

Scratch ユーザのコンピューテーショナル・シンキング 習熟順序把握に向けた作品ジャンルの特徴分析

木内 正浩^{1,a)} 榎原 絵里奈^{2,b)} 小野 景子^{3,c)} 花本 凪^{1,d)}

概要：様々な先行研究から Scratch では制作する作品ジャンルにより、ユーザが習熟可能なコンピューテーショナル・シンキングが異なることが明らかにされている。そのため、ユーザの学習過程を分析するためには、そのユーザが今までどのジャンルに属するプログラムを作成したか分析する必要がある。しかしながら、現状の Scratch コミュニティサイトでは直近に投稿された作品しかジャンルを確認することができず、ユーザがこれまでに投稿した作品全てのジャンルを把握することは困難である。そこで本稿では、Scratch コミュニティサイトから収集可能なデータを用いて、作品ジャンルを分類するモデルを構築し、さらに作品ジャンルの分類に寄与する要因を分析した。34,400 件の投稿作品を対象に評価実験を行った結果、分類モデルは、適合率、再現率ともに約 0.962 の精度で作品ジャンルを分類することができた。また、作品内で利用されるブロックの種類や数よりも、作品のタイトルやメモ・クレジットに含まれるテキスト情報の方が分類精度に強く寄与することがわかった。

キーワード：Scratch, プログラミング教育, 作品ジャンル, コンピューテーショナル・シンキング

Characteristics of each category in Scratch community aiming to analyze the proficiency of a user's computational thinking

Abstract: Previous studies have shown that the proficiency of computational thinking skills in Scratch depend on which category a user has been created her/his program. However, current Scratch community site, only recent submitted program can be tagged its category, then analyzing the category of all programs a user has been submitted are difficult. In this paper, we construct a model for categorizing a Scratch program using data that collected from the Scratch community site, and analyzed important factors to categorize a program. We conducted evaluation by using 34,400 programs posted on the Scratch community site and categorization model archives accuracy in approximately 0.962. Furthermore, this study finds the textual information contained in the title and notes/credits of a program contributed more strongly to the classification accuracy than the type and number of blocks used in a program.

Keywords: Scratch, programming education, program category, computational thinking

1. はじめに

初学者を対象としたプログラミング教材の1つに、ビジュアルプログラミング学習環境 Scratch^{*1}がある。Scratch はテキストプログラミングと同様に、学習者がプログラムの実装を通してコンピューテーショナル・シンキング（以下、CT）を習熟可能であるため、様々な場面で利用されている。CT とは、問題を抽象化し分析することで効率的な問題解決を可能にする考え方であり、1980 年代に Paper[1]

^{*1} <https://scratch.mit.edu>

¹ 同志社大学大学院理工学研究科
Doshisha University, Graduate School of Science and Engineering

² 立命館大学情報理工学部
Ritsumeikan University, College of Information Science and Engineering

³ 同志社大学理工学部
Doshisha University, Faculty of Science and Engineering

a) kiuchi.masahiro@mikilab.doshisha.ac.jp

b) makihara@fc.ritsumei.ac.jp

c) ono.keiko@mikilab.doshisha.ac.jp

d) hanamoto.nagi@mikilab.doshisha.ac.jp

によって提唱され、近年 Wing[2] によって一般化された。Scratch におけるユーザの CT を評価する方法の一つに、Dr.Scratch[3] がある。Dr.Scratch は、作品中のプログラムで使用されるブロックを解析し、作品制作で使用された7つの CT 概念を0点から3点、合計21点満点で評価する。Dr.Scratch による評価は、専門家らの評価と相関があることが示されており、様々な先行研究 [4][5][6] で利用されている。

Scratch は教育機関においてプログラム導入教材として使用されることも多いが [7]、一方教育現場以外において、学習者が自ら進んで学習を行うインフォーマルラーニング環境も提供している [8]。そのため、Scratch には数学や英語とは異なり、決められた学習順序が存在せず、学習者の CT 習熟段階を把握することが困難である。インフォーマルラーニングにおけるユーザの習熟度把握を目的とした Yang らの研究では、Scratch におけるユーザは、複数の作品制作を経て、使用ブロックの種類が増加することを明らかにした [9]。また、安東らの研究では、高い CT スキルを必要とする作品を制作するまでに、複数の作品制作を通して習熟が必要な CT 概念を明らかにした [10]。上記の結果から、ユーザは複数の作品制作を通して、CT 概念を習熟することが示唆されるが、高い CT スキルを必要とする作品を制作できるユーザがこれまでにどのような順序で各 CT 概念を学んできたかは明らかにされていない。なお、本稿で述べる順序とは、複数のユーザが複数回作品制作を行うことで獲得する、各 CT 概念のスコアの傾向を含むこととする。以上より本研究では、ユーザがこれまでに制作した作品で獲得した CT 概念の点数の遷移を分析することで、インフォーマルラーニング環境におけるユーザの CT 習熟順序を明らかにすることを目的とする。

著者らは事前調査として、ユーザがこれまでに獲得した各 CT 概念の点数の推移から、次に習熟可能な CT 概念を予測するモデルを作成することで、学習過程を明らかにすることを試みた。しかし、制作する作品の規模やジャンルによって作品制作に必要なスキルが異なるため、単純に作品制作で獲得した各 CT 概念の点数の遷移のみから、学習過程を明らかにすることは困難であった。

