療の改善に貢献することを目的とする．アプローチとして，医療指示データを対象として，シーケンシャルパターンマイニングを用いて，実際の電子カルテデータベースから頻出治療パターンの抽出を行う．また，頻出治療パターンをシーケンスバリアントとして，最長共通サブシーケンスバリアント（Longest Common Subsequence Variant, 以下 LCSV）の算出と併合シーケンスバリアント（Merged Sequence Variant, 以下 MSV）の作成を行い，医療機関間の治療パターンの差分を示す．

電子カルテには医療支援のための解析の対象となる様々な情報が含まれるが，ここでは医療指示の情報に着目する．それぞれの患者の医療指示はシーケンスとなるため，疾患毎に複数の患者の医療指示にシーケンシャルパターンマイニングを適用することで、疾患に対する頻出医療指示パターンを抽出できる．しかし，これまでの研究は一つの病院の頻出パターンしか抽出せず，医療機関の違いの影響も考慮していない．本研究は異なる病院の治療パターンの比較をすることで医療行為の改善に貢献することを目指す．

1.3 本稿の構成

本稿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を述べる．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アント（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 を以下のように定義する．

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

$i_j \in I$ 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．

$$S = (\{s_1, s_2, \ldots, s_m\}, \prec_S)$$

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

2.1.2 SV

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

$$SV = (\{sv_1, sv_2, \ldots, sv_l\}, \prec_{SV})$$

$sv_j = (id, i), i \in I$．id は SV の中でユニークなインデックスとなる．$\prec_{SV}$ 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，シーケンスバリアントになる．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，注射と処方がバリアントと呼ばれ，この二つのシーケンスをまとめてシーケンスバリアントと考える．SV のバリア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，バリア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行為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，$A \prec B, B \prec C, B \prec D, C \prec E, D \prec E$ を，$< A, B, (C, D), E >$ と表す．

SV を可視化する方法として，シーケンスバリアントグラフを定義する．

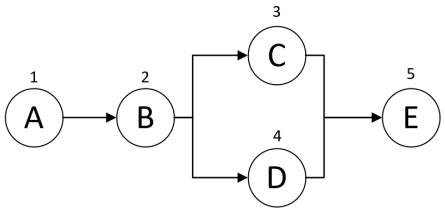

図 1　SV のシーケンスバリアントグラフ

定義 4 (シーケンスバリアントグラフ)．シーケンスバリアント SV に対し，有向非巡回グラフ G をシーケンスバリア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，$\forall e = (sv_{out}, sv_{in}) \in E, sv_{out} \prec_{SV} sv_{in}$．

シーケンスバリアントグラフを可視化することで，簡単にシーケンスバリア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，$SV =< A, B, (C, D), E >$ のシーケンスバリアントグラフが図 1 のように示されている．

シーケンスバリアントとシーケンスバリア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．

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

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，$|CS| \leqq |LCS|$．$|CS|$ 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 [15]．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 [3]，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 [13]．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．

アプリオリに基づいたアルゴリズム [3] 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 [6] 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 [8] 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 [4] は，実際の電子カルテデータに、タイムインターバル SPM [5]（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 [7] は，効率的な SPM である CSpan [8] 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 [9] 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 [12] 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の拡充し，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 [14] 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 [10] は，実電子カルテの医療指示のシーケンスバリアントを抽出し，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 [11] は，シーケンスバリアントに対して，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アント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 の算出による頻出治療パターンの比較方法を提案する．前提として [9] の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，$sv.\text{nextList} = \{sv' \in SV | (sv, sv') \in E\}$.

Algorithm 1 LCSV

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

3.1 LCSV

医療機関の間の差異を示すために，まずそれぞれの医療機関の SV の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アント (CSV) と最長共通サブシーケンスバリアント (LCSV) を提案する．

定義 7 (共通サブシーケンスバリアント (CSV))．シーケンスバリアント SV_α と SV_β に対し，共通サブシーケンスバリアント CSV を下記の条件を満たすシーケンスバリアントと定義する．$\forall csv_i, csv_j \in CSV, csv_i, csv_j \in SV_\alpha \cap SV_\beta$ かつ $csv_i \prec_{CSV} csv_j \iff csv_i \prec_\alpha csv_j \wedge csv_i \prec_\beta csv_j$.

