

完全 e-learning 型講義における データの解析と各コンテンツの改善・運用に関する提案

米田 毅浩[†] 板橋 朋宏[†] 遠藤 亮輔[†] 高岡 詠子[†]
千歳科学技術大学大学院 光科学研究科[†]
千歳科学技術大学 光科学部 光応用システム学科[†]

概要

本学の単位認定型完全 e-learning の科目で情報系の講義である「ソフトウェアデザイン」において、昨年度までに各コンテンツの利用状況と成績との間に相関がみられるか解析を行ってきた。その結果、コンテンツの利用状況と成績の間には関係があることが明らかになった。その結果を用いて今年度の授業に向けてコンテンツの改善・追加が行われてきたが、より正確なデータを得るためさらに細かい解析を行う必要がでてきた。そのため本研究では昨年度までに得られた結果からさらに追跡調査を行いコンテンツ作成に役立てていくと共に、必要な解析結果をすぐに得ることができるような自動解析ツールを設計し、授業の実施から解析・コンテンツの改善までを一連の流れとした運用モデルを提案する。この運用モデルにより、完全 e-learning 型講義の内容を学習者の学習状況にあわせて迅速に対応させていくことが可能となる。

1、 研究の背景

これまでにない学びの形態として、1990年代に米国を中心に普及し始めた e-learning は、ここ数年のパソコンとインターネットの著しい進歩によって急速に日本でも普及してきており、多くの教育機関や企業などでも導入されている。このような流れの中で、e-learning で単位認定を行う大学も増えてきた。しかし、e-learning は「学習者は時間や場所に囚われずに自由な時間に自由な場所で学習を行うことができる」、「学習者が自分のペースで学習を行うことができる」、などの利点がある反面、「学習者のモチベーションの維持が難しい」、「質問などがある場合、リアルタイムでの問題解決ができない」、「講師は学習者の学習状況をデータからしか把握することができない」、というような欠点も併せ持っており、中でも学習者のモチベーションの維持と従来の対面式の授業が無くなることによる学生へのケアをどうするかが重要な問題である。

このような背景を受け、本学でも平成 16 年から全学的に e-learning プロジェクトが進められている。本学でのプログラミングの授業は、システム学科においては 2 年春学期に C 言語プログラミングの必修授業がある。その後「ソフトウェアデザイン」で Java 言語

プログラミング入門の授業を行っている。この授業では平成 16 年度には従来の e-learning を用いない対面授業を、平成 17 年度にはブレンド型授業を、平成 18 年度には完全 e-learning 型というように年ごとに授業形態を変遷してきた。

2、 研究の目的

昨年度までの研究では平成 16 年度から 18 年度にかけての「ソフトウェアデザイン」の授業形態の移行に伴い、各コンテンツを改良しながら運用し、様々なデータから解析を行ってきた[1,2,3]。その結果、以下のような結果が得られた。

- (ア) 学生のコンテンツ利用状況と中間・期末の各成績の間には相関が見られる
 - (イ) (ア)で相関が見られた項目を授業に役立てることができる
 - (ウ) 様々なデータやアンケート結果を円滑な授業を行うために役立てることができる
- しかし、同時に以下にあげる問題点が生じた。
- (ア) より細かなサポートを行うために、多面的なデータの解析を行う必要がある
 - (イ) 取り扱うデータの量が膨大になってしまふ
 - (ウ) 解析を手動で行うため、時間と手間がかかってしまふ

Takahiro Yoneta, Tomohiro Itabashi, Ryosuke Endo and Eiko Takaoka
Chitose Institute of Science and Technology

(エ) 年度が変わる際、解析手順などの引継ぎが不十分なことがある

これらの問題点を解決するためにいくつかの解決策をたてた。

(ア) より細かい解析を行い、コンテンツの修正・サポートに役立てる。

(イ) 自動解析用のツールを作成し、時間の短縮と引継ぎの際の不手際を防ぐ

(ウ) データと解析結果をデータベース化することにより、目的のデータを探しやすくする。また、過去の結果との比較が容易になる。

本研究の目的は上述の問題点を解決し、次年度以降も同じような解析が行えるように自動解析ツールを作成し、そのツールを効果的に利用した完全 e-learning 型授業における運用モデルを提案することである。これにより「より学習効果の高いコンテンツの作成を行うことが出来る」・「学習者に対して細かなサポートを行う」・「解析にかかる時間と手間を減らす」などの利点が考えられる。