作品ジャンルの違いに注目した Adams らや LeÓN ら、Vourletsis らの研究では、制作する作品ジャンルによって、ユーザが習熟しやすい CT 概念の違いがあることを明らかにした [4][5][6]。また、Troiano らの研究では、1つのゲーム作品の制作を通して、ユーザが各 CT 概念を習熟する過程を明らかにした [11]。

事前調査や先行研究の結果から、ユーザの学習過程を明らかにするためには、作品制作で獲得した各 CT 概念の点数の推移だけでなく、これまでにユーザが制作した作品ジャンルも考慮する必要があると考える。しかしながら、現状の Scratch コミュニティサイトでは直近に投稿された

作品しかジャンルを確認することができず、ユーザがこれまでに投稿した作品全てのジャンルを把握することは困難である。

そのため、本研究では、自動で Scratch プログラムの作品ジャンルを分類するモデルを提案する。コミュニティサイトに投稿された作品から取得可能な情報を用いて作品ジャンルを分類するモデルを構築し、分類モデルに寄与する特徴量を分析する。具体的には、作品ジャンルごとに異なる特徴が示唆されるソースコードの情報 [5] と、作品から取得可能なテキスト情報を用いて分類モデルを構築し、分類に寄与する要因を分析する。本研究により、過去にユーザが制作した作品ジャンルと作品制作を経て獲得した各 CT 概念の点数の推移を分析することで、ユーザの CT 習熟順序の把握や、学習者の習熟段階に応じたフィードバック、作品推薦等に貢献できると考える。

本稿の構成は以下の通りである。2章で Scratch、3章で先行研究について述べ、本研究の立ち位置を示す。次に、4章で分類モデルの学習に使用したデータセットについて述べる。また、5章では使用する特徴量や分類モデルについて述べ、6章で分類モデルの評価結果について述べる。最後に、7章で評価結果の考察を述べ、8章で本研究をまとめる。

2. Scratch を用いたプログラミング学習

2.1 プログラミング学習環境：Scratch

Scratch とは、MIT メディアラボが開発したビジュアルプログラミング学習環境である。ゲームやアニメーションの作成に特化しており、実行結果が動画形式で表示されるため初学者を対象としたプログラミング教育で利用されることが多い。また、C や Java などの一般的なテキストプログラミング言語とは異なり、自由度が高く文法エラーなどが発生しない。そのため、学習者は難しいプログラミング言語や正しい構文の書き方を覚える必要がなく、データ構造やアルゴリズムの学習に専念することができる。

制作した作品は、アニメーション、アート、ゲーム、音楽、物語、チュートリアルなど、1つまたは複数のジャンルに分類される。ジャンルによっては、特定のプログラミング概念に関する知識が他のジャンルよりも必要であることが明らかにされている [4][5][6]。

2.2 静的解析ツール：Dr.Scratch

Moreno-LeÓN らは、Scratch における学習者の CT を評価するためのツールとして、Dr.Scratch を開発している [3]。Dr.Scratch は、作品中のソースコードに使用されるブロックやプログラムの構造を解析し、7つの CT 概念（フロー制御、データ表現、抽象化、ユーザ対話性、同期、並列、論理）を計測する。Dr.Scratch による CT7 概念の評価方法を表 1 に示す。Dr.Scratch は作品制作で使用され

表 1 Dr.Scratch による CT7 概念の評価方法 [3]

Table 1 Dr.Scratch assessment score for seven CT concepts[3]

CT の概念	0 点	1 点	2 点	3 点
フロー制御	-	2 つ以上のブロックを連結して使用	for ブロック/無限回繰り返すブロックを使用	while ブロックを使用
データ表現	-	定数を入力するブロックを使用	変数ブロックを使用	配列ブロックを使用
抽象化	-	2 つ以上のスクリプトを使用	定義ブロックを使用	クローンブロックを使用
ユーザ対話性	-	開始ブロックを使用	ユーザの入力を伴うブロックを使用	インタラクションを伴うブロックを使用
同期	-	待機ブロックを使用	プログラムを停止する機能を実装	指定条件でプログラムが動作する処理の実装
並列	-	開始ブロックを複数使用	2 つ以上のスクリプトを同時に実行する機能を実装	イベントブロックを実装
論理	-	if ブロックを使用	if-else ブロックを使用	論理演算ブロックを使用

るブロックの種類や数に基づき、各概念を 0 点から 3 点、合計 21 点満点で評価する。例えば、論理では、if 文に対応するブロックを利用することで 1 点、if-else 文に対応するブロックの利用で 2 点、論理演算に対応するブロックの利用で 3 点である。本研究では、ユーザがこれまでに制作した作品で獲得した各 CT 概念の点数の推移を分析することで、ユーザの CT 習熟過程や傾向などの順序を明らかにする。

