

# 空欄補充問題の自動ラベリングによる非修得項目の明確化 Clarifying Unacquired Learning Items with Automatic Labeling in Fill-in-the-blank Test

五島僚佑<sup>†</sup>

Ryosuke Goshima

島川博光<sup>†</sup>

Hiromitsu Shimakawa

## 1. はじめに

指導者が学習者を指導するさい、学習者のプログラミング理解度に合わせた適切な指導を与えることが求められる [2]. 学習者の理解度を把握するために、指導者は空欄補充問題を採用することが望ましい. 空欄補充問題は採点個所を限定することができるため、自動採点が可能である [1]. 空欄補充問題の採点だけでなく、その結果の分析を自動化できれば、即座に学習者の多様な理解度を明らかにできる. 本研究では、空欄補充問題の採点結果の体系的分析を自動化する手法を提案する. 本手法を用いることで、各問題の理解度ごとに学習者を分類することができるため、特定の問題に対する理解度が低い学習者を抽出できる.

## 2. プログラミング学習における理解度の推定

### 2.1 理解度を推定する空欄補充問題の生成

空欄補充問題の作問は、空欄に設定する位置が重要である. 柏原ら [3] は、プログラムにおける処理の意味内容には踏み込まず処理の要所を同定し、形式的に空欄を設定する手法を提案している. 学習者の理解度を明らかにしていないため、生成された空欄補充問題に対応する理解度推定の方法を考える必要がある.

### 2.2 空欄補充問題を用いた理解度推定の方法

菅沼ら [4] は、学習者が空欄補充問題を回答する毎に学習者の理解度を定量的に評価する手法を提案している. 学習者に問題をいくつか与えて、問題を解くごとに理解度を積算する. 約 50 問の問題を解くことで、学習者の理解度は、本来の理解度の値に収束する. 学習者は問題を少なくとも 50 問解かなければならないため、早期に理解度を判別することは容易ではない. 指導者は多くの問題を用意するのに多大な労力を必要とする. また学習者の理解度は定量的に評価されるが、この手法は、不理解点を精緻に同定していない. これらの問題解決を踏まえた理解度推定方法が必要である.

## 3. 空欄のラベル付けによる非修得項目の抽出

### 3.1 プログラミング理解度の推定手法

本研究では、プログラム空欄補充問題の空欄にラベルを付け、学習者の非修得項目を明確化する手法を提案する. 本手法は、学習者の非修得項目を明確にするために空欄補充問題に回答するさいの学習履歴を分析する. 学習履歴は、学習者の空欄補充問題への回答、その正誤の組の時系列である. この時系列内の要素数から採点回数が判明する. 空欄ごとに記録された学習履歴に基づき、各空欄にラベルを付ける. 空欄のラベル付けは、学習者の課題への得点の変化を視覚的に表す理解度増進グラフの作成に利用される. 理解度増進

グラフの形状をクラスタリングにより分類し、課題に対する理解度別に学習者を分ける. これにより、課題ごとに理解度の低い学習者が分類されるクラスタができるため、学習者の非修得項目を明確にできる.

### 3.2 B2A 図に基づく空欄のラベル付け

本手法は、空欄の価値を正しく評価するために空欄補充問題の空欄を基礎ないしは応用の 2 種類のラベルを付ける. 空欄に適切なラベルを付けるために B2A 図を用いる. B2A 図は学習者の基礎力を横軸、応用力を縦軸として学習者のプログラミングの能力を平面上にプロットした図である. 応用、基礎とラベル付けられた空欄の集合を、それぞれ  $A, B$  としたとき、学習者  $i$  の基礎力  $B_i$  と応用力  $A_i$  は次式で評価される:

$$X_n = \begin{cases} \log \frac{1}{R_n} & (\text{正答時}) \\ 0 & (\text{誤答時}) \end{cases}$$

$$A_i = \sum_{k \in A} X_k \quad B_i = \sum_{k \in B} X_k$$

$X_n$  は学習者が回答した空欄  $n$  の点数である. すべての空欄を同じ点数としないために、空欄  $n$  の正答率  $R_n$  を用いて点数に重みをつける.  $A_i$  は応用とラベルを付けた空欄の合計点数である. 同様に、 $B_i$  は基礎とラベルを付けた空欄の合計点数である. 応用力を得るためには基礎力が必須であるため、応用力が基礎力を上回る学習者は存在しない. したがって、 $A_i \leq B_i$  を満たすように各空欄にラベルを付けることが適切である. 与えられたラベル付けの適切さ  $L$  は次式で評価される:

$$L = \sum_{t=1}^3 \sum_{i \in S} \frac{1}{2^{t-1}} \left( \frac{A_{it}}{A_{MAX}} - \frac{B_{it}}{B_{MAX}} \right)$$

ラベル付けの評価には学習者の 3 回目までの採点の学習履歴が使用される. 初回の採点結果はそのまま使われ、2 回目は  $1/2$ 、3 回目は  $1/4$  の重みがかかけられる.  $S$  は応用力が基礎力を上回る学習者の集合を表している. 与えられたラベル付けのパターンの下での  $A_{it}$  は  $t$  回目の回答における学習者  $i$  の応用力、 $B_{it}$  は  $t$  回目の回答における学習者  $i$  の基礎力である. また、 $A_{MAX}$  は課題における応用力の最大値で、応用とラベルが付けられた空欄の点数の合計値である. 同じく、 $B_{MAX}$  は課題における基礎力の最大値で、基礎とラベルが付けられた空欄の点数の合計値である.  $L$  は、B2A 図において応用力が基礎力を上回る学習者が存在すると大きくなる. ゆえに、最も適切な B2A 図は、 $L$  が最小化される場合である. 課題中の空欄に対応するすべてのラベル付けのパターンから  $L$  を算出し、最小の  $L$  を得る. このように、学習者の学習履歴から空欄に対する最適なラベルを見つけることで、問題作成者の主観なしに空欄を基礎と応用に分類することができる.

