

# 集中度と操作情報を用いたプログラミング学習の躓き検出に基づく支援システム

## A Support System for Detecting Learning Difficulties in Programming Using Concentration and Interaction Data

大政 悠暉<sup>\*1</sup>, 加島 智子<sup>\*1</sup>

Yuki OMASA<sup>\*1</sup>, Tomoko KASHIMA<sup>\*1</sup>

<sup>\*1</sup> 近畿大学大学院システム工学研究科

<sup>\*1</sup> Graduate School of Systems Engineering, Kindai University

Email: 2433850038d@hiro.kindai.ac.jp

**あらまし**：近年、プログラミング教育の拡大を背景に授業外でのプログラミング学習の機会が増え、教員からの直接的な支援によらない支援手法が必要とされている。本研究では、集中度と操作情報を統合して学習者の躓きを検知し、支援を提供する手法とその手法を利用したシステムの実装を行った。その結果、一定タイミングで支援を行うシステムと比較して支援の適時性や学習者のモチベーション、スキルの向上値の有意な上昇が確認された。

**キーワード**：躓き検出, 学習支援, 停滞度, 集中度

### 1. はじめに

近年、情報化の進展を背景にプログラミング教育が拡大し、学習者は授業外でプログラミングの自主学習や課題演習を進める必要がある。しかし、授業外での学習は教員からの支援は得られず、学習の停滞が生じやすい。そのため、教員からの直接的な支援に依存しない支援の仕組みが求められている。

学習者の状態を自動的に把握する手法として、操作情報を用いるもの<sup>(1)</sup>や生体情報を用いるもの<sup>(2)</sup>があるが、躓きのパターンには学習者の理解や内的要因に起因するものなど複数あり<sup>(3)</sup>、単一の手法では支援提示の要否やタイミングを適切に決めることは難しい。そこで、本研究ではプログラミング初学者が文法の理解をより深めるためのコーディング演習において、学習者の停滞を検知し、熟考を阻害しない形で支援を行うことを目的として、集中度と操作情報から支援が必要な躓きを抽出し、過度な干渉を与えない支援へ接続するシステムを構築する。

### 2. 躓き検出手法

提案手法では、Webカメラから取得した学習者の集中度とプログラミング学習中の操作情報から学習者の躓き状態を検出する。提案手法の概要は図1の通りである。

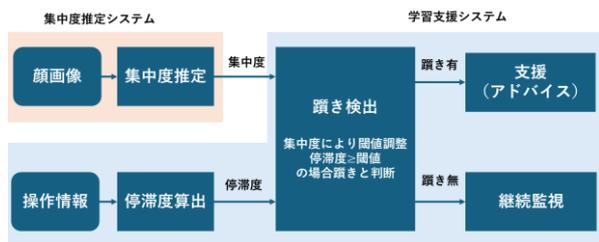


図1 躓き検出手法の概要

まず、学習者が条件を満たすプログラムをコーディングしている場面において、Webカメラを用いてリアルタイムに学習中の顔画像を取得し、機械学習モデルを用いて集中度を算出する。

次に、コーディング中の操作情報として、過去一定区間の無操作の時間の割合、前回の進捗からのプログラム実行や提出回数の割合、前回の進捗から停滞している時間の割合を算出し、0~1で正規化する。このとき、進捗はプログラムの評価が上昇、エラーの解消、提出コードと模範解答コードの類似度上昇を指す。その後、これら3つの指標のうち最も値の大きいものを停滞度として算出する。

そして、取得した集中度を用いて躓きの閾値調整を行う。具体的には、初期閾値に補正值を足すことで閾値を決定し、補正值は-1~1に正規化した集中度、欠損率、補正量を掛け合わせることで算出する。

最後に、バッチ処理及びプログラム実行や提出時に躓きの判定を行い、算出された停滞度が補正後の閾値以上の場合にその学生は現在躓いていると判断する。

### 3. 提案システム

提案する躓き検出手法を利用したプログラミング学習支援システムを構築する。

提案システムでは、集中度推定システムによって集中度をリアルタイムに取得し、学習支援システムのWebブラウザから操作情報を取得、バックエンドで躓き検出やシステム管理を行う。また、学習者の提出したプログラムに対してインターネット上のOpenAI APIを用いてLLMによる5段階の評価を実施して評価を行う。さらに、学習の際に得られた学習履歴はログとして出力される。

