

Webアプリケーションを用いた情報技術演習における 学習履歴の類似度算出方法の提案

木村 友哉¹ 新村 正明²

概要: 本研究は、オンライン形式の情報技術演習において、つまづいている学習者群の抽出を目的とする。つまづいている学習者には類似した行動パターンがあると推測し、行動パターンから解析して学習者の学習履歴間の類似度を算出する手法を提案する。なお、本研究においては Web アプリケーションを用いた演習を対象とすることとし、Web アクセス履歴の解析手法を拡張することで、学習履歴間の類似度を算出する方法を考案した。さらに、C 言語のオンライン演習に対して、提案した類似度算出に基づいた学習者のグループ化を行い、分類したグループがつまづいている学習者群を抽出できているか、演習後のアンケートから評価を行った。

キーワード: 学習状況可視化, 情報技術演習, 学習行動履歴, 類似度, クラスタリング

1. はじめに

プログラミング教育を目的とした講義では、教授者が学習者に対して課題を与え、学習者はその課題に個別で取り組むといった演習形式が一般的である。また、大学で行われる多人数講義の場合、教授者一人に対して学習者の人数が多いことから、学習者の学習状況を把握することが困難である。教授者はつまづいている学習者に対して状況に応じて学習指導を行う必要がある。

このような教授者の学習指導を支援するものとして、先行研究 [1][2] において学習者の学習状況を可視化するアプリケーションが開発された。しかし、この可視化アプリケーションでは、学習者の学習状況を可視化しただけであって、つまづいている学習者を検出するには教授者の主観によって判断しているといった問題点がある。

本研究では、つまづいている学習者群を抽出するために、学習者の学習履歴から類似度を算出する手法を提案する。

2. 先行研究

井垣ら [1] はプログラミング演習時におけるコーディング過程を記録し、可視化して教授者に提示するシステムを提案した。このシステムは Web 上で動作するオンラインエディタとコーディング過程ビューから構成されている。

オンラインエディタでは学習者のコーディング過程における、文字入力、コンパイル、実行、提出の行動を記録し、コーディング過程ビューは課題の進み具合や、学習者の進捗遅れを可視化して教授者に提示できる。

また、横山ら [2] は Web ブラウザ上で動作する可視化アプリケーションを開発した。これは学習者側の演習システムから xAPI[3] を用いて学習履歴を収集しデータベースに蓄積する。その学習履歴を教授者側に可視化することによって、学習者全体の進捗状況の把握ができるといったシステムである。この概要図を図 1 に示す。この可視化アプリは学習者が行ったコマンドの種類と時間がグラフ上に表示されるようになっている。また、このシステムでの演習進捗状況の可視化画面を図 2 に示す。この画面では 5 つのコマンドが異なる色で定義され、学習者ごとに時間軸に行動履歴が可視化されている。このように学習者全体の学習状況を把握することを可能とした。

この 2 つの先行研究は、共に情報技術系の演習を Web アプリケーションとして実現し、その学習進捗状況の可視化を行っている。そこで、本研究でも、Web アプリケーションを用いた情報技術演習を対象とする。

¹ 信州大学
Shinshu University

² 信州大学大学院工学系研究科
Department of Computer Science, Shinshu University

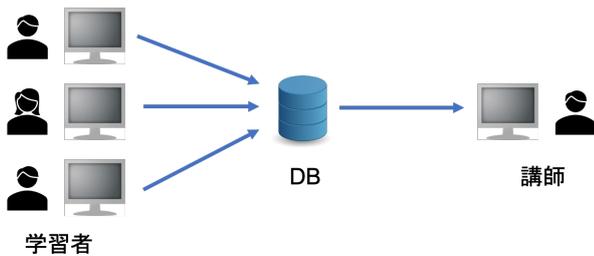


図 1 横山らの先行研究の概要図

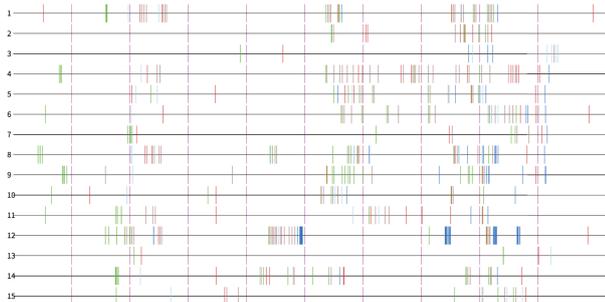


図 2 可視化画面 [eg. [2] の図 5 より転載]