CSV のサブシーケンスバリアントも 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アントを定義する．

定義 8 (最長共通サブシーケンスバリアント (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アント $LCSV$ を下記の条件を満たす共通サブシーケンスバリアントと定義する．SV_α と SV_β に対する任意の共通サブシーケンスバリアント CSV に対し，$|CSV| \leqq |LCSV|$．$|CSV|$ は CSV のノード数を表す．

LCSV を計算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 1 により算出され，中のノードがリナンバーされた．

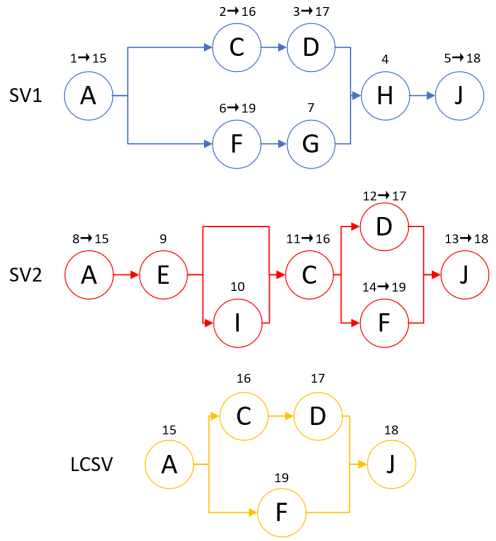

図 2　SV_1 と SV_2 の LCSV を計算する

Algorithm 2 MSV

1: input: SV_1, SV_2
2: output: MSV
3: $LCSV \leftarrow LCSV(SV_1, SV_2)$
4: $map \leftarrow NodeMappingGeneration(SV_1, SV_2, LCSV)$
5: restore the MSV from map (map contains label and order information)

3.2　MSV

複数の医療機関間の頻出医療指示シーケンスパターンを比較するために，MSV を導入する．

定義 9 (併合シーケンスバリアント (MSV))．シーケンスバリアント SV_α, SV_β とその $LCSV$ に対し，併合シーケンスバリアント MSV を下記の条件を満たすシーケンスバリアントと定義する．$\forall sv_i, sv_j \in SV_\alpha \cup SV_\beta, sv_i, sv_j \in MSV$ かつ $sv_i \prec_{MSV} sv_j \iff sv_i \prec_\alpha sv_j \wedge sv_i \prec_\beta sv_j$. $\forall msv_i, msv_j \in MSV, msv_i \prec_{MSV} msv_j \Longrightarrow (msv_i, msv_j \in SV_\alpha \wedge msv_i \prec_\alpha msv_j) \vee (msv_i, msv_j \in SV_\beta \wedge msv_i \prec_\beta msv_j)$. 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 を算出するためのアルゴリズムを以下に述べる．

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，SV_1 を青で，SV_2 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 4 でサーチをするとき，一度出現し

Algorithm 3 NodeMappingGeneration

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

Algorithm 4 SearchNext

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

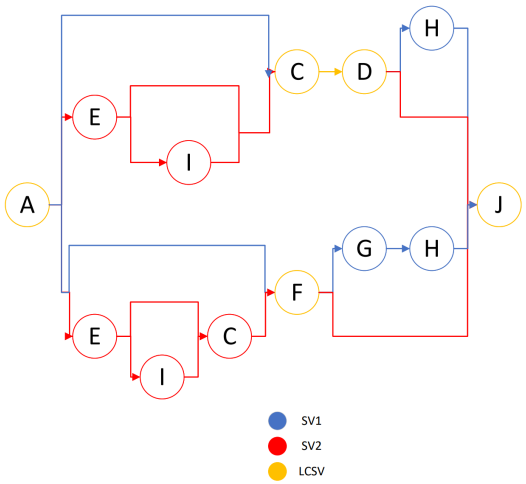

図 3　SV_1 と SV_2 の MSV

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

た *LCSV* に属しているノードがまた見つかると，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，宮崎大学医学部附属病院と宮崎市郡医師会病院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，頻出パターンを求める．得られた頻出パターンをシーケンスバリアントに併合する．次に LCSV のアルゴリズムを適用し，各病院の 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，この実験は最長のパターン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病院で 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，本研究で宮崎大学医学部附属病院ならびに宮崎市郡医師会病院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 [16] 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