3、各年度における e-learning 運用方法の変遷

平成 16 年度から 18 年度にかけての「ソフトウェアデザイン」の授業形態の移行に伴い、その授業形態が以下のように変化してきた。

(ア) 平成 16 年度

春学期にシステム学科 3 年生 130 人を対象とし、対面式で行われ e-learning は使用していない。学生には冊子の教科書が用意され、全 13 週で行われた。

(イ) 平成 17 年度

春学期にシステム学科 3 年生 144 人を対象とし、講師による授業に加え、e-learning を教材としたブレンド型で行った。平成 16 年度と同じく全 13 週で行われた。

(ウ) 平成 18 年度

春学期にはシステム学科 3 年生 128 人を、秋学期にはシステム学科 2 年生 123 人を対象として完全 e-learning 型の授業を行った。学生が出席する必要があるのはスクリーニング 2 回と中間・期末試験であり、それ以外は教室での授業は行わず、学生は学習スケジュールに従って学習を進めていった。

各年度通じて、中間・期末試験は基本的な語句確認するためのペーパーテストと実際に

コーディングを行うオンラインテストが行われた。背景で述べたように授業形態が完全 e-learning 型講義へと推移していく中で平成 16 年度に実施したアンケートに基づき学習者が必要と感じている範囲について平成 17 年度の授業にあわせて新規にコンテンツの作成を行った(表 1・2 参照)。

また、4-1 節で述べるように平成 17 年度において e-learning コンテンツの利用状況と成績の間に相関があるか調査を行い、学生アンケートの結果を踏まえ、平成 18 年度の授業開始前にコンテンツの充実化を図った。大項目として以下に示すように「変数」、「基本的なプログラミング」の項を増やし、変数の説明用の教科書の追加と、17 年度には対面授業で行ったプログラミング実践の授業を映像コンテンツ用にシナリオ化した 3 本の映像を追加した。また、それに伴い教科書やドリルの内容の追加・修正も行った。その結果 17 年度と比べ、解析結果とアンケートをもとにコンテンツの作成・改善を行った 18 年度の各学期に行われた授業において、学習者の成績上昇が見られ、さらにコンテンツ利用状況と成績の間に高い相関を見ることが出来た。

表 1 平成 17 年度コンテンツ一覧

カテゴリ	種類	題名
オブジェクト指向とJava	教科書	オブジェクト指向
	教科書	Java
クラスの概念とインスタンス	映像	オブジェクト指向・クラスの概念
	教科書	クラスの概念
クラス・フィールド・メソッド	アニメーション	クラスの概念
	教科書	インスタンスとコンストラクタ
	教科書	クラスの定義
	アニメーション	パッケージ
	アニメーション	実際にクラスを定義
	教科書	フィールドの宣言
	アニメーション	publicとprivateの違い
	アニメーション	フィールドの宣言
	教科書	メソッドの定義
	アニメーション	メソッドの定義
まとめ	教科書	プログラムの使用し説明
	映像	まとめ
継承①	教科書	継承の概念
	アニメーション	スーパークラスとサブクラスの関係
	アニメーション	継承におけるクラスの持つ性質
	教科書	継承の方法
	映像	継承①
継承②	アニメーション	メソッドのオーバーライド
	教科書	継承におけるコンストラクタの取り扱い
配列	映像	継承②
	アニメーション	プリミティブ型変数
	アニメーション	参照型変数
	映像	配列
	映像	static

