

卒業論文 2012年度（平成24年度）

履修科目推薦システムの設計と実装

慶應義塾大学 総合政策学部

氏名：鴻野弘明

担当教員

慶應義塾大学 環境情報学部

村井 純

徳田 英幸

楠本 博之

中村 修

高汐 一紀

Rodney D. Van Meter III

植原 啓介

三次 仁

中澤 仁

武田 圭史

平成25年1月22日

履修科目推薦システムの設計と実装

近年、科目概要の書かれたシラバスは多くの大学でインターネットから閲覧することが可能となってきた。現状における履修科目の選択は、学生がシラバスの検索システムに興味のある分野のキーワード等を検索クエリとして入力し、検索にヒットした科目の中から自身の履修したい科目を探す手法が一般的である。しかし、検索システムを用いた現状の手法では、検索システムが検索クエリを要するため、学生が履修したい科目について漠然としていて有効な検索クエリが思いつかない場合に、適切な手法とは言えない。本論文では、シラバス検索システムにおける欠点を補完するための履修科目推薦システムを実装し、学生がより快適に履修科目選択を可能とすることが目標である。それを実現するために、単純に既存の汎用的な推薦手法を用いるのではなく、履修科目の推薦に特化した推薦手法を提案した。そして提案した手法の有効性について検証するため、学生に履修科目推薦システムに自身の履修履歴の登録と、各科目に対する評価値を設定してもらい、そのデータを用いてどの程度の精度で推薦を行うことが可能か評価実験を行った。その結果、検索クエリを必要とせず、また一般的に検索システムでは得られにくい意外性のある科目を提示する能力に優れる情報推薦技術を用いた際にも、十分に高い精度で履修科目の推薦を行うことが可能であることを確認した。本論文の成果により、履修科目選択の支援システムとして、検索システムの他に推薦システムを用いることも有用であることが確認され、今後、大学からシラバスの検索システムと共に推薦システムも提供されることによって、学生にとってより快適な履修科目選択環境が構築されることが期待される。

キーワード:

1. 情報推薦システム, 2. 嗜好の予測, 3. 協調フィルタリング, 4. 内容ベースフィルタリング, 5. 科目選択

慶應義塾大学 総合政策学部

鴻野 弘明

Design and implementation of
Course Selection Recommendation System

In many universities, online syllabus has been widely used. A common selection method of courses with online syllabus is to search key words form in it. However, when a user uses the syllabus keyword search, a user needs to input some keywords as a search query. But some students may not come up with any keywords to type in. The purpose of this study is to implement a recommendation system which complements such disadvantages of the syllabus search system. In this study, we used an original specialized algorithm to recommend the course accurately and found that the recommendation system is able to generate recommendation of courses accurately enough for students. Thus we anticipate that if the universities offer the course selection recommendation system as an asisstant program to select courses with more advantages over syllabus search system.

Keywords :

1. Recommendation System, 2. Prediction of Preferences, 3. Collaborative Filtering,
4. Contents-based Filtering, 5. Course Selection

Keio University, Faculty of Policy Management

Hiroaki Kono

目次

第1章	序論	1
1.1	履修科目選択の現状	1
1.2	問題点	1
1.3	本研究の目的	1
1.4	本論文の構成	2
第2章	背景	3
2.1	シラバス	3
2.1.1	シラバスの役割	3
2.1.2	シラバスの問題点	3
2.2	意思決定の際の選択肢	4
2.2.1	選択肢の数についての実験	5
2.2.2	豊富な選択肢の影響	6
2.3	情報抽出技術	6
2.3.1	シラバスの検索システム	7
2.4	情報推薦手法	7
2.4.1	情報推薦システムの処理	8
2.5	推薦システムにおけるユーザデータ獲得	8
2.5.1	明示的なデータ獲得	8
2.5.2	暗黙的なデータ獲得	9
2.6	推薦システムにおける推薦アルゴリズム	9
2.6.1	協調フィルタリング	9
2.6.2	内容ベースフィルタリング	14
2.7	推薦システムにおける推薦アイテム提示手法	15
2.8	推薦システムの評価手法	16
2.8.1	履歴適合度	16
2.8.2	ユーザ評価適合度	17
2.8.3	順位評価適合度	18
2.8.4	セレンディピティ	18
2.9	本論文の着眼点	19
2.10	まとめ	19

第3章	関連研究	20
3.1	履修科目選択支援に関する研究	20
3.1.1	大学教養教育における科目選択支援	20
3.1.2	ダイナミックシラバスの開発	21
3.2	推薦システムに関する研究	22
3.2.1	内容に基づく音楽データの探索・推薦システム	22
3.2.2	協調フィルタリングとコンテンツ分析を利用した観光地推薦手法	24
3.3	まとめ	27
第4章	推薦システムによる履修科目選択支援の提案	28
4.1	提案手法	28
4.1.1	既存研究との違い	28
4.1.2	推薦システムを使用する利点	28
4.1.3	ノーフリーランチ定理	29
4.2	要求事項	29
4.2.1	各科目の推薦度合いの明快さ	29
4.2.2	学生のニーズに合った推薦	30
4.3	まとめ	30
第5章	実装	31
5.1	実装環境	31
5.2	実装したシステムの構成要素	31
5.2.1	認証部分	32
5.2.2	評価入力部分	33
5.2.3	推薦部分	35
5.2.4	アイテム提示部分	37
5.3	まとめ	38
第6章	評価	39
6.1	評価手法	39
6.1.1	$RMSE$ と MAE	39
6.1.2	$RMSE$ 値の評価	40
6.1.3	実際の測定手法	41
6.2	評価用データの準備	42
6.3	評価結果	42
6.3.1	全体データに対する評価	42
6.3.2	評価済み科目数と $RMSE$ 値の関係	43
6.4	まとめ	44

第7章	結論	45
7.1	本研究における目的の達成	45
7.1.1	本研究の目的	45
7.1.2	達成された目的	45
7.2	まとめ	46
7.3	今後の展望	46
	謝辞	48

目 次

2.1	シラバスの例	4
2.2	ジャム試食実験	5
2.3	アイテム情報付加時と非付加時の比較	15
2.4	システム透過性の有無の比較	16
3.1	類似度の可視化	21
3.2	作成された決定木	23
3.3	観光地情報入力フォーム	25
3.4	観光地推薦画面	27
5.1	ログイン後の画面	33
5.2	評価入力画面	34
5.3	推薦結果表示画面	38
6.1	推薦結果表示	43
6.2	評価済み科目数と $RMSE$	44

表 目 次

2.1	ユーザデータ獲得法の長所と短所	9
2.2	ユーザの行動とそのアイテムへの嗜好の強さ	10
2.3	記号の定義	11
2.4	記号の定義	16
2.5	記号の定義	17
2.6	心理状態階層モデルの定義	18
3.1	音楽の特徴量	23
3.2	観光地の特徴量	24
3.3	観光地特徴行列	25
3.4	ユーザ特徴行列	26
5.1	実装環境	31
5.2	システムの構成要素	32
5.3	認証用テーブル	32
5.4	履修済み科目評価テーブル	34
5.5	特徴量定義テーブル	34
5.6	科目情報テーブル	35
5.7	記号の定義	36
6.1	サンプルデータ	40
6.2	Netflix dataset	41
6.3	実際の評価手法の解説に用いる表	42
6.4	最新学期履修科目とそのユーザ評価値データ	43

第1章 序論

本章では、履修科目選択の現状を述べる。その後、現状の履修科目選択に一般的に用いられているシラバス検索システムを使用する際に生じる問題点を挙げる。そして、挙げた問題点を踏まえ、本研究の概要と目的について記述する。最後に本論文の構成を記す。

1.1 履修科目選択の現状

学生が履修する科目の選択を行う手法は各大学により細部は異なり、最初の一週間を試用期間としその期間中に学生が興味のある科目の授業に自由に参加し実際に履修するかどうかを決定したり、各学生にメンターを設け、学生とメンターで相談したりすることにより履修する科目の選択やメタな視点で履修方針を決定したりする手法がある。しかし現状では学生がシラバスを参照して履修する科目の選択を行うのが一般的である。シラバスを参照するには、シラバスの検索システムを用いて何かしらのキーワード等を検索クエリとして検索し、その検索にヒットした結果の中から自身の興味のある科目を探す手法が一般的である。

1.2 問題点

1.1節での述べた現状における問題点は、シラバスの検索システムを利用するには学生が検索クエリを考えて入力する必要があることに起因する。目標や目的がまだ漠然としている学生には検索クエリが思いつかず適切な履修科目選択ができない場合がある。しかし現状では、履修科目選択の手法として大学が学生に対して提供している手法はシラバスの検索システムを利用する手法のみであることが多い。そのため、学生によっては快適な履修科目選択ができていないことが問題点として挙げられる。

