

Moodle を対象とした教員向けダッシュボードにおける xAPI 利用の検討

森本容介¹ 古川雅子²

概要 : Moodle を対象とした教員向けダッシュボードで使用するため、Moodle の学習履歴データを xAPI のステートメントとして表現する手法を設計した。Moodle のイベントログから学習行動を同定し、それぞれに対応するステートメントを、相互運用性を考慮して設計した。設計を通して明らかになった課題を共有し、考察を行う。

キーワード : xAPI, Moodle, 学習履歴データ, 学習分析

Investigation for Using xAPI in a Moodle Dashboard for Teachers

YOSUKE MORIMOTO^{†1}

MASAKO FURUKAWA^{†2}

1. はじめに

1.1 学習分析と xAPI

コンピュータを用いた学習が一般的になり、e ラーニングシステムや、学習端末に学習履歴データが蓄積されるようになった。それらのデータを使って、教育、学習の改善に生かす学習分析 (learning analytics) が注目され、研究が行われている。代表的な研究内容として、学期の途中での成績予測と介入、学習パターンと成績の関連の分析、教材の推薦、成績評価の自動化、分析結果の視覚化と教員、学生へのフィードバック等があげられる。

学習分析が登場した初期の事例として、アメリカ合衆国インディアナ州のパデュー大学における、Course Signals と呼ばれるシステムと学生支援の取り組みがあげられる。この取り組みでは、LMS における学習履歴、高校の成績、学生の属性などの各種データから、学生の成績を予測する。そして、学生の画面に信号機に見立てたアイコンで予測を表示するとともに、落第等のリスクが高い学生に早期に介入することによって、学生を支援する。その結果として、在学の継続率や授業の成績が向上したことが報告されている[1]。2010 年代の初頭に、このような事例がいくつかの大学から報告されたことにより、学習分析の有効性が急速に認知されることになった。

この取り組みのように、学習履歴データの集約結果や学習分析の結果を、教員や学生にフィードバックして、教育、学習の改善に生かすことが求められる[2]。データ分析の専門家でないユーザに視覚的に提示するシステムを、ダッシュボードと呼ぶことがある。ダッシュボードについても多様な研究が行われている[3]。

e ラーニングシステムには、アクセス履歴のほか、テス

トへの解答と結果、電子掲示板への投稿内容、課題の提出物など多様な形式の学習履歴データが蓄積される。また、e ラーニングシステム以外での学習履歴データとして、Web 上の教材閲覧履歴、ソーシャルメディア上での活動履歴、電子書籍の閲覧履歴なども考えられる。システムや組織をまたいで学習履歴データを統合し、学習分析を行うためには、学習履歴データの表現形式や意味を統一し、相互運用性を確保する必要がある。そのような目的に使用できる標準規格として、ADL (Advanced Distributed Learning) により、Experience API (xAPI) が策定された[4]。xAPI では、学習履歴データを、JSON で記述されるステートメント (statement) として表現する。ステートメントは、LRS (learning record store) に記録される。xAPI では、ステートメントのデータ構造と、ステートメントを LRS と送受信する方法を定めている。xAPI を用いることにより、学習履歴データの相互運用性が確保でき、組織を超えた利活用や、データの長寿命化も期待できる。

1.2 xAPI の課題

xAPI では、ステートメントのデータ構造を定めているが、学習履歴データの粒度や、ステートメントで用いる語彙は定めていない。簡単な例では、ビデオの視聴履歴を、“10:25 から 15 分視聴しました” というステートメントで記録する方法と、“10:25 に視聴を開始しました”、“10:40 に視聴を終了しました” という 2 つのステートメントで記録する方法が考えられるが、その指針は決められていない。また、“視聴しました” の類語として、“再生しました” や “見ました” など、複数の表現が考えられる。xAPI では自然言語ではなく IRI で表現するが、その語彙は定められていない。それらは利用者が定める必要があり、その設計をプロファ