表1 平成18年度コンテンツ一覧

平成18年度カテゴリ	種類	題名
オブジェクト指向	教科書	オブジェクト指向
	映像	オブジェクト指向
Java	教科書	Java
	映像	Java、プログラムファイルの構築
変数	教科書	変数(プリミティブ型)
	アニメーション	プリミティブ型変数
	教科書	プリミティブ型のキャスト
クラス概念とインスタンス	教科書	クラス概念
	アニメーション	クラス概念
	教科書	プリミティブ型変数のキャスト
クラスの定義	教科書	クラスの定義
	アニメーション	パッケージ
	アニメーション	実際にクラスを定義
フィールドの宣言	教科書	フィールドの宣言
	アニメーション	public/privateの違い
	アニメーション	フィールドの宣言
メソッドの定義	教科書	メソッドの定義
	アニメーション	メソッドの定義
基本的なプログラミング	教科書	インスタンスとコンストラクタ
	アニメーション	仮引数・実引数・戻り値
	映像	クラス概念
	教科書	変数(参照型)
	アニメーション	参照型変数
	映像	変数について
	教科書	プログラミング作法
	映像	プログラミング作法
	教科書	動くプログラムを作る(前編)
	教科書	動くプログラムを作る(後編)
	映像	プログラムを使用し説明
映像	コンストラクタについて	
今までのまとめ	教科書	まとめ
	映像	まとめの映像
	映像	プログラミング実践1
継承概念と方法	映像	プログラミング実践2
	教科書	継承概念
	アニメーション	スーパークラスとサブクラスの関係
	アニメーション	継承におけるクラスの持つ性質
	教科書	継承の方法
継承におけるコンストラクタ	アニメーション	メソッドのオーバーライド
	映像	継承(1)
	教科書	継承におけるコンストラクタの取り扱い
配列	映像	継承(2)
	教科書	配列の宣言と生成
	教科書	配列の使い方
	映像	配列
	映像	static

(表中網掛け部が新規に追加したカテゴリおよびコンテンツ)

4、コンテンツと成績間の相関

ここでは各コンテンツと成績間にどのような相関があったのかについて述べる。ここでのコンテンツとは実際に本学のe-learningに搭載されている「教科書」・「ドリル」であり、成績とは学期中において2度行われる定期考査のうちの期末試験での学習者の得点のことである。期末試験は基本知識を問うペーパーテストと実際にコーディングを行うオンラインテストに分かれている。本稿ではペーパーテストについての調査結果について述べる。

4-1 コンテンツ利用状況と成績との相関

本研究に先駆け、本研究室では昨年度、平成18年度春学期・秋学期におけるe-learningコンテンツ利用状況と成績との

関連性について解析を行った。コンテンツ利用状況とは、

- ・ 教科書閲覧回数
各単元へのアクセス回数をカウントしたものの。
 - ・ 教科書閲覧率
各単元の教科書を一度でも閲覧していれば100%としたもの。
 - ・ ドリル達成率
学生が課題として出されたドリルを全て解いていれば100%としたもの。
 - ・ ドリル正解数
学生がどれだけドリルで正解したかをカウントしたもの。
- 以上の4つの項目であり、それぞれに成績との相関を求めた結果を表2に示す。

表2 コンテンツ利用状況と成績との相関

相関係数	期末ペーパーテスト得点	
	春学期	秋学期
教科書閲覧回数	☆0.536642120	*0.771624869
教科書閲覧率	*0.666239865	*0.757666498
ドリル平均達成率	☆0.434427950	*0.701251804
ドリル平均正答数	0.145182684	☆0.511118175

(表中の*は高い相関を、☆は中程度の相関を得たもの)

これらの解析を行った結果、ほぼ全ての項目で相関を見ることができた。しかし、成績は様々な要因を含んでいることを考慮すると、これまでの解析ではまだ十分なデータが得られているとはいえない。そこでペーパーテストの問題をカテゴリごとに分け、別々に分析を行うこととした。

4-2 コンテンツとテストの各間における相関

前項で述べたように、各コンテンツと成績の間には相関を得ることができた。

しかし、実際のペーパーテストでは付録1のような教科書・ドリルに解答が記載されている語句問題と、付録2のようなプログラムの読解問題の二つのカテゴリが存在する。そこで、前項の解析結果から、さらにコンテンツ利用状況と語句問題での点数・読解問題での点数・合計点での点数について相関があるかどうかを追跡調査した。今回の調査で用いたコンテンツ利用状況は、

表3 教科書閲覧率と期末テスト素点

教科書閲覧率 期末ペーパー 素点	春学期			秋学期		
	期末テスト ペーパー総合点	期末テスト ペーパー語句点	期末テスト ペーパー読解点	期末テスト ペーパー総合点	期末テスト ペーパー語句点	期末テスト ペーパー読解点
教科書閲覧1	*0.666239865	*0.601259006	*0.705090560	*0.757666498	*0.766365663	*0.658012971
教科書閲覧2	*0.764070079	*0.748934274	*0.682208748	*0.731425244	*0.722404987	*0.665283713
教科書閲覧3	*0.664636860	*0.694360159	*0.502129907	*0.624570593	*0.592356057	*0.610395150
教科書閲覧4	*0.649256715	*0.680927522	*0.484900459	*0.540265241	*0.490129078	*0.566437212
教科書閲覧5	*0.517494658	*0.568143713	0.332412347	*0.489672865	*0.447459603	*0.507823100
教科書閲覧6	*0.481405145	*0.536889236	0.291418895	*0.406385848	0.346789735	*0.463839906

