

プログラミング学習時における認知的負荷の発生要因の特定 Identification of Factors Affecting Cognitive Load in Programming Learning

浅井 創[‡]
So Asai

島川 博光[†]
Hiromitsu Shimakawa

1. はじめに

情報教育を目的とした教育機関では、プログラミング経験のない学習者が多数を占める。プログラミングの概念はそのような学習者にとっては難しく、学習者によっては理解が及ばず躓くことがある。学習者の不理解の原因として高い認知的負荷がかかっていることが挙げられる。指導者は学習者の認知的負荷を適切な状態に抑え、不適切な認知的負荷がかかる学習者が理解しやすい状況を作らなければならない。しかし、高い認知的負荷がかかる学習者を多数の中から見つけ出すのは指導者にとって困難である。本研究では、プログラミング学習における学習行動に基づき、学習者の認知的負荷の発生要因を推定する手法を提案する。空欄補充問題の学習行動から決定木を生成し、学習者の認知的負荷を同定する。これにより、学習者の認知的負荷の状態が明らかになり、指導者は学習者の高い認知的負荷を適切にする措置を講じることができる。

2. プログラミング学習における認知的負荷

2.1 プログラミングの不理解の原因

プログラミングの能力の要件は、概念を理解して適切なプログラムを書けることと読めることである。これらのためには多くの知識が学習者の頭の中で体系化されている必要がある。達成できない原因は、プログラミングの概念や考え方の難しさが念頭にある [4]。そのため十分に達成できない学習者は課題や試験に取り組むとき、問題を与えられても何を書けばよいのか分からない。また、プログラミングの知識は把握している場合でも、実際に書いたプログラムの具体的な処理過程を考慮することができず、なぜそのプログラムが動いているのか分からない状況が起こる。このような学習者はプログラミング学習に躓き、プログラミングに対する忌避を起こすおそれがある。指導者は躓くおそれのある学習者を早期に発見し、躓きを防止する必要がある。

2.2 学習に影響を与える認知的負荷

プログラミングの概念の難しさによる躓きには認知的負荷が影響する [8]。認知的負荷は、学習者が新しい知識を習得したり、課題に取り組んだりするさいに、その処理能力を表すワーキングメモリをどの程度割り当てるかを指す概念である [6]。ワーキングメモリは個人差があるため、学習の許容量は個人間で異なる。認知的負荷は次の 3 種類に分類される [5]。

Intrinsic Load (IL) 学習者自身に起因する認知的負荷である。個人が持つ仕事の処理能力を示すワーキングメモリに依存する。

Extraneous Load (EL) 周囲の環境に起因する認知的負荷である。学習者が利用する教材や授業そのものの完成度に依存する。

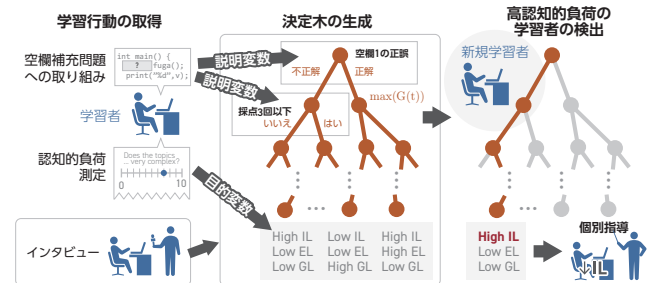


図 1: 手法全体図

Germane Load (GL) ワーキングメモリの内容を長期記憶化するときにかかる認知的負荷である。

認知的負荷は IL, EL が低く, GL が高い状態にあることが望ましい。指導者は学習者の認知的負荷をこのような状態に抑える必要があるが、高い認知的負荷がかかる学習者を見つ找出すのは困難である。

2.3 認知的負荷の検出

Morrison ら [3] はプログラミングの授業における学習者の認知的負荷を測定した。認知的負荷を測るために、IL, EL, GL それぞれを問う質問に対して、学習者は 0 から 10 の 11 段階で評価する。評価の対象が学習全体であり、認知的負荷の精緻な原因が明らかになっていない。

Yousoof ら [7] はプログラミングの授業における視覚情報と文字情報を分割して、その蓄積から認知的負荷の測定と軽減をする手法を提案している。この手法は EL に重点を置いており、学習者自身に起因する IL や GL は十分に考慮されていない。3 種類の認知的負荷を考慮して、学習者の行動に基づき本質的な学習者の躓きを明らかにする必要がある。

3. 学習行動による認知的負荷の要因推定

3.1 認知的負荷の要因推定手法

本研究では学習者が抽象的なプログラミング言語を学習するときの認知的負荷の発生要因を推定する手法を提案する。手法の全体図を図 1 に示す。本手法は、過去の学習者が空欄補充問題を回答するときの学習行動に基づき認知的負荷を推定する。空欄補充問題による学習行動を裏付けるために、学習者に対して回顧法によるインタビューを実施する。さらに、学習者に対し既存手法を用いて学習者の認知的負荷を測定する。空欄補充問題における学習行動とインタビューの実施結果、学習者の認知的負荷をもとに決定木を生成する。これにより、学習者の認知的負荷の発生要因が明らかになる。さらに決定木に対して新規学習者の学習行動を当てはめることでその学習者の認知的負荷が明らかになる。指導者はただちにその学習者に対し別途指導し、学習を妨げる認知的負荷の最適化を図ることができる。