1.3 本研究の目的

本研究は履修科目選択において、シラバス検索システムに入力する検索クエリが思いつかないような目標や目的がまだ漠然としている学生にとって快適な履修科目選択を可能とすることが目的である。目標や目的の漠然としている学生にとって快適な履修科目選択を行うためには学生が自身の履修すべき科目を明確に示し、且つその科目が実際に学生にとって満足度の高いものである必要である。

1.4 本論文の構成

本論文は全 7 章から構成される。第 2 章では、シラバスとシラバスの検索について詳細に述べ、その問題点を明らかにする。その後、本論文で提案する推薦システムに関連した技術を体系的に述べる。第 3 章では、履修科目選択の支援に取り組む関連研究と、推薦システムを実用した研究を紹介する。第 4 章では、本研究の目的を達成するための手法を提案する。第 5 章では、第 4 で述べた手法に基づき実装したシステムについて詳しく述べる。第 6 章では、第 5 章で述べたシステムについての評価を行う。最後に、第 7 章では、本論文における結論と今後の展望について述べる。

第2章 背景

本章では、大学などの教育機関において学生が科目を履修する際に必要となるシラバスやシラバスの検索システムについて述べ、そこに生じる問題点を示し、その解決のために本論文で提案する推薦システムに関連した技術について述べる。

2.1 シラバス

シラバスは科目概要を示したものであり、学生はそれを参照して履修する科目を決める。本節ではシラバスの役割と、その問題点を示す。

2.1.1 シラバスの役割

シラバスとは科目の内容、授業計画、評価方法など、科目に関する情報を文章で示されたもので、学生に対して科目を履修する以前に科目についての情報を公開することができる。従来シラバスは冊子などの形式で学生に配布されていたが、現在は図 2.1 (慶應義塾大学湘南藤沢キャンパス・ネットワークプログラミング [1] より) のようにウェブ上で公開されている大学も多くある。ウェブ上で公開されているシラバスは検索システムによって条件を絞ってマッチした科目を表示できる仕組みになっていることが多い。検索により担当教員名や授業開講曜日などの検索クエリにマッチした授業を表示することが可能である。学生は事前にシラバスを参照することで科目の概要を把握しどの科目を履修するか検討することができるため、履修計画の立てやすさなどの点でメリットがある。また、担当教員は学生に対して事前に科目の内容を周知できるため、シラバスに記述された内容に興味を惹かれた学生が集まり、より専門性の高い授業を展開することができるというメリットがある。

2.1.2 シラバスの問題点

履修科目の選択に用いられるシラバスの問題点を以下に挙げる。

(1) 膨大な科目数

科目数は大学により数百以上も存在し、学生がそれぞれ全てのシラバスを参照し比較分析を行い、大学としてどのような授業があるのかその全容を把握してから履修選択をすることは困難である。シラバスには図 2.1 のように関連した科目、推奨若

13070 - ネットワークプログラミング (C) NETWORK PROGRAMMING		創造技法科目-プログラミング	
開講日程	2012年度 春学期 月曜日2時限	担当教員	山本 隆
関連科目	【前提科目(推奨)】13010:コンピュータ基礎とプログラミング 【前提科目(推奨)】13030:データ構造とプログラミング 【関連科目】95014:フューチャーインターネットテクノロジー:アーキテクチャと構成要素 【関連科目】04180:インターネット		
開講場所	SFC	授業形態	講義、実習・演習
履修者制限	履修人数を制限しない。	履修条件	この科目は、2004年度以降入学者は認定
使用言語	日本語		
連絡先		授業ホームページ	
科目概要	UNIX上で動作する小・中規模アプリケーションを題材に、ネットワークアプリケーションの知識とプログラミ:		
科目概要(詳細)			
ネットワークを介したソフトウェアの基本的な構造を理解し、ネットワークを用いたソフトウェアが作成できるようにする。基本的なブロック提出を通して、プログラミングに慣れてもらう。なお今期は IPv6 アドレスを極力利用し、プログラミングをする。			
授業シラバス			
主題と目標/授業の手法など			
□ 主題と目標/授業の手法など □			

図 2.1: シラバスの例

しくは必須の前提となる科目が挙げられていることがあるが、これらは独立度の高い科目に対して弱く、またその科目同士がどのように関連しているのかが不明瞭であることが多いため、履修選択の手助けとして十分とは言えない。

(2) 授業の雰囲気

シラバスは一般的に文字情報のみであるため、文字情報では表現のしづらい情報、すなわち授業の雰囲気などを事前に把握することが困難である。大学により科目の履修者に対して授業についてのアンケートを行い、その回答をシラバスに公開している場合がある。該当科目を次期履修しようと考えている学生はそのアンケート結果を見て学生視点でどのような授業が行われるのかといった情報を得ることができる。しかし、実際にはアンケートへの回答率が低かったり、アンケートの回答が文章形式で数値化しづらかったりといった問題がある。

2.2 意思決定の際の選択肢

意思決定を行う際、選択肢が豊富にあることは一見魅力的なことに思えるが、2.1.2 項で述べたように選択肢の豊富さが選ぶ際の手間を増大させる要因ともなりうる。本節では

一般的に選択肢の増大がユーザに与える影響についての実験を挙げ、それらの実験からわかる選択肢の豊富さに起因する問題点を述べる。

2.2.1 選択肢の数についての実験

本項では選択肢の多い場合と少ない場合のどちらでより多くコンバージョンに至るかを示す、2つの実験を挙げる。

(1) ジャム試食コーナーにおけるジャムの種数と購入者数 [2]

これはジャム販売店において 6 種類のジャムと 24 種類のジャムを置いた試食コー

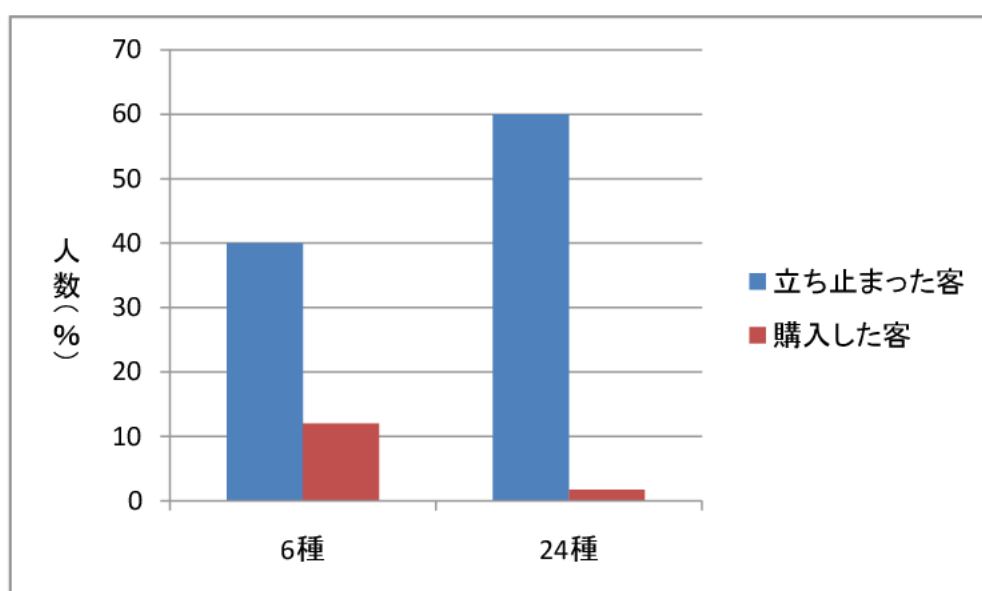


図 2.2: ジャム試食実験

ナーをそれぞれ設け、どちらでより多くの客に購入されるかを実験したものである。実験の結果に影響する時間帯や場所などを同じにして実験を行った結果は、図 2.2 ([2] によるデータを基に作成) のように示された。24 種類のジャムを提示した際は 6 種類提示した場合に比べてより多くの客を引き止めることに成功した。しかし購入まで至った客は 24 種類提示した場合より 6 種類提示した場合の方が多かった。

(2) 退職貯蓄制度における投資信託提案数と加入率 [3]

この実験は投資信託運用会社のヴァンガード・グループにおいて行われたもので、約

100 万人の退職者に対して退職貯蓄制度として利用する投資信託の提案を行い，提案する投資信託の数が 2 から 59 の間の約 650 のプランでそれぞれにおける加入率を比較したものである．実験の結果，提案する投資信託の数を 10 増やすごとに加入率は 1.5% から 2.0% 落ちた．