¹ 放送大学
The Open University of Japan

² 国立情報学研究所
National Institute of Informatics

イル (profile) と呼ぶ。利用者が独自にプロフィールを作成するのではなく、広く共有されるプロフィールを使用することにより、相互運用性の向上や実践の効率化が期待できる。xAPI を用いて異種の学習履歴データを収集し、教育、学習の改善を行う取り組みは、e ラーニング企業やユーザーグループから報告されている。しかし、用いたプロフィールが公開されることは少ない。

xAPI Vocabulary and Profile Server[5]と The Registry (The Experience API Registry) [6]では、xAPI で使用できる語彙やプロフィールを収集し、公開している。また、xAPI の関連規格として、プロフィールの記述方法や、共有方法に関する仕様が定められている[7]。しかし現時点では、共有されるプロフィールや語彙の不足、片寄りや、同じ意味を持つ語彙の重複登録などが見られ、十分整備されているとは言い難い。Berg らは、権威づけられた (authoritative) プロフィールを提供する必要性を指摘している[8]。実践の共有やそれに基づく改良により、広範な学習に対応できるプロフィール群を整備することが望ましい。

1.3 本研究の目的

現在、第一著者の所属する大学において、xAPI を用いた LRS の構築を進めている。まずは、Moodle を用いた授業の学習履歴データを蓄積し、教員を対象としたダッシュボードを構築する。データの視覚化や学習分析の実践を通して、xAPI の表現可能性、ステートメント設計の妥当性、ステートメントのデータ加工や分析の容易性などを検証したい。そのうち本研究では、Moodle における学習履歴を xAPI のステートメントとして表現する方法を設計し、考察を行うことを目的とする。

本研究は、[9][10]をベースに、Moodle のバージョンを更新するとともに、実運用システムの学習履歴データへの適用と、ダッシュボードにおける使用方法を考慮し、再設計を行ったものである。

2. 前提とステートメント設計の方針

2.1 対象システム

対象とする Moodle は、フルオンラインの遠隔授業で用いている。ビデオの視聴や小テストへの解答で学習し、評価用の小テストやレポートなどで成績評価が行われる。ビデオ教材のうち、ビデオファイルのみのものは Vimeo から配信する。ビデオまたはオーディオファイルとスライドを組み合わせるものは、SCORM パッケージとして制作し、Moodle から配信する。Vimeo に掲載するビデオ教材のために、Moodle の活動モジュールとして mod_vimeo を開発し、使用している。mod_vimeo は、Vimeo のビデオを表示する機能のほか、視聴ログを取得する機能を持っている。活動モジュールとして標準的な閲覧の履歴に加えて、ビデオの再生、一時停止、終了、シーク、再生速度やボリュームの変更、字幕表示の切り替えを記録する。また、Moodle への

ログイン時にユーザーエージェントを記録する機能を開発し、使用している。

ステートメントは、Moodle のデータベースからバッチ処理により生成し、LRS を介さずに、MongoDB に記録する。ダッシュボードとして、Metabase を使用する。

2.2 xAPI のステートメント

xAPI では、学習履歴データを、学習者を表す actor、行動を表す verb、対象を表す object、結果を表す result、文脈を表す context の組み合わせとして表現する。ステートメントは、これらの actor や verb をキーとする JSON のオブジェクトであり、actor や verb の値も JSON のオブジェクトである。ステートメントの設計は、actor、verb、object、result、context の表現方法を定める作業に相当する。

2.3 ステートメント設計の方針

本研究では、異種の学習履歴データを組み合わせて用いることを考慮し、次の方針でステートメントを設計する。

ステートメントには、学習分析に有用と考えられるデータのみを含める。object、result、context には、JSON のオブジェクトとして任意のデータを含められる拡張方法が定義されているが、その使用は必要なものに限る。

語彙は同種のものと同種のを区別するために用いられる。同種概念には同じ語彙を、異種概念には異なる語彙を使用する。また、広く使われ、寿命が長いと考えられる語彙を使用する。新しい語彙は、極力作らない。

そこで、xAPI の仕様で規定されたもの、および xAPI Vocabulary and Profile Server に掲載されているものから語彙を選定する。プロフィールで使用する語彙として、次の形式の IRI を用いることが推奨されている。

```
https://w3id.org/xapi/ [profile name] /  
[concept type] / [concept]
```

例えば、Video プロファイル ([profile name] = video) で用いる、“再生しました” ([concept] = played) を表す動詞 ([concept type] = verbs) は、https://w3id.org/xapi/video/verbs/played である。また、ビデオ ([concept] = video) を表すアクティビティの種類 ([concept type] = activity-type) は、https://w3id.org/xapi/video/activity-type/video である。xAPI Vocabulary and Profile Server には、本稿の執筆時点で、10 のプロフィールが登録されている。xAPI Vocabulary and Profile Server から選定する語彙は、ADL が定めたと考えられるもの、および“ADL Vocabulary”と“Video”プロフィールで規定されたもののみを用いる。