3. 本研究の目的

教授者にはつまづいている学習者に対し状況に応じて学習指導することが求められるが、前節で紹介した先行研究 2 つは、学習者の学習状況を可視化しただけであって、つまづいている学習者を検出するには教授者が可視化画面を見て判断していた。そのため、つまづいている学習者を検出しづらいといった問題点がある。本稿で「つまづいている学習者」は教授者の想定よりも遅れていて学習指導が必要な状態の学習者と定義する。

先行研究のつまづいている学習者を検出しづらいという問題点の解決のため、本研究では、つまづいている学習者をグループ化することで、つまづいている学習者を検出し易くすることを目的とする。この目的を達成するために、学習者の学習履歴間における類似度算出方法を提案する。

4. 提案手法

つまづいている学習者は類似した行動パターンをとっているのではないかと考え、学習行動のパターンが似ている学習履歴を分類し、つまづいている学習者をグループ化することとする。グループ化することで、つまづいている学習者群だけを抽出し、学習指導が必要とされる学習者群だけを表示することが可能になる。これにより、先行研究の問題点を解決することができると思われる。

4.1 Web アクセス解析

本研究では Web アプリケーションを用いた情報技術演習を対象に可視化を行っているため、Web アクセスの解析

に使われる手法を用いる。

Web アクセス解析において、複数の Web ページが相互にリンクする構造が多く、ページ間の遷移を記録し利用者の閲覧パターンを解析するが多い。この Web ページを閲覧した Web アクセスログを一つのシーケンスとみなし、このシーケンスの解析によって利用者の特徴を抽出する研究が行われている [4][5]。

4.2 Event Vector Sequence 法

赤塚らはこの Web アクセス解析の手法として Event Vector Sequence 法 [6] (以下 EVS 法と略す) を提案した。EVS 法とは膨大な Web アクセスログを効率よく分類するために、個々の Web アクセスログをイベントとして連続的に存在する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算出する手法である。EVS 法の簡単な概要図を図 3 に示す。以下、EVS 法の説明を行う。

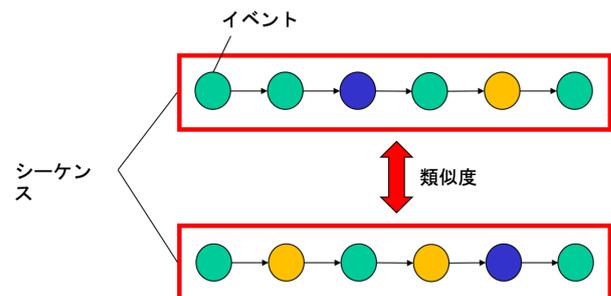


図 3 EVS 法の概要図

4.2.1 イベントベクトルへの変換

Event Vector Sequence とは、1 つの Web ページへのアクセスをイベントとみなし、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シーケンスのことである。例えば、 a, b, c のイベントがありシーケンスサイズ 4 の a, b, c, c のシーケンスが存在した場合、 $[1, 0, 0], [0, 1, 0], [0, 0, 1], [0, 0, 1]$ とイベントベクトルに変換できる。

4.2.2 ウィンドウ処理

イベントの前後関係を距離に反映するためにウィンドウ処理を行う。このウィンドウ処理は、まずウィンドウサイズ w を定め、あるイベントの次元の値が 1 であった場合、前後 w の範囲内にある k 番目 ($0 < k < w$) のイベントの該当する次元に $(w - k)/w$ を加算する処理を行う。イベントベクトル変換の一連の流れを図 4 に示す。