	宮崎大学医学部附属病院	宮崎市郡医師会病院
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宮崎大学医学部附属病院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．両病院で共通しているため，PCI を選んだ．実験に関する情報は表 2 に示す．

4.3　実 験 結 果

4.3.1　シーケンスバリアントの抽出

両病院のデータセットが違うため，同じ処置でも用語が違う場合がある．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は表 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．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 [18] 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，実際に宮崎大学医学部附属病院では 0.15 と 0.2，宮崎市郡医師会病院では 0.25 と 0.3 を採用した．抽出されたシーケンスバリアントは図 4，図 5，図 6 と図 7 に示す．

病院毎に総点数を算出する．宮崎大学医学部附属病院の 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．宮崎大学医学部附属病院の minsup=0.2 の場

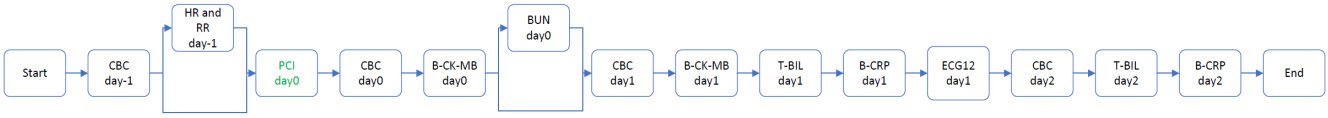

図 4　宮崎大学医学部附属病院の minsup=0.15 の SV

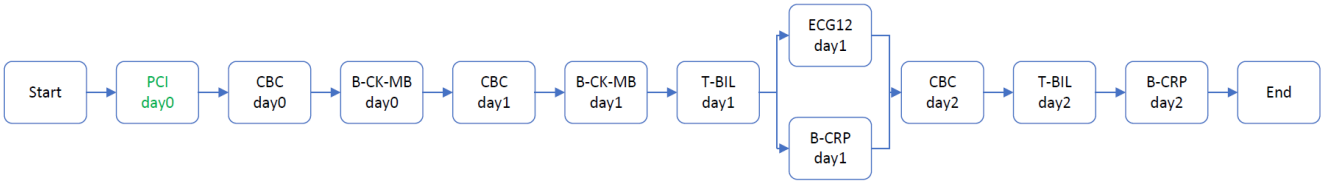

図 5　宮崎大学医学部附属病院の minsup=0.2 の SV

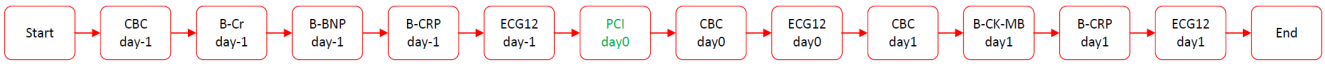

図 6　宮崎市郡医師会病院の minsup=0.25 の SV

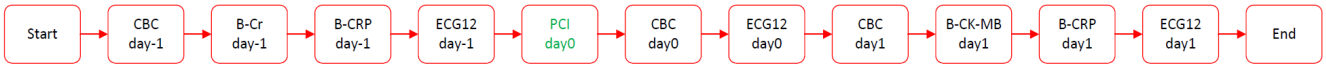

図 7　宮崎市郡医師会病院の minsup=0.3 の SV

表 3　処置の詳細

ノードネーム	レセプト電算コード	点数	UMH [1]	MMAH [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	PCI	PCI	経皮的冠動脈インターベンション
	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			

[1] UMH は宮崎大学医学部附属病院の処置の名称を指す
[2] MMAH は宮崎市郡医師会病院の処置の名称を指す

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．宮崎市郡医師会病院の minsup=0.25 の場合，総点数は 25,524 となる．宮崎市郡医師会病院の minsup=0.3 の場合，総点数は 25,388 となる．今回の場合は，シーケンスバリアント間および医療機関間の差は 0.5%以下で，比較的小さい．