[‡]立命館大学大学院情報理工学研究所

[†]立命館大学情報理工学部

3.2 学習行動の取得

認知的負荷の要因推定のために、学習者の学習行動を取得する。本手法では空欄補充問題の回答に取り組むさいの学習行動に着目する。空欄補充問題を用いる学習はプログラミングの授業でよく実施されている。また空欄補充問題は回答する部分が限定されているので、不理解を特定しやすい。さらに空欄補充問題は、選択肢による問題よりも回答を憶測される可能性が低く、学習者の理解度を問うのに効果的である [2]。本手法における空欄補充問題による学習では、学習行動として空欄補充問題の回答時間、回答変更履歴、採点回数を取得する。そのために学習者は、空欄補充問題の自動採点システム [1] を利用する。このシステムを用いることで回答の修正がすべて記録される。制限時間内であれば正答するまで何度も回答を修正することができる。

一方で学習者の認知的負荷を測定するために、認知的負荷の測定手法 [3] を用いる。学習者は認知的負荷を測る 10 項目の質問に回答する。IL, EL, GL を測る質問項目がそれぞれ 3 項目、3 項目、4 項目で構成されている。学習者はおのおのの質問項目に対しについて 0 から 10 の 11 段階で評価する。数値が大きいほど高い認知的負荷として判断される。また、空欄補充問題の回答および認知的負荷の測定について、回顧法によるインタビューを実施する。学習行動と認知的負荷を同定するために、学習行動の裏付けが必要である。インタビューにより学習者は空欄補充問題の回答時の行動および認知的負荷測定の項目の選択の原因を追求される。インタビューの結果は認知的負荷の高い学習者と低い学習者を分類するための指標とする。

3.3 認知的負荷を同定する決定木の生成

空欄補充問題の回答時の学習行動と測定された認知的負荷を同定するために、学習行動を説明変数、認知的負荷を目的変数にとる決定木を生成する。空欄補充問題の学習行動に基づき、説明変数として与える学習者を二分する特徴を決定する。学習者を正確に二分できる特徴を選定する必要があるため、インタビューの結果の併用により空欄補充問題における行動の原因を考慮する。決定木アルゴリズムを用いてこれらの特徴をノードとして構成する木を生成する。決定木は不純度が低くかつ情報利得が最大化するように構成する。そのためにノードのデータ数によって得られる、データの不純性を表すエントロピーを用いる。ノード t におけるエントロピー $I_H(t)$ は、 $P(i|t)$ をノード t から分岐した子ノード i のデータ数の割合とすると、次式で求められる。

$$I_H(t) = - \sum_{i=1}^2 P(i|t) \log_2 P(i|t)$$

また、情報利得の最大化のために分岐前後のエントロピーの差分を用いる。ノード t における情報利得 $G(t)$ は次式で求められる。

$$G(t) = I_H(t) - \sum_{i=1}^2 \frac{|t_i|}{|t|} I_H(t_i)$$

決定木の過剰適合を防止するために、情報利得が閾値未満になるノードを葉ノードとする。決定木の各葉ノードには、学習行動が根からの分岐にすべて合致する学習者の認知的負荷が目的変数として与えられる。ここで高い IL および EL となる分岐が高い認知的負荷の要因となる学習行動となる。

3.4 学習者の認知的負荷の検出

新規の学習者は同じく、自動採点システムを用いた空欄補充問題に取り組む。空欄補充問題の回答で得られる学習行動と決定木を照合し、決定木の葉ノードまで辿る。ここで合致する葉ノードにおける認知的負荷がその学習者の認知的負荷となる。検出された学習者の認知的負荷は、指導者に通知される。IL または EL が高い場合、指導者は当該学習者へ認知的負荷が適切な状態となるように再指導する。これにより学習者が不理解を抱く可能性が低下し、躓きの発生を防止できる。

4. おわりに

本論文では決定木により学習者の認知的負荷の要因の推定をする手法を提案した。学習者の学習行動から認知的負荷が明らかになるので、指導者は認知的負荷が高い学習者を早期に発見することができる。今後は、本手法の有用性を実験により検証する。

参考文献

- [1] S. Asai and H. Shimakawa, "Automatic scoring system of fill-in-the-blank tests to measure programming skills," in *Proc. of the 6th the International Conference on Information Technology and Its Applications*, 2017, pp. 23 – 29.
- [2] R. S. H. B. Medawela, D. R. D. L. Ratnayake, W. A. M. U. L. Abeyasinghe, R. D. Jayasinghe, and K. N. Marambe, "Effectiveness of "fill in the blanks" over multiple choice questions in assessing final year dental undergraduates," *Educación Médica*, 2017.
- [3] B. Morrison, B. Dorn, and M. Guzdial, "Measuring cognitive load in introductory cs: adaptation of an instrument," in *ICER '14 Proceedings of the tenth annual conference on International computing education research*, 2014, pp. 131 – 138.
- [4] S. Shuhidan, M. Hamilton, and D. D'Souza, "Understanding novice programmer difficulties via guided learning," in *ITiCSE '11 Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*, 2011, pp. 213 – 217.
- [5] J. Sweller, "Element interactivity and intrinsic, extraneous, and germane cognitive load," *Educational Psychology Review*, vol. 22, no. 2, pp. 123 – 138, 2010.
- [6] J. Sweller, P. Ayres, and S. Kalyuga, *Cognitive Load Theory*. Springer, 2010.
- [7] M. Yousoof, M. Sapiyan, and K. Kamaluddin, "Measuring cognitive load - a solution to ease learning of programming," in *World Academy of Science, Engineering and Technology International Journal of Computer and Systems Engineering*, vol. 1, no. 2, 2007, pp. 32 – 35.
- [8] 岡本雅子 and 喜多一, "プログラミングの「写経型学習」における初学者のつまずきの類型化とその考察," *パイディア : 滋賀大学教育学部附属教育実践総合センター紀要*, vol. 22, pp. 49 – 53, 2014.