2004－DBS－134　(I)　(1)
2004／7／13

Web アクセスログ分類のための Event Vector Sequence 法とその評価

赤塚厚司[†]　鈴木　優[††]　川越恭二[††]

近年インターネットの利用者の増加に伴い，Web アクセスログが膨大な量へと増加している．ここで，Web サイトの運営者にとって，Web アクセスログから利用者の閲覧パターンを知ることは Web サイトを構築，運営する際に重要な要素である．ところが，現在これら大量の Web アクセスログが十分に分析されているとはいえない．そこで本稿では，膨大な Web アクセスログを効率よく良く分類するために，個々の Web アクセスログをイベントとし連続的に存在する Web アクセスログをシーケンスとみなし，そのシーケンス間の類似度を高速に求めるための手法である Event Vector Sequence を提案する．本手法は，従来のシーケンス間の類似度を算出する手法である Edit distance とは異なり，シーケンスに含まれるイベントをベクトルに変換することにより，Web アクセスログの分類を高速に行う．実験により Edit distance に比べて同等の再現率を維持しながら，処理時間の短縮が可能となった．また本手法の Web アクセスログへの適用を確認した．

Evaluation of Event Vector Sequence for The Web accecss Log Classification

Atsushi Akatsuka ,[†] Yu Suzuki [††] and Kyoji Kawagoe[††]

Currently the number of Internet users has rapidly increased. For the web manager, it is an important element to get to know a user's perusal pattern from a Web access log, in case a website is built and managed. A lot of Web access log is not fully used. In this paper, we propose a new advanced method to make efficient clasification of such web access log. That is, a set of web access sequences can efficientlly be clasified using our new sequence mining method. In our new method, Event Vector Sequence is introduced, which is different from the existing method called Edit distance method. The main point of the difference is the way of calculation of similarity between sequences. In our preliminary experiments, our method is more efficient than the Edit distance method with equal values of recall ratio.

1. はじめに

インターネットの普及によって，Web サイトを閲覧，利用する人が増加し，それに伴い Web サイトに蓄積されていく Web アクセスログも膨大な量へと増加している．また，Web サイトの運営者にとって，Web アクセスログから利用者の閲覧パターンを知ることは Web サイトを構築，運営する際に重要な要素である．このため Web アクセスログの有効な利用法が必要とされている．有効な利用を行うための一つの手法として，Web アクセスログの分類がある．

本稿では分類を行う際に Web アクセスログが連続して存在することに着目する．そして Web ページを閲覧した Web アクセスログを一つのイベントとして扱うシーケンスを Web アクセスシーケンスログとする．この Web アクセスシーケンスログを分類の対象とする．

現在までにシーケンスの分類を行う方法としては，シーケンス間の類似度を求めその類似度を基に分類を行う方法が提案されている．しかし，類似度の算出方法によって分類の精度や，分類にかかる処理速度に大きな変化が生ずる．そこで本稿では，Web アクセスシーケンスログ間の類似度を求めるための手法として，Event Vector Sequence を提案する．

2. 従来方式とその問題点

シーケンス間の類似度を求める代表的な方式として Edit distance[1] がある．Edit distance とは，あるシーケンスを他のシーケンスに変換する際の変換作業（すなわちシーケンスへのイベントの挿入，シーケンスに含まれるイベントの削除）をコストとし，そのシーケンス間の最小の変換

† 立命館大学大学院 理工学研究科
Graduate School of Science and Engineering, Ritsumeikan Univ.
†† 立命館大学 理工学部
Faculty of Science and Engineering, Ritsumeikan Univ.