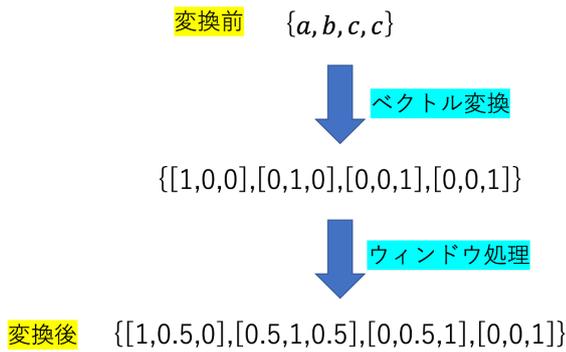


図 4 イベントベクトル変換例

4.2.3 類似度算出方法

シーケンス間の類似度は、各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し、以下に示す数式で定義されている。なお、シーケンスは m 個の n 次元ベクトルにより構成されるものとし X_{ij}, Y_{ij} はそれぞれ、イベントベクトルシーケンス X, Y の i 番目のベクトルの第 j 次元の値とする。

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

4.3 本研究における EVS 法の拡張

本研究では学習履歴間の類似度算出に、先に説明した EVS 法を用いることとする。しかし、学習行動と Web アクセスには異なる点があるため、EVS 法の拡張を行うこととした。以下拡張方法を説明する。

4.3.1 シーケンスサイズの単位の変更

本研究では、情報技術演習として C 言語のプログラミング演習を対象とする C 言語のオンライン形式の演習で課題の提出期限が設定されており、期限内に各自で取り組むため、取り組む時間帯と合計時間が学習者によって異なる。学習者が行う行動は C 言語の演習のものであるため、コンパイル成功、コンパイル失敗、の 2 種類である。今回 EVS 法に用いる要素は、コンパイル成功、コンパイル失敗、の 2 種類の行動と、その行動が行われた時間の履歴である。

C 言語のプログラミング演習では課題がいくつか出題されているが、本研究では一つの課題に対する学習履歴を扱う。本研究での EVS 法におけるシーケンスは、学習者一人の学習履歴とする。オンライン形式での演習であるため、演習開始時間が学習者によって異なる。そのため、学習者それぞれの最初の行動を演習開始時間とみなす。

また、EVS 法的前提として、シーケンスサイズが一定でないとこの方法を適用できない。しかし、本研究で扱うデータでは、学習者によって行動数が大幅に違うため、行動自体をイベントとすることが難しい。そのため、シーケンスサイズを一定時間で揃えることにより、単位時間で区

切った際の行動をイベントとする。そこで、学習者それぞれの最初の行動を演習開始時間とみなし、最初の行動からの経過時間をシーケンスサイズとする。学習者が一度に連続して演習に取り組む時間として 1 時間程度が妥当だと考え、シーケンスサイズを 1 時間に揃えて EVS 法を適用した。

本研究での EVS 法を図 5 に示す。シーケンスはオレンジ色の四角で囲まれた部分、横軸が時間軸、行動が楕円形、イベントが青色で囲まれた部分である。なお、行動はコンパイル成功が水色、コンパイル失敗が赤色で表現している。

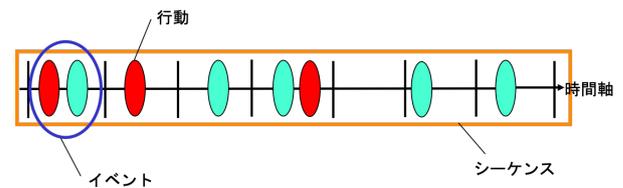


図 5 本研究での EVS 法

4.3.2 イベントベクトルの拡張

イベントを単位時間で区切ることにより行動のないイベントが生まれる。そのため、従来の EVS 法に加え、新たに行動なしイベントを追加することとする。行動がコンパイル成功、コンパイル失敗の 2 種類であるため、行動なしイベントを追加する用に次元を 1 足して 3 次元のベクトルで表す。行動をそれぞれ、コンパイル成功を $[1,0,0]$ 、コンパイル失敗を $[0,1,0]$ とすると、単位時間内にコンパイルの双方を実行している場合には $[1,1,0]$ のように対応する要素全ての値を 1 として表現する。行動がないイベントは $[0,0,1]$ と表現する。その後ウィンドウサイズを設定しウィンドウ処理を行う。このイベントベクトルへの変換例を図 6 に示す。類似度の算出は従来の EVS 法と同じ算式で行った。