2.2.2 豊富な選択肢の影響

2.2.1 項の実験から，必ずしもユーザに多くの選択肢を与えることが高いコンバージョン率に繋がるわけではないことが示された．また，実験を通して多すぎる選択肢は次のような悪影響を与えるという結果を得られた．

(1) 選択の先延ばし

ユーザは与えられる選択肢が増大するにしたがって，選択を先延ばしにする傾向が強くなる．これは，ユーザが与えられた選択肢の中から選択を誤ることを恐れる心理が作用し，またそれぞれの選択肢を比較分析をすることは困難であるために選択行為自体を先延ばしにしてしまうためである．

(2) 不適切な選択

ユーザは与えられる選択肢が増大するにしたがって，適切でない選択をする傾向が強くなる．これは，選択の先延ばしと同じように誤りを恐れる心理作用と選択肢比較分析の困難さ故に，最終的にユーザ自身にとって意味のわかりやすい選択肢を選択してしまうためである．ユーザ自身にとって意味のわかりやすい選択肢が必ずしもユーザにとって適切であるとは限らない．

(3) 満足感の減少

ユーザは客観的に見て平均より良い選択をしているにも関わらず，多すぎる選択肢の中から選んだアイテムに対してその良さに見合った満足感を得られない傾向が強くなる．これは，多すぎる選択肢の存在がユーザのアイテムに対しての期待値を釣り上げ，結果として選択したアイテムがユーザの期待値と比較して低くなってしまったために起こる．ユーザの期待値を釣り上げる要因は，多すぎる選択肢がその中に完璧なアイテムがあるという考えを植え付けてしまうためである．選択肢の増大はユーザの期待値を釣り上げることを余儀なくさせることを示している．

2.3 情報抽出技術

情報抽出とは多くの情報から特定の情報を抽出することであり，2.2.2 項において述べた影響を緩和することができる．情報の抽出を行う手法として情報推薦や情報検索が挙げられる．情報推薦と情報検索の違いは，情報推薦は事前に用意されたユーザのプロファイリングデータと他ユーザのプロファイリングデータやアイテムの特徴量からユーザ視点で受動的に行われる情報抽出であり，情報検索はユーザのアドホックな検索クエリに

よってユーザ視点で能動的に行われる情報抽出である。例として、情報推薦については Amazon[4] の「おすすめ商品リスト」や「この商品を買った人はこんな商品も買っています」などが挙げられ、情報検索としては Google[5] の Google 検索や、Amazon の商品検索などが挙げられる。

2.3.1 シラバスの検索システム

学生が膨大な選択肢の中から自分に合った科目を選択することは大きな手間となりうる。そのため多くの場合は全体の一覧からではなくシラバスの検索システムを用いて条件を指定し、選択肢を絞って選択を行う。しかし、シラバスの検索システムを用いた場合には、以下のような問題が発生する。

(1) 分野知識

検索システムでは、自分の全く知らないことを検索することはできない。たとえば、その検索システムが一般的な検索システムのように検索クエリの文字列が含まれるシラバスの科目を列挙する仕組みの場合、学生が学習したいと考えている分野に該当する科目のシラバスに学生の入力した検索クエリの文字列が含まれていない場合にヒットしない。

(2) 学生評価

シラバスには大学によって科目の履修者に対して授業についてのアンケートを送り、その回答を公開している場合がある。該当科目をこれから履修しようと考えている学生はそのアンケート結果を見て学生視点でどのような授業なのかといった情報を得ることができる。しかし、このようなアンケートの回答は検索システムによって検索することができない仕組みになっていることが多く、学生視点からの評価を基に科目を探すことが困難である。

(3) セレンディピティ

セレンディピティとは、直接調べているアイテムとの関係はないが価値のあるアイテムを発見する能力のことを言う。シラバスの検索システムによって得られる結果は、アドホックに与えられた検索クエリのみを考慮した結果となっているため、セレンディピティに欠ける。

以上のように、履修選択において科目を絞る際に単純な検索システムのみを用いることが必ずしも最適ではない。

2.4 情報推薦手法

本節以降では本論文で提案する推薦システムについて関連した技術を述べる。

2.4.1 情報推薦システムの処理

一般的な情報推薦システムにおける処理は大きく 3 つのフェーズに分割することができる。すなわち、ユーザプロフィール獲得フェーズ・嗜好の予測フェーズ・推薦アイテム提示フェーズである。本項ではこれらについて述べる。

(1) ユーザプロフィール獲得フェーズ

情報推薦を行う際には対象ユーザのプロファイリングデータを事前に取得する必要がある。ユーザのプロファイリングデータとはユーザ自身の関心や好みについてのデータのことであり、過去のユーザの購買履歴やユーザの評価に加えユーザ自身の年齢や性別などの情報なども指す。ユーザのプロファイリングデータの獲得手法には明示的手法と暗黙的手法がある。明示的手法はユーザに対してアイテムについての好き嫌いなどを質問する方法で、暗黙的手法はユーザの行動履歴や購買履歴をトラックし、購買した事実や閲覧時間の長いものには関心が強いと見なす方法である。これらについては 2.5 節で詳しく述べる。

(2) 嗜好の予測フェーズ

ユーザプロフィール獲得フェーズに於いて獲得したデータを基に他ユーザのプロファイリングデータを用いたり推薦対象となるアイテムの特徴量を用いたりする手法を用いて嗜好の予測を行う。嗜好の予測は大きくわけて協調フィルタリングとコンテンツベースフィルタリングという手法がある。これらについては 2.6 節で詳しく述べる。

(3) 推薦アイテム提示フェーズ

予測されたアイテムを目的に応じた形で利用者に提示する。提示手法としてはこれらについては 2.7 節で詳しく述べる。

本節以降では上記のそれぞれのフェーズについて詳しく述べる。

2.5 推薦システムにおけるユーザデータ獲得

本節では推薦システムの実行過程における最初の段階である「ユーザデータの獲得」について述べる。ユーザの嗜好データの獲得手法は明示的な手法と暗黙的な手法に大別される。神島氏 [6] は明示的なデータ獲得と暗黙的なデータ獲得について表 2.1 のようにまとめている。

2.5.1 明示的なデータ獲得

明示的なデータ獲得とはユーザに対して何らかのアイテム群についての評価の入力を求める手法である。暗黙的な手法と比較したとき、多くのデータは収集しづらいが、精度の高いデータを収集できるという特徴がある。これは、明示的な手法はユーザの手間を取ら

表 2.1: ユーザデータ獲得法の長所と短所

	明示的	暗黙的
データ量	少ない	多い
データの正確さ	正確	不正確
未評価と不支持の区別	明確	不明確
利用者の認知	認知	不認知

せるためにユーザが面倒に感じてしまうが、入力するデータには基本的に誤りはないものと考えられるためである。ユーザに入力を求める評価について、「好き」「嫌い」「どちらでもない」といった評価を求める手法と 1 (嫌い) ~ 5 (好き) までの値の何れかを選択といった評価を求める手法が考えられる。Cosley 氏 [7] によれば、ユーザはアイテムの評価について、より細かく評価が可能な手法を好む傾向がある。

明示的なデータ獲得には、嗜好データ以外にユーザのデモグラフィックを獲得することもある。デモグラフィックとはユーザの性別や年齢、住んでいる地域、職業などを指す情報のことである。これらは、推薦されるアイテムがデモグラフィックに依存する可能性がある場合に有効である。

2.5.2 暗黙的なデータ獲得

暗黙的なデータ獲得とは、ユーザの行動履歴をユーザデータとして獲得する手法である。行動履歴とは、推薦されるアイテムをクリックしたり、そのアイテムを購入したりといった行動の情報である。Bruke 氏 [8] は行動と嗜好度の関係を、嗜好の強さについて降順で表 2.2 のように挙げている。

2.6 推薦システムにおける推薦アルゴリズム

本節では推薦システムの中核部分である「推薦アルゴリズム」について述べる。推薦アルゴリズムは用いるデータによって協調フィルタリングと内容ベースフィルタリングに大別される。本節では協調フィルタリングを用いた一般的な手法と内容ベースフィルタリングを用いた一般的な手法について述べる。なお、以降で用いる言葉として「活動ユーザ」は推薦システムを使い、推薦されようとしているユーザを意味する。

2.6.1 協調フィルタリング

協調フィルタリングとは推薦システムに用いられるフィルタリング手法の 1 つである。フィルタリングとは、ある集合を篩に掛けてその一部を抽出することを言う。ここでは、

表 2.2: ユーザの行動とそのアイテムへの嗜好の強さ

	Action
1	Purchase
2	Assess
3	Repeated Use
4	Save/Print
5	Delete
6	Refer
7	Reply
8	Mark
9	Terminate Search
10	Examine/Read
11	Consider
12	Glimpse
13	Associate
14	Query