3. ステートメントの設計

3.1 学習行動の同定

Moodle 上で何らかの行動を行うと、イベントが発生する。Moodle の既定では、イベントはログとしてデータベースに

記録される。イベントには、学習者の履歴のほか、管理者や教員の履歴、管理ツールの動作記録などがある。そこで、まずは学習行動としてステートメント生成の対象とするイベントを選定する。Moodle 3.9.3 の標準配付物が生成するイベント、ユーザエージェントを記録するためのイベント、mod_vimeo が生成するイベントは、計 513 種類である。これらから、ステートメント設計の対象とする学習行動を抽出した。活動モジュールとしては、対象システムで使用していない、H5P (mod_h5pactivity), IMS コンテンツパッケージ (mod_imscep), Wiki (mod_wiki), チャット (mod_chat), データベース (mod_data), レッスン (mod_lesson), 外部ツール (mod_lti), 投票 (mod_choice), 用語集 (mod_glossary), 調査 (mod_survey) を対象外とした。結果、本研究では 38 のイベントを対象とする。

3.2 Moodle の要素の階層構造と表現

Moodle の要素は、図 1 のような階層構造をしていると考える。

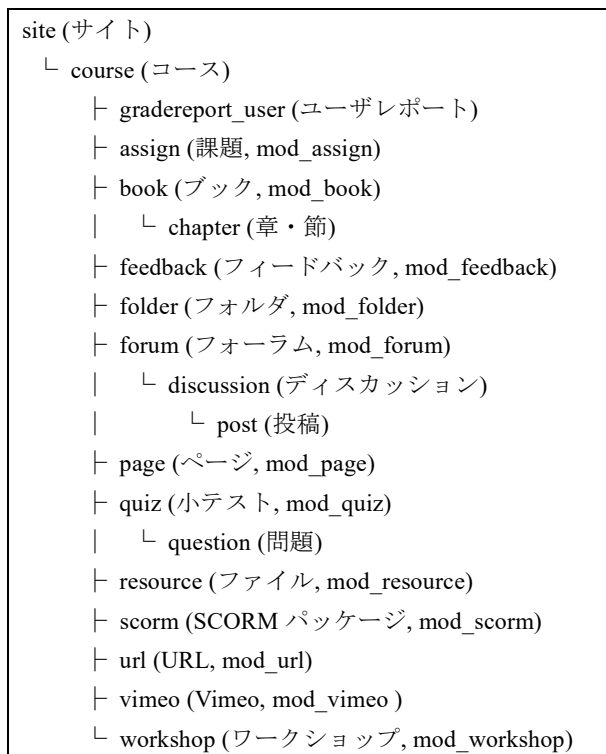


図 1 Moodle の要素の階層構造

サイトの下には複数のコースがあり、コースの下には複数の課題やブックが配置される。ブックは複数の章・節からなり、小テストは複数の問題からなる。なお、Moodle の問題は小テストの子要素ではないが、ここでは子要素と考える。フォーラムの下にはスレッドに相当するディスカッションがあり、ディスカッションの下には複数の投稿がある。投稿はさらに、図 2 のような階層構造をしている。Moodle ではディスカッションの作成と 1 件目の投稿は同時に行われる。つまり、discussion 1 の作成者と post 1 の投

稿者は同じである。post 1 に対して post 2, post 4, post 5 の返信があり、post 2 に対して post 3 の返信、post 5 に対して post 5 の返信があったことを表している。

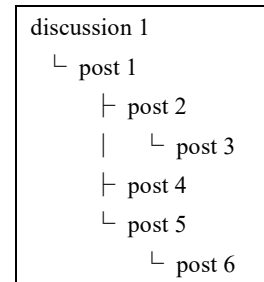


図 2 投稿の階層構造

これらの Moodle の要素は、学習対象や関連要素としてステートメントに記載される。本稿では、xAPI の用語を用いて、これらの要素をアクティビティ (activity) と呼ぶ。まず、アクティビティの表現方法を設計した。図 3 にコースの表現の例を示す。id の値にアクティビティの識別子 (IRI) を指定する。本研究では、識別子に該当ページの URL を用いる。ディスカッションのページには複数の投稿が含まれるが、フラグメント識別子により、それぞれの投稿に固有の URL を作成できる。小テストの問題の識別子は、問題のプレビューページの URL とする。

```

{
  "id": "https://example.com/course/view.php?id=xxx",
  "definition": {
    "name": {
      "ja": "XXXXXX"
    },
    "type": "http://adlnet.gov/expapi/activities/course"
  }
}
    
```