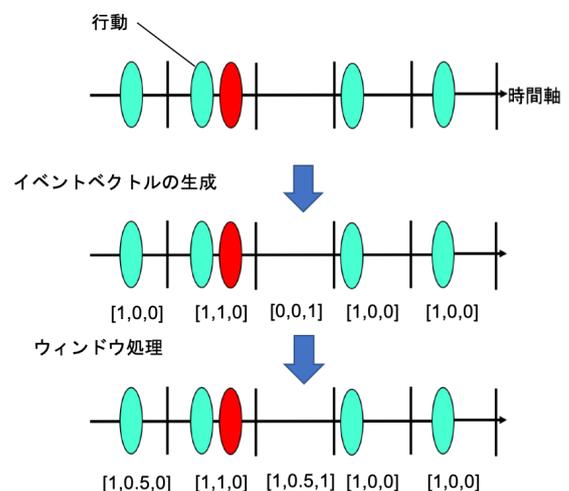


図 6 イベントベクトル生成の流れ

5. 結果

イベントの単位時間、ウィンドウサイズのパラメータを変えて算出した類似度でクラスタリングを行った。なおクラスタリングには最遠隣法で行った。その結果を図 7, 図 8, 図 9, 図 10 に示す。図 7 が単位時間 60 秒ウィンドウサイズ 2, 図 8 が単位時間 60 秒ウィンドウサイズ 3, 図 9 が単位時間 120 秒ウィンドウサイズ 2, 図 10 が単位時間 120 秒ウィンドウサイズ 3 で行ったものである。

図の左がクラスタリングの結果に基づくデンドログラムの樹形図を示し、右が 1 時間分の学習履歴を示したものである。右の学習履歴では横軸が時間軸、縦軸が学習者、プロットされた点が行動を表す。行動のコンパイル成功が水色、コンパイル失敗が赤色でプロットされている。図の A, B, C は次章で説明する。

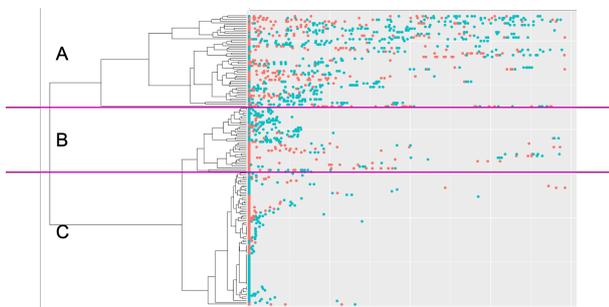


図 7 単位時間 60 秒, ウィンドウサイズ 2

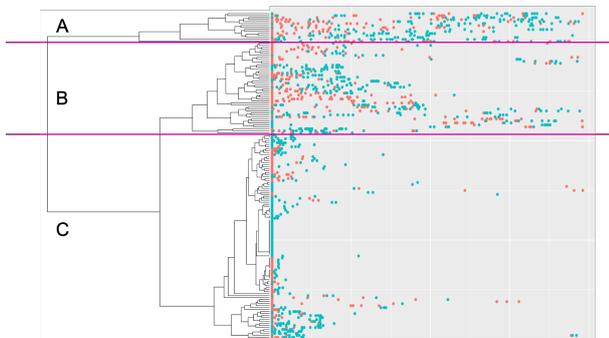


図 8 単位時間 60 秒, ウィンドウサイズ 3

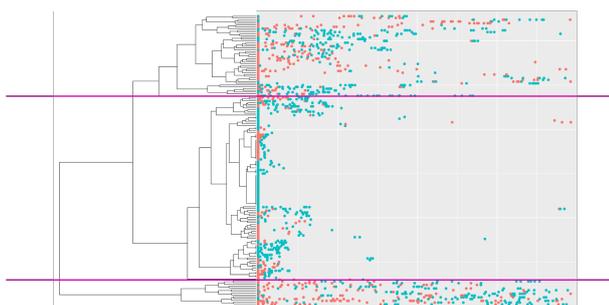


図 9 単位時間 120 秒, ウィンドウサイズ 2

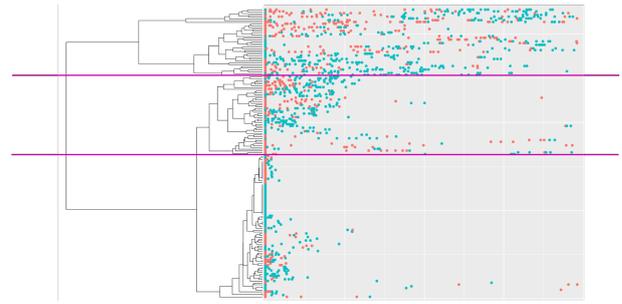


図 10 単位時間 120 秒, ウィンドウサイズ 3

6. 評価

6.1 評価方法

前節で提案した類似度の算出方法を評価するため、結果として示した図をデンドログラムの分類によって 3 グループに分けた。図に示した A, B, C はグループの名前とする。