アイテム集合の中から推薦対象となるアイテムを抽出するための手法という意味で述べている。協調フィルタリングとは、活動ユーザと似たユーザデータをもつ他ユーザの嗜好情報から活動ユーザの嗜好するであろうアイテムをフィルタリングする手法である。協調フィルタリングは内容ベースフィルタリングと比較したとき、セレンディピティとドメイン知識の面で優れていると言われている。それぞれについて詳細を述べる。

(1) セレンディピティ

セレンディピティとは「何かを探索する際に、探索している対象とは別の価値あるものを発見する能力」のことを言う。協調フィルタリング手法はセレンディピティについて内容ベースフィルタリングと比較して優れている。それは、内容ベースフィルタリングはアイテムの属性に注目したフィルタリングを行うために、ユーザにとって意外と感じられるアイテムが推薦されにくく、一方で協調フィルタリングは他ユーザとの相関を用いてフィルタリングを行うため、他ユーザが活動ユーザの知らないアイテムを知っている場合にそのアイテムが活動ユーザに対する推薦対象と成りうるためである。

(2) ドメイン知識

推薦システムを構築する際に、推薦されるアイテムについての情報を収集する必要がある場合にはその分コストが高くなってしまふ。内容ベースフィルタリングはアイテムの属性や特徴を利用したフィルタリングなのでアイテムについての情報を事前に収集するが、協調フィルタリングはユーザ同士の相関を利用したフィルタリン

グなので、アイテムについての情報は一切必要なく、推薦システムの構築コストは低くなる。

協調フィルタリングはメモリベース法とモデルベース法の 2 つに大別することができる。それぞれについて詳細を述べる際に、数式を用いるため、数式に用いられる記号の意味を表 2.3 のように定義する。

表 2.3: 記号の定義

記号	定義
a	活動ユーザを意味する添字
k	評価予測値を計算されるアイテムを意味する添字
I	全アイテムの集合
U	全ユーザの集合
I_x	ユーザ x が評価したアイテムの集合
U_i	アイテム i を評価したユーザの集合
V	全ユーザの、全アイテムに対する評価値行列
m	総アイテム数
n	総ユーザ数
$v_{x,i}$	ユーザ x のアイテム i への評価値
$\hat{v}_{x,i}$	ユーザ x のアイテム i への評価値の予測値
\overline{v}_x^u	ユーザ x の評価の平均値
\overline{v}_i^l	アイテム i への評価の平均値

(1) メモリベース法

メモリベース法とは活動ユーザに対して推薦を行う段階になってからユーザデータを直接利用して推薦されるアイテムを算出する手法である。ユーザデータとは、式 2.1 のような行列の形式になっており、各要素はユーザのアイテムに対する評価を示す。

$$V = \begin{pmatrix} v_{1,1} & \cdots & v_{1,m} \\ \vdots & \ddots & \vdots \\ v_{n,1} & \cdots & v_{n,m} \end{pmatrix} \quad (2.1)$$

メモリベース法は更に、ユーザ間型とアイテム間型に分けることができる。ユーザ間型とは似ているユーザを探す手法であり、アイテム間型とは似ているアイテムを探す手法である。それぞれについて詳細を述べる。

(a) ユーザ間型

ユーザ間型のメモリベース法は、ユーザ間、すなわち式 2.2 の四角形で囲われる

単位で相関を求めることにより，活動ユーザと似たユーザを発見し，そのユーザが嗜好するアイテムを活動ユーザも同様に嗜好すると予測する手法である．

$$V = \begin{pmatrix} \cdots & \cdots & \cdots & \cdots \\ \boxed{v_{i,1}} & \boxed{v_{i,2}} & \cdots & \boxed{v_{i,m}} \\ \boxed{v_{i+1,1}} & \boxed{v_{i+1,2}} & \cdots & \boxed{v_{i+1,m}} \\ \cdots & \cdots & \cdots & \cdots \end{pmatrix} \quad (2.2)$$

ユーザ型のメモリベース法の例として Resnick 氏ら [9] の GroupLens という推薦システムに用いられる手法を式 2.3 及び式 2.4 に示す．

$$p_{x,y} = \frac{\sum_{i \in I_x \cap I_y} (v_{x,i} - \bar{v}_x^u) (v_{y,i} - \bar{v}_y^u)}{\sqrt{\sum_{i \in I_x \cap I_y} (v_{x,i} - \bar{v}_x^u)^2} \sqrt{\sum_{i \in I_x \cap I_y} (v_{y,i} - \bar{v}_y^u)^2}} \quad (2.3)$$

$$v'_{a,k} = \bar{v}_a^u + \frac{\sum_{\substack{z \in U \\ z \neq a}} p_{a,z} (v_{z,k} - \bar{v}_z^u)}{\sum_{\substack{z \in U \\ z \neq a}} |p_{a,z}|} \quad (2.4)$$

式 2.3 の $p_{x,y}$ はピアソンの積率相関係数と言われ，ユーザ x とユーザ y の相関を求めるために用いられる．式 2.4 は，式 2.3 で求めた相関係数を係数とした加重平均を用いて $v'_{u,j}$ を求める式である．なお，相関係数を求める際に総和を求めるアイテムの範囲として $I_x \cap I_y$ ではなく， I_x とし，ユーザ y の評価が ϕ のアイテムについて \bar{v}_y^u を用いる手法も一般的である．また，式 2.4 の総和の範囲について， $i \in U, i \neq u$ ではなく相関係数が一定の閾値以上の近傍ユーザに対してのみに限定した場合でも，精度を落とさずに推薦することが可能である．これにより，推薦に必要な計算を減らしてシステムを高速化させることが可能である．

(b) アイテム間型

アイテム間型のメモリベース法は，アイテム間，すなわち式 2.5 の四角形で囲われる単位で相関を求めることにより，多くの人から同じように評価されているアイテム同士は似ているアイテムと見なし，活動ユーザの好むアイテムと似たアイテムを発見し，そのアイテムを好むと予測する手法である．

$$V = \begin{pmatrix} \cdots & \boxed{v_{1,j}} & \boxed{v_{1,j+1}} & \cdots \\ \cdots & \boxed{v_{2,j}} & \boxed{v_{2,j+1}} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ \cdots & \boxed{v_{n,j}} & \boxed{v_{n,j+1}} & \cdots \end{pmatrix} \quad (2.5)$$

アイテム間型のメモリベース法の例として Sarwar 氏ら [10] や Ali 氏ら [11] において用いられている手法を式 2.7 に示す .

$$p'_{x,y} = \frac{\sum_{x \in U_i \cap U_j} (v_{x,i} - \bar{v}_i) (v_{x,j} - \bar{v}_j)}{\sqrt{\sum_{x \in U_i \cap U_j} (v_{x,i} - \bar{v}_i)^2} \sqrt{\sum_{x \in U_i \cap U_j} (v_{x,j} - \bar{v}_j)^2}} \quad (2.6)$$

$$v'_{a,k} = \frac{\sum_{j \in I_a} v_{a,j} p'_{j,k}}{\sum_{j \in I_a} |p'_{j,k}|} \quad (2.7)$$

ここで、式 2.6 と式 2.3 はそれぞれアイテム同士についての相関とユーザ同士についての相関を意味している . アイテム間型のメモリベース法は、似たアイテムを発見し推薦する手法なので、ユーザ間型のメモリベース法と比較してセレンディピティが劣ると考えられる .

(2) モデルベース法

モデルベース法とは活動ユーザに対して推薦を行う以前の段階で、ユーザデータからモデルを生成し、推薦時の計算量を減らして高速化することを目的とした手法である . ユーザの入力した評価値からユーザの嗜好傾向を得るにはいくつかの計算を経る必要がある . モデルとは、その嗜好傾向を得るために行う計算のうち、事前に行えるものを行って導いたものである . 活動ユーザに対する推薦時にはそのモデルを利用する . モデリング手法のうち、代表的な手法を以下に 2 つ挙げる .

(a) クラスタリングモデル

クラスタリングモデルとは、嗜好傾向の似ているユーザ同士を同じクラスタに分類する手法である . クラスタリング手法の 1 つとして凝集法 [12] がある . 凝集法の手順について解説する .

- (1) 各ユーザをそれぞれ単一要素のクラスタに分割する .
- (2) 最も近似しているクラスタ同士の要素 (ユーザ) を新しいクラスタに分類し、元々所属していたクラスタを削除する .
- (3) (2) を希望するクラスタ数になるまで繰り返す .
- (4) 新しいユーザを追加したとき、もっとも近似しているクラスタに分類する .

