

WebCTにおけるアセスメント作成作業の効率化を目的とした 支援ツールの構築

岡山大学 教育開発センター 長瀧 寛之

nagataki@cc.okayama-u.ac.jp

1 はじめに

本稿では、コース管理システム (Course Management System, 以下 CMS) の活用における、コンテンツ登録、特にアセスメント作成にかかる作業負担を軽減させるための、筆者が作成し利用した一連のツール群について紹介する。具体的には、柔軟な自動採点を行うための正規表現を自動生成するツールと、テキストの羅列を入力として大量の小問を含むアセスメントを自動生成するツールについて、それらが必要となった背景とツール構築における工夫点を中心に紹介する。関連して、コンテンツ登録における文字コード上の問題点についても議論する。

本稿で対象とする CMS は WebCT (Blackboard learning system)¹ である。筆者は大阪大学において 2007 年度から 2008 年度にかけて、教師の WebCT 活用を技術面から支援する STS (Student Technical Staff) 業務を行っており、本稿はその実践例に基づいた報告である。

2 背景

2.1 WebCT アセスメント機能

WebCT では、テストやアンケートなどの作成と実施、成績管理が可能な“アセスメント”機能を提供している。問題登録から採点まで全作業が Web ブラウザ上で完結しており、特別な技術がなくてもオンラインテストを実施できる点が利点である。また解答の自動採点や成績管理機能も提供されており、テスト実施における業務負担の軽減が期待できる。反面、様々なテスト形式に対応するため非常に多くのパラメータが用意されており、それでも対応可能な問題形式には限界がある [1] ため、望み通りのテストを行うには教師側にある程度の慣れが必要である。

解答の自動採点は、選択式の解答だけでなく自由記述解答に対しても可能である。自由記述の採点方法と

しては、正答との完全一致や部分一致の他、Perl 正規表現によるマッチングを利用することができる。また、複数の解答候補を登録し、いずれかの候補に該当すれば正解とする採点方法も可能である。

2.2 対象テスト概要

本稿では、ツール作成の対象となるアセスメントとして、ある語学系科目で行った演習問題 (以下本テスト) を取り上げる。本テストは、30–40 個の英単語それぞれの意味を日本語で解答するというものであり、毎週難易度別に 2 種類実施する。

本テストを WebCT 上で実現する際の手順について説明する。まず教師から作業スタッフへ、問題となる英単語と解答となる日本語訳が列挙されたテキストデータが送付される。作業スタッフは一英単語ごとに、一行入力欄に英単語の意味を解答として記入させる“穴埋め形式”の質問として WebCT に登録し、全質問をまとめたものをアセスメントとして登録、学習者に公開する。採点は学習者のテスト終了後自動的に行い、採点結果と正答を学習者に即時フィードバックする。

本テストの解答形式については、教師より以下の条件が提示されている。

- 一つの英単語に対して複数の日本語訳が可能なものがある
- 日本語訳のうち、どれか一つが解答に記入できていれば正解とする
- 日本語訳が複数個列挙されていても正解とする。ただし誤訳が含まれている場合は不正解とする。
- 括弧で括られた部分、また記号“~”は、解答に含まれていなくても正解とする

例として、英単語“issue”に対する日本語訳として、“問題 (点) 課題、(~ を) 発行する”が指定されている場合、解答として「問題」「問題点」「課題」「~ を発行する」「を発行する」「発行する」のうち一つ以上が列挙されていれば正解とする。

¹Blackboard.inc: Blackboard Home,
<http://www.blackboard.com/>

2.3 本テストにおける採点方法

本テストでは解答として複数の日本語訳を列挙可能であることから、 n 個の日本語訳があれば $n!$ 個の正答候補が存在するため、全ての正答候補を登録するのは現実的ではない。よって、正規表現による採点方法を採用する。具体的には、学習者が複数の解答を記入する場合の区切り文字として、半角/全角スペース、半角カンマ(,)、全角カンマ(、)、全角読点(、)のいずれかを用いてもよいこととし、「文字列の先頭に日本語訳のいずれかが一つが現れ、その後、区切り文字 + 日本語訳が 0 回以上繰り返し現れる」解答にマッチするように正規表現を作成する。先ほどの英単語 settle の例の場合、採点条件を満たす正規表現は以下のとおりとなる。

`[^(問題 (点)|課題|((~|)を)発行する)((,|,、|、|、|、|、|^2)(問題 (点)|課題|((~|)を)発行する)){0,}$]`³

以上の正規表現を問題ごとに作成することで、条件を満たす自動採点が可能なアセスメントを作成できることになる。なおアセスメントの採点結果画面では、“正解”欄に正規表現がそのまま表示されてしまうため、“フィードバック”欄に、正解の日本語訳を別途表示することとする。