学習者が演習を終えた後に、自己評価でアンケートに回答してもらっている。このアンケートの結果から、デンドログラムによって分類したグループ A, B, C が、つまづいている学習者をグループ化できているかどうかの評価を行う。

自己評価のアンケートは以下の内容で行った。アンケートは「この課題はうまく解けたか?」という質問に対して、

- 解答方針がすぐに分かり、プログラムもすぐに作成できた。
- 解答方針はすぐに分かったが、プログラムの作成に手間取った。
- 解答方針はすぐに分かったが、プログラムにできなかった。
- 解答方針に手間取ったが、プログラムはすぐに作成できた。
- 解答方針に手間取って、プログラムの作成にも手間取った。
- 解答方針に手間取って、プログラムできなかった。
- 解答方針が立たなかった。

の 7 項目の回答があり、どれか一つを選択するようになっている。

上記の回答に対して、最終的にプログラムを書くことができたかどうかによって評価したため、この 7 項目の回答は「プログラムがすぐに作成できた」、「プログラムの作成に手間取った」、「プログラムができなかった・解答方針が立たなかった」の 3 段階で表現することにした。3 段階に分けた学習者群をそれぞれ学習者群 1, 学習者群 2, 学習者群 3 と定義する。また、アンケート無回答者は学習者群 4 とした。この中で、学習指導が必要であると考えられる学習者群の優先順位は学習者群 3, 学習者群 2 である。

デンドログラムで分類したグループの学習者に対して、

この3段階での対応を見て評価する。評価として、グループA,B,Cに含まれる学習者群1,2,3の比率を全体の学習者群1,2,3の比率と比べて検定を行う。検定方法は比率の差の検定を有意水準5%で両側検定で行う。

6.2 評価結果

前節で定義した3段階評価と色の対応を表1に示す。今回アンケート無回答者は比率に考慮しない。

評価の結果を単位時間60秒の場合と単位時間120秒の場合の2つに分けて記述する。

表1 3段階評価と色の対応表

| 名前 | 3段階評価 | 色 |
|-------|--------------------------|-----|
| 学習者群1 | プログラムがすぐに作成できた | 青色 |
| 学習者群2 | プログラムの作成に手間取った | 黄緑色 |
| 学習者群3 | プログラムができなかった・解答方針が立たなかった | 赤色 |
| 学習者群4 | アンケート無回答 | 白色 |

6.2.1 単位時間60秒における評価結果

評価の結果を図11, 図12に示す。前節で回答項目を3段階に分けたものを色で区別したものと学習履歴を対応させている。

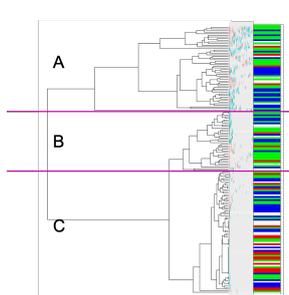


図11 単位時間60秒
ウィンドウサイズ2

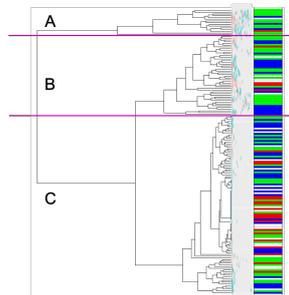


図12 単位時間60秒
ウィンドウサイズ3

全体の学習者群1, 2, 3の比率とグループA, B, Cに属する学習者群1, 2, 3の比率をウィンドウサイズ2, 3の場合において表2, 表3に示す。また、検定を行った時のp値を表4, 表5に示す。このp値が有意水準5%を下回っていれば有意さが見られる。しかし、ウィンドウサイズ2, 3の場合双方において有意さは見られなかった。

表2 単位時間60秒 ウィンドウサイズ2

| | 学習者群1 (青) | 学習者群2 (黄緑) | 学習者群3 (赤) |
|----|-----------|------------|-----------|
| 全体 | 37.5% | 44.9% | 17.7% |
| A | 40.0% | 51.1% | 16.7% |
| B | 31.3% | 59.4% | 12.5% |
| C | 38.9% | 32.2% | 28.8% |