(表中の*は高い相関を、^は中程度の相関を得たもの)

表4 教科書閲覧回数と期末テスト素点

教科書閲覧回数 期末ペーパー素点	春学期			秋学期		
	期末テスト ペーパー総合点	期末テスト ペーパー語句点	期末テスト ペーパー読解点	期末テスト ペーパー総合点	期末テスト ペーパー語句点	期末テスト ペーパー読解点
教科書閲覧回数	*0.536642120	*0.597016468	0.350054238	*0.771624869	*0.722566901	*0.770090454

(表中の*は高い相関を、^は中程度の相関を得たもの)

表5 ドリル利用状況と期末テスト素点

ドリル 期末ペーパー素点	春学期			秋学期		
	期末テスト ペーパー総合点	期末テスト ペーパー語句点	期末テスト ペーパー読解点	期末テスト ペーパー総合点	期末テスト ペーパー語句点	期末テスト ペーパー読解点
達成率平均	*0.434427950	*0.427134874	0.385089637	*0.701251804	*0.692107386	*0.638695284
平均正答数	0.145182684	0.191261334	0.025418161	*0.511118175	*0.512612402	*0.451441445

(表中の*は高い相関を、^は中程度の相関を得たもの)

・ 教科書閲覧率 X

X の値によって 100%とする条件を変えたもの。X が 1 の場合は教科書を一度でも閲覧していれば 100%となり、X が 2 から 6 は教科書を X 回まで閲覧している場合のみ 100%とした。

・ 教科書閲覧回数

各単元へのアクセス回数をカウントしたもの。

・ ドリル達成率

各授業時に課題として出されるドリルを全問正答で 100%としたもの。

・ ドリル正答数

ドリルの解答のなかで正答だけをカウントしたもの

以上の 4 項目であり、それぞれに期末ペーパーテスト平均総合点数・平均語句問題点数・平均読解問題点数との相関を求めた(表 3,4,5)。この結果、教科書閲覧率においては合計点・語句問題点・読解問題点においてほぼ全ての結果で相関を得ることができた。教科書閲覧回数においては、総合点・語句点において春学期は教科書閲覧 3 以下、秋学期は教科書閲覧

2 以下の相関が教科書閲覧回数との相関を上回った。これらの結果から、教科書閲覧率の方が教科書閲覧回数よりも成績との間に強い相関関係があるということになり、教科書は何度も閲覧したからといって必ずしも成績に結びつくものではないという結果を得ることができた。つまり、教科書は閲覧する回数よりもすべての単元を満遍なく見るのが重要であると考えられることができる。

以上より、教科書の改善を行うことによってペーパーテストにおいて語句問題の点数だけではなく、読解問題に関しても点数を上げていくことができると考えられる。しかし、表 5 の結果をみると、ドリルに関しては必ずしも点数を上げることに貢献しているとは言えない。このため、ドリルの使用方法や内容について、見直しが必要になってくることがわかる。また、この調査の結果、春学期よりも秋学期のほうの相関が全体的に高くなった。これは秋学期に講義を受けた 2 年生は春学期に C 言語の講義を行っていたため、プログラムの基本的な考え方に多少は慣れた状態で講義を受けることができたからで

はないかと考えられる。対して春学期に講義を受けた3年生がC言語の講義を受けたのは半年前の2年生春学期であり、2年生の秋学期にはプログラム系の講義を受けておらず、基本的な考え方を忘れてしまったためこのような結果につながったのではないかと考えられる。

さらに、コンテンツの改善を行う際、語句問題に関してコンテンツが実際に学習の役に立っているかを調べるため、語句問題(穴埋め全28問)とコンテンツ利用状況の間に相関が得られるかどうか解析を行った。語句問題は問1～問10までがJavaの基本語句の穴埋めであり、問11～18までがプログラム作成の手順についての穴埋め、問19～問28が変数の型や範囲、配列の扱いについての穴埋め問題である(表6参照)。