宮崎市郡医師会病院のシーケンスバリアントにはパスが一つしかない．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．

4.3.2　LCSV の導出

いくつかの minsup の組み合わせを試し，LCSV を算出する．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」は違う．

実際に，宮崎市郡医師会病院の minsup=0.25 と minsup=0.3 の SV には細かい違いしかないため，導出した LCSV は同じである．

図 8　宮崎大学医学部附属病院 (0.15) と宮崎市郡医師会病院 (0.25&0.3) の LCSV

図 9　宮崎大学医学部附属病院 (0.2) と宮崎市郡医師会病院 (0.25&0.3) の LCSV

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 に対応するノードの上部に示している．

図から簡単に処置の情報が得られる．黄色の部分は両病院で共通のパターンで，赤と青の部分から両病院それぞれがどのように診療を行うかも分かる．赤と青の部分を比較することで，両病院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)」は宮崎大学医学部附属病院では何回か出てくるが，宮崎市郡医師会病院では出てこない．また，「ECG12(12 誘導心電図)」は宮崎市郡医師会病院では毎日実施されるが，宮崎大学医学部附属病院では一回しか実施されない．

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，宮崎大学医学部附属病院では 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，宮

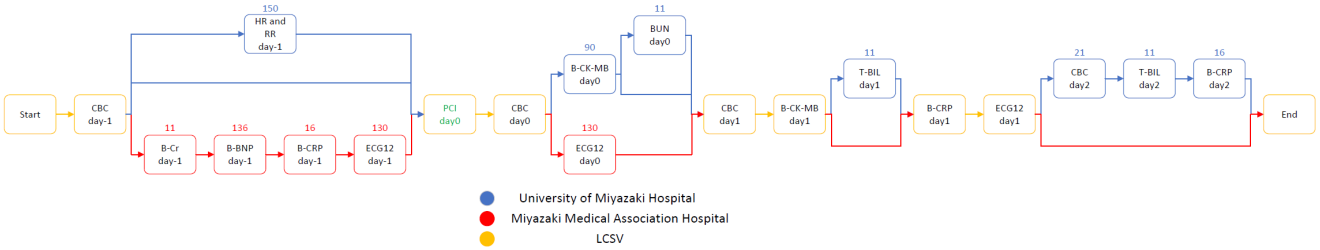

図 10　宮崎大学医学部附属病院 (0.15) と宮崎市郡医師会病院 (0.25) の MSV

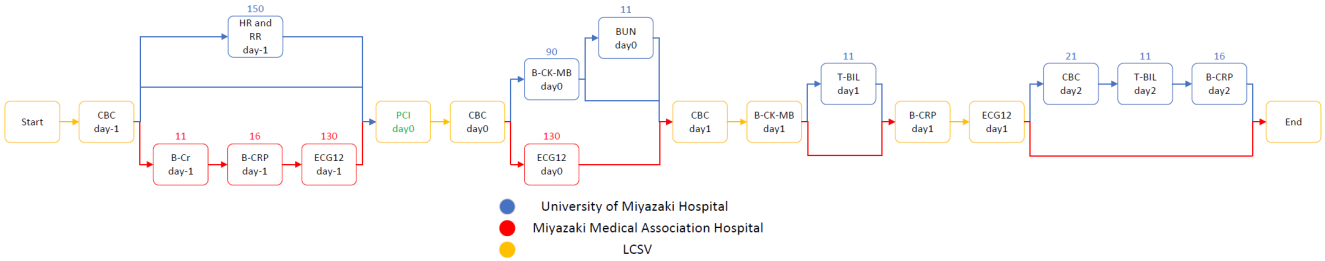

図 11　宮崎大学医学部附属病院 (0.15) と宮崎市郡医師会病院 (0.3) の MSV

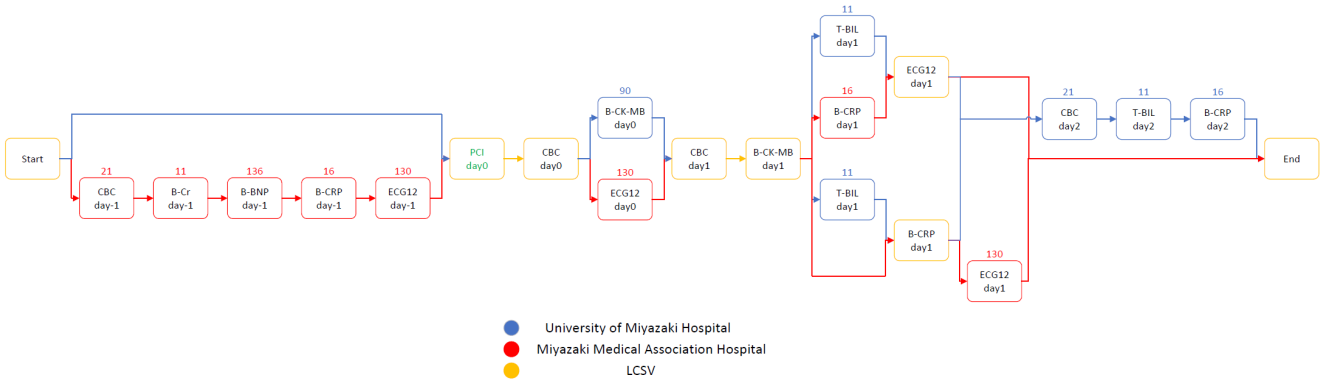

図 12　宮崎大学医学部附属病院 (0.2) と宮崎市郡医師会病院 (0.25) の MSV

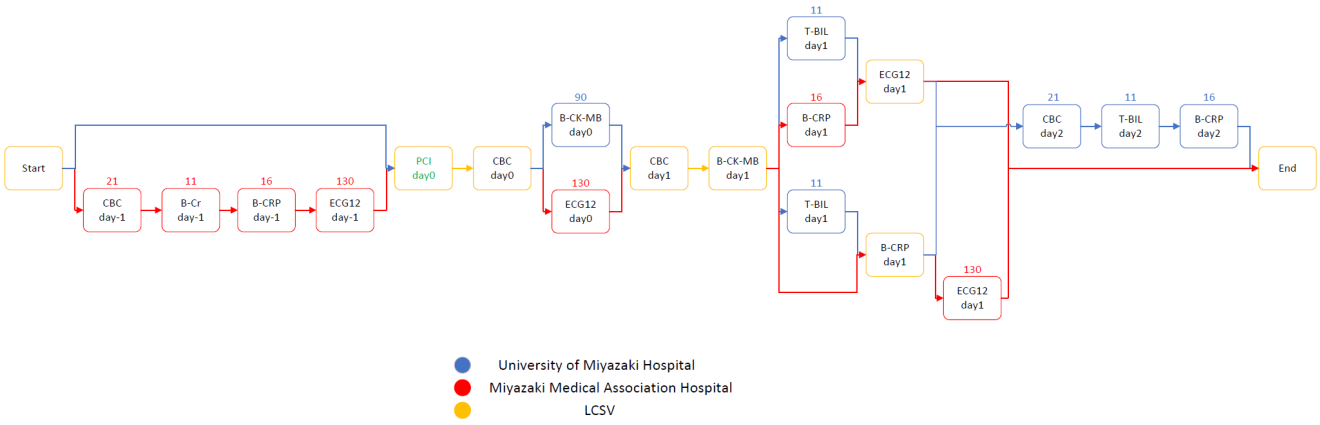

図 13　宮崎大学医学部附属病院 (0.2) と宮崎市郡医師会病院 (0.3) の MSV

崎市郡医師会病院では，3 点では不十分で毎日 12 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，宮崎市郡医師会病院ではたまに実施されるが，頻度が低いため抽出されなかった．宮崎大学附属病院の場合に，パッケージとして含まれている可能性が考えられる．

5　考　　察

一つの病院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，複数の病院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二つのシーケンスバリアント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

本研究は複数の医療機関の電子カルテを解析し，医療の改善支援に貢献することを目指した．電子カルテ中の医療指示に着目し，医療機関の共通のシーケンスを示す LCSV と，LCSV に基づき差異を示すための MSV の概念とその算出アルゴリズムを提案した．