3. 先行研究

3.1 Scratch ユーザの分析

Yang らは、インフォーマルラーニング環境における学習者の習熟段階把握を目的として、3852 人のユーザが 1 番目から 50 番目までに制作した作品で使用されるブロックの種類数の遷移を分析した [9]。分析の結果、ユーザは繰り返し作品制作を行うことで、使用するブロックの種類数が増加することを明らかにした。この結果から、インフォーマルラーニング環境においても、ユーザは繰り返し作品制作を行うことで、高い CT を習熟可能であることが示唆される。

安東らは、学習者の CT 習熟段階把握を目的に、習熟度到達予測モデルを構築し、特定の習熟度到達に重要である CT 概念を分析した [10]。分析の結果、ユーザが高い CT を必要とする作品を制作するには、同期やフロー制御に関する CT 概念の学習が有効であることを明らかにした。しかし、ユーザが同期やフロー制御などの CT 概念をどの段階や順序で学ぶべきかは明らかにされていない。そのため、本研究では高い CT を保持するユーザがこれまでにどのような順序で各 CT 概念を習熟したかを明らかにする。

3.2 作品ジャンルによる影響

Adams らは、ビジュアルプログラミング言語である Alice と Scratch で制作された 322 件の作品を対象に、ゲーム作品やミュージック作品、物語作品から学べるプログラミングスキルの違いを定量的に分析した [4]。分析の結果、ゲーム作品はミュージック作品と比較して、変数や if 文、ループ文の使用頻度に有意な差があることを明らかにした。

Moreno-LeÓN らは、Scratch コミュニティサイトから

ジャンルの異なる 250 件の作品を収集し、Dr.Scratch を用いて、各ジャンルの違いを定量的に分析した [5]。分析の結果、全ての CT 概念で作品ジャンル間の有意な差を確認し、異なる作品ジャンルは異なる CT 概念を習熟するために有効であることを明らかにした。

Vourletsis らは、物語作品やゲーム作品のリミックスが、学習者の CT 習熟に影響を与えるか分析した [6]。具体的には、静的解析ツールである Dr.Scratch を用いて、リミックス前後の CT スコアの推移を分析した。分析の結果、物語作品のリミックスではフロー制御、ゲーム作品のリミックスではデータ表現とユーザ対話性に関する CT 概念を習熟しやすいことを明らかにした。

先行研究の結果より、ユーザが過去に制作した作品ジャンルを特定することが、ユーザの成長を分析する上で重要な要素である可能性が示唆される。そのため、本研究では作品ジャンルを自動で分類するモデルを作成することで、ユーザが過去に制作した作品ジャンルを明らかにする。

4. 準備

4.1 Scratch プログラムの収集

本稿では、Scratch コミュニティサイト*2に投稿された 5 つの主要ジャンル（アニメーション、アート、ゲーム、音楽、物語）に分類される作品を分析対象とした。まず最初に、Scratch コミュニティサイトから API を用いて、各ジャンルに関する作品データをそれぞれ 20,000 件ずつ取得した。これらの作品の半分は 2022 年 10 月 5 日に、残りの半分は 2023 年 6 月 6 日に取得した。取得するデータには、作品のソースコードだけでなく、図 1 に示すような作品タイトル、ユーザが任意で記述できる作品の使い方やメモ・クレジットに関するテキストデータも含まれる。取得した 20,000 件の作品データには、同一ユーザが同じ作品を複数回投稿したものが含まれていたため、それらを除外した。また分析の都合上、複数ジャンルに分類される作品、処理ブロックを 1 つも含まない作品、英語以外の言語で制作された作品を除外した。最終的に本研究では、各ジャンル 6880 件、合計 34,400 件の作品データを分析対象とした。

*2 <https://scratch.mit.edu/explore/projects/>



図 1 ゲームジャンルに分類される作品 (ID:806660914)*3

Fig. 1 Projects classified in the game genre

4.2 データセットに含まれるプログラム

収集した作品を対象に、各ジャンルによるソースコードの規模間を分析した結果を図2に示す。横軸は5種類の作品ジャンル、縦軸はそれぞれの規模を対数軸で示す。作品内で使用されるブロックの種類数の平均は、アート作品、音楽作品、アニメーション作品、物語作品の順で多く、ゲーム作品が一番多い。また、作品内で使用されるブロックの個数の平均も、アート作品、音楽作品、アニメーション作品、物語作品の順で多く、ゲーム作品が一番多い。この結果は Adams ら [4] の分析結果と一致する。

次に、1作品当たりで使用される10種類のブロックカテゴリ（動き、見た目、音、イベント、制御、調べる、演算、変数、リスト、ブロック定義）に注目する。各ジャンルで頻繁に使用されるブロックカテゴリの違いを図3に示す。横軸は5種類の作品ジャンル、縦軸は1作品あたりに占めるブロックカテゴリの割合を示す。図3より、演算に関する処理を行うブロックは音楽作品では使用されにくく、物語作品やゲーム作品でよく使用されていることがわかる。また、変数に関する処理を行うブロックはアート作品、音楽作品では使用されにくく、物語作品やゲーム作品でよく使用されていることがわかる。上記の結果より、作品ジャンルによって使用されやすいブロックの種類に違いがあるから、作品ジャンルごとに習熟しやすいプログラミングスキルに違いがあることが示唆される。