2.4 問題点

本テストのアセスメント登録を 3 人のスタッフで分担あるいは交代で作業を行ったが、実際作業を開始すると非常に困難を伴うことがわかってきた。

まず、正規表現が複雑なため記入間違いが頻発し、テストのプレビューと間違い修正を繰り返すことに問題作成のかなりの時間を費やすこととなった。さらに、解答文を更新するたびに採点方式が“正規表現”から“完全一致”にリセットされる、フィードバック文字列の入力欄がオプション扱いのため毎回入力欄を表示する操作が必要であるなど、インタフェース仕様上必要な作業ステップが多く、しばしばパラメータ設定忘れなどのミスの要因となった。

また WebCT では、複数の問題を一括で登録する機能が標準で提供されていないため、一問ずつ質問文と解答を地道に入力していくしかなく、大量の小問で構成される本テストの作成は非常に手間がかかった。

結果、アセスメントの完成まで早くても 2–3 時間、特に初回は 6–8 時間かかってしまい、他の業務時間を圧迫しかねない状況になってきた。WebCT の仕様そのも

²□: 半角空白, □: 全角空白

³この正規表現の場合、“問題、問題、問題”など同じ訳を列挙する不自然な解答も正解となるが、誤答が含まれているわけではないので、特に対策はとらなかった。

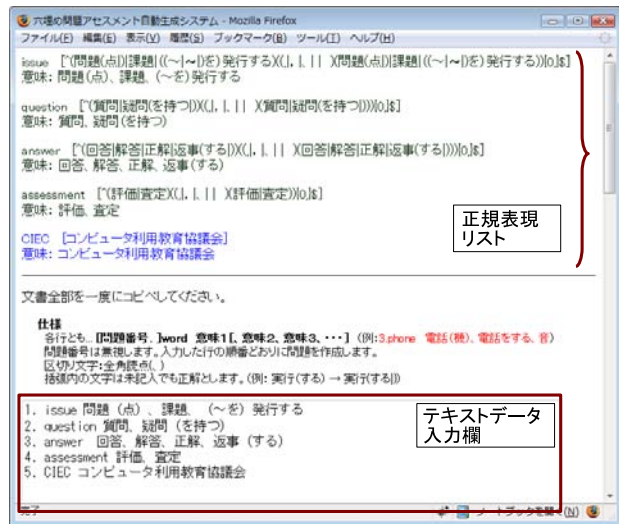


図 1: 正規表現自動生成ツール：実行画面

のを改変することは容易ではないため、WebCT 外で作業効率化のための支援策を講じる必要があった。

3 正規表現自動作成ツール

まず、正規表現作成における作業負担を軽減するため、正規表現の文字列を自動生成するツールの作成を行った。教師から渡されるテキストデータは、1 行ごとに“問題番号”、“英単語”、“日本語訳リスト”が列挙された形式であったため、この形式のテキストデータ全行をそのままコピーするだけで、全問題の正規表現を自動作成するという仕組みとした。

作成したツールの画面例を図 1 に示す。作業スタッフが、画面に表示された複数行入力欄に問題のテキストデータを貼り付け、Submit ボタンをクリックすると、(a) 解答の正規表現を含む WebCT の“穴埋め問題”形式の文字列、(b) フィードバック欄に表示する日本語訳の文字列が、問題の個数分画面上に列挙される。作業スタッフは、各問題の (a) を WebCT アセスメントの“質問文”欄に、(b) を“フィードバック”欄にそのまま貼り付ければよい。

ツール内部の処理について説明する。まずインタフェースから入力された文字列が、各行とも“問題番号. 英単語 □日本語訳 1[、日本語訳 2、日本語訳 3、...]”の形式であると仮定して、問題番号、英単語 (*word*)、日本語訳リスト (*wordlist* = “*ans*₁, *ans*₂, ...”) に分割する。問題番号は WebCT 側で自動的に付与されるため、以降の処理では使わない。次に各日本語訳 (*ans*₁, *ans*₂, ...) について、(1) 全角/半角空白の削除、(2) 括弧つき表現の置換 (‘(aaa)’ → ‘(aaa|)’)、(3) 記号“~”の置換 (‘~’ → ‘(~|)’) の文字列操作を行い、変換後の文字列 (*ans*₁’, *ans*₂’, ...) を、正規表現の OR 記号で結合する

```
[^(問題(点)|課題|((~|~|)|を)発行する)  
(,|,|,|)|)|(問題(点)|課題|((~|~|)|を)  
発行する)){0,}$]
```

～:波ダッシュ, ~:全角チルダ

図 2: 波ダッシュ/全角チルダ両対応の正規表現

(*anslist'* = "*ans*₁|*ans*₂|...").最後に区切り文字 (*delimiters*=" , | , | , | _ | ")と合わせて,穴埋め問題用の文字列 "*word*_□[^{^(*anslist'*)((*delimiters*)(*anslist'*)){0,}\$}]"を生成する.フィードバック文字列には,日本語訳リスト *anslist* をそのまま表示する.

