

科学研究費助成事業 研究成果報告書

平成 29 年 6 月 8 日現在

機関番号：12604

研究種目：基盤研究(B) (一般)

研究期間：2013～2016

課題番号：25282051

研究課題名(和文) プログラミング学習プロセスに着目した学習履歴分析手法の開発と状況把握支援への適用

研究課題名(英文) Development of a Learning History Analysis Method Focusing on Programming Learning Processes and Its Applications in Learning Situation Recognition Support

研究代表者

宮寺 庸造 (Miyadera, Youzou)

東京学芸大学・教育学部・教授

研究者番号：10190802

交付決定額(研究期間全体)：(直接経費) 13,600,000円

研究成果の概要(和文)：プログラミング演習授業において、学習者のソースコードの編集パターンと学習状況の関連をあらかじめ明らかにしておくことで、ソースコード編集過程に基づいた学習状況推定の自動化が期待できる。そこで本研究では、ソースコードの編集パターンと学習状況との関連を明らかにするために、ソースコード編集過程分析手法とそのツールの開発を行った。本研究が提案する分析手法では、系列パターンマイニングにより、過去の学習者がソースコードに対して加えた編集を類型化し、頻出する編集パターンを抽出し、さらに編集パターンと学習状況を関連付けた。本手法を用いることにより、高い精度で学習状況推定が可能であることが示された。

研究成果の概要(英文)：In the programming exercise lesson, it is expected that estimating learning situations based on the source code editing process can be automated by clarifying the relation between source code editing patterns and learning situations. Therefore, to clarify the relationship between the editing pattern of the source code and the learning situation, this study developed a method and tools for analyzing source code editing processes. Using this method, we extracted frequent editing patterns, by using Sequence Pattern Mining, from the editing history that previous learner applied to the source code. Then, we associated the edit pattern with the learning situation. By using this method, it was shown that learners' situations can be estimated with high accuracy.

研究分野：教育工学

キーワード：プログラミング学習 学習履歴 データマイニング 状況把握 系列パターンマイニング

1. 研究開始当初の背景

プログラミング演習における学習者の多くはプログラミング初学者である。そのため、多くの学習者が途中でいくつかのエラーに直面する。そして、エラーの原因を探りながらソースコードの修正を繰り返すことで、最終的なプログラムの完成を目指している。

プログラミング学習の状況把握に関する先行研究の多くは、演習時の学習者から収集した学習履歴をもとに、エラーの発生状況、課題の提出の有無、課題達成までに要した時間、などの情報を教授者に提示することで学習状況の把握を支援している。しかし、状況把握の判断材料として用いられているこれらの情報は、課題提出時などといったある時点のみで得られた学習履歴の利用に止まっている。そのため、経験の積み重ねにより進められるプログラミング学習には、従来手法では詳細な学習状況の把握には限界が生じている。

学習者のソースコードの編集過程は、演習中の試行錯誤の過程が反映されているため、よりの確かな学習状況推定が可能になると考えられる。過去の学習者の編集過程を分析し、ソースコードの編集パターンと学習状況との関連をあらかじめ明らかにしておくことで、新規の学習者の学習状況を、編集過程に基づき自動的に推定することが可能となると考えられる。しかし、ソースコードの編集パターンと学習状況との関連は未だに明らかにされておらず、これらの関連を明らかにするための分析手法やその環境は確立されていない。

2. 研究の目的

そこで本研究では、「このような学習プロセスを経た学習者はこういった状況に陥っている可能性が高い」といったように、学習プロセスと学習状況との関連を明らかにし、学習プロセスから学習状況を自動推定する手法の開発を目的とする。さらにその手法を適用した学習状況の自動推定の可能性について検討する。

3. 研究の方法

本研究では、学習プロセスをプログラムコードの編集過程としてとらえ、ソースコードの編集履歴と学習状況を関連付ける手法とそれを支援するためのツールの開発、および学習状況自動推定システムの開発を行う。以下の手順で進める(図1)。

- (1) 学習履歴収集手法およびそのシステムの開発
- (2) 系列パターンマイニング手法[1]を用いたプログラムコードの編集過程分析手法の開発
- (3) 教育データマイニング手法に基づいた分析支援ツールの開発
- (4) 学習状況把握への適用