また、作品ジャンルによるテキスト量の違いを図4に示す。横軸は5種類の作品ジャンル、縦軸は各項目に含まれる単語数を対数軸で示す。図4より、作品規模と同じく他のジャンルと比較して、物語作品やゲーム作品は、使い方やメモ・クレジットの項目で文中に含まれる単語数が多いことがわかった。また、全ての項目において物語作品は文中に含まれる単語数の平均が多いことがわかった。

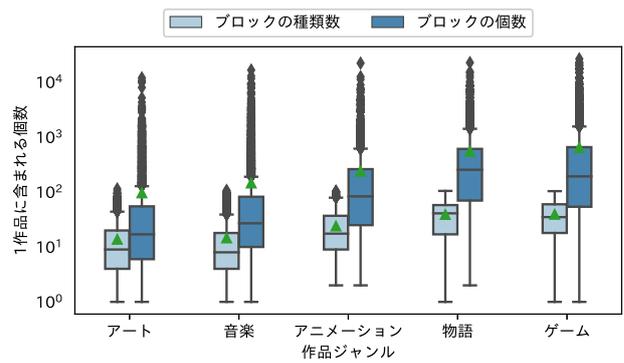


図 2 作品ジャンルによる作品規模の違い

Fig. 2 Differences in project size by genre

5. 分類モデル

5.1 特徴量の抽出

本研究では、Scratch コミュニティサイトから収集した作品のソースコードやテキスト情報を用いて分類モデルを作成する。収集したデータを分類モデルに適用するためには、全ての特徴を数値、またはカテゴリカル変数に変換する必要がある。そのため、ソースコードは全種類のブロックに対して、作品内での使用回数をカウントすることで特徴量として変換する。また、作品の規模感を考慮するため、作品で使用された全ブロック数の逆数を重みとして割り当てる。次に、テキスト情報は Bag-of-Words (以下、BoW) により数値化することで特徴量に変換する。BoW とは語順を考慮せず、文中に存在する全ての単語の出現回数をカウントすることで数値表現に変換する手法である。テキスト情報も、ソースコード同様に文章の長さを考慮するため、文中に出現する全単語数の逆数を重みとして割り当てる。

5.2 ランダムフォレスト

本研究では、分類モデルの構築にランダムフォレスト [12] を用いる。ランダムフォレストは、アンサンブル学習による機械学習アルゴリズムの一つである。アンサンブル学習とは、複数の弱識別機を用い、その結果を結合することでより優れた予測モデルを作成する方法である。ランダムフォレストは、凡化性能の高いモデルを作成するため、複数の決定木を用いることで決定木の過学習しやすい性質を克服している。また、アンサンブル学習の特徴を利用することで、予測結果から分類制度に強く影響した特徴量を特定することが可能である。本研究では分類制度に強く影響した特徴量を分析することで、自動で作品ジャンルの分類するための基準を明らかにする。

予測モデルの検証には層化 K 分割交差検証法を用いる。層化 K 分割交差検証法とは、データセットを K 個に分割し、K-1 個を学習データ、残りを検証データとして使用する手法である。また、K 分割交差検証法と異なり、データ

*3 <https://scratch.mit.edu/projects/806660914/>

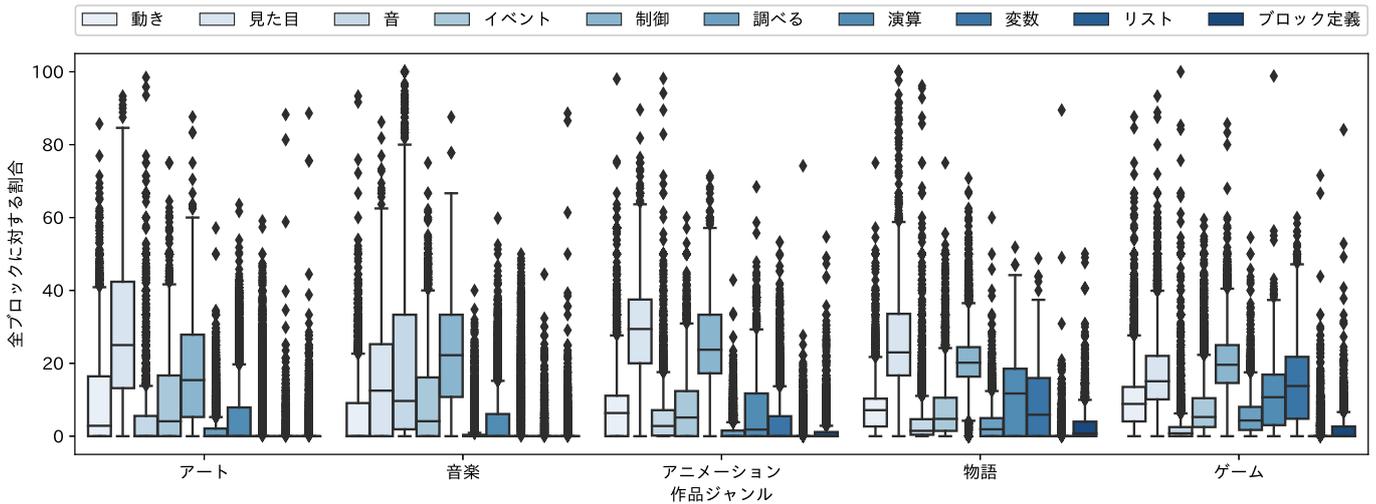


図 3 作品ジャンルによる利用されやすいブロックカテゴリの違い
 Fig. 3 Differences in block categories used by project genres