にかかる総コストである．

たとえば，二つのシーケンス $A = \{a, b, c\}$ と $B = \{a, b, d\}$ の類似度を算出する場合，A に含まれるイベントを B に含まれるイベントに変換する際にかかったコストを計算する．つまりシーケンス A，B 間の類似度は A からイベント c を削除し，イベント d を挿入した際の総コストとなる．仮に挿入，削除のコストをそれぞれ 1 に設定した場合，シーケンス A，B 間の類似度は 2 となる．

ところが，この方式の問題点としてシーケンス間の類似度を算出するために，膨大な計算量が必要となることが挙げられる．すなわち，シーケンスの長さであるサイズ l の二つのシーケンス間の類似度を求める際に $(l+1)\times(l+1)$ のテーブルを作成し，そのテーブルからシーケンス間の最小の変換コストを求める必要がある．このためシーケンスのサイズが増加するに従い，シーケンス間の類似度を求めるための計算量が膨大なものとなる．

そこで本稿では，この問題点を改善するために，より高速に類似度を求める手法として Event Vector Sequence を提案する．

3. Event Vector Sequence

3.1 基本的な考え方

Event Vector Sequence とは，シーケンスに含まれるイベントをベクトルに変換したシーケンスである．シーケンス間の類似度は，そのシーケンスを構成する全てのベクトル間の距離の総和とする．

シーケンスに含まれる個々のイベントは，これ以上分解することができない基本単位であると考える．

3.2 イベントのベクトル変換方式

イベントをベクトルに変換することにより，同等に扱われるべきイベントが同等に扱われない可能性がある．そのためベクトルへの変換前のシーケンスと，変換後のシーケンスに差が発生しないようにする必要がある．そこで，シーケンスに含まれるイベントの種類数をベクトルの要素数とし，イベントごとに違う要素に 1 を挿入し，その他の要素には 0 を挿入する．例えばイベントの種類 3，シーケンスのサイズ 4 のシーケンス $\{a, b, c, c\}$ が存在した時，ベクトルに変換後のシーケンスは $\{[1,0,0],[0,1,0],[0,0,1],[0,0,1]\}$ となる．なお, シーケンス間の類似度は，各々のイベントに対応するベクトル間のユークリッド距離の総和とする．ここで，サイズ m 個，種類 n 個の二つのシーケンス X，Y の間の類似度を次式で定義する．

$$\sum_{j=1}^{m}\sqrt{\sum_{i=1}^{n}(X_{ij}-Y_{ij})^2} \tag{1}$$

ただし, X_{ij}，Y_{ij} は各々シーケンス X，Y の j 番目のイベントのベクトルの i 番目の要素を示す．

3.3 シーケンス的要素を含めた変換方式

3.2 節で述べた基本的な変換方式では，変換によって各イベント間に差が発生しない変換方式は実現できた．しかし，シーケンスに含まれるイベントの前後関係を示す要素は含まれていない．また，ベクトルが相違する場合の距離は常に 1 となり，シーケンス間の類似度は必ず相違するイベント数と等しくなる．これでは十分に類似性を示すことができないと考えられる．一方，イベントの前後関係をベクトルに反映させることにより，シーケンスをベクトルとして正確に表現することが可能となるとが考えられる．そこで，イベントをベクトルに変換する際に，変換する対象のイベントの前後に存在するイベントを考慮する拡張方法を以下に示す．

Step 1. ベクトルへの変換の際に，前後関係を考慮するイベント数をウインドウサイズ w として設定する．

Step 2. 変換の対象となっているイベントに対応しているベクトルの要素に 1 を挿入する．その他の要素には 0 を挿入する．つまり 3.2 節で述べた基本的な変換方法である．

Step 3. 変換の対象となっているイベントの $(0 < k < w)$ 個前のイベントに対応しているベクトルの要素に $(w-k)/w$ を挿入する．ここで挿入を行う要素が Step 2 で行った作業によって 1 が挿入されている場合にはこの作業は行わない．

Step 4. 変換の対象となっているイベントの $(0 < k < w)$ 個後のイベントに対応しているベクトルの要素と $(w-k)/w$ の和を挿入する．ここで挿入を行う要素が Step 2 で行った作業によって 1 が挿入されている場合にはこの作業は行わない．