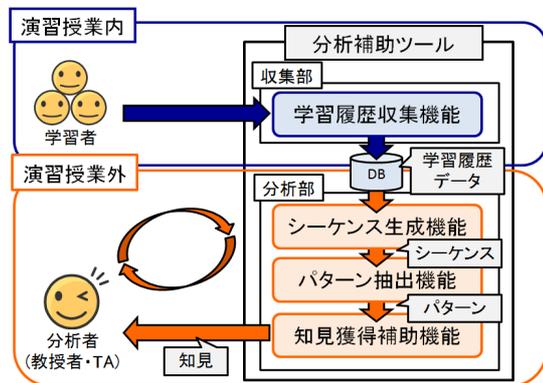


図1 プログラミング学習履歴分析

4. 研究成果

4.1 ソースコード編集過程分析手法の開発

4.1.1 分析方針

本分析の目的は、ソースコードの編集パターンと学習状況の関連を明らかにすることである。本分析では、系列パターンマイニング(Sequential Pattern Mining, 以下, SPM)により、多くの学習者が共通して行っている編集を抽出し類型化する。このような編集パターンは学習状況と関連のある可能性が高いと考えられる。また、膨大なパターンの抽出に対する配慮として分析する編集の抽象度を調整可能にする。

本研究では、2通りの分析手法を開発し、それぞれについて検証した。1つ目は、プログラムのソースコードのエラー要因に着目しソースコードが編集される毎のエラー要因の発生と解消をシーケンスとしてSPMに適用したものである(図2)。2つ目は、ソースコード編集における追加、削除、置換の操作をシーケンスとしてSMPに適用したものである。1つ目の場合は、エラー要因を抽出する手間がかかるため、2つ目の手法の方が手間の軽減が図れる。

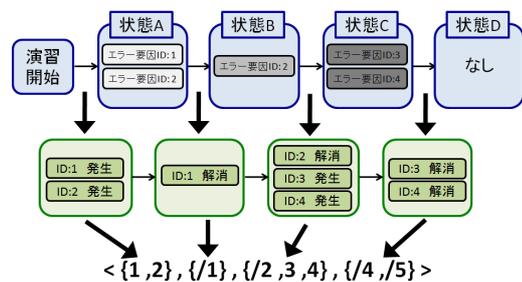


図2 プログラミング学習履歴分析

いずれの手法においても、学習状況把握への適用の可能性が示されたが、本稿では紙面の関係から2つ目の手法について報告する。

4.1.2 本分析におけるSPMの利用方法

本分析でSPMを利用するために、ソースコードの編集過程をシーケンスとして表現する。はじめに、演習中の学習者から表1に示すデータを収集する。次に、収集したソースコードを時系列順に列挙し、隣り合うソースコードの差分を抽出する。差分抽出には木

構造データ間の最小編集距離を表す Tree Edit Distance (以下 TED) を算出するアルゴリズム[2]を用いる。これにより、解析木の差分を追加・削除・置換の三つの編集操作で表現できる(図3)。解析木間の差分を時系列順に並べたものを本分析におけるシーケンスとする。また、抽出した差分の中の編集操作を取捨選択することで、編集の抽象度を調整する(図4)。

表1 収集する学習履歴データ

収集タイミング	収集データ
コンパイル時	学習者 ID, 課題 ID, エラーメッセージ, コンパイル日時, ソースコード
実行時	入力データ, 出力データ
課題提出時	理解困難な箇所とその理由についての自由記述アンケート

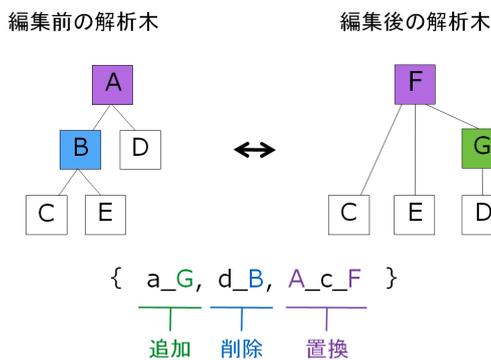
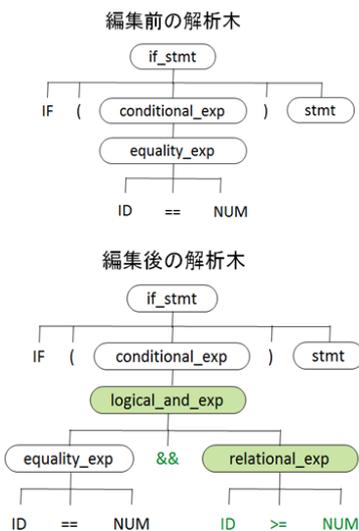