ここで、希望するクラスタ数は、多すぎた場合はコールドスタート問題に対して弱くなり、少なすぎた場合はクラスタとユーザとの一致度が低くなり適切な推薦を行うことができなくなるため、最適なクラスタ数は目的や状況によって変化する . また、クラスタリング手法の欠点として、ユーザはクラスタのうちどれか一つに分類されなくてはいけないため、どのクラスタとも一致しない嗜好傾向を持つユーザにとっては適さない . これを灰色の羊問題 [13] という .

(b) 関数モデル

関数モデルとは，ユーザの嗜好傾向を基に関数を導く手法で，独立変数としてアイテム i を与えると従属変数として評価予測値 $v'_{a,i}$ が得られるような関数を用いる手法である．すなわち，式 2.8 のような関数のことを言う．

$$v'_{a,i} = f(i) \quad (2.8)$$

式 2.8 に当てはまるような関数を統計的手法によって導くことを機械学習の分野では回帰と呼ぶ．式 2.8 のような関数 f を回帰で解く場合，教師信号として独立変数である $i (i \in I)$ と従属変数である $v_{x,i}$ を与える．ここで，アイテム i とはアイテム i の特徴量という意味であることに注意されたい．アイテム i の特徴量とは協調フィルタリングに於いてはユーザ同士の評価値から得られるアイテムの特徴量のことである．そして，与えられた独立変数と従属変数のセットの全ての組み合わせを満たすような関数 f のうち最も適切なものを求める．

協調フィルタリングには本項冒頭で述べた利点がある．しかし欠点として式 2.1 の行列 V にユーザのデータを集めたとき，ほとんどのセルが ϕ になってしまう場合が多いことが挙げられる．これはユーザが評価できるデータは既に利用したことのあるアイテムのみであるのに対して，全体のアイテム数が圧倒的に多いために発生する． ϕ の率が高くなるにつれ，精度の高い推薦は難しくなる．

2.6.2 内容ベースフィルタリング

内容ベースフィルタリングとは，アイテムの属性情報や特徴量を用いて活動ユーザの嗜好するであろうアイテムをフィルタリングする手法である．内容ベースフィルタリングは協調フィルタリングと比較したとき，コールドスタート問題とカバレッジの面で優れていると言われている．それぞれについて詳細を述べる．

(1) コールドスタート問題

コールドスタート問題とは，システムを運用させ始めたばかりの段階でユーザ数が少ないとき，協調フィルタリングの場合はユーザ同士の嗜好情報を用いたフィルタリングを行うためにユーザ数自体が少なければ推薦精度が下がってしまう．一方で内容ベースフィルタリングはアイテムの属性情報や特徴量を用いるため，ユーザ数の多寡に関係なく一定の精度で推薦することができるため，コールドスタート問題に対して強い．

(2) カバレッジ

カバレッジとは推薦されるアイテム群が，推薦されるべきアイテムをカバーしている率を表す．協調フィルタリングは未評価のアイテムについて活動ユーザに対して推薦することができない．一方で内容ベースフィルタリングは全アイテムについて推薦行うか否かの判定をすることができるため，カバレッジに優れる．

内容ベースフィルタリングによってフィルタリングする手法は獲得できるアイテムの属性・特徴量に大きく依存するため、代表的な手法は確立されていないが、一般的には機械学習を用いる手法と人手によってルールを決める手法がある。機械学習を用いる手法と人手のルールによる手法で双方共に推薦するアイテムの内容によって手法は大きく異なる。

2.7 推薦システムにおける推薦アイテム提示手法

本節では 2.6 節で述べた手法を用いて出力された結果を、活動ユーザに対して提示する手法について述べる。Swearingen 氏の論文 [14] では、推薦されるアイテムの提示について 7 つの点に注目している。それらのうち、特に重要な 2 点について以下にまとめる。

(1) 推薦されるアイテムの情報

図 2.3 (Swearingen 氏らの論文 [14] より引用) にあるように、ユーザは推薦されるアイテムと共にそのアイテムに関する情報をより多く提示した時にしなかった時と比べて利便性が高いと答える傾向がある。Swearingen 氏らはこれ対し、推薦アイテム

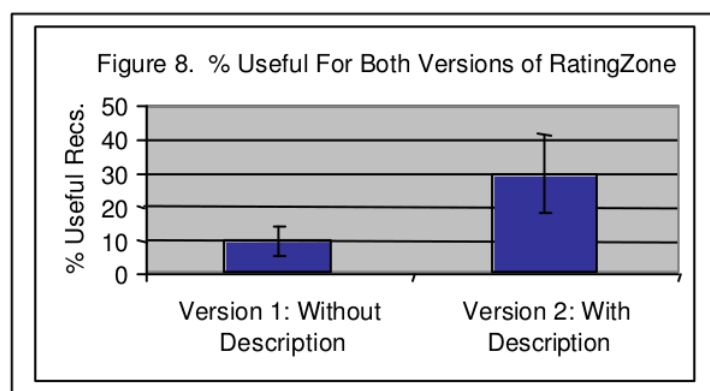


図 2.3: アイテム情報付加時と非付加時の比較

を提示する際には、そのアイテムの情報や、情報にアクセスするための手法を明確にすることを提案している。また、それぞれのアイテムについてのフォーラムを用意することで推薦システムの効果を向上させることも推奨している。

(2) システムの透過性

ユーザは推薦されるアイテムがどのような経緯で推薦されているのかを知ることができる場合に、知ることができない場合と比べて良い推薦であると答える傾向がある。これは特に内容ベースフィルタリングの際に、どの属性・特徴量が影響して推薦されたかを提示する手法が適切である。それらの情報を提示した場合としなかった場合のユーザの評価を図 2.4 (Swearingen 氏らの論文 [14] より引用) に示す。

Swearingen 氏らはこれ対し、ユーザは推薦結果が提示された際に、自身の評価したアイテムとの関係が無いと混乱してしまうため、少なくともいくつかの推薦結果アイテムはユーザの評価との関係が明快であると良いと述べている。

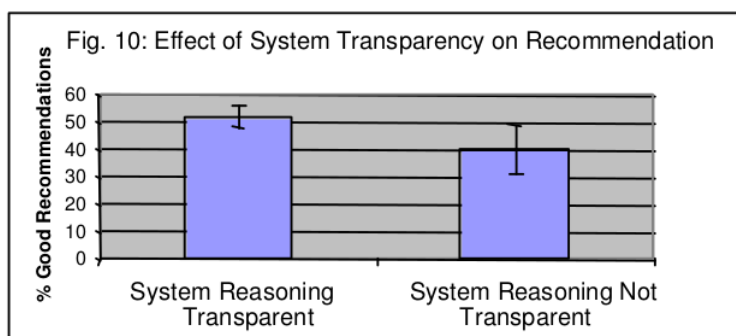


図 2.4: システム透過性の有無の比較

2.8 推薦システムの評価手法

推薦システムの評価手法には様々な手法があり，推薦システムの評価手法についての研究も存在する．本節ではその中の代表的な評価手法について記述する．

2.8.1 履歴適合度

履歴適合度評価とは，ユーザの利用履歴データからある時系列点以降をマスクし，マスクされていない利用履歴データを基に推薦を行った結果，マスクされる前の実際の利用履歴データとどの程度適合するかを指標とする，Cyril 氏ら [15] の提案した手法である．本手法を式 2.9，式 2.10 及び 2.11 に示す．

$$P = \frac{S \cap T}{S} \quad (2.9)$$

$$R = \frac{S \cap T}{T} \quad (2.10)$$

$$F = \frac{2PR}{P + R} \quad (2.11)$$

ここで，各記号は表 2.4 のように定義する．また，一般的に式 2.9 の P を「精度 (Precision)」，

表 2.4: 記号の定義

記号	定義
S	マスクしたデータの正解集合
T	マスクしたデータの予測集合

式 2.10 の R を「再現率 (Recall)」，式 2.11 の F を「f-measure」と呼ぶ．以下に本手法に用いられる「精度」と「再現率」，及び「f-measure」について詳しく述べる．

(1) 精度

精度とは，予測集合 S のうち正解集合 T の要素に含まれている割合のことを言う．一般的に，推薦されるアイテム数を増やすと精度は落ちる．

(2) 再現率

再現率とは，正解集合 T のうち予測集合 S の要素に含まれている割合のことを言う．一般的に，推薦されるアイテム数を減らすと，再現率は落ちる．

(3) f-measure

精度と再現率は単純に一方の数値を上げようとするともう片方が下がるといった関係にあるが，精度が高く（すなわち余計なアイテムは推薦されない）尚且つ再現率も高い（すなわち推薦されるべきアイテムは推薦する）推薦システムを高く評価するための指標の一つが f-measure である．f-measure は精度と再現率の調和平均を求めることで，精度と再現率の両立された推薦システムを高く評価することが可能である．