表3 単位時間60秒 ウィンドウサイズ3

| | 学習者群1 (青) | 学習者群2 (黄緑) | 学習者群3 (赤) |
|----|-----------|------------|-----------|
| 全体 | 37.5% | 44.9% | 17.7% |
| A | 21.4% | 71.4% | 7.1% |
| B | 41.5% | 48.8% | 9.8% |
| C | 38.3% | 38.3% | 23.5% |

表4 単位時間60秒 ウィンドウサイズ2のp値

| | 学習者群1 (青) | 学習者群2 (黄緑) | 学習者群3 (赤) |
|---|-----------|------------|-----------|
| A | 0.7647 | 0.4655 | 0.1590 |
| B | 0.5082 | 0.1388 | 0.2516 |
| C | 0.8445 | 0.099 | 0.0787 |

表5 単位時間60秒 ウィンドウサイズ3のp値

| | 学習者群1 (青) | 学習者群2 (黄緑) | 学習者群3 (赤) |
|---|-----------|------------|-----------|
| A | 0.2329 | 0.0579 | 0.3152 |
| B | 0.6474 | 0.6581 | 0.2248 |
| C | 0.8909 | 0.3426 | 0.2990 |

6.2.2 単位時間120秒における評価結果

単位時間120秒の評価の結果を図13, 図14に示す。グループA, B, Cに対応した赤色の割合をウィンドウサイズ2, 3の場合において表6, 表7に示す。また、検定を行った時のp値を表8, 表9に示す。単位時間120秒においてもウィンドウサイズ2, 3の場合双方において有意さは見られなかった。

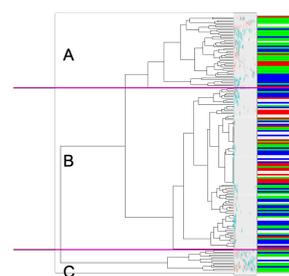


図13 単位時間120秒
ウィンドウサイズ2

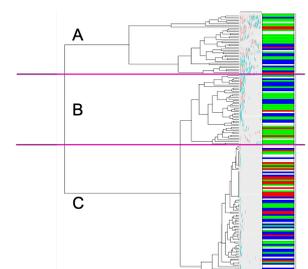


図14 単位時間120秒
ウィンドウサイズ3

表6 単位時間120秒 ウィンドウサイズ2

| | 学習者群1 (青) | 学習者群2 (黄緑) | 学習者群3 (赤) |
|----|-----------|------------|-----------|
| 全体 | 37.5% | 44.9% | 17.7% |
| A | 27.7% | 58.1% | 14.0% |
| B | 43.9% | 34.1% | 22.0% |
| C | 27.3% | 72.7% | 0.0% |

表 7 単位時間 120 秒 ウィンドウサイズ 3

| | 学習者群 1 (青) | 学習者群 2 (黄緑) | 学習者群 3 (赤) |
|----|------------|-------------|------------|
| 全体 | 37.5% | 44.9% | 17.7% |
| A | 35.3% | 53.0% | 11.8% |
| B | 34.2% | 57.9% | 7.9% |
| C | 40.6% | 32.8% | 26.6% |

表 8 単位時間 120 秒 ウィンドウサイズ 2 の p 値

| | 学習者群 1 (青) | 学習者群 2 (黄緑) | 学習者群 3 (赤) |
|---|------------|-------------|------------|
| A | 0.2509 | 0.1285 | 0.5719 |
| B | 0.3497 | 0.1192 | 0.4351 |
| C | 0.4985 | 0.0747 | 0.1277 |

表 9 単位時間 120 秒 ウィンドウサイズ 3 の p 値

| | 学習者群 1 (青) | 学習者群 2 (黄緑) | 学習者群 3 (赤) |
|---|------------|-------------|------------|
| A | 0.8117 | 0.3976 | 0.4081 |
| B | 0.7101 | 0.1547 | 0.1421 |
| C | 0.6718 | 0.1063 | 0.1451 |

7. 考察

単位時間 60 秒, ウィンドウサイズ 2 の場合において, グループ C の学習者群 2 の p 値が 0.0990 であることより, 有意さは見られなかったが, 低い値になったためグループ C に含まれる学習者群 2 の比率が他に比べて小さいと推測できる. また, グループ C の学習者群 3 の p 値が 0.0787 であることより, 有意さは見られなかったが, 低い値になったためグループ C に含まれる学習者群 2 の比率が他に比べて大きいと推測できる. よって, グループ C はプログラム作成に手間取った学習者が少なく, プログラムが作成できなかった学習者が多いグループと考えられる. したがって, 最も支援すべき学習者群を抽出することはできたと推測できる.