図 3 コースの表現

以降、JSON のオブジェクトが入れ子になっている場合、キーをドット (.) でつないで表現する。definition.name には、コース名、教材名、投稿の件名など、アクティビティの適切な名前を図 3 のような形式で記載する。definition.type にはアクティビティの種類を記載する。2.3 節で述べた方針で語彙を選択した結果、サイト、ユーザレポート、フィードバック、フォーラム、フォーラムのディスカッション、フォーラムの投稿、ワークショップを表す適切な語彙が見つからなかった。definition.type は必須ではないため、これらのアクティビティでは記載を省略する。

小テストの問題は、definition.interactionType に問題の種

類を記載する。Moodle の問題タイプと interactionType の対応を、表 1 に示す。

表 1 Moodle の問題タイプと interactionType の対応

Moodle の問題タイプ	interactionType
description	ステートメント生成の対象とし
essay	ない。
gapselect	
match	matching
multianswer	other
multichoice	choice
random	受験回によって問題が異なる。 出題された問題タイプに対応する interactionType を記載する。
shortanswer	fill-in
truefalse	true-false

小テストの問題の definition.description には問題文を記載し、interactionType に応じて correctResponsesPattern (正答)、choices (選択肢) などを記載するが、仕様通りであるため説明を省略する。

3.3 ステートメントの基本構造

基本的には、対象となる Moodle のイベント 1 つごとに、ステートメントを 1 つ生成する。ステートメントの基本構造を図 4 に示す。図中の①~⑪は、次のとおりである。

- ① ステートメント ID として、UUID v4 を生成して記載する。
- ② 学習者 (actor) の識別に大学のアカウントを用いるため、大学を表す適切な URL を記載する。
- ③ ソルトに相当する文字列と学生番号を組み合わせ、UUID v5 を生成して記載する。
- ④ イベントに対応した verb の IRI を記載する。verb の語彙については、次節で説明する。
- ⑤ verb の最後の “/” 以下の文字列を記載する。例えば、verb が <https://w3id.org/xapi/adl/verbs/logged-in> であれば、“logged-in” とする。
- ⑥ 学習行動の対象を、前節で説明した形式で記載する。
- ⑦ result は、必要がある場合のみ記載する。3.5 節、3.7 節、3.9 節で説明する。
- ⑧ 図 1 に記載した階層構造に従って、親のアクティビティを記載する。配列であるが、記載するアクティビティは直接の親である 1 つのみである ([4] の 2.4.6.2 ContextActivities Property 参照)。図 2 の例では、post 2 の親として post 1 を、post 1 の親として discussion 1 を、discussion 1 の親として該当するフォーラムを記載する。object がサイトの場合、context.contextActivities.parent を記載しない。

- ⑨ 関連するアクティビティの配列である。object がサイトの場合、サイトのアクティビティを記載する。そうでなければ、サイトとコースのアクティビティを記載する。
- ⑩ プラットフォームの名称として、“Moodle” と、ステートメント生成時の Moodle のバージョンを記載する。バージョンは、Moodle の config テーブルから取得できる。
- ⑪ イベントが記録された時刻を、RFC 3339 の形式で記載する。

```

{
  "id": " ① ",
  "actor": {
    "account": {
      "homePage": " ② ",
      "name": " ③ "
    }
  },
  "verb": {
    "id": " ④ ",
    "display": {
      "en": " ⑤ "
    }
  },
  "object": {
    ⑥
  },
  "result": {
    ⑦
  },
  "context": {
    "contextActivities": {
      "parent": [
        ⑧
      ],
      "grouping": [
        ⑨
      ]
    },
    "platform": " ⑩ "
  },
  "timestamp": " ⑪ "
}