2.8.2 ユーザ評価適合度

ユーザ評価適合度による評価とは，ユーザのアイテムに対する評価値の一部をマスクし，マスクされていないユーザ評価データを基に推薦を行った結果，マスクされる前の実際の評価値とどの程度近似するかを指標とする手法である．これについては Hyndman 氏ら [16] の論文に詳しく記載されている．本評価手法の代表的な手法として MAE (Mean Absolute Error) と $RMSE$ (Root Mean Squared Error) について述べる．

(1) MAE (Mean Absolute Error)

MAE とは実際の値に対して予測値がどの程度外れているかを示す値である．その式 2.12 に示す．

$$MAE = \frac{1}{|X|} \sum_{i \in X} |v'_i - v_i| \quad (2.12)$$

ここで，各記号は表 2.5 のように定義する． MAE は各評価予測値と実際の評価値

表 2.5: 記号の定義

記号	定義
X	マスクしたデータの集合
v_i	データ i についての実際の評価値
v'_i	データ i についての評価予測値

との差の絶対値の平均である．したがって， MAE の値は小さければ小さいほど予測の精度は高く，推薦システムとして高く評価される．

(1) *RMSE*(Root Mean Squared Error)

RMSE とは, *MAE* と同様に, 実際の値に対して予測値がどの程度外れているかを示す値である. その式 2.13 に示す.

$$RMSE = \sqrt{\frac{1}{|X|} \sum_{i \in X} (v'_i - v_i)^2} \quad (2.13)$$

記号の意味は表 2.5 に準ずる. *MAE* は差の絶対値の平均であるのに対して, *RMSE* は差の 2 乗の平均の平方根である. したがって, *RMSE* は *MAE* と比較したとき, 実際の値からの外れ方の大きい値は評価に大きく影響する.

2.8.3 順位評価適合度

順位評価適合度評価とは, アイテムに対するユーザ評価が順位形式である場合に用いられることがある. 例として, 神蔦氏 [17] の論文がある. 本手法では式 2.3 にあるピアソン積率相関係数を用いる. ユーザ評価を順位形式にする手法では, 予測されるアイテムも順位形式で提示されるため, ユーザの評価した順位と予測された順位とのピアソン積率相関係数を求め, その相関係数が高ければ高いほど推薦システムを高く評価する.

2.8.4 セレンディピティ

セレンディピティは一般的に, 定量的に測定することは難しいと言われているが, 推薦システムの評価基準としてセレンディピティを用いた例として小出氏の研究 [18] がある. 小出氏は表 2.6 のようにユーザの心理状態の階層モデルを定義し, システムの利用前と利用後でどのように心理状態が遷移したかを測定している.

表 2.6: 心理状態階層モデルの定義

	状態	説明
Action	行動段階	選択・購入したことがある
Desire	欲求段階	選択・購入したいと思っている
Search	強い感情段階	興味があり, 能動的に知ろうとしている
Interest	弱い感情段階	興味はあるが, まだ何もしていない
Attention	認知段階	存在・名前は知っている
Unknown	認知以前	存在自体を知らない

2.9 本論文の着眼点

2.2.2 項で述べた，多すぎる選択肢がユーザに対して負の影響を及ぼすという実験結果を受けて，履修選択の手法としてシラバス一覧から各科目の情報を参照するという手法が必ずしも最適な履修選択の手法ではないことがわかった．そのため，一般的にはシラバスの検索システムを用いて選択肢を絞って履修選択をする．本論文では，2.1.2 項で述べたシラバス検索システムの問題にあるように，選択肢を絞る手法として検索システムのみを用いることが必ずしも最適ではないことに着目している．このことから，本論文では，履修選択を支援するシステムを開発し，学生がより少ない手間で満足度の高い履修選択を行うことができるようになるかについて検証する．

2.10 まとめ

本章では，大学などの教育機関において一般的に履修選択に用いられるシラバスの役割と問題点を述べ，シラバスの選択肢が多すぎるという問題点から，多すぎる選択肢が一般的にどのような影響を及ぼすかの実験を挙げた．実験からは多すぎる選択肢の影響として3つの影響，すなわち選択の先延ばし，不適切な選択，満足度の減少が示された．そこで，一部のシラバスを抽出する手法として現在用いられているシラバス検索システムを挙げ，その検索システムを用いた際の問題点として3つの問題点，すなわち分野知識の必要性，学生評価の不活用，セレンディピティの無さを挙げた．次に，本論文において提案する推薦システムの3つのフェーズ，すなわちユーザプロフィール獲得フェーズ，嗜好の予測フェーズ，推薦アイテム提示フェーズについて，それぞれのフェーズにおける手法の紹介を行った．最後にその推薦システムを評価する手法について述べた．

第3章 関連研究

本章では，履修科目選択の支援における既存研究と，本論文で提案する推薦システムについての既存研究について述べる．

3.1 履修科目選択支援に関する研究

本節では，履修科目選択の支援について述べている既存研究を分析，記述する．

3.1.1 大学教養教育における科目選択支援

森氏らの研究 [19] では，科目間の類似度を可視化することにより，学生が突き当たる2つの困難を軽減するための支援法を提案している．以下にその2つの困難となる点を示す．

- (1) どのような科目があるのかわからない．
- (2) 提供されている科目間にどのような関係があるのかわからない．

科目の類似度を算出する方法として，各科目に対して tfidf 値を求め，科目ごとに単語の tfidf 値を並べたベクトルを作り，ベクトル空間モデルにより余弦値を基にした類似度を，科目間の類似度とした．tfidf 値とは科目 d における単語 t の出現頻度 $tf(d, t)$ と単語 t を含む科目数 $df(t)$ を用いて，式 3.1 で計算される．

$$tfidf(d, t) = \frac{tf(d, t)}{N} * \left(\log \left(\frac{M}{df(t)} \right) + 1 \right) \quad (3.1)$$

ここで N は単語の総数， M は科目の総数を表す．単語 t は，シラバス中の「科目名」，「授業のテーマと目的」，「授業計画と内容」の中にある名詞を意味する．

科目間の類似度の可視化画面を図 3.1 に引用する．氏らの論文では，上記の方法で算出した類似度を可視化することにより，検索されにくい科目の検出を可能にし，学生が自分の学習計画を主体的に考えながら時間割を作成するまでの科目選択を支援できることを示している．しかし，シラバスの記述量が科目によって変わるため，実際には類似している科目が類似科目として表示されない場合があることや，独立性の高い科目について類似する科目が表示されないこと，類似していない科目でも学生にとって適している科目について考慮されていないといった疑問が残る．

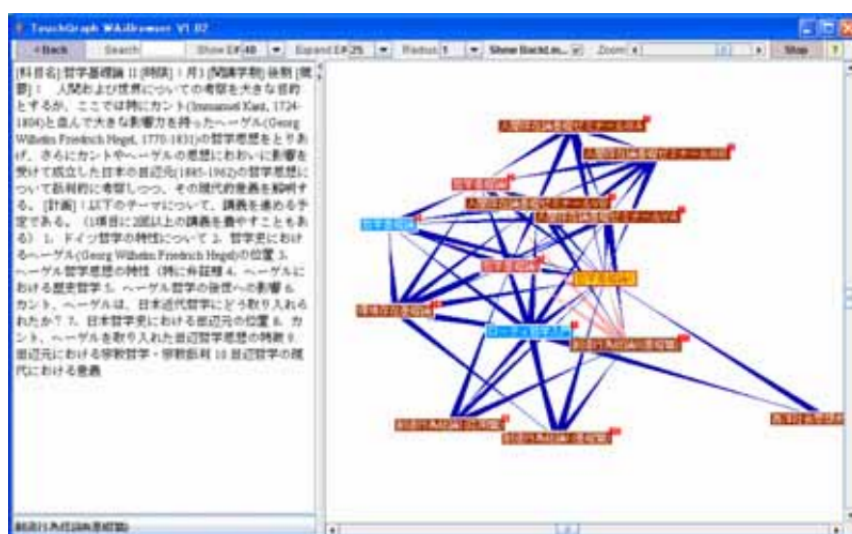


図 3.1: 類似度の可視化

3.1.2 ダイナミックシラバスの開発

土肥氏らの研究 [20] では、学生が将来の目標を掲げるために必要な科目を系統的に表示し、無理のない順序でそれらを学習できるように、時間割を作成するための複雑な作業を支援するシステムとしてダイナミックシラバスを開発した。ダイナミックシラバスには主に以下の 9 つの機能を実装している。

