

プログラミング学習状況推定のための教師データ作成支援システムの開発

川口 翔大⁺¹ 西川 直志⁺¹ 佐藤 克己⁺¹ 大沼 亮⁺² 中山 祐貴⁺³ 中村 勝一⁺² 宮寺 庸造⁺¹

⁺¹ 東京学芸大学 ⁺² 福島大学 ⁺³ 早稲田大学

⁺¹ { m183305w@st., a151417k@st., satoyo@, miyadera@ } u-gakugei.ac.jp

⁺² { onuma, nakamura }@sss.fukushima-u.ac.jp, ⁺³ nakayama@aoni.waseda.jp

我々が以前に開発したプログラミング学習における学習状況推定手法に必要な教師データ作成支援ツールの開発を行なった。本システムでは、学習者らの振り返り作業をもとに教師データを作成する。本稿では、本システムの開発について報告する。

1. はじめに

プログラミング学習では、学習者は様々な躓きに直面する。また、学習者によっては、その躓きを解消することができないことも少なくない。そこで我々は、躓きの原因などといった学習者の学習状況を自動的に推定する手法を開発した⁽¹⁾。しかし、このとき機械学習手法の一つである教師あり学習を用いたため、依然として教師データ作成時における負担が残っている。

そこで本研究では、教師データを作成する際の負担軽減を行うための教師データ作成支援システムの開発を目的とする。

2. 研究概要

2.1 開発した学習状況推定手法

我々は、すでにソースコード編集履歴から該当する学習状況を自動的に推定する手法を開発している⁽¹⁾。

本手法では、機械学習手法の一つである教師あり学習を用いているため、教師データが必要となる。このとき、教師データは「ソースコードの編集履歴」と「学習状況」とを関連づけたデータとした。ソースコードの編集履歴とは、「演習活動中のソースコードデータのシーケンス」とし、学習状況とは、「有限個に教授者が定めたもの」としている。しかし本手法では、大量の教師データを用意しなければならず依然として負担は残ってしまっている。

2.2 研究概要

従って、本研究では教師データ作成支援を行うシステムの開発を目的とする。アプローチとして、教師データ作成の作業を学習者にも課すことを考える。しかし、この教師データ作成の作業が学習の妨げとなっては本末転倒である。故に、本来行なっていた課題の振り返り作業中において、半自動的に教師データを作成させることを考える。このときの具体的な振り返り作業としては、「課題の

提出時に、システムに推定された自身の行き詰まり区間に関して、選択・自由記述形式で考察する」ものとする。この振り返り作業から得られたデータを基に、教師データの作成を行う。

3. 教師データ作成支援システムの開発

開発したシステムでは、次の6つの機能を実装している。これらの機能を通して、開発した学習状況推定手法に適用可能な教師データを作成する。

1. 課題登録機能
2. ソースコード作成・提出機能
3. ソースコード収集機能
4. 行き詰まり区間推定機能
5. 振り返り機能
6. 教師データ作成・整形機能

3.1 課題登録機能

本機能では、教授者が実際に学習者に出題する課題の登録を行うことができる。

課題を登録する際には、表1に記述した6つの項目を入力する。このうち入出力テストとは、当該課題における正答となる入出力データのペアのことである。また、学習項目テストとは、当該課題における新たに学ばせたい学習項目とその指定使用回数のペアのことである。これらのテストは、学習者が課題を提出しようとしたタイミングで、自動的に行われるようにしている。また、これらのテストに通過しない場合には、課題の提出は行えないものとしている。

表1 課題登録時のデータ

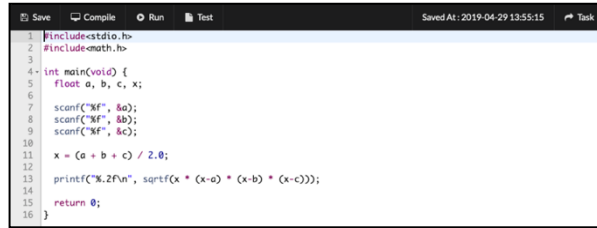
課題登録時のデータ
・タイトル
・課題の内容
・入出力の例
・入出力テスト
・入出力テスト時における入力制限
・学習項目テスト