ここで，w，k はともに整数である．

たとえば，シーケンス $\{a, b, c, c\}$ が存在し，ウ

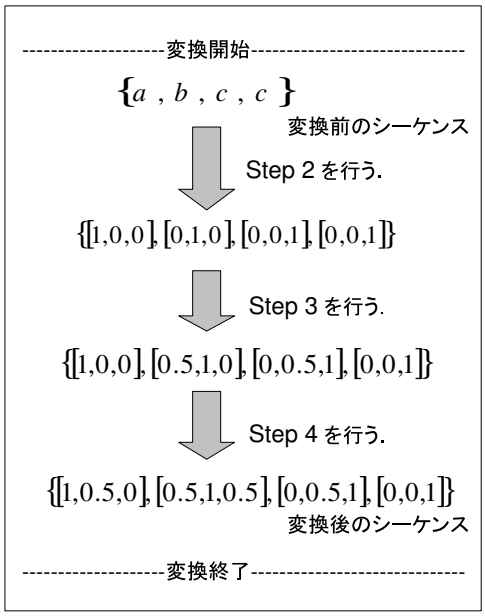

図 1 拡張方法を用いたベクトル変換例
Fig. 1 An example of vector conversion using our extended method.

インドウサイズが 2 の場合の変換の流れを図 1 に示す．ここで示されている Step は上記の拡張方法の Step と対応している．変換を行ったシーケンス間の類似度は 3.2 節で述べた式 (1) を用いる．

上記の拡張方法により，シーケンスに含まれるノイズを除去する効果を得ることができると考える．

4. 試作システムの実験と評価

本手法の有用性を検証するために，試作システムを作成した．試作システムは，複数存在するシーケンスをファイルから読み込み，従来手法，本手法のいずれかを用いてシーケンス間の類似度を求め，その類似度から分類[2] を行い，処理時間も表示するシステムである．なお，従来手法とは 2 章で説明した Edit distance を類似度とする手法である．求められた類似度からの分類を行う手法としては，階層的クラスタリングの一手法である最長距離法[3)4)] を用いた．

本実験で用いたテストデータはランダムに出現するイベントの集合をシーケンスとしたものであるが，二つのシーケンスごとに同じような偏りを持たせることにより，正解集合を判断できるように作成した．本実験で用いた再現率の

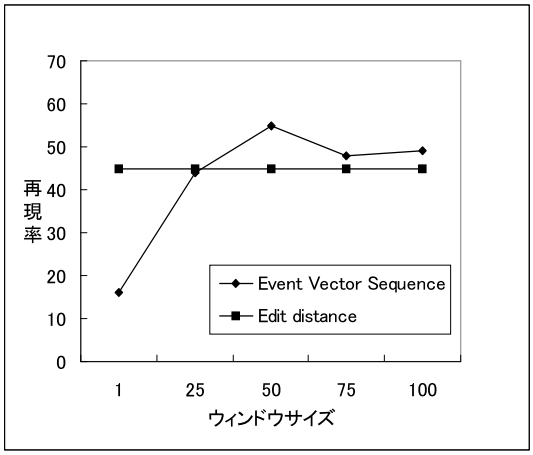

図 2 ウインドウサイズの違いによる平均再現率の変化
Fig. 2 Average recall ratio with window size.

定義を以下の式に示す．

$$R = \frac{O}{C} \tag{2}$$

ここで，R を再現率，O は抽出された正解集合数，C は正解集合数とする．正解集合数はテストデータ作成時に決定しており，正解集合数は完全に正解として分類されたものとする．

4.1 再現率を用いた類似度の精度の評価

4.1.1 実 験 概 要

シーケンス数 10，サイズ 100，種類 100 のテストデータを 20 集合用いて実験を行った．ウインドウサイズを変化させた場合の再現率の平均の変化を Event Vector Sequcnce と Edit distance を用いて調べその結果の比較を行った．

4.1.2 実験結果および考察

ウインドウサイズの違いによる再現率の変化を示している図 2 からウインドウサイズが 50 の時が最もよい値を示していることがわかる．また，20 件のテストデータの再現率の標準偏差が図 3 に示されており，この場合も再現率と同様にウインドウサイズが 50 の時が最も標準偏差が小さく良い値を示しているといえる．よって，上記の結果からウインドウサイズには 50 つまり，サイズの 50 %が最も適しているといえる．以降，本章の Event Vector Sequence を用いる実験では，ウインドウサイズをサイズの 50 %とする．