本ツールは Web アプリケーションとして PHP 言語で実装しており,実装行数は 120 行となった.

本ツールによって,正規表現の記述ミスがなくなり,作業時間は 1 テストあたり 30 分程度に短縮できた.また,元となるテキストデータ自体の表記ゆれが度々起こったが,変換後の正規表現の一覧を見比べるとすぐに異常が発見できたことと,フォーマットが単純なため修正が容易であったことから,作業時間への影響はほとんどなかった.

4 自動採点における文字コード問題

正規表現自動生成ツールのデバッグ過程において,正答に全角記号“~”が含まれる場合,Windows 上で解答を入力した場合と Linux 上で解答を入力した場合で採点結果が異なるという問題に遭遇した.

原因は波ダッシュと全角チルダの扱いに関する,文字コードごとの違いにあった.波ダッシュと全角チルダは,EUC-JP や UTF-8 では異なる文字コードが割り当てられており(UTF-8 の場合,それぞれ U+301C, U+FF5E),字体も微妙に異なる.WebCT も内部文字コードに UTF-8 を用いるため両者を区別して扱う.一方 Shift-JIS では波ダッシュと全角チルダが同じ文字コード(0x8160)に割り当てられているため,Windows からは原則全角チルダしか入力できず,波ダッシュはアプリケーションによっては表示すらできない場合がある.その結果,正答文字列に波ダッシュが含まれる場合には Windows で正解を入力することができず,逆に全角チルダが正答に含まれる場合は,Linux や MacOS X では間違えて波ダッシュを解答に入力してしまう可能性がある.

有効な対策の一つとして,波ダッシュと全角チルダどちらの文字でも採点可能のように正規表現を作成する方法があり,本ツールでも同様の対処を行うこととした(図 2).しかし正規表現自体に馴染みが薄い WebCT 利用者は少なくない上,そもそも Windows では波ダッ

シュの入力自体に手間がかかることもあり,WebCT でアセスメントを作成する全ての教師へ同様の対応を求めるには限界がある.また波ダッシュ以外にも同様の問題を抱える全角記号がいくつか存在する(例:Em Dash(U+2014))ことから,複数 OS が学内に混在する教育機関では,解答にはできる限り全角記号を含まないよう各自に注意を促すのが現実的な対策と言える.

5 アセスメントパッケージ自動作成ツール

正規表現の自動生成によって作業時間は短縮されたが,依然としてマウスクリックを多用する質問作成の作業負担は大きいものであった.しかし本テストは,問題と解答の文字列を除いて,各問題の形式やアセスメントの試験時間など全体の設定項目は毎回同じであったため,問題と解答が一括入力でき,かつ残りの設定を過去の雛型から流用できれば,作業の一層の効率化を図ることが期待できる.

WebCT では,アセスメントを IMS コンテンツパッケージ⁴として取り出し(エクスポート),そのパッケージを他コースに流用(インポート)できる機能が提供されている.コンテンツパッケージを使うことで,既存のアセスメントのパラメータ設定を引き継いで別テストを容易に作成することが可能となる.しかし本テストの場合,アセスメント全体のパラメータはコピーできても,各問題の情報を編集するためマウスクリックを多用する手間は変わらない.そこで,WebCT 上での問題作成作業を削減するため,パッケージをインポートするだけでアセスメント作成が完了できるよう,雛型となるコンテンツパッケージに問題情報を自動的に含めてしまうツールを構築することにした.

ツールの実装に先立って,雛型となるコンテンツパッケージの準備を行った.コンテンツパッケージは zip アーカイブファイルで,中身はアセスメントのパラメータや各問題の情報が記述された XML ファイルと XML Schema ファイルの集合である.そこで,まず既に作成した複数の本テストからエクスポートしたコンテンツパッケージを展開し,アセスメントごとに相違がないファイル(不変ファイル)と相違があるファイル(可変ファイル)を確認するとともに,可変ファイルについてはファイル中の相違箇所を特定した.その結果,以下の点を確認した.(1)XML Schema ファイル(*.xsd)は全て不変ファイルである,(2)可変ファイルにはアセスメント全体のマニフェストファイル(imsmanifest.xml)とリソースファイルが一つずつ,問題ごとにマニフェス