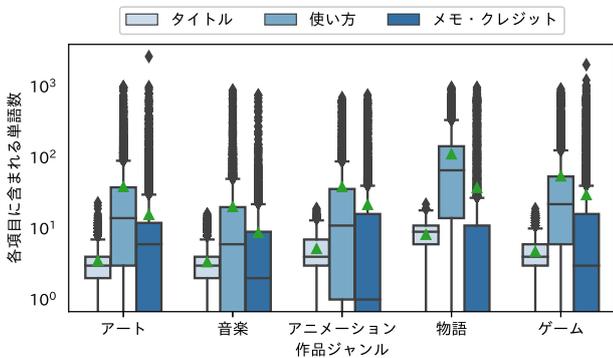


図 4 作品ジャンルによるテキスト量の違い
 Fig. 4 Differences in text volume by project genre

分割時、各クラスのデータが均等になるように分割することで、データの偏りによる精度のブレを抑えている。本研究では K=10 として検証を行う。

5.3 Research Question の設定

本研究では、分類モデルに有効な特徴量を分析するため、以下 2 つの Research Question (以下, RQ) を設定した。各 RQ について詳細に述べる。

RQ1 Scratch コミュニティサイトから収集可能な情報のうちの情報が作品ジャンルの分類に影響を与えているか

RQ2 作品ジャンルを分類する潜在的な要素は何か

RQ1 では Scratch コミュニティサイトから収集した 4 つの情報 (ソースコード、タイトルや使い方、メモ・クレジットに記載されたテキスト情報) を用いて複数の分類モデルを構築し、作品ジャンルの分類に有効な特徴量を明らかにする。具体的には、表 2 に示す複数の分類モデルを構築し、

表 2 各モデルで使用する特徴量

Table 2 Feature values employed in each model

	ソースコード	タイトル	使い方	メモ・クレジット
モデル 1	○	-	-	-
モデル 2	-	○	-	-
モデル 3	-	-	○	-
モデル 4	-	-	-	○
モデル 5	○	○	-	-
モデル 6	○	-	○	-
モデル 7	○	-	-	○
モデル 8	-	○	○	-
モデル 9	-	○	-	○
モデル 10	-	-	○	○
モデル 11	○	○	○	-
モデル 12	○	○	-	○
モデル 13	○	-	○	○
モデル 14	-	○	○	○
モデル 15	○	○	○	○

最も精度が高くなる特徴量の組み合わせを明らかにする。

RQ2 では RQ1 で分類精度が高かったモデルに対して、分類に寄与した特徴量の重要度を分析することで、作品ジャンルを分類する潜在的な要素を明らかにする。本研究では、層化 10 分割検証法によって構築された 10 個の分類モデルに寄与した特徴量を明らかにするため、順位傾向のクラスタ分析手法である Scott-Knott Effect Size Difference (以下, Scott-Knott ESD) 検定 [13] を用いる。Scott-Knott ESD 検定とは階層的クラスタリングを利用して複数のスコアを統計的に異なるグループに分割する手法であり、モデルに寄与する特徴量の分析にも利用される。本研究では層化 10 分割検証法によって構築された 10 個の分類モデルから得られた特徴量の重要度から Scott-Knott ESD 検定を用いていずれのモデルにも強く寄与した特徴量を特定する。

表 3 各モデルの分類精度

Table 3 Prediction results of each model

	適合率	再現率	F 値	Accuracy
モデル 1	0.713	0.700	0.701	0.700
モデル 2	0.783	0.778	0.779	0.778
モデル 3	0.632	0.587	0.595	0.587
モデル 4	0.705	0.547	0.562	0.547
モデル 5	0.836	0.828	0.829	0.828
モデル 6	0.745	0.731	0.732	0.731
モデル 7	0.824	0.818	0.818	0.818
モデル 8	0.818	0.813	0.813	0.813
モデル 9	0.962	0.962	0.962	0.962
モデル 10	0.796	0.767	0.772	0.767
モデル 11	0.839	0.830	0.831	0.830
モデル 12	0.934	0.930	0.931	0.930
モデル 13	0.839	0.833	0.833	0.833
モデル 14	0.957	0.957	0.957	0.957
モデル 15	0.927	0.923	0.924	0.923

6. 結果

6.1 RQ1：Scratch コミュニティサイトから収集可能な情報のうちの情報が作品ジャンルの分類に影響を与えているか

表 2 で定義した分類モデルの精度（適合率，再現率，F 値，Accuracy）を表 3 に示す。紙面の都合上，表 3 に示す適合率，再現率，F 値は各ジャンルに対する適合率，再現率，F 値の平均値である。表 3 より，適合率，再現率の値が極端に異なるモデルは存在しなかった。またモデル 9 は，適合率，再現率，F 値，Accuracy の全ての評価指標において一番高い精度を示した。このことから，タイトルとメモ・クレジットに記述されたテキスト情報を特徴量として用いることで，高い精度で作品ジャンルを分類可能であることがわかった。