4.2 類似度の算出にかかる処理時間の評価

4.2.1 実 験 概 要

シーケンス数 10，種類 100 のテストデータを用いて実験を行った．サイズを変化させた場合の

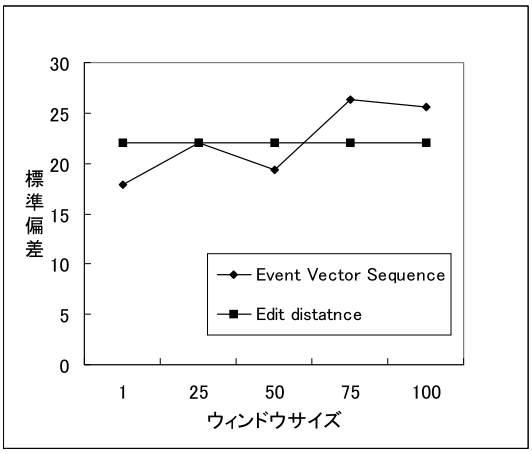

図 3 ウインドウサイズの違いによる再現率の標準偏差の変化
Fig. 3 Standard deviation of recall ratio with window size.

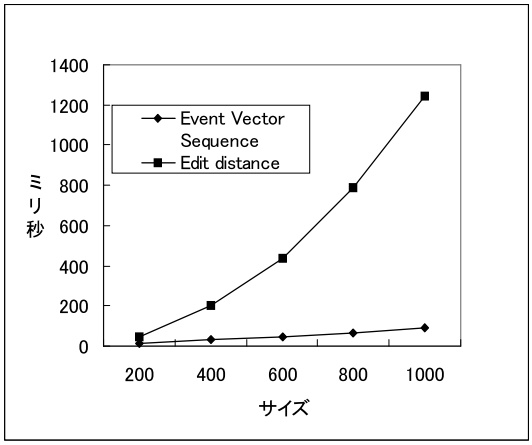

図 4 サイズの違いによる処理時間の変化
Fig. 4 Processing time with data size.

処理時間の変化を Event Vector Sequcnce と Edit distance 用いた場合において調べその結果の比較を行った.

次にシーケンス数 10 , シーケンスサイズ 1000 のテストデータを用いて種類の変化による処理時間の変化を Event Vector Sequcnce と Edit distance を用いた場合において調べその結果の比較を行った.

4.2.2 実験結果および考察

処理時間の変化の実験の結果を示している図 4, 図 5 からシーケンスのサイズ, シーケンスに含まれるイベントの種類数を変化させても, Event Vector Sequence を用いた場合に処理時間が短縮したことが分かる. 特にシーケンスサイズが増加した場合に Edit distance を用いた場合と比較した場合, 短い処理時間を維持している.

4.3 シーケンス数の違いによる類似度の精度の評価

4.3.1 実 験 概 要

サイズ 100 , 種類 100 のテストデータを 20 集合用い, シーケンス数の違いによる再現率の変化を Event Vector Sequcnce と Edit distance を用いて調べ, 比較を行った.

4.3.2 実験結果および評価

シーケンス数を変化した場合にどのように変化するかを検証した実験結果が図 6 に示されているが, シーケンスが増加するにしたがい再現率が低下するのは, シーケンス数が少なすぎたための結果であり, ある一定以上のシーケンス数が存在した場合に, 一定の再現率を保っていることが確認できる.

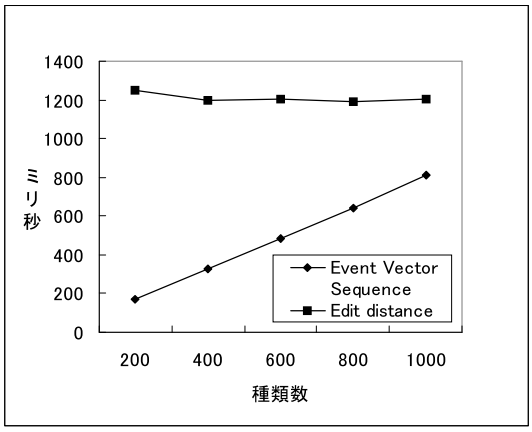

図 5 イベントの種類数の変化にによる処理時間の変化
Fig. 5 Processing time with the number of event types.