実際に，2 種類の minsup の設定を用いて，宮崎大学医学部付属病院と宮崎市郡医師会病院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．

6.2 今後の課題

今後の課題として，まず適切な minsup を選択する指針に関して検討を進める必要がある．また，今後利用を想定している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ことから，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 に関しても検討を行う必要がある．また，提案手法は様々な施術を対象にすることができるが，時間の関係で PCI のみの適用となった．今後他の施術に対する適用も必要である．

さらに，シーケンスバリアント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必要がある．また，千年カルテプロジェクトに参加するより多くの医療機関に対して，本手法を適用するとともに，3 医療機関以上の扱いに関しても検討する．

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．本研究は宮崎大学医学部附属病院と宮崎市郡医師会病院の電子カルテデータを用いている．これは宮崎大学の HP [16] 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．

文　　献

[1] 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 年 3 月.

[2] 吉原博幸. 千年カルテプロジェクト：本格的日本版 EHR と医療データの 2 次利用に向けて. 情報管理，vol.60，no.11，pp.767-778，2018.

[3] R. Agrawal, R. Srikant. Fast algorithms for mining association rules in large databases. Proceeding of the 20th International Conference on Very Large Data Bases，pp. 487-499，1994.

[4] 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. DEIM Forum 2015，G5-1，2015.

[5] Y. Chen，M. Chiang，M. Ko. Discovering time-interval sequential patterns in sequence databases. Expert Systems with Applications 25，pp. 343-354，2003.

[6] 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. Proceeding of 2001 international conference on data engineering，pp. 215-224，2001.

[7] Hieu Hanh Le, Henrik Edman, Yuichi Honda, Muneo Kushima, Tomoyoshi Yamazaki, Kenji Araki, Haruo Yokota. Fast Generation of Clinical Pathways Including Time Intervals in Sequential Pattern Mining on Electronic Medical Record Systems. Proceeding of the fourth International Conference on Computational Science and Computational Intelligence (CSCI 2017)，pp. 1726-1731，2017.12.

[8] V. P. Raju，G. S. Varma. Mining Closed Sequential Patterns in Large Sequence Databases. International Journal of Database Management Systems，vol.7，no.1，pp.29-39，2015.

[9] 山田達大，本田祐一，萱原正彬，Le Hieu Hanh，串間宗夫，小川泰右，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アント分析. DEIM Forum D1-1, 2019.

[10] Y. Honda, M. Kushima, T. Yamazaki, K. Araki, H. Yokota. Detection and visualization of variants in typical medical treatment sequences. Proceeding of the 3rd VLDB workshop on data management and analytics for medicine and healthcare. Springer，pp. 88-101，2017.

[11] 本田祐一，山田達大，萱原正彬，Le Hieu Hanh，串間宗夫，小川泰右，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. DEIM Forum D1-5, 2019.

[12] 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. DEIM Forum C25-1, 2021.

[13] P. Fournier-Viger, J. C.-W. Lin, R. U. Kiran, Y. S. Koh, and R. Thomas. A Survey of Sequential Pattern Mining. Data Science and Pattern Recognition, vol. 1, no. 1, pp. 54–77, 2017.

[14] Le, H.H., Kushima, M., Araki, K., Yokota, H. Differentially private sequential pattern mining considering time interval for electronic medical record systems. Proceedings of the 23rd International Database Engineering and Applications Symposium, pp. 95–103, 2019.

[15] David Maier. "The Complexity of Some Problems on Subsequences and Supersequences". J. ACM. ACM Press. 25 (2): 322–336, 1978.

[16] 宮崎大学医学部附属病院臨床研究支援センター. http://www.med.miyazaki-u.ac.jp/home/crsc/patient/notice/

[17] 医科診療行為告示・通知情報明細. https://shinryohoshu.mhlw.go.jp/shinryohoshu/paMenu/doPaDetailSpNext&100/

[18] 医科診療報酬点数表. https://www.ichikawa568.com/ika-sinryouhousyu-tensuuhyo.html/