表6 語句問題・問題番号と正答

問題番号	正答	問題番号	正答
1	オブジェクト指向	15	小文字
2	フィールド	16	static
3	メソッド	17	クラス変数
4	クラス	18	final
5	インスタンス	19	-128
6	メッセージ	20	127
7	カプセル化	21	0
8	スーパークラス	22	length
9	サブクラス	23	30
10	継承	24	コンストラクタ
11	中間コード	25	デフォルトコンストラクタ
12	VM	26	mainメソッド
13	区別	27	String
14	大文字	28	void

また、それぞれの項目について語句問題一問ごとの平均正答数との間の相関を調査した。表6に語句問題各問の正答を、表7から表10に語句問題とコンテンツ利用状況の相関を調査した結果をグラフ化したものを示す。

表7 春学期教科書平均閲覧率と問題ごとの平均点

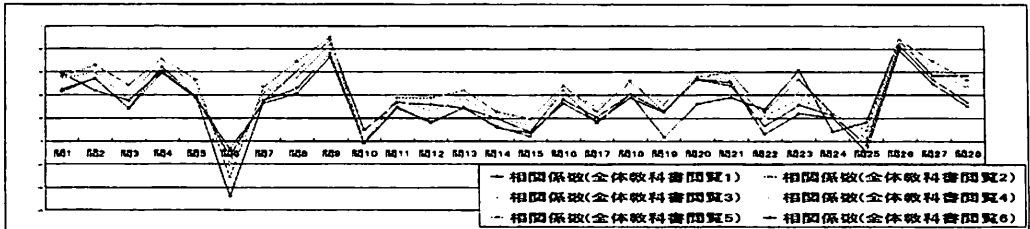


表8 秋学期教科書平均閲覧率と問題ごとの平均点

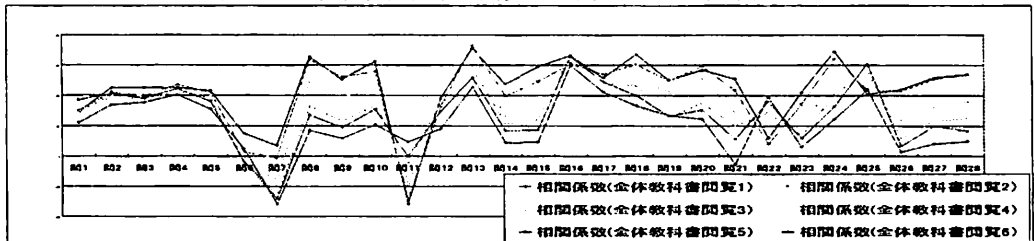


表9 春学期教科書平均閲覧回数・ドリル達成率・ドリル正答率と問題ごとの平均

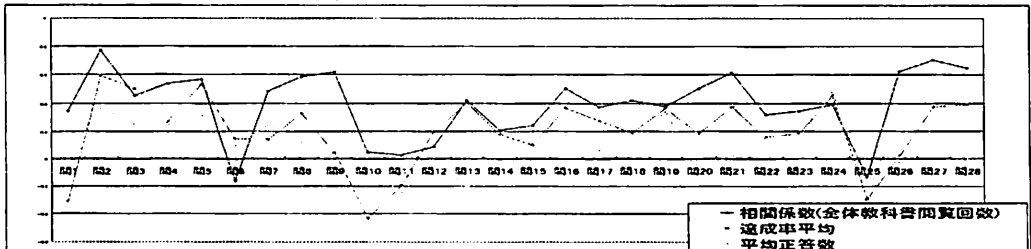
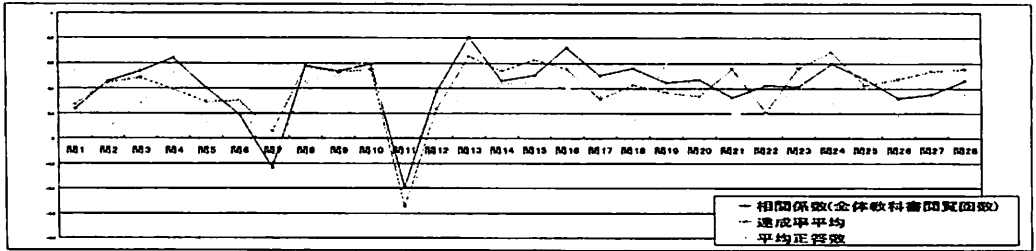