4.4 Web アクセスシーケンスログテストデータを想定した予備実験

4.4.1 Web アクセスログへの適用

本実験を行う際に, 本手法の Web アクセスログへの適用を考慮する必要がある. 本稿では適用方法の一つに閲覧者の分類を挙げる. 閲覧者ごとの分類を可能とするため, Web アクセスログを抽出して, ベクトルに変換する際に, WWW サーバが提供している一つの Web サイトに存在する一つのページを一つのイベントに関連付ける. そしてアクセスしてきた IP アドレスごとにシーケンスとして扱い分類を行う.

4.4.2 実 験 概 要

これまでの実験において用いていたテストデータは偏りのあるランダムなテストデータであった

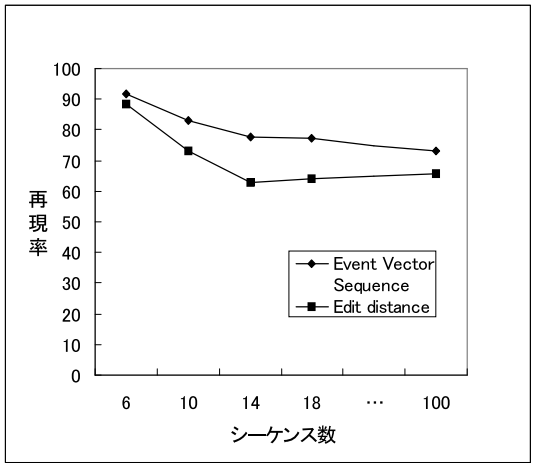

図 6 シーケンス数の違いによる平均再現率の変化
Fig. 6 Average recall ratio with the number of sequences.

表 1 パターンを含んだテストデータを用いた平均再現率と標準偏差
Table 1 Average recall ratio and its standard deviation when test data with some patterns is used

	Event Vector Sequence	Edit distance
平均再現率	77.2%	75.6%
標準偏差	5.35	11.38

が，Event Vector Sequence を用いて算出された類似度の Web アクセスシーケンスログに対しての適応性を考慮する必要がある．そこで Web アクセスシーケンスログを想定したデータ，つまり一定のパターンをランダムなシーケンスに含ませたテストデータを用いて実験を行った．サイズ 50，種類 100，シーケンス数 50 個のテストデータを用いた．ここで，テストデータのサイズは Web アクセスのシーケンスの長さ，種類は Web サイトあるいはページ数，シーケンス数は Web アクセスシーケンス数をそれぞれ想定したものである．このテストデータを 10 集合用いて実験を行い再現率を調べた．

4.4.3 実験結果および考察

Web アクセスシーケンスログテストデータを想定して行った実験結果は表 1 に示されているが，Event Vector Sequence を用いた場合と Edit distance を用いた場合を比較した場合，再現率，標準偏差ともによい値を示している．また 4.3 節の実験結果も踏まえて本実験の結果を検証した場合も Event Vector Sequence が有効であるということがいえる．

5. Web サイトを想定した評価

実際の Web サイトを想定して作成したシーケンスを用いて Event Vector Sequence と Edit distance の評価を行った．ここで実際の Web サイトの想定とは，Web サイトが階層構造を持っていることに着目し，階層構造を持ったサンプルの Web サイトを想定することである．その Web サイト内のページを参照した Web アクセスログをシーケンスとして扱った．Web サイト内のページの参照に偏りを持たせることにより，Web アクセスシーケンスログの正解集合を判断できるように作成した．

5.1 評価尺度

この実験において評価を行う際に用いた再現率は，4 章で示した式 (2) を用いるが，一つの正解集合に多数のシーケンスが含まれた際に，抽出された正解集合数を 1 もしくは 0 で判断したのでは，正確に評価を行うための再現率は算出できない．例えばシーケンス S_1，S_2，S_3，S_4，S_5，S_6，S_7 が存在した場合，正解集合が (S_1，S_2，S_3，S_4) と (S_5，S_6，S_7) であった場合，分類結果が (S_1，S_2，S_3，S_4，S_5) と (S_1，S_2，S_3，S_5) では，前者と後者と比較した場合，前者の方が適切な類似度を用いて分類を行ったといえる．そこでこの実験に用いた抽出された正解集合数を以下のように定義する．例に用いているシーケンスの集合は，上記に示す S_1 から S_7 シーケンスの集合と同一である．ここで O を抽出された正解集合数とする．