また，モデル 9，モデル 12，モデル 14 の結果を比較すると，タイトルとメモ・クレジットに記述されたテキスト情報に加え，ソースコードや使い方に記載されたテキスト情報も特徴量として使用すると分類精度が低下することがわかった。続く RQ2 では，高い精度で分類できたモデル 9 に寄与する特徴量の重要度を分析し，作品ジャンルを分類する潜在的な要素を明らかにする。

6.2 RQ2：作品ジャンルを分類する潜在的な要素は何か

RQ1 で最も精度が高かったモデル 9 において，分類精度に寄与した特徴量の上位 20 件を表 4 に示す。表中には，分類に寄与した特徴量を { 項目/ブロックの種類または文中に含まれる単語 } のように示す。表 4 より，モデル 9 ではタイトルに記述された文中に単語：music が含まれるか否かが分類精度に最も寄与し，続いてタイトルの文中に単語：story が含まれるか否か，タイトルの文中に単語：art が含まれるか否かが分類精度に寄与していることがわかった。このことから，文中にジャンルを指す単語を含むか否

表 4 モデル 9 における重要度の高い特徴量上位 20 件

Table 4 Top 20 most important feature values in Model 9

	重要度	特徴量
グループ 1	0.100	{ タイトル/music }
グループ 2	0.089	{ タイトル/story }
グループ 3	0.085	{ タイトル/art }
グループ 4	0.084	{ タイトル/animation }
グループ 5	0.072	{ タイトル/game }
グループ 6	0.053	{ メモ・クレジット/art }
グループ 7	0.050	{ メモ・クレジット/game }
グループ 8	0.049	{ メモ・クレジット/music }
グループ 9	0.042	{ メモ・クレジット/animation }
グループ 10	0.027	{ メモ・クレジット/story }
グループ 11	0.019	{ タイトル/remix }
グループ 12	0.013	{ タイトル/trend }
グループ 13	0.006	{ タイトル/dump }
グループ 14	0.006	{ タイトル/0-9 (数字) }
グループ 15	0.005	{ メモ・クレジット/code }
グループ 16	0.005	{ タイトル/clicker }
グループ 17	0.005	{ タイトル/platformer }
グループ 18	0.004	{ タイトル/minecraft }
グループ 19	0.004	{ メモ・クレジット/trend }
グループ 20	0.004	{ タイトル/tutorial }

かが分類精度に直結していることが示唆される。

しかし，必ずしも作品のタイトルやメモ・クレジットにジャンルを指す単語が記載されているわけではない。また，Scratch コミュニティサイトの分類とは異なるジャンルを指す単語が含まれている場合もある。具体的には，アニメーション作品に分類された作品のタイトルやメモ・クレジットの文中に単語：art や単語：music など異なるジャンルを指す単語が含まれている場合がある。

そこで，animation, art, game, music, story の 5 つの単語が各ジャンル作品にどの程度含まれているか，また単語の有無によって分類精度が変化するか調査した結果を表 5 に示す。表中のカッコ内は該当する作品のモデル 9 による分類精度を示す。表 5 より，アニメーション作品 6880 件中，6780 件のタイトルやメモ・クレジットの文中に単語：animation を含むことがわかった。そのうち，1513 件の作品では，単語：art や単語：music など別のジャンルを指す単語もタイトルやメモ・クレジットの文中に含まれていることがわかった。一方，100 件の作品では文中に単語：animation を含まず，内 13 件の作品では単語：art や単語：music など別ジャンルを指す単語が含まれていた。この結果より，Scratch コミュニティサイトのジャンル分類は，ユーザが任意で記述する文章，すなわちユーザの思考に強く影響を受けていることがわかった。

また，タイトルやメモ・クレジットに正しいジャンルを指す単語を含む場合は，いずれの作品ジャンルでも 100 % に近い精度で正しく分類される。しかし，異なるジャンルを指す単語を含む場合は，分類精度が低下することがわかった。また，正しいジャンルを指す単語が含まれておらず，異なるジャンル名を指す単語を文中に含む場合は，1 件も正しく分類することができなかった。

表 5 各ジャンルのタイトルやメモ・クレジットに含まれる特定の単語の有無（%：分類精度）
Table 5 Presence or absence of specific words in the title or notes/credits for each genre

作品ジャンル	正しいジャンル名を含む	正しいジャンル名のみを含む	異なるジャンル名も含む	正しいジャンル名を含まない	異なるジャンル名のみを含む	ジャンル名を含まない
アニメーション	6780 (95.8%)	5267 (100%)	1513 (81.1%)	100 (24.%)	13 (0.00%)	87 (27.5%)
アート	6817 (96.5%)	6068 (99.9%)	749 (68.2%)	63 (19.0%)	7 (0.00%)	56 (19.6%)
ゲーム	6769 (97.0%)	6043 (99.9%)	726 (72.2%)	111 (45.9%)	6 (0.00%)	105 (48.6%)
音楽	6812 (97.1%)	6131 (99.9%)	681 (71.7%)	68 (22.1%)	7 (0.00%)	61 (24.6%)
物語	6633 (99.2%)	1096 (100%)	5537 (99.1%)	247 (56.7%)	47 (0.00%)	200 (70.0%)