これに対して, 単位時間 60 秒, ウィンドウサイズ 3 の場合において, グループ A の学習者群 2 の p 値が 0.0579 であることより, 有意さは見られなかったが, 低い値になったためグループ A に含まれる学習者群 2 の比率が他に比べて大きいと推測できる. よって, グループ A はプログラム作成に手間取った学習者が多いグループだと考えられ, 2 番目に支援すべき学習者群は抽出できたが, 最も支援すべき学習者群は抽出できなかったと言える.

次に, 単位時間 120 秒, ウィンドウサイズ 2 の場合において, グループ C の学習者群 2 の p 値が 0.0747 であることより, 有意さは見られなかったが, 低い値になったためグループ C に含まれる学習者群 2 の比率が他に比べて大きいと推測できる. よって, グループ C はプログラム作成に手間取った学習者が多いグループだと考えられ, 2 番目に支援すべき学習者群は抽出できたが, 最も支援すべき学習

者群は抽出できなかったと言える.

また, 単位時間 120 秒, ウィンドウサイズ 3 の場合においては, 全ての p 値が大きかったため有用性は見られなかった.

単位時間が 60 秒と小さい方がつまづいている学習者群の抽出には有用性が見られたため, イベントの単位時間は小さい方が良いと考えられる. また, 行動回数の多い学習者は学習意欲があるが手間取っている学習者だと考えられるため, このような学習者を支援すべきだと推測できる. さらに, 今回評価を学習者の自己評価によるアンケート結果で行ったため, 主観に基づく部分が大きく評価基準として有用であったとは必ずしも言えない. そのため, 学習者のソースコードが模範解答に近づいているかというソースコードの進捗を見て評価を行うべきであったと考えられる.

8. まとめ

本研究では, つまづいている学習者をグループ化するために, 学習行動パターンによって類似度の算出方法を提案した. 今回は目的とするグループ化はできなかったが, EVS 法のイベントの単位時間が小さい時の場合には, つまづいている学習者の抽出はできた. また, 行動回数が多く学習意欲があるが手間取っている学習者への支援にもつながると考えられる. 今後の展望として, ソースコードの進捗と合わせて見ることでより細かく分類できるのではないかと考える.

参考文献

- [1] 井垣宏, 齊藤俊, 井上亮文, 中村亮太, 楠本真二. プログラミング演習における進捗状況把握のためのコーディング過程可視化システム c3pv の提案. 情報処理学会論文誌, Vol. 54, No. 1, pp. 330-339, jan 2013.
- [2] 横山貴志, 國宗永佳, 新村正明. 情報技術演習における演習状況可視化手法の提案 (教育工学). 電子情報通信学会技術研究報告 = IEICE technical report : 信学技報, Vol.117, No. 469, pp. 119-124, mar 2018.
- [3] 特定非営利活動法人日本イーラーニングコンソシアム. Tincan プロジェクト. 2017.
- [4] 宇根田純治, 横田治夫. Web ログの共通シーケンス解析. 電子情報通信学会技術研究報告. DE, データ工学, Vol. 102, No. 64, pp. 7-12, may 2002.
- [5] 新村正明, 長谷川理, 國宗永佳. アクセスログを対象とした特徴抽出支援システムの開発 (医療・看護・福祉分野における ict 利用教育および ict を活用した教育の質保証). 教育システム情報学会研究報告, Vol. 27, No. 7, pp. 237-244, mar 2013.
- [6] 赤塚厚司, 鈴木優, 川越恭二. Web アクセスログ分類のための event vector sequence 法とその評価. 情報処理学会研究報告. データベース・システム研究会報告, Vol. 134, No. 1, pp. 1-8, jul 2004.