図3 解析木の差分表現の例



解析木全体の編集:

a_logical_and_expression, a_&&, a_relational_expression, a_ID, a_>=, a_NUM

トークンレベル単位の編集:

a_logical_and_expression, a_relational_expression, a_&&, a_ID, a_>=, a_NUM

文/式レベル単位の編集:

a_logical_and_expression, a_relational_expression, a_&&, a_ID, a_>=, a_NUM

図4 2つの抽象度の例

4.2 ソースコード編集過程分析支援ツールの開発

4.2.1 分析支援ツールの概要

分析の各手順を支援するツールを開発する。分析支援ツールは以下の4つの機能を持つ。

- ・学習履歴データ収集機能
- ・シーケンス生成機能
- ・パターン抽出機能
- ・知見獲得補助機能

以下これらの機能について述べる。

4.2.2 学習履歴データ収集機能

「プログラミング演習」の授業で既に利用されている学習環境に基づき、表1に示したデータを演習中の学習者から逐次収集する。

4.2.3 シーケンス生成機能

分析者が選択した課題を解いた学習者の編集過程を、分析者が指定した編集の抽象度で、シーケンスとして外部ファイルに出力する。編集の抽象度は「トークンレベル」と「文/式レベル」の2つが選択可能である(図5)。



図5 シーケンス生成画面

4.2.4 パターン抽出機能

シーケンス生成機能により出力された外部ファイルを読み込み、多くの学習者間に共通する編集パターンを抽出し分析者に提示する。分析者は提示された編集パターンの長さや出現頻度を元に特徴的な編集パターンを一つ選択する。SPMにはSPADEアルゴリズム[3]を利用する。

システムはパターン抽出機能で選択された編集パターンを持つ学習者を抽出し提示する(図6)。分析者は提示された学習者の中から一名を選択する。選択後、システムは選択された学習者のソースコード、実行結果、アンケートの回答を提示する。そして、分析者は、提示された学習履歴データから、学習者群に共通する学習状況を考察し、編集パターンと学習状況を関連付ける。

パターン	アイテム数	支持度
<[a_relational_exp],[d_for_stmt,d_relational_exp]>	3	0.12
<[a_for_stmt],[d_for_stmt,d_relational_exp]>	3	0.12
<[d_for_stmt,d_relational_exp],[d_for_stmt]>	3	0.12

図6 編集パターン抽出結果画面

4.2.5 知見獲得補助機能

システムはパターン抽出機能で選択された編集パターンを持つ学習者を抽出し提示する．分析者は提示された学習者の中から一名を選択する．選択後，システムは選択された学習者のソースコード，実行結果，アンケートの回答を提示する（図 7）．そして，分析者は，提示された学習履歴データから，学習者群に共通する学習状況を考察し，編集パターンと学習状況を関連付ける．

4.3 ソースコード編集過程分析

4.3.1 分析の概要

今回の分析の目的は，本分析手法による編集パターンと学習状況との関連付けに関する知見を得ることである．分析は，以下の 3 つの課題に対し演習を行った 112 名の情報系の学習者のデータを対象とした．

A: 10 進数と 2 進数の対応表を作成せよ．

B: 1 から 1000 の間にある完全数を印字するプログラムを作成せよ．

C: 整数値を受け取り，三角形を印字する再帰関数を作成せよ．ただし，for 文や while 文などの繰り返し構文を用いないこと．



図 7 知見獲得補助機能画面

4.3.2 分析結果

課題 A の分析では，異なる編集の抽象度を 3 つ設定した．

抽象度：追加・削除・置換操作全てトークンレベル

抽象度：追加・削除操作のみ文 / 式レベル

抽象度：追加・削除・置換操作全て文 / 式レベル

表 2 に分析結果を示す．

表 2 分析結果

	抽象度	抽象度	抽象度
シーケンス数	56	56	56
最大シーケンス長	148	36	22
最小シーケンス長	1	1	1
平均シーケンス長	33.2	7.6	4.5
support 値	0.1	0.1	0.1
パターン数	231,660	78	62
関連付け可否	不可	可	不可