```

図 4 ステートメントの基本構造

3.4 verb の語彙

対象とする Moodle のイベントに対応する verb の語彙を

選定した。まず、2.3 節で述べた方針に基づき、xAPI Vocabulary and Profile Server に登録された次の 31 種類の中から選定した。

- <http://adlnet.gov/expapi/verbs/> で始まる語彙のうち、<http://adlnet.gov/expapi/verbs/experienced> (experienced) と <http://adlnet.gov/expapi/verbs/interacted> (interacted) を除いた 23 種類
 - <https://w3id.org/xapi/adl/verbs/> で始まる 5 種類
 - <https://w3id.org/xapi/video/verbs/> で始まる 3 種類
- experienced と interacted を除いた理由は、以前は catch-all verb とされていたためである。その結果、適切な語彙が見つかったのは、対象とする 38 種類のイベントのうち、サイトへのログイン (<https://w3id.org/xapi/adl/verbs/logged-in>) や、ビデオの再生 (<https://w3id.org/xapi/video/verbs/played>) など、7 種類のみであった。基本的な学習行動と考えられる“閲覧した”に対応できる語彙がないことが原因の 1 つである。次に残りの 31 種類のイベントの verb を、experienced と interacted から選定した。その結果、experienced が適切と考えられるイベントは 17 種類であった。これらはすべて“閲覧した”である。interacted が適切と考えられるイベントは、ワークショップでの評価に関する 2 種類である。ここまでの結果、12 種類のイベントに対応する verb を選定できなかった。例として、課題における“提出した”、フォーラムにおける“投稿した”、小テストにおける“受験を開始した”などがあげられる。フォーラムにおけるディスカッションでは、“閲覧した”、“作成した”、“削除した”を区別しなければならず、これらに catch-all verb を使うことはできない。そこで今回は、The Registry に登録されている <http://activitystrea.ms/schema/1.0/> で始まる語彙を追加で用いることにした。その結果、すべてのイベントに対応する verb を選定できた。

次節以降で、前節で述べた基本構造への追加項目や、追加ステートメントを説明する。

3.5 活動完了の更新

コースモジュールの活動完了が更新されたときに、イベント `core:event/course_module_completion_updated` が発生する。コースモジュールの初回の起動においても、このイベントが記録される。活動完了が完了に変わった場合のみ、ステートメントを生成する。また、図 5 のように、`result.completion` の値を `true` としてステートメントに記載する。

```
"result": {  
  "completion": true  
}
```

図 5 活動完了の更新時の `result.completion`

3.6 フォーラムに関するイベント

フォーラムに関するイベントのうち、ディスカッション

の作成、削除、閲覧、および投稿の作成、削除、更新は、`context.contextActivities.grouping` にフォーラムとディスカッションのアクティビティを追記する。つまり、サイト、コース、フォーラム、ディスカッションの 4 つのアクティビティの配列とする。

3.7 小テストに関するイベント

小テストに関するイベントは受験回を区別することが必要である。しかし、受験回の記載に適したデータ項目は定義されていない。そこで、図 6 のように、`context.extensions` に受験回を記載することにした。

```
"context": {  
  "extensions": {  
    "urn:x-moodle-quiz-attempt": 3  
  }  
}
```

図 6 受験回の記載

`extensions` の値として記載するオブジェクトのキーは IRI である。本研究では、仮に `urn:x-moodle-quiz-attempt` とした。Moodle のプロファイル、もしくはテストのプロファイルで定義されることが望ましいと考える。小テスト受験の開始、送信、レビューに対応するステートメント、および小テストの送信に付随して生成する後述のステートメントに、受験回を記載する。

小テストの送信に対応するステートメントは、`result` に得点を記載する。これは仕様通りであるため、説明を省略する。

小テストの送信に付随して、小テストの問題ごとにステートメントを生成する。ただし、表 1 で対象外とした問題は除く。verb は <http://adlnet.gov/expapi/verbs/answered> を用いる。受験回を記載する点以外は、仕様通りであるため、説明を省略する。