表 10 秋学期教科書平均閲覧回数・ドリル達成率・ドリル正答率と問題ごとの平均点



教科書閲覧率との結果より、問題によって相関にばらつきがはっきりと見られた。高い相関を得られた問題は問2から問5・問21から問23・問24、問25のように、正答が「フィールド」や「メソッド」、「String」などといった Java の基本的な語句に関する問題や、配列の使い方・変数の型について出題したものである。これらの問いはプログラムを記述する上で最低限覚えておく必要があり、教科書などにも強調して書かれている語句であるため相関を得ることができたと考えられる。反対に相関を得られなかった問題は問6、問11・問12、問14・問15である。これは正答が「メッセージ」や「中間コード」となるものであり、プログラムを記述する際には必ずしも覚えておく必要が無い語句であるため、教科書中に強調して書かれていなかったことが原因として考えられる。

また、春学期と秋学期ではコンテンツは全く同じものを使用したにも関わらず相関が一致しなかった問題もいくつか見られた。この原因として考えられるのは、春学期の結果を踏まえて秋学期の授業が行われたことである。ガイダンスや合計2回行われるスクーリングにおいて春学期に相関を得られなかった問題に力を入れて説明が行われたため、学習者が盲目的にその分野のみを集中して学習した結果、このような結果になったのではないかと考えられる。

教科書閲覧回数・ドリル達成率・ドリル正答率等に関する相関係数の推移に関しても同じような理由が考えられるため教科書・ドリルともに改善が必要である。以上を踏まえ、今年度秋学期からの授業に備え、相関の低かった分野のコンテンツの充実など改善を行っていく。

5、 運用モデルの提案

4-2 節で述べたようにさらに細かな解析を行った結果をコンテンツの改善・追加に用いることで、次年度の運用において高い効果を期待できる。しかし、各コンテンツと成績との相関において昨年度までは解析を手動で行ってきたため、データの収集などに非常に時間がかかる。また、前年度との比較を行う際に用いるデータを探し出すのに時間がかかってしまう。そのため、今年度までに行ってきた解析のノウハウを元に、各コンテンツから取得できるデータと必要になるデータを自動的に収集し、相関を求めたいデータを選択することで解析・比較・分析を自動的に行えるツールが必要となった。以上より本研究では各コンテンツと成績間の解析結果を用いたコンテンツの改善と現在開発を行っている自動解析ツールを使用した、データ解析からコンテンツ改善までの流れを完全 e-learning 型講義における運用モデルとして提案する。

5-1 運用の流れ

実際の運用においては以下の①から④を繰り返し行うことによってコンテンツの質を高めていく。

- ① 作成したコンテンツを実際に授業で使用する。
- ② 中間・期末に行われるペーパー・オンライン両テストの学生ごとの点数、各コンテンツの利用状況・成績を取得しデータベース化する。
- ③ データベース化されたデータから相関の調査・利用状況の分析を行う。
- ④ 解析結果と期末テスト時に行うアンケートの結果に応じて、相関の高かった

コンテンツに関してはそれを維持し、相関の低かったコンテンツおよび必要であると判断されたコンテンツに関しては新規作成・改善を行う。

この運用モデルを用いることによって、以下のような利点が見込まれる。

- ・ 講義を繰り返し行うほどコンテンツの質を高めていくことができる。
- ・ 学習効果の高い教科書やドリルがはっきりとわかるため、そのコンテンツに沿ってテンプレートを作ることができる。
- ・ 学習者が苦手としている分野がはっきりとするためフォローしやすい。

また、現段階では②・③において手動でデータを収集し相関をとっているが、ここを5-2節で述べる自動解析ツールによって自動化を行う。

5-2 自動解析ツール

このツールを実現することにより、解析に費やされる時間・作業手順が大幅に削減でき、「引継ぎが容易になる」、「講師の時間に余裕ができ、学生のサポートに割ける時間が増える」、「学習者のニーズに迅速に対応できる」など、様々なメリットを得ることができると考えられる。