- 完全な正解集合: $O = 1$
  例: (S_1，S_2，S_3，S_4) が抽出された場合の正解集合数は $O = 1$ となる．
- 正解の部分集合: $O = \dfrac{sc}{nc}$
  ただし $\dfrac{sc}{nc} \geq \dfrac{1}{2}$ の場合に限る．
  例: (S_1，S_2，S_3) が抽出された場合の正解集合数は $O = \dfrac{3}{4}$ となる．
- 正解集合が合併した集合: $O = \dfrac{1}{mc}$
  例: (S_1，S_2，S_3，S_4，S_5，S_6，S_7) が抽出された場合の正解集合数は $O = \dfrac{1}{2}$ となる．
- 正解と不正解の集合: $O = 1 - \dfrac{ni}{no}$
  ただし，正解集合は一つの場合のみとする．
  例: (S_1，S_2，S_3，S_4，S_5) が抽出された場合

の正解集合数は $O = \frac{4}{5}$ となる．

- 正解の部分集合と不正解の集合: $O = \frac{sc}{nc} - \frac{ni}{no}$ ただし $\frac{sc}{nc} \geq \frac{1}{2}$ かつ $\frac{ni}{no} < \frac{1}{2}$ に限る．また $\frac{sc}{nc} \geq \frac{1}{2}$ となる正解の部分集合は一つの場合のみとする．
  例: (S_1, S_2, S_3, S_5) が抽出された場合の正解集合数は $O = \frac{1}{2}$ となる．

ここで，O は抽出された正解集合数，sc は抽出された正解の部分集合の要素数，nc は正解集合の要素数，mc は合併された正解集合数，ni は抽出された集合に含まれる不正解数，no は抽出された集合の要素数である．

想定した Web サイトを二種類を基に Web アクセスシーケンスログを作成し実験を行った．作成した Web アクセスシーケンスログは参照する目的のページにランダムな偏りを持たせた．また正解集合数での分類と，正解集合数以外での分類も行い，その変化を調べた．実験概要と結果を以下に示す．

5.2 実　　験 1

実験 1 ではページ数 50 の最高階層が 6 の Web サイトを想定して実験を行った．ここで最高階層とは想定した Web サイトの最も深い階層数である．データは，アクセス数つまりシーケンス数が 50，サイズが 50，ウインドウサイズ 50 で六通りの偏り，つまり正解集合数が 6 のデータを用いた．

5.3 実　　験 2

実験 2 ではページ数 30 の最高階層が 5 の Web サイトを想定して実験を行った．データは，アクセス数つまりシーケンス数が 30，サイズが 20，ウインドウサイズ 10 で五通りの偏り，つまり正解集合数が 5 のデータを用いた．

5.4 実験結果および評価

実験 1，実験 2 による，想定した Web サイトの Web アクセスシーケンスログの分類を行った結果を図 7，図 8 に示す．この図から Edit distance を用いた結果が Event Vector Sequence を用いた結果を上回ったのは，実験 1 の分類数 4 で分類した場合のみである，この場合の再現率の差も，非常に小さい．このことから，想定した Web サイトの各ページを参照した Web アクセスログの集合である Web アクセスシーケンスログ の分類

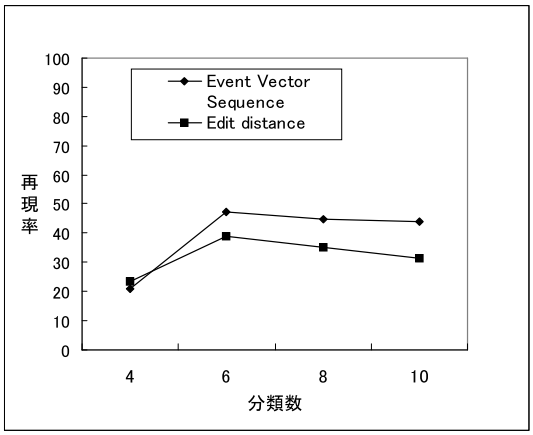

図 7　実験 1 の分類数の変化による再現率の変化
Fig. 7　Recall ratio of experiment 1 with number of classificaiton.