抽象度 で分析を行った結果，有用な知見が得られた．抽象度 で SPM を行った結果

得られた 78 件の編集パターンから 56 名中 6 名が発生させている編集パターン「<{>_c_>= }>{>=_c_>}>」に着目した．このような編集パターンを持つ学習者の編集過程の例を図 8 に示す．

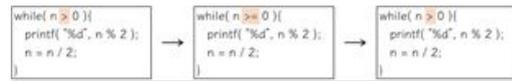


図 8 編集過程の例

このような編集過程を持つ学習者は繰り返し処理の継続条件を整理できていないと考えられる．注目した編集パターンには「繰り返し処理の継続条件が整理できていない」という学習状況を関連付けた．

4.3.3 構築された関連テーブル

分析により構築されたソースコードの編集パターンと学習状況との関連テーブルを表 3 に示す．

表 3 編集パターンと学習状況との関連テーブル

課題	関連	編集パターン	学習状況
A	1	<{>_c_>= }>{>=_c_>}>	繰り返し処理の継続条件が整理できていない
	2	<{ 3_c_4 }>	使用可能な配列の範囲を理解していない
B	3	<{<=_c_> }>{<_c_<= }>	繰り返し処理の継続条件が整理できていない
	4	<{=_c_== }>	=と==の違いを理解していない
C	5	<{ a_param_dec }, { a_param_dec }>	引数が足りないことに後から気づいた
	6	<{ int_c_void }>	作成した関数が整数値を返す必要が無いことに後から気づいた
	7	<{ a_jump_statement	関数には戻り値が必要であると勘違いをしている

「relational_exp」は関係式，
「assignment_exp」は代入式，
「param_dec」は引数宣言を表す．

4.4 学習状況把握への適用

4.4.1 アンケート調査の実施

本分析手法による編集パターンと学習状況との関連付けの適切さを評価する．前章で構築した関連テーブルを用いて，平成 28 年度の学習者に対して学習状況を推定し，課題提出後にアンケートを行う．アンケートの項目は，「推定された学習状況が当てはまっていたか」，「課題を通して躓いた箇所とその理由」の 2 項目とし，アンケート結果を元に推定結果の適合率と再現率を算出する．

4.4.2 結果

実験の結果，課題 A，B において推定された学習状況の件数は 0 件であったため，課題 C の結果を表 4 に示す．

4.4.3 考察

課題 A，B では，過去に解いた類似する課題のソースコードを再利用し課題を解いていた学習者が多く，編集履歴が少なかったため適合する編集パターンが無かったと考えられる．しかし，課題 C における適合率と再

現率がある程度高いため、本手法により適切な関連付けが可能であることが示された。

以上のことから、提案手法による学習状況把握への適用の可能性が示された。

表4 課題Cにおける実験結果

関連ID	5	6	7	計
推定された件数	7件	3件	5件	15件
推定結果が的中した件数	6件	2件	3件	11件
実際の件数	9件	2件	3件	14件
適合率	0.86	0.67	0.60	0.73
再現率	0.67	1.00	1.00	0.79

<引用文献>

- [1] R.Agrawal, R.Srikant, "Mining Sequential Patterns", Proc. the Int. Conf. on Data Engineering (ICDE'95), pp.3-14, 1995.
- [2] Pawlik, Mateusz, Nikolaus Augsten, "RTED: a robust algorithm for the tree edit distance", Proc. the VLDB Endowment 5(4), pp.334-345, 2011.
- [3] M.J.Zaki, "SPADE: An Efficient Algorithm for Mining Frequent Sequences", Proc. Machine Learning, 42, pp.31-60, 2001.