また、前述した問題点を解決するために自動解析ツールには以下のような機能を実装する。

- ・ 相関を求めたいデータの選択機能
- ・ 3変数間での相関
- ・ 過去のデータとの比較
- ・ データの出力
- ・ コンテンツ使用状況の分析

これらを実現するためデータをデータベースに登録する際、春学期・秋学期にカテゴリ化する。さらにこれを「中間テストまで」・「期末テストまで」と細かく分けて保存することにより、過去のデータとの比較を容易にする。

また、昨年度まで「教科書閲覧率と成績の相関」・「ドリル達成率と成績の相関」などといったように2変数間で行っていた解析を「教科書閲覧率・成績・ドリル達成率の相関」といったように3変数間に対応させ、より詳細な解析を行う。このように細かい分析を行うことで最終的には成績優秀者は教科書を何回閲覧し、ドリルを何回解いているかなど

いった結果を得ることが可能となる。これにより最も学習効果を期待することのできるコンテンツの使用法の提示や、コンテンツの作成・改善などに役立てていくことができる。

6、 おわりに

今回の研究における解析の結果平成18年度におけるコンテンツ利用状況と成績との間に相関が得られ、次回以降の授業において学習効果を期待できるようなコンテンツの追加・改善を行うことが出来た。また、過去のデータを元に本研究で行ったようなさらに細かい解析を行うことによってより細かな解析結果を得ることができ、この解析を次年度以降も行うことによってさらに正確な相関関係を調査することが可能になると考えられる。そのため、

- ① コンテンツの使用
- ② 利用状況と成績との相関の調査・分析
- ③ コンテンツの追加・改善

といった一連の流れを運用モデルとした提案を行った。モデルを実現するためにデータ解析を自動的に行うツールを実現することでより完全e-learning型講義を学習者の学習状況にあわせて迅速に改善していくことができる。

今後、期末時に行うアンケートの結果や補助教材として開発したJavaプログラミング初等教育の専用統合環境「JavaEditor」[4]などからデータを取得し、これをコンテンツ利用状況と併せて成績との相関を調査することによって実際に学習者が何を求め、何を必要としているのかを把握することができ、よりよい授業を行うことが可能になる。

参考文献

- [1] 高岡詠子、石井和佳奈、"Javaプログラミング入門単位認定型完全e-learningへ向けての試み～コンテンツ構築および実践バージョン～"、情報処理学会研究報告、CE-81, pp.73—80(2005).
- [2] 高岡詠子、石井和佳奈、"Javaプログラミング入門単位認定型完全e-learningへ向けての試み～評価バージョン～"、情報処理学会研究報告、CE-82, pp. 53—60(2005).
- [3] 高岡詠子、"大学におけるデジタルコンテンツ開発の変遷およびe-learningプロジェ

クト", 情報教育シンポジウム SSS2005 プレカンファレンス論文集, pp.46-52, 2005
 [4] 高岡詠子、米田毅浩、澤田英敏、山本啓介、"Java プログラミング教育統合環境「JavaEditor」", 情報処理学会研究報告, CE-83, pp. 149—156(2006).

付録1 ペーパーテスト、語句問題(抜粋)

問い	解答
オブジェクトの状態のことを(1)、オブジェクトの行動のことを(2)と呼ぶ	(1) フィールド (2) メソッド
オブジェクト同士のやり取りを行うために、オブジェクト自身やほかのオブジェクトに対して(1)を送ることができる。 (1)を受け取ったオブジェクトは、それに対応した(2)を実行することで、(3)の値を変更することができる。 基本的に、(1)の受け渡しを行わずに、オブジェクトが保持する値を直接変更することはできない。	(1) メッセージ (2) メソッド (3) フィールド
(1)で宣言された変数は、そのクラスにひとつしか存在せず、(2)と呼ばれ、そのクラスのすべてのインスタンスによって共有される変数である。こう宣言することで、そのクラスのすべてのインスタンスごとに重複した変数のためのメモリ領域を使う必要がなくなるので、処理速度が上がる。 (2)に対して、インスタンスごとに持つ変数を(3)変数という。 (4)で宣言された変数は定数で、変数の値を初期値から変更することはできない。	(1) static (2) クラス変数 (3) インスタンス (4) final

付録2 ペーパーテスト、プログラム読解問題(抜粋)

問い	解答
以下のプログラムの空欄を埋めなさい	
<pre>class OneDimension{ private double xAxis; public OneDimension(double x){ (1); } Public double xAxis(){ (2) } }</pre>	(1) xAxis = x (2) return xAxis