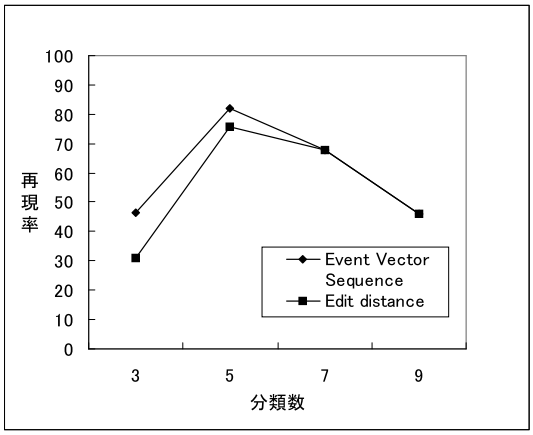

図 8　実験 2 の分類数の変化による再現率の変化
Fig. 8　Recall ratio of experiment 2 with number of classification.

を行う際に，Event Vector Sequence を用いて分類を行う手法は有効であると考える．

6. 関 連 研 究

2 章で述べた Edit distance を用いて二つのシーケンス間の類似度を DP 法[5] で求め分類を行う方法以外にも，Web アクセスログへの適用が可能と考えられるマイニング手法としては，これまでに以下の方法が提案されている．本稿で提案した Event Vector Sequence を用いてシーケンスの分類を行う手法は，以下の方法と異なる新たな手法と考える．

6.1 Association Rule

代表的なマイニング手法に，Association Rule を発見する手法がある．この Association Rule 手法をシーケンスに応用した場合，シーケンスに

含まれるイベントの中から同時に存在する比率の高いイベント組合せを Association Rule として抽出する．すなわち，多数のシーケンスから頻度の高い部分シーケンスを抽出する方法である．この Association Rule を発見するためのアルゴリズムとして，Apriori Algorithm[6] が提案されており，シーケンスに対する応用も提案されている[7]．

Apriori Algorithm ではサポート値とコンフィデンス値を用いて Association Rule を抽出する．シーケンスに応用した場合について以下に例を示す．

イベント e_1，e_2，e_3 が存在した場合を考える．サポート値は，シーケンスにイベントの組合せが存在する確率である．すなわち e_1，e_2 のサポート値とは，e_1，e_2 がすべてのシーケンスに存在する確率となる．コンフィデンス値とは，あるイベントが存在した時に，他のイベントが存在する確率である．e_1 かつ e_2 なら e_3 という Association Rule が存在した場合のコンフィデンス値は e_1 と e_2 が存在した場合の e_3 が存在する確率である．Apriroi Algorithm を高速化するためには，存在するイベントの組合せを発見していく際に，最小サポート値を設定し，最小サポート値以下の組合せに対して枝刈りを行う．これにより，不必要と考えられる組合せを排除することができ，処理時間の短縮を図る．すなわちサポート値が高くなれば処理時間は短縮されるが，有効な Association Rule が枝刈りにより排除されてしまう恐れがある，一方サポート値を低くすると処理時間は増加するが，有効な Association Rule が排除される可能性は低くなる．そのため適切なサポート値を設定する必要がある．

Event Vector Sequence を用いて分類を行ったシーケンスに対して，Apriori Algorithm を適用することによってより有効な部分シーケンスを抽出できると考えられる．

6.2 Matrix Clustering

Web アクセスログへ適用可能な有効なマイニング手法として Matrix Clustering[8] が挙げられる．Matrix Clustering は，マイニングの対象となるデータを 2 値行列で表現し，密な部分行列を抽出する手法である．Web アクセスログを対象とした場合，Web ページを行，閲覧者を列とし Web ページを閲覧した場合に 1 閲覧していない場合を 0 とする 2 値行列を作成し，その 2 値行列から密な部分行列を作成し，有効な部分集合を抽出する．