- (1) 履修モデルの提示
- (2) GPA と履修制限の判定
- (3) 事前履修条件の提示および判定
- (4) 科目の一覧と科目配当表の提示
- (5) 科目の内容（シラバス）の提示
- (6) 授業内容に関する質問と議論
- (7) レポート関連
- (8) 卒業条件と卒業申告
- (9) 履修登録

また、ダイナミックシラバスの利用は以下の手順に従って行う。

- (1) 個人別カリキュラム設定
個人別カリキュラム設定画面では、科目が一覧になって表示され、各科目について

「科目名」「シラバス」「事前履修条件」が設定され、各科目についての概要や履修条件などを簡単に閲覧することができる仕組みになっている。

(2) 時間割作成

科目の一覧から興味のある科目を選択し終えたら時間割画面に移動し、それぞれの時限に何を受講するかを選択し、時間割を完成させていく。

(3) 時間割決定

時間割に時限による重複や事前履修条件による矛盾がなくなると、時間割を確定させることができる。

氏らの論文では、授業の最終日にアンケートを行った結果、ダイナミックシラバスを用いることで約 8 割以上の学生から役立ったという回答を得ることができたとして、ダイナミックシラバスの有効性を示している。しかし、ダイナミックシラバスは科目を選択する際に多くの選択肢から選ぶことになり、2.2.2 項において述べた影響を受けざるを得ない。そのため、少ない手間で満足度の高い履修選択を実現することができるかについては疑問が残る。

3.2 推薦システムに関する研究

本節では、推薦システムを活用している既存研究を分析、記述する。推薦システムは 2.4 節で述べたように、3 つのフェーズからなる。それぞれのフェーズには推薦されるアイテムや状況に合ったフェーズ毎の手法を選択する。本節に挙げる既存研究については、特に上記の点に着目して述べる。

3.2.1 内容に基づく音楽データの探索・推薦システム

竹川氏らの研究 [21] では、自分の嗜好に合った音楽を探す労力を軽減させる手段として、MIDI データを対象とした音楽探索・推薦システムを提案し、その有効性を示した。音楽探索・推薦システムは以下の手順に従って行う。

(1) 音楽の試聴・評価

ユーザプロフィール獲得フェーズにあたるこのフェーズでは、ユーザがリストから選んだいくつかの音楽を視聴し、それぞれを「好き」「嫌い」「どちらでもない」の何れかから選び評価する。評価を基に予め音楽から抽出した 20 種類の特徴量を用いて決定木を作成する。特徴量を表 3.1 に示す。ここで作成された決定木はユーザから閲覧及び編集することが可能になっている。作成された決定木の例を図 3.2 に引用する。

(2) 推薦手法

ユーザプロフィール獲得フェーズで作成した決定木を基にユーザの好みに当てはまる音楽を推薦アイテムとする。

表 3.1: 音楽の特徴量

曲全体	1. 拍子, 2. 調性, 3. 平均テンポ, 4. リズム, 5. キー, 6. メジャ - コードの割合 7. マイナーコードの割合 8. sus4 コードの割合
CH ごと	9. 音色 (メロディ) 10. 平均音高差 (メロディ, ベース, コード) 11. 平均音長 (メロディ, ベース, コード, ドラム) 12. 平均音長差 (メロディ, ベース, コード, ドラム)

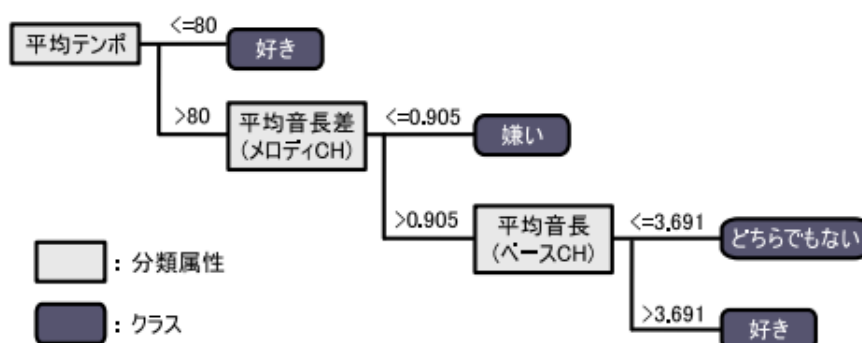


図 3.2: 作成された決定木

(3) 提示方法

推薦アイテムとなった音楽を画面に列挙する。

(4) ユーザプロファイルの調整

ユーザは自らの評価を基に作成された決定木を編集することができ、編集を施した決定木を用いて再び推薦することができる。決定木の編集を行う際には、決定木の作成に用いた音楽の特徴量を表示し、編集を行い易くしている。

氏らの論文では、ユーザプロファイルの可視化を行いユーザが直接決定木を編集することを可能にしユーザに編集させることで、期待する音楽を推薦し、嗜好に合った音楽を探す労力を軽減させることに成功したとしている。

決定木を用いた推薦は、推薦されるアイテムに対して「推薦する」か「推薦しない」かなどの離散的な値のみ出力として得られることや、決定木を作成する際に与えられるデータから、学習すべきでないデータまで学習してしまう過学習が発生しうることが特徴として挙げられる。決定木を用いた履修科目の推薦は、推薦する判定を受ける科目数が不定であり、推薦される科目間に上下が存在しないため、科目間の推薦度合いの数値を必要とするシステムには適さないと考えられる。また、プロフィールデータのユーザによる直接編集は決定木のアルゴリズムがユーザにとって理解しやすいために可能であったため、より

複雑なアルゴリズムを用いる場合にはユーザによるプロフィールデータの直接的な編集は難しいと考えられる。

3.2.2 協調フィルタリングとコンテンツ分析を利用した観光地推薦手法

樽井氏らの研究 [22] では、協調フィルタリングの利点とコンテンツ分析法の利点を組み合わせた観光地推薦手法の基本的なアイデアについて述べている。協調フィルタリングの利点とコンテンツ分析法の利点を組み合わせた手法では、観光地の特徴ベクトルとユーザの履歴行列からユーザ特徴ベクトルを計算する。ユーザ特徴ベクトルとはユーザの嗜好情報を表すベクトルである。ユーザ特徴ベクトルを用いてユーザ同士の類似度を計算し、類似度の高いユーザの履歴から活動ユーザのまだ訪れていない観光地を推薦する。氏らの提案したシステムの推薦アルゴリズムを以下に詳しく述べる。

(1) 観光地情報の入力

観光情報提供者が観光地推薦システムに観光地情報を入力する。観光地情報とは観光地の基本的な情報に加えて観光地の特徴量も指す。氏らは観光地の特徴量として表 3.2 の 16 の特徴量を用意した。図 3.3 ([22] より引用) に観光地情報の入力フォー

表 3.2: 観光地の特徴量

	特徴量
1	自然
2	保養(地)
3	海水浴
4	レジャー
5	歴史遺産
6	文化(施設)
7	スポーツ
8	農村風景
9	町並
10	温泉
11	山寺(社寺)
12	建造物
13	避暑地
14	別荘地
15	特産
16	グルメ(味覚)

ムを示す。観光情報提供者は入力フォームを埋めて送信する。送信された情報は表

図 3.3: 観光地情報入力フォーム

3.3 (例として特徴量の数を 6 にしているが実際には 16) のように行列の形に整理されて保存される。

表 3.3: 観光地特徴行列

	自然 e_{i1}	保養 e_{i2}	レジャー e_{i3}	文化 e_{i4}	歴史遺産 e_{i5}	グルメ e_{i6}
草津 E_1	1	1	0	0	0	0
渋川 E_2	1	1	0	0	0	0
佐野 E_3	1	0	0	0	1	0
桐生 E_4	1	0	0	1	1	0
山の内 E_5	1	1	1	0	0	0

(2) ユーザ履歴入力

ユーザは今まで訪れたことのある観光地を入力する。システムは観光地入力フォームを用意している。観光地入力フォームはユーザが膨大な数の観光地の中から手探りで探さずに済ますために地方によって分類され表示される。

(3) ユーザ特徴ベクトルの計算

ユーザ特徴ベクトルの計算はユーザの履歴から各ベクトルの構成要素割合を計算して求める．ユーザ t に対するユーザ特徴ベクトルを計算する式を式 3.2，及び式 3.3 に示す．

$$F_t = \{ f'_{t1} = \frac{f_{t1}}{s_t} \quad f'_{t2} = \frac{f_{t2}}{s_t} \quad \dots \quad f'_{tn} = \frac{f_{tn}}{s_t} \} \quad (3.2)$$

$$s_t = \sum_{j=1}^n f_{tj} \quad (3.3)$$

ここで n は観光地特徴量の要素数を表す．ユーザ特徴行列の例を表 3.4 に示す．ユー

表 3.4: ユーザ特徴行列

	自然 f'_{t1}	保養 f'_{t2}	レジャー f'_{t3}	文化 f'_{t4}	歴史遺産 f'_{t5}	グルメ f'_{t6}
A さん F_1	0.43	0.29	0.00	0.14	0.14	0.00
B さん F_2	0.40	0.20	0.10	0.10	0.20	0.00
C さん F_3	0.44	0.33	0.11	0.00	0.11	0.00
D さん F_4	0.40	0.20	0.10	0.10	0.20	0.00
E さん F_5	0.42	0.25	0.08	0.08	0.17	0.00