7. 考察

7.1 CT 習熟順序分析への適用

RQ1 より、作品ジャンルは作品のタイトルやメモ・クレジットに記載されるテキスト情報を用いた際に分類精度が高くなることが明らかになった。Moreno-LeÓN ら [5] の研究より、作品ジャンルごとに獲得可能な CT スコアに有意な差があることから、ソースコードに特徴があることが示唆されるため、ソースコードの情報を用いた際に分類精度が低下することは著者らの予想とは異なった。

RQ2 より、作品のタイトルやメモ・クレジットの文中にジャンルを指す単語 (animation, art, game, music, story) を含む場合は、分類精度が向上し、ジャンルを指す単語を複数種類含む場合や、ジャンルを指す単語を一つも含まない場合は分類精度が低下した。また、約 9 割以上の作品でタイトルやメモ・クレジットの文中に Scratch コミュニティサイトが分類したジャンルと同じジャンルを指す単語が 1 つ以上含むことがわかった。

作品のタイトルやメモ・クレジットはユーザが任意で文章を記述するため、Scratch コミュニティサイトはユーザの意思通りに作品ジャンル进行分类している可能性が高いことが明らかになった。しかしながら、その場合ソースコードが類似している作品でも、ユーザが記述するタイトルやメモ・クレジットの内容によっては、他のジャンルに分類されてしまう可能性がある。実際、作品タイトルに単語：animation を含み、Scratch コミュニティサイトがアニメーション作品に分類した作品を著者らが目視で確認したところ、ソースコードの特徴は限りなくゲーム作品に近い作品が複数存在した。

このように、Scratch コミュニティサイトが定義する作品ジャンルは、作品制作者の意思が強く反映されるため、第三者が作品を確認して、同じ作品ジャンルに分類されるには限らない。著者らはユーザの CT 習熟順序を分析するために、過去にユーザが制作した作品ジャンルを考慮する必要があると考えていたが、Dr.Scratch が作品のソースコードから、定量的にユーザの CT を分析していることを鑑みると、ユーザの主観により変化する可能性がある Scratch の作品ジャンルを CT 習熟順序の分析に用いるのことは不適切である可能性がある。そのため今後は作品ジャンルでは

なく、作品の規模や構造など、ソースコードから定量的に評価できる指標を取り入れることで、ユーザの CT 習熟傾向を分析する。具体的には、ユーザが過去に複数の作品を経て獲得した各 CT 概念の点数の推移と作品制作で使ったブロックの数と種類数の遷移から、今後ユーザが各 CT 概念で 3 点に到達可能か予測するモデルを構築する。本研究で構築する予測モデルは、Scratch ユーザの CT の傾向把握や、Scratch コミュニティサイトに投稿された作品の中から、ユーザの CT 習熟段階に合わせた作品推薦に貢献できると考える。

Scratch では、1 からプログラムを作成することもできるが、他のユーザが公開したプログラムを複製し、編集することで、新しいプログラムを作成することもできる。Scratch ではこの過程をリミックスと呼び、Scratch コミュニティはリミックスを推奨している [14]。また、リミックスを通して異なる知識やスキルを持った他のユーザが作成したプログラムを理解することが、プログラミング学習に有効であると示されている [6][15]。しかし、ユーザがリミックス対象の作品を選択する際、現状 Scratch コミュニティサイトは、閲覧数や評価などが多い人気作品を検索結果の上位に表示するため、必ずしもユーザの CT 習熟段階にあった作品が上位に表示されるわけではない。そのため、ユーザは自身の CT 習熟段階に適した作品を見つけることが容易でない。この課題の解決方法として、本研究で構築する予測モデルを用いることで、よりユーザに適したリミックス作品の推薦が可能になると考える。

7.2 妥当性の脅威

本研究でデータセットに用いた作品が過去に複数ジャンルに分類されていた可能性がある。本研究では Scratch コミュニティサイトの各ジャンルタブから直近に公開された取得可能な作品データを全て取得し、データを取得時に複数ジャンルのタブに掲載されている作品、すなわち複数ジャンルに分類された作品を除き、学習データとして利用した。しかし、作品ジャンルによりタブの更新頻度が異なるため、例えばデータを取得した時点ではアートタブにしか掲載されていなかった作品が、過去にアートタブとゲームタブなどの複数のジャンルタブに掲載されていた可能性がある。本研究でデータセットとして利用した作品に複数

ジャンルに分類されるデータが含まれていた場合、分類モデルの精度低下につながる恐れがある。しかし、本研究のデータセットの作成方法は、先行研究 [5] と同じ手法であるため、一定の妥当性があると考えられる。

8. おわりに

Scratch では、制作する作品ジャンルにより、ユーザが習熟可能な CT 概念が異なるため [4][5][6]、ユーザがこれまでに制作した作品ジャンルは、ユーザの学習過程を分析する上で重要な要素であることが示唆される。しかしながら、現状の Scratch コミュニティサイトでは直近に投稿された作品しかジャンルを確認することができず、ユーザがこれまでに投稿した作品全てのジャンルを把握することは困難である。