3.2 ソースコード作成・提出機能

本機能では、学習者が登録された課題に対して、

実際にソースコードを作成し、その作成したソースコードを提出することができる。

本機能における課題演習中のソースコードエディタ画面の一例を図1に示す。



```

1 #include <stdio.h>
2 #include <math.h>
3
4 int main(void) {
5     float a, b, c, x;
6
7     scanf("%f", &a);
8     scanf("%f", &b);
9     scanf("%f", &c);
10
11     x = (a + b + c) / 3.0;
12
13     printf("%f\n", sqrt(x * (x-a) * (x-b) * (x-c)));
14
15     return 0;
16 }

```

図1 ソースコードエディタ画面

3.3 ソースコード収集機能

本機能では、システムが、学習者がソースコードを作成している際に、ソースコードを自動的に収集する。

収集するタイミングとしては、「ソースコードエディタに focus がされている間 30 秒ごと」としている。

3.4 行き詰まり区間推定機能

本機能では、システムが、学習者が課題を提出した際に、自動的にその課題に対する学習者の行き詰まり区間を推定する。

この行き詰まり区間推定アルゴリズムには、藤原らの手法⁽²⁾を基に、本システム用に若干の変更を加えたものを取り入れた。

3.5 振り返り機能

本機能では、学習者が課題の提出後、自身の演習活動に対する振り返り作業を行うことができる。

これまでの機能により、学習者が課題を提出したタイミングで、演習活動中のソースコードと行き詰まり区間のデータが収集できている。そこで、本機能における振り返り内容としては「各行き詰まり区間に対して、なぜ行き詰まってしまったのか?」を考察させることとしている。より具体的には、各行き詰まり区間に対してあらかじめ教授者が課題ごとに登録している学習状況から選択する形式、またそれらのどれにも当てはまらない場合には自由記述での形式をとっている。

3.6 教師データ作成・整形機能

本機能では、教授者が前節までで得られたデータを教師データとする。

前節までで得られたデータをまとめると以下のようになる。

- ・学習者の行き詰まり区間と教授者が課題ごとに設定した学習状況とのペア
- ・学習者の行き詰まり区間と自由記述形式の学習状況とのペア

ここで再度、収集する教師データについて整理する。収集したい教師データは、「演習活動中のソ

ースコードデータのシーケンス」と「教授者が有限個に定めた学習状況」であった。従って、前節までで得られたデータにおいて、前者はすでに教師データとして扱えるものとなっているが、後者はまだそうはなっていない。本機能では、これらのデータを教師データに整形する。

教授者は学習者の振り返りのデータを基にして、自由記述形式の学習状況がすでに登録している既存の学習状況に当てはまるのか、それともまったく別の新しい学習状況として登録するのかを考察する。前者であれば、既存の学習状況との関連づけを行い、後者であれば新しい学習状況を登録した上で、その学習状況と関連づけを行い、既存の学習状況推定手法に適用可能な教師データに整形する。

4. 考察

従来の手法では、全て手動で教授者が教師データを作成する必要があったため、実際に開発した学習状況推定手法を実用することは非現実的であった。しかし本システムを開発することにより、それらの作業を学習者にも課すことができるようになり、学習状況推定が現実的なものとなったといえる。教授者は、3.6 節で述べたような教師データの整形作業のみをするだけでよくなったため、教授者の負担は大きく軽減された。また、本システムは、本学の実際のプログラミング演習授業ですでに実際に運用をしており、教師データを逐次収集できている。

5. おわりに

本研究では、我々が開発した学習状況自動推定手法における教師データ作成時の負担軽減を行うシステムを開発した。また本システムにより、既存の学習状況自動推定手法に適用可能な教師データを作成することができた。

今後は、得られた教師データをもとに実際に学習状況の推定を行い、その結果を学習者に提示するシステムの開発を行う。

参考文献

- (1) Shota Kawaguchi et al., Machine Learning Model for Analyzing Learning Situations in Programming Learning, IEEE Conference on Big Data and Analytics, pp.74-79, 2018.
- (2) 藤原 賢二 他, スナップショットを用いたプログラミング演習における行き詰まり箇所の推定, コンピュータソフトウェア, 第 35 巻, 1 号, pp.3-13, 2018.