3.8 ユーザーエージェントの記録

2.1 節で述べた通り、対象とする Moodle へのログイン時にユーザーエージェントを記録するためのイベントが発生する。そこで、このイベントをログインとみなし、Moodle 本来のログインのイベント (`core:event/user_loggedin`) はステートメント生成の対象にしない。ユーザーエージェントは学習分析に必要なデータ項目であるが、ステートメントへの記載箇所は定義されていない。そこで、図 7 のように、`context.extensions` にユーザーエージェントを記載することにした。受験回同様、仮のキーとして `urn:x-user-agent` を用いる。

```
"context": {
  "extensions": {
    "urn:x-user-agent":
      "Mozilla/5.0 . . . "
  }
}
```

図 7 ユーザーエージェントの記載

3.9 Vimeo に関するイベント

Video プロファイルに従って、ステートメントに表 2 の項目を追加する。プロファイル通りであるため、詳細は省略する。

表 2 Vimeo に関するステートメントへの追加項目

追加項目	対象イベント
context.extensions にビデオの長さ (https://w3id.org/xapi/video/extensions/length)	開始, 一時停止, 終了, シーク
result.extensions にビデオの再生位置 (https://w3id.org/xapi/video/extensions/time)	開始, 一時停止, 終了
result.extensions にシーク先の再生位置 (https://w3id.org/xapi/video/extensions/time-to)	シーク

4. 設計したステートメントの課題と考察

本章では、ステートメント設計で得られた課題や考察を述べる。

4.1 学習者の識別子の匿名化

学習履歴データの活用にあたり、個人情報の保護が求められる。xAPI の仕様では、学習者の匿名性を保つ方法は規定されていない。本研究では、k-匿名性等が要求されるような利用方法は想定しておらず、ソルトに相当する文字列と学生番号から UUID v5 を生成し、学習者の識別子とした。ただし、本データの活用にあたっては、識別子から学生を同定する処理を禁止する条項を含む学内規定を設け、利用申請、許可のプロセスを経て使用可能とする予定である。

4.2 語彙

本研究では、verb, アクティビティの種類 (activity type), extensions のキーについて、語彙を選定、作成した。

4.2.1 verb の語彙

選定方針に基づき、catch-all verb 以外から verb を選定できたイベントは、38 種類中 7 種類のみであった。catch-all verb を使っても、12 種類のイベントに対応する verb を選定できなかった。選定対象を広げて、[6]で公開されている語彙を使えばすべて選定できた。

本研究では、“評価した”を表す語彙として、catch-all verb

である <http://adlnet.gov/expapi/verbs/interacted> を用いたが、専用の語彙を使うことが適切と考えられる。[6]には <https://w3id.org/xapi/acrossx/verbs/evaluated> と <https://w3id.org/xapi/dod-isd/verbs/evaluated> が登録されているが、それぞれ AcrossX と DOD IMI (アメリカ合衆国防総省の Interactive Multimedia Instruction) のプロファイルで定義された語彙であり、Moodle における学習行動に適用することは適切ではない。

[5]と[6]とで同じ学習行動を表す verb が異なる例や、同じ学習行動を表すと考えられる verb が重複登録されている例が見られる[10]。CLA toolkit では、Facebook のシェアと Twitter のリツイートに同じ verb を割り当てるなど、システムをまたいだ verb の統一を試みている[11][12]。学習分析の効率化のためには、同種の学習行動に同一の語彙を割り当てるべきである。ステートメントの高寿命化のためにも、一般的な学習行動を表す語彙が整備されることが望まれる。

4.2.2 アクティビティの種類語彙

3.2 節で述べた通り、7 種類のアクティビティに対する適切な語彙が見つからなかった。アクティビティの種類は、学習分析に有用と考えられるため、語彙の整備が望まれる。

ブック、ブックの章・節 (chapter)、ページに対して、<http://adlnet.gov/expapi/activities/media> を割り当てた。この場合、ブックと、その章・節が区別できない。IEEE LOM における Aggregation Level のように、粒度の概念を導入することにより解決できる可能性がある。

SCORM のプロファイルが用意されている[13]が、本研究では用いなかった。これは、対象システムではビデオ教材を SCORM パッケージとして配信しており、それらの教材ではデータモデルを活用していないからである。SCORM のプロファイルでは、アクティビティの種類として、SCORM パッケージは <http://adlnet.gov/expapi/activities/module>、SCO は <http://adlnet.gov/expapi/activities/lesson> と規定している。しかし、本研究では、SCORM に <https://w3id.org/xapi/video/activity-type/video> を割り当てた。一般的な SCORM パッケージを併用する場合などでは、SCORM パッケージに用いる語彙を固定することができない。この点に関しては今後検討したい。なお、対象システムで用いている SCORM のビデオ教材は、現在のところ再生や停止などを記録する機能を有していない。