学習中に躓きが検出された場合は、3段階のアドバイスを提示する。アドバイス提示時の様子は図2の通りである。



図2 アドバイス到着時の様子

アドバイスはトーストによって通知され、画面下部に具体的な内容が表示される。また、徐々に具体的な内容のアドバイスが提示される。

#### 4. 実験

提案システムの効果を確認するため、近畿大学工学部の学生16名（有効データ13名）に対して実験を行う。実験ではシステムを利用してC言語の課題に取り組んでもらい、固定のタイミングでアドバイスを提示する比較手法と、躓き検出を行いつつアドバイス提示を行う提案手法でアドバイスの提示タイミングの適切性と学習者への影響を確認する。問題数は6問で各問題の制限時間は15分、演習全体の制限時間を40分とし、両手法ともアドバイス回数は最大3回とする。

#### 5. 結果

実験より、アドバイス提示タイミングの妥当性、学習行動への影響、学習成果に関して評価を行う。

##### 5.1 アドバイス提示タイミングの妥当性

学習者が問題毎に自己申告した躓き区間と一致している割合とアドバイスが躓き区間より早期提示されている割合を示した結果が表1の通りである。

表1 自己申告の躓き区間に対する提示位置

	一致割合	早期提示割合
比較手法(n=33)	0.273	0.684
提案手法(n=26)	0.615	0.182
p 値	0.016	0.021

表1より、アドバイスが自己申告の躓き区間内に位置する割合は提案手法が有意に高く、早期提示も有意に抑えられている。このことから、提案手法は不要なアドバイスによる負担を減らし、学習者が試行錯誤する機会を提供できていると考えられる。

##### 5.2 学習行動への影響

次に、問題進行に伴うアドバイス回数や問題ごとの回答時間の変化は表2の通りである。

表2 問題進行に伴う変化

群	アドバイス回数 の傾き	回答時間 (秒) の傾き
比較手法	0.143	20.68
提案手法	-0.21	-89.14

比較手法は問題が進むにつれてアドバイス回数や回答時間が増える一方で、提案手法はどちらも減少する傾向である。

#### 5.3 学習成果

次に、演習事前事後に実施したプログラミング学習へのモチベーションに関するアンケートとスキルテストの結果は表3の通りである。アンケートは計6問で解答を0-1に正規化し、スキルテストではC言語の基礎的な4択問題を7問出題する。

表3 事前事後アンケートとテストの比較

群	n	指標	増加量	p 値
比較手法	4	スキル得点	0.00	1.000
		モチベーション	0.38	0.109
提案手法	8	スキル得点	1.00	0.047
		モチベーション	0.50	0.033

比較手法ではスキルおよびモチベーション向上に有意差はない一方で、提案手法ではスキルとモチベーションが有意に上昇することが分かる。このことから、提案手法は学習者のプログラミングへの興味関心や能力によりよい影響を与えられられる。

#### 6. おわりに

本研究では、プログラミング学習において学習者の停滞を検知し学習者の熟考を阻害しないように支援を行うために、集中度と操作情報を統合した躓き検出手法を提案し、提案手法を活用したシステムを構築した。システムを利用した実験では、固定のタイミングで支援を行う手法と比較してより適切なタイミングでアドバイスを提供できた。また、学習者のモチベーションやスキルにおいても有意な上昇が見られた。今後は、躓き区間の検知率の改善や、より人数を増やした実験を行い、システムの効果をより厳密に検証する必要がある。

#### 参考文献

- (1) T. W. Price, D. H. Hovemeyer, K. Rivers, G. Gao, A. C. Bart, A. M. Kazerouni, B. A. Becker, A. K. Petersen, L. Gusukuma, S. H. Edwards, D. S. Babcock; "ProgSnap2: A Flexible Format for Programming Process Data," ITiCSE '20 (2020)
- (2) H. Oka, A. Ohnishi, T. Terada, M. Tsukamoto; "System for Detecting Learner Stuck in Programming Learning," Sensors, 23巻, 13号 (2023)
- (3) Y. Qian, J. Lehman; "Students' Misconceptions and Other Difficulties in Introductory Programming: A Literature Review," ACM Transactions on Computing Education (TOCE), Volume 18, Issue 1 (2017)