⁴IMS Global Learning Consortium: Content Packaging Specification, <http://www.imsglobal.org/content/packaging/>

表 1: コンテンツパッケージのファイル群

分類	ファイル名
不変	/* .xsd (XML Schema)
可変	/imsmanifest.xml
	/X-WEBCT-VISTA-V0/aaa.ccc.R.xml (アセスメントリソースメタデータ)
	/X-WEBCT-VISTA-V0/aaa.nnn.xml (問題マニフェスト&リソースファイル)

aaa: 任意の英数字列 (アセスメントで共通)

ccc, nnn: 任意の数字列 (問題ごと一意)

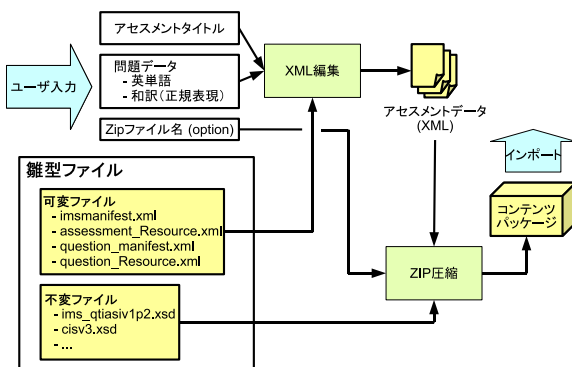


図 3: パッケージ自動生成ツール: 処理の流れ

トファイルとリソースファイルがセットで存在する (表 1), (3) 可変ファイルのファイル名は, imsmamifest.xml を除いて, 前半がアセスメント共通の英数字列, 後半が問題ごとに一意の数字列で構成されている, (4) ファイル中で編集が必要なパラメータは, アセスメントタイトル, 問題ファイルのリスト, 各問題ファイル内の問題文と解答文, 解答形式 (正規表現/完全一致) である.

以上の結果をもとに作成したパッケージ自動生成ツールの処理の流れを図 3 に示す. まずユーザから入力された問題データをもとに, 解答の正規表現を生成し, 次に可変ファイルの雛型をもとに, 各問題のデータをファイル内の該当箇所に挿入したアセスメント用 XML ファイル群を生成する. この際可変ファイルのファイル名は, 基本的に WebCT が生成する XML ファイル名の規則に従い, 前半をアセスメント共通の英数字列, 後半を問題ごとに一意な数字列として, ランダムに生成した文字列を用いることとした. 可変ファイルを作成後, 不変ファイルと合わせて zip 圧縮を行い, コンテンツパッケージファイルを生成する.

本ツールは, 正規表現自動生成ツールを拡張する形で実装した. 具体的には, 英単語と日本語訳のテキス

トデータを一括入力すると, 正規表現に変換した文字列の表示とともに, 自動生成した zip ファイルへのリンクを表示する仕様とした. つまり作業スタッフは, 教師から受け取った問題データをコピー & ペーストで入力すれば, 後は自動生成された zip ファイルを WebCT でインポートするだけでアセスメントを作成できることになる. 本ツールのソースコードの行数は, 正規表現の自動生成処理と合わせて 688 行となった.

本ツールの完成により, 実質的な作業時間はインポートしたアセスメントのプレビューのみとなり, アセスメント登録における作業ミスはほぼ生じなくなった. 結果, 2 テストの作成作業に 30 分もかからない状態まで改善した.

今回作成したツールは本テストに特化した実装となっているが, 同じ形式の質問が何個も続くようなアセスメントであれば, 同様のアプローチで実装が可能である. また, アセスメント全体のパラメータは, 雛型ファイルの入れ替えだけで変更可能である.

6 おわりに

本報告では, WebCT でのアセスメント作成の効率化を目指した, 正規表現自動生成, アセスメントパッケージ自動生成の各ツールの概要について紹介した.

本稿で紹介したツールは特定のアセスメント作成に特化したものであり, その応用範囲は限定的である. しかし, アセスメント作成の汎用性を高めていくと, 次第にユーザが設定すべきパラメータも増加し, 結果的に作業効率が現在の WebCT のインタフェースと同程度に落ちてしまう可能性がある. 今後の展開として, 作業効率と汎用性を両立させたアセスメント作成ツールを検討するとともに, むしろ特定用途のアセスメント作成ツールを積極的に構築することを前提とし, 必要に応じて容易かつ迅速にツールを作成できる環境の実現を, 技術と運用体制の両面から検討していく予定である.

参考文献

- [1] 長瀧寛之: "CMS 上でリスニング試験問題を作成する際の問題点とその解決手段の検討", 情報処理学会研究グループ報告 第 9 回 CMS 研究発表会, pp.48-55 (2008-09)