4.2.3 extensions のキー

本研究では、小テストの受験回、ログイン時のユーザーエージェント、Video プロファイルで規定されたビデオに関するデータのために extensions を用いた。Video プロファイルでは、表 2 のようにキーの語彙が規定されている。前者 2 つの語彙も xAPI Vocabulary and Profile Server で見つかったが、選定方針に合致しないため使用しなかった。

4.3 context.contextActivities.grouping の使用法

サイトとコースの情報を、context.contextActivities.grouping に含めた。当面、ダッシュボードでは、Moodle の学習履歴データのみを使用し、教員は自身のコースの学習履歴データのみを使用する。context.contextActivities.grouping の情報は、サイト内、およびコース内の学習履歴データのみを集計する場合に使用することを意図している。3.6 節において、フォーラムとディスカッションのアクティビティを追記した理由も同様である。特定のフォーラムやディスカッションごとに投稿を集約するなどの操作を意図している。

4.4 学習セッションに関する検討

既存のプロファイルでの学習セッションの表現を3つ述べる。

xAPI のプロファイルの1つである cmi5[14]では、教材(AU)を起動するときに、LMS がセッション ID を発行する。セッション ID は、図 8 のように context.extensions に記載する。

```
"context": {
  "extensions": {
    "https://w3id.org/xapi/cmi5/context/extensions/sessionid": "3b8a2e74-2778-4b3d-8b6d-ce3648270395"
  }
}
```

図 8 AU のセッション ID

SCORM プロファイル[13]では、SCO の初期化時にアテンプトを表す IRI を生成する。この IRI は、図 9 のように context.contextActivities.grouping に記載する。

```
"context": {
  "contextActivities": {
    "grouping": [
      {
        "id": "http://adlnet.gov/courses/compsci/CS204/lesson01/01?attemptId=50fd6961-ab6c-4e75-e6c7-ca42dce50dd6",
        "definition": {
          "type": "http://adlnet.gov/exapi/activities/attempt"
        }
      }
    ]
  }
}
```

図 9 SCO のアテンプト ID ([13]より抜粋)

Video プロファイルでは、セッション ID を生成し、図 10 のように context.extensions に記載する。

```
"context": {
  "extensions": {
    "https://w3id.org/xapi/video/extensions/session-id": "241675c7-7405-11eb-9fc4-e0db550d04aa",
  }
}
```

図 10 Video プロファイルにおけるセッション ID

小テストの受験回もセッションと考えることができる。本研究では、図 6 のように context.extensions に受験回を記載した。受験回を actor, object と組み合わせることにより、受験セッションを一意に識別できる。なお、actor と object から生成される UUID は registration であり、ステートメントに含めることもできるが、本設計では使用しない。受験回の代わりに、図 8~図 10 の例の UUID 等、一意な ID を発行することも考えられる。学習セッションは一般的な概念と考えられることから、統一的な手法が用意されることが望ましい。

4.5 その他

その他、本研究での設計における細かい課題を述べる。

mod_vimeo では、セッションの識別を行っておらず、図 10 のようなセッション ID を含めることができなかった。複数の端末で同時にビデオを再生したり、開始や終了のイベントが正しく記録されなかったりした場合に、視聴時間が正しく集計できない。セッション ID を扱う機能が必要であることがわかった。

対象とする Moodle では、学期ごとにコースを作成している。しかし、設計したステートメントでは学期を識別できない。現在の運用では、学期の期間外にコースにアクセスされることはないが、分析、視覚化の手法によっては学期の識別が必要になる可能性がある。

ユーザーエージェントをログイン時にのみ記録している。Moodle は複数の端末から同時にログインすることができる。端末によって学習する教材や学習行動に違いがあることも考えられる。すべてのステートメントにユーザーエージェントを記載することが望ましい。その場合は、Moodle のカスタマイズ、プラグイン開発や、Web サーバソフトウェアのアクセスログとの照合が必要となる。

result.duration には、学習行動の時間の長さを記載できる。学習分析のためには、学習時間も重要な情報である。しかし、Moodle のログをベースとしたため、本設計では result.duration は使用できなかった。