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文](計8件)

石和田圭, 森本康彦, 中村勝一, 中山祐貴, 宮寺庸造, プログラミング演習における学習状況推定のためのソースコード編集過程分析手法の開発, 電子情報通信学会技術研究報告, vol.116, no.438, ET2016-92, pp.75-80, 2017. (査読無)

Hiroki Nakayama, Kodai Yamamoto, Katunori Ogane, Hiroaki Kaminaga, Shoichi Nakamura, Youzou Miyadera, Estimation of the Error Factors in Programming Exercises Based on the Dynamic Revision of Case-base, Proc. The 13th IEEE International Conference on Advanced and Trusted Computing, IEEE CS Press, pp.475-481, 2016. (査読有)
DOI: 10.1109/UIC-ATC-ScalCom-CBD Com-IoP-SmartWorld.2016.0085

石和田圭, 森本康彦, 中村勝一, 宮寺庸造, プログラミング学習における学習状況推定のためのソースコード編集履歴分析ツールの開発, 電子情報通信学会技術研究報告, vol.116, no.126, ET2016-25, pp.21-26, 2016. (査読無)

伊藤明裕, 森本康彦, 北澤武, 宮寺庸造, eポートフォリオを活用した学習における相互評価活動の効果, 日本教育工学会報告集, JSET16-1, pp.539-542, 2016. (査読無)

Shoichi Nakamura, Kaname Nozaki, Hiroki Nakayama, Yasuhiko Morimoto, Youzou Miyadera, Sequential Pattern Mining System for Analysis of Programming Learning History, Proc. 2015 IEEE International Conference on Data Science and Data Intensive Systems, pp.69-74, 2015. (査読有)
DOI: 10.1109 /DSDIS.2015.120

Takuhei Ejiri, Yasuhiko Morimoto, Youzou Miyadera, EWMS: E-Worksheet Management System for Accumulating and Using Learning Records, Proc. Society for Information Technology & Teacher Education International Conference, AACE, pp.1610-1617, 2015. (査読有)

Shoichi Nakamura, Kaname Nozaki, Yasuhiko Morimoto, Youzou Miyadera, Sequential Pattern Mining Method for Analysis of Programming Learning History Based on the Learning Process, Proc. the International Conference on Education Technologies and Computers (ICETC2014), pp.55-60, 2014. (査読有)
DOI: 10.1109/ICETC. 2014.6998902

大橋旭雄, 野崎要, 伊藤雄貴, 森本康彦, 中村勝一, 宮寺庸造, 学習履歴を活用した支援ツールのためのプログラミング学習基盤環境, 電子情報通信学会技術研究報告, vol.113, no.316, ET2013-60, pp.15-20, 2013. (査読無)

[学会発表](計3件)

Hiroki Nakayama, Shoichi Nakamura, Kaname Nozaki, Yasuhiko Morimoto, Hiroaki Kaminaga, Youzou Miyadera, Learning History Transition Graphs for understanding the programming learning situations, Proc. 11th International Conference for Internet Technology and Secured Transactions, pp.476-477, IEEE, 2016.12.06. 「Barcelona, (Spain)」

石和田圭, 森本康彦, 中村勝一, 宮寺庸造, プログラミング学習における学習者の苦手構造把握のための決定木構築, 日本教育工学会第31回全国大会講演論文集, pp.303-304, 2015.09.21. 「電気通信大学(東京都・調布市)」

大橋旭雄, 野崎要, 森本康彦, 中村勝一,

宮寺庸造, 学習履歴を活用したプログラミング学習支援システムのための基盤環境の開発, 第38回教育システム情報学会全国大会講演論文集, pp.159-160, 2013.09.03.「金沢大学(石川県・金沢市)」

6. 研究組織

(1) 研究代表者

宮寺 庸造 (MIYADERA, Youzou)
東京学芸大学・教育学部・教授
研究者番号: 10190802

(2) 研究分担者

中村 勝一 (NAKAMURA, Shoichi)
福島大学・共生システム理工学類・准教授
研究者番号: 60364395

喜久川 功 (KIKUKAWA, Isao)
常葉大学・環境学部・准教授
研究者番号: 10440611

(3) 連携研究者

櫛山 淳雄 (HAZEYAMA, Atsuo)
東京学芸大学・教育学部・教授
研究者番号: 70313278

森本 康彦 (MORIMOTO, Yasuhiko)
東京学芸大学・情報処理センター・准教授
研究者番号: 10387532

栗田 るみ子 (KURITA, Rumiko)
城西大学・経営学部・教授
研究者番号: 60383171

夜久 竹夫 (YAKU, Takeo)
日本大学・文理学部・名誉教授
研究者番号: 90102821