そこで本稿では、Scratch コミュニティサイトに投稿された作品データから、作品ジャンルを分類するモデルを構築し、作品ジャンルの分類に寄与する要因を分析した。具体的には、Scratch コミュニティサイトに投稿された 34,400 件の作品データを対象に、Scratch コミュニティサイトから取得可能な 4 つの情報（ソースコード、タイトルや使い方、メモ・クレジットに記載されたテキスト情報）を用いて、どの情報を特徴量として利用した際に分類モデルの精度が高くなるか分析した。

分析の結果、作品内で使用されるブロックの種類や数ではなく、作品のタイトルとメモ・クレジットのテキスト情報を特徴量として用いた際、分類モデルの精度が一番高くなることがわかった。その要因として、animation, art, game, music, story の 5 つの単語が分類精度に強く寄与していることが明らかになった。この結果から、Scratch コミュニティサイトは、作品ジャンルを分類する際にソースコードの特徴ではなく、ユーザの意向を重視している可能性が示唆される。

著者らはユーザが過去に制作した作品ジャンルを考慮することでユーザの成長過程を分析可能だと考えていたが、上記の結果踏まえ、ユーザの主観により変化する可能性がある Scratch の作品ジャンルを成長過程の分析に用いるのは不適切であると考えられる。そのため今後は作品ジャンルではなく、作品の規模や構造など、ソースコードから定量的に評価可能な指標を取り入れることで、ユーザの CT 習熟過程や傾向などの順序を分析する。本研究によりユーザの習熟過程や傾向が明らかになれば、Scratch コミュニティサイトに投稿された作品の中から、ユーザの CT 習熟段階に合わせた作品推薦が可能になると考える。

謝辞 本研究は JSPS 科研費 20K14101 の助成を受けたものです。

参考文献

- [1] Papert, S.: *Mindstorms: Children, Computers, and Powerful Ideas*, Basic Books, Inc., USA (1980).
- [2] Wing, J. M.: Computational Thinking, *Communications of the ACM*, Vol. 49, No. 3, p. 33–35 (2006).
- [3] Moreno-León, J., Robles, G. and Román-González, M.: Dr. Scratch: Automatic analysis of scratch projects to assess and foster computational thinking, *RED. Revista de Educación a Distancia*, No. 46, pp. 1–23.
- [4] Adams, J. C. and Webster, A. R.: What Do Students Learn about Programming from Game, Music Video, and Storytelling Projects?, *In proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, Association for Computing Machinery, p. 643–648 (2012).
- [5] Moreno-León, J., Robles, G. and Román-González, M.: Towards Data-Driven Learning Paths to Develop Computational Thinking with Scratch, *IEEE Transactions on Emerging Topics in Computing*, Vol. 8, No. 1, pp. 193–205 (2020).
- [6] Vourletsis, I. and Politis, P.: Exploring the effect of remixing stories and games on the development of students' computational thinking, *Computers and Education Open*, Vol. 3, pp. 1–8 (2022).
- [7] 文部科学省：小学校プログラミング教育に関する研修教材, https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1416408.htm (Accessed on 6/18/2022).
- [8] Maloney, J., Resnick, M., Rusk, N., Silverman, B. and Eastmond, E.: The Scratch Programming Language and Environment, *ACM Transactions on Computing Education*, Vol. 10, No. 4 (2010).
- [9] Yang, S., Domeniconi, C., Reville, M., Sweeney, M., Gelman, B. U., Beckley, C. and Johri, A.: Uncovering trajectories of informal learning in large online communities of creators, *In proceedings of the Second ACM Conference on Learning@Scale*, pp. 131–140 (2015).
- [10] 安東亮汰, 伊原彰紀: コンピュータショナル・シンキング・スコアに基づく Scratch ユーザの習熟度到達予測, *情報処理学会論文誌*, Vol. 63, No. 4, pp. 928–937 (2022).
- [11] Troiano, G. M., Snodgrass, S., Argimak, E., Robles, G., Smith, G., Cassidy, M., Tucker-Raymond, E., Puttick, G. and Harteveld, C.: Is My Game OK Dr. Scratch? Exploring Programming and Computational Thinking Development via Metrics in Student-Designed Serious Games for STEM, *In proceedings of the 18th ACM International Conference on Interaction Design and Children*, p. 208–219 (2019).
- [12] Breiman, L.: Random Forests, *Machine Learning*, Vol. 45, No. 1, p. 5–32 (2001).
- [13] Tantithamthavorn, C., McIntosh, S., Hassan, A. E. and Matsumoto, K.: The Impact of Automated Parameter Optimization for Defect Prediction Models, *IEEE Transactions on Software Engineering (TSE)* (2018).
- [14] Monroy-Hernández, A. and Resnick, M.: Empowering Kids to Create and Share Programmable Media, *Interactions*, Vol. 15, No. 2, p. 50–53 (2008).
- [15] Dasgupta, S., Hale, W., Monroy-Hernández, A. and Hill, B. M.: Remixing as a pathway to computational thinking, *In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pp. 1438–1449 (2016).