Matrix Clustering は部分集合を抽出することにより複数の閲覧者が持つパターンを抽出することができる．一方 Event Vector Sequence はシーケンス間の類似度を算出するための手法であり，シーケンスに含まれるイベントの前後関係に焦点を置き類似度を算出して，その類似度からシーケンスすなわち閲覧者の分類を行う．このように考えることで，目的に応じた選択が可能である．

6.3 Self-Organizing Map

Web アクセスログの分類が可能な方法としては，教師なしのニューラルネットワーク[9] である Self-Organizing Map[10] を用いた分類がある．Self-Organizeng Map は，多次元の要素を 2 次元のマップに映し出したものであり，このマップを用いて分類を行う．しかしこの Self-Organizing Map[10] は，イベントの前後関係を考慮することが困難である．シーケンスを扱うには，イベントの前後関係を考慮した Event Vector Sequence を用いて分類を行うほうがよいと考える．

7. おわりに

本稿では Web アクセスログを効率的に分類することが可能となる新しい手法を提案した．提案した手法では，Web アクセスログの集合を Web アクセスシーケンスログとして扱い，そのシーケンスに含まれるイベントをベクトルに変換した．ベクトルに変換したシーケンスを Event Vector Sequence とし，Event Vector Sequence 間の距離を Web アクセスシーケンスログ間の類似度として算出した．そして，算出した類似度から Web アクセスシーケンスログを効率的に分類することが可能となった．今後は以下の課題を解決する予定である．

- **大量の Web アクセスログへの適用:** 本稿では実験を行うためにテストシーケンスを作成して分類を行い，分類の精度や処理速度について検証を行った．今後，大量のシーケンスを本稿で提案した手法に適用した場合の処理時間の変化を調べ，問題点の発見および改善を行う必要がある．
- **他のシーケンスデータ分野への応用:** 本稿では Web アクセスログを対象とし研究を行った．今後，音声データや，ストリーミングデータなど，Web アクセスログ以外のデータをシーケンスとして扱い，本稿で提案した

手法に適用することが可能であると考えられるため，対象となるデータ分野を検討する必要がある．

- **他の既存手法との比較:** 6 章で関連研究について述べたが，6 章で紹介した代表的なマイニング手法と本稿で提案した手法との比較および組合せを考えていく必要がある．

参 考 文 献

1) Moen, P.: *Attribute, Event Sequence, and Event Type Similarity Notions for Data Mining*, PhD Thesis, University of Helsinki, Finland (2000).
2) Chen, M.-S., Han, J. and Yu, P. S.: Data Mining:An Overview from a Database Perspective, *IEEE Transactions on knowledge and data engineering*, Vol. 8, No. 6, pp. 866–883 (1996).
3) R., A. M.: クラスター分析とその応用, 内田老鶴圃 (1988).
4) マイケルJ.A. ベリー, ゴードン・リノフ: データマイニング手法, 海文堂 (1999).
5) V., A. A.: *Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity*, Elsevier Science Publishers B.V, chapter Algorithms for finding patterns in strings, pp. 255–400 (1990).
6) Agrawal, R. and Srikant, R.: Fast Algorithms for Mining Association Rules, *Proceedings of the 20th International Conference on Very Large Databases*, pp. 487–499 (1994).
7) Agrawal, R. and Srikant, R.: Mining Sequential Patterns, *Proceedings of the Eleventh International Conference on Data Engineering*, pp. 3–14 (1995).
8) 小柳滋, 久保田和人, 沖瀬明彦: Matrix Clustering:CRM 向けの新しいデータマイニング手法, 情報処理学会論文誌, Vol. 42, No. 8, pp. 2156–2166 (2001).
9) Sun, R. and Giles, L.(eds.): *Sequence Learning: Paradigms, Algorithms and Applications*, Springer LNAI 1828 (2001).
10) T. コホネン: 自己組織化マップ, シュプリンガー・フェアラーク東京株式会社 (1996).