5. まとめと今後の課題

本研究では、Moodle の学習履歴データを xAPI で表現する手法を設計した。まず、Moodle のイベントログから学習行動を選定した。次に、各イベントに対するステートメントを設計した。その結果、特に verb に関する語彙の不足が明らかになった。その他、設計を通して明らかになった課題と考察を述べた。

著者らは、類似の研究を 4 年前に行っている[10]。xAPI のプロファイルに関する状況は、当時とあまり変わっていない。個々の実践者がプロファイルを作成するのではなく、広く使われるプロファイルが広範にわたって準備されることが望ましい。[10]では、Moodle のプロファイルの設計を目指した。本研究における Video プロファイルのようにスコープの狭いプロファイルが用意され、それらを組み合わせる Moodle のプロファイルとすることがよいかもしれない。Moodle を対象としたステートメント設計の参考にできるものとして、Berg らによるレシピ[8]、Jisc (Joint Information Systems Committee) によるレシピ[15]、Moodle のプラグインである logstore_xapi[16]などがある。これらとの比較も必要である。xAPI のプロファイルについては、引き続き研究を行いたい。

また、本研究での設計を実データに適用し、ダッシュボードによる分析、視覚化を試行する予定である。その実践を通して、設計の妥当性などを検討したい。

謝辞 本研究の一部は、JSPS 科研費 15K12424, 17K18665 の助成を受けた。

参考文献

- [1] Arnold, K. E., Pistilli, M. D.. Course Signals at Purdue: Using Learning Analytics to Increase Student Success. Proc. 2nd International Conference on Learning Analytics and Knowledge, 2012, p.267-270.
- [2] EDUCAUSE. Analytics for Student Success. 2020 EDUCAUSE Horizon Report Teaching and Learning Edition, 2020, p.20-22.
- [3] Schwendimann, B. A. et al.. Perceiving Learning at a Glance: A Systematic Literature Review of Learning Dashboard Research. IEEE Transactions on Learning Technologies, 2017, vol.10, no.1, p.30-41.
- [4] ADL. “Experience API Specification v1.0.3” . <https://github.com/adlnet/xAPI-Spec/tree/1.0.3>, (参照 2021-02-22).
- [5] “xAPI Vocabulary and Profile Server” . <http://xapi.vocab.pub/>, (参照 2021-02-22).
- [6] “The Registry” . <https://registry.tincanapi.com/>, (参照 2021-02-22).
- [7] ADL. “Experience API Profiles Specification v1.0” . <https://github.com/adlnet/xapi-profiles>, (参照 2021-02-22).
- [8] Berg, A. et al.. The Dutch xAPI Experience. Proc. Sixth International Conference on Learning Analytics & Knowledge, 2016, p.544-545.
- [9] 森本容介, 古川雅子, 山地一禎. Moodle 上での活動を Experience API で表現する手法の検討. 教育システム情報学会研究報告, 2016, vol.30, no.5, p.99-106.
- [10] 森本容介, 古川雅子, 山地一禎. Moodle を対象とした xAPI のプロファイルの策定. 情報処理学会研究報告, 2017, vol.2017-

CLE-21, no.20, p.1-8.

- [11] Kitto, K. et al.. Learning Analytics beyond the LMS : the Connected Learning Analytics Toolkit. Proc. Fifth International Conference on Learning Analytics & Knowledge, 2015, p.11-15.
- [12] Bakharia, A. et al.. Recipe for Success - Lessons Learnt from Using xAPI within the Connected Learning Analytics Toolkit. Proc. Sixth International Conference on Learning Analytics & Knowledge, 2016, p.378-382.
- [13] ADL. “Experience API SCORM Profile” . <https://github.com/adlnet/xAPI-SCORM-Profile/blob/master/xapi-scorm-profile.md>, (参照 2021-02-22).
- [14] ADL. “cmi5 Specification Profile for xAPI” . https://github.com/AICC/CMI-5_Spec_Current/blob/quartz/cmi5_spec.md, (参照 2021-02-22).
- [15] Jisc. “xAPI recipes for Jisc Learning Analytics v1.2.0” . <https://github.com/jiscdev/xapi>, (参照 2021-02-22).
- [16] “Moodle plugins directory: Logstore xAPI” . https://moodle.org/plugins/logstore_xapi, (参照 2021-02-22).