<sup>†</sup>立命館大学情報理工学部

<sup>‡</sup>立命館大学大学院情報理工学研究科

### 3.3 理解度増進グラフの生成

学習者の課題への理解度を評価するために、最適な空欄のラベル付けを得た後に、理解度増進グラフを生成する。理解度増進グラフとは、空欄補充問題の採点ごとの学習者の得点の変化を評価したものであり、採点回数を横軸、得点を縦軸にとるグラフである。学習者の得点は次式で定量的に評価される：

$$Y_{it} = A_{it} + wB_{it}$$

学習者の得点は、3.2 で述べた基礎力、応用力により評価される。  $w$  は応用力に対する基礎力の重みであり、  $0 < w < 1$  を満たす。基礎力の得点に重みをつけることで、基礎問題の点数がグラフに小さく影響する。これは、学習者が基礎問題と応用問題のどちらに苦戦しているのかをグラフに反映させるためである。グラフの形が類似している学習者同士は、課題に対する理解度が等しいと仮定できる。

### 3.4 学習者のクラスタリング

学習者を理解度別に分類するために、理解度増進グラフの形に基づきクラスタリングを適用する。これにより、クラスタごとに理解度が異なる学習者が分類されるため、課題への理解度が低い学習者を抽出することができる。本研究では、グラフの位置と変化の二点について学習者の理解度増進グラフにクラスタリングを適用する。本手法では、どちらの場合においてもクラスタ数が3になるようにクラスタに区切る。課題ごとにクラスタリングの結果が与えられるため、個々の学習者に対し、その学習者の理解度が低いと判定される学習項目を、指導者はフィードバックすることができる。しかも、そのフィードバックは、これまでに説明した自動化のために十分に速い。これにより、学習者は自らの非修得項目を中心に学習することができる。

## 4. 結果と考察

本手法の有効性を検証するために、学習履歴のデータを用いて分析を実施した。分析に用いるデータは、立命館大学情報理工学部の学生 80 名のプログラミングの学習履歴である。本分析において、位置と変化のそれぞれに着目して作成したクラスタの組合せにより、学習者は 9 個のクラスタに分かれた。各クラスタの理解度増進グラフの形とその特徴から図 1 のように学習者の理解度状況は六つの型に統合される。

クラスタリングの正当性を明らかにするために、無作為に抽出した学習者 A~J が基礎停滞型に分類された場合をより深く調査した。これら学習者の、他の問題に対する学習履歴から実際にその問題に対し苦戦しているのかを確認した結果を表 1 に示す。本手法で基礎停滞型と判断された学習者が、実際に理解度が低い割合を示す適合率  $P$  を、基礎停滞型の問題の集合を  $F$ 、不適当な回答を繰り返し採点していた問題の集合を  $W$  とし、集合  $X$  の数を表す  $n(X)$  を用いて以下の式で評価した：

$$P = \frac{n(F \cap W)}{n(F)}$$

本手法を用いて検出した学習者の非修得項目は高い適合性をもつ。

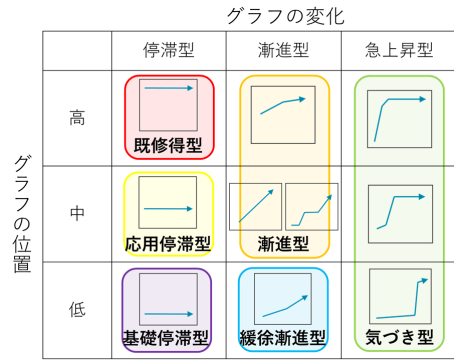


図 1: 各クラスタの理解度増進グラフの概形

表 1: 基礎停滞型に属する問題の適合率

student	$n(F)$	$n(F \cap W)$	$P$
A	7	3	0.429
B	3	2	0.667
C	27	24	0.889
D	4	4	1.000
E	13	10	0.769
F	14	14	1.000
G	5	5	1.000
H	20	19	0.950
I	9	8	0.889
J	13	9	0.692

## 5. おわりに

本研究では、空欄補充問題の空欄にラベルを付けることで学習者の理解度状況を示すグラフを生成し、それらの形状の違いから理解度の低い学習者を抽出する手法を提案した。本手法を用いることで、個々の学習者に対し、その学習者の理解度が低いと判断される項目を、指導者はフィードバックすることができる。今後は本手法の使用性を評価する。

## 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] T. Crow, A. Luxton-Reilly, and B. Wuensche, “Intelligent tutoring systems for programming education: a systematic review,” in *Proceedings of the 20th Australasian Computing Education Conference*. ACM, 2018, pp. 53–62.
- [3] A. Kashihara, A. Terai, and J. Toyoda, “Making fill-in-blank program problems for learning algorithm,” *Adv. Res. Computers and Communications in Education*, 1999.
- [4] A. Suganuma, T. Mine, and T. Shoudai, “Dynamic evaluation of both students’ and questions’ levels for generating appropriate exercises to students automatically,” in *Knowledge-Based Software Engineering: Proceedings of the Fifth Joint Conference on Knowledge-Based Software Engineering*. IOS Press, 2002, p. 325.