ザ特徴行列はユーザ同士の類似度の計算に用いるため，各ベクトルの構成割合を示す数値にすることで正規化されている．このことは各ユーザの特徴行列が合計で 1 になっていることからわかる．

(4) ユーザ同士の類似度を計算

ユーザ同士の類似度は第 2 章で述べたピアソン積率相関係数を用いる．

(5) 活動ユーザに対して観光地を推薦

観光地の推薦はユーザ同士の類似度の計算によって高い類似度を示したユーザの訪れた観光地のうち，活動ユーザの訪れていない観光地を推薦対象とする手法を用いている．推薦結果の表示は図 3.4 ([22] より引用) のような画面になる．

一般的な協調フィルタリングではユーザの各アイテムに対する評価を基にユーザ同士の類似度を計算するが，氏らの提案したアイデアでは観光地の特徴ベクトルを事前に設定することに加え，訪れたことのある観光地を良いと評価したと暗黙的に認識することによりユーザの各観光地についての評価を入力する手間を省いた．氏らの提案したシステムは実験には至っていないが，観光地特徴量の検討，より正確な推薦のためのユーザの旅行満足度の収集の検討，推薦アルゴリズムの妥当性の検証などを今後の課題として挙げている．氏らの提案する推薦手法を本研究の履修科目推薦に適用する場合，各科目のデータ入力が必要となるため，開発コストが増大する．また，ユーザ同士の類似度計算にユーザ特徴ベクトルを用いる手法の妥当性が検証されていないため，精度の高い推薦を行うことが可能か不明である．

■観光地推薦リスト

※ 推薦一覧表を表示する地方名を選択して下さい。

全国 北海道 東北地方
 関東地方 中部地方 近畿地方
 中国地方 四国地方 九州地方
 沖縄地方

推薦地一覧表			
推薦NO	地方名	観光地名	コメント
1	北海道	釧路・臼田	アロハン海野、曹達温泉
2	北海道	臼田	ワイン畑など
2	北海道	臼田	高山植物(レンゲスユキキノ)自生地など

図 3.4: 観光地推薦画面

3.3 まとめ

本章では履修科目選択の支援における既存研究と、推薦システムについての既存研究の2つの種類の研究について紹介し、その手法について述べた。履修科目の選択を支援する手法の既存研究は、何れも選択肢の豊富さにつきまとう問題については考慮されておらず、学生の状態も自分の目標とする事柄が明確であることを前提としており、アプローチとしてはシラバス検索システムの利便性を向上させるという方向性である。推薦システムについての既存研究は、2.4節で述べた3つのフェーズについて着目し、様々な推薦アイテム群に対してどのような手法を用いることが有効であるか分析した。本論文では、学生が学びたい対象についてまだ漠然とした状態においても、選択肢の多さに圧倒されずに、適切な履修科目の選択を可能とさせる手法として、履修科目の選択に推薦システムを用いる手法を提案する。

第4章 推薦システムによる履修科目選択支援の提案

本論文では、履修科目選択の手段として推薦システムを利用することで、少ない手間による満足度の高い履修科目選択を実現する手法を提案する。本章ではまず、提案手法について述べ、既存研究との違いや、推薦システムを使用する利点を述べる。そして、本提案手法によってどのようにユーザが少ない手間で満足度の高い履修科目選択が可能となるかについて、システムに不可欠な機能要件を挙げ、その具体的な達成手法にを述べる。

4.1 提案手法

推薦システムを用いることで、履修科目選択の支援をする手法を提案する。本節では、提案する手法と既存研究の違いについて述べ、その後履修科目選択に推薦システムを利用する利点について述べる。

4.1.1 既存研究との違い

本研究と既存の履修科目選択支援に関する研究の違いは、推薦システムを用いている点である。第3章に挙げた既存研究のうち、履修科目選択支援に関する研究では、シラバスの検索システムを機能拡張するものが主であった。例えば、森氏らの研究[19]では科目間の類似度を可視化した。これは内容ベースフィルタリングによる推薦と見なすこともできるが、システムの利用には検索クエリを入力することが求められているため、基本的には検索システムの機能拡張されたものである。

4.1.2 推薦システムを使用する利点

本研究で実装するシステムは推薦システムであるため、検索システムを用いる際に生じる問題点を避けることができる。検索システムの問題点は2.3.1項で述べたとおり、分野知識が必要なこと、他学生の評価を参考にできないこと、セレンディピティに欠けることが挙げられる。推薦システムを用いることは検索システムと比較して分野知識を必要とせずセレンディピティに優れることが利点として挙げられる。本研究では、目標がまだ漠然として、どの科目を履修したらよいか判断できない学生が主たる対象ユーザであるため、検索クエリを考えるための知識を必要としないこと、すなわち分野知識を必要としな

いことが重要である。また、既に自分の知っている分野についての科目ばかりが推薦されるのでは、検索システムを用いた場合との差別化ができないため、セレンディピティの優れたシステムであることも重要である。

4.1.3 ノーフリーランチ定理

本論文で提案する推薦システムには、2.4 節以降で述べた汎用的な手法をそのままは適用しない。汎用的な手法をそのまま適用しない理由としてノーフリーランチ定理が挙げられる。以下にノーフリーランチ定理について述べる。

- ノーフリーランチ定理

ノーフリーランチ定理は問題の集合に対して、どのアルゴリズムも平均的に完全に同じ性能であることを示している。すなわち、あらゆる問題に対して平均的に他よりも高い性能を持ったアルゴリズムは存在しない。

ノーフリーランチ定理については Wolpert 氏ら [23] の論文が詳しい。推薦システムに用いられる手法は、推薦されるアイテム、システムを利用するユーザ、利用可能なデータなどの様々なパラメータによって変わる。既存の汎用性の高い手法をそのまま適用したとき、それは履修科目選択の支援として決して優れてはいないため、本論文では履修科目選択の支援に特化させた推薦手法を用いる。

4.2 要求事項

履修科目選択の支援システムとして推薦システムを提案する上で、いくつかの要求事項がある。本節ではその要求事項について詳しく述べる。

4.2.1 各科目の推薦度合いの明快さ

豊富すぎる選択肢がユーザにどのように影響するかについて 2.2.2 項で述べた。そのため、一般的に推薦システムがユーザにとって多すぎる選択肢を提示することは適切とは言えない。しかし、豊富な選択肢を与えた際に現出する問題点は与えた選択肢が全て並列に与えられた場合について述べられている。したがって、与えた選択肢ごとに推薦度合いを表示すればこれらの問題は避けることができると考えられる。本研究では科目の推薦結果を提示する際に、タグクラウドを用いる。タグクラウドとはサービスのコンテンツに関する短いフレーズをタグとし、そのタグを利用頻度の高さによってフォントのサイズなどを変えたものを一覧表示したものである [24]。ここでは、推薦結果に表示される科目名をタグとし、その推薦度合いが大きいほどタグのフォントサイズを大きくすることによって推薦度合いが明快になり、多くの推薦結果が表示されている場合にも 2.2.2 項で述べた影響は出にくいと考えられ、尚且つ科目名から興味が湧いた時には比較的推薦度合いの高くない科目でもその情報を見ることが残すことができる。

4.2.2 学生のニーズに合った推薦

ユーザが検索システムの代わりに推薦システムを利用するとき、推薦結果としてユーザのニーズに沿わない科目に対して高い推薦度合いを提示した場合、履修科目選択後の授業の際にユーザが自身には合わない科目を選択してしまったことに対して後悔してしまう問題が起きる。そのため、ユーザのニーズに合った推薦結果を提示する必要がある。本システムでは特にこの項目について評価を行っている。すなわち、ユーザが高く評価したアイテムや低く評価したアイテムを推薦システムもその他のユーザデータから同じように高く・低く評価することのできる推薦システムを実装する。

4.3 まとめ

本章では、少ない手間による満足度の高い履修科目選択を実現するための提案手法について述べ、既存の研究との違いと、提案する手法として、推薦システムを用いることの利点を述べた。その後、本提案手法についての要求事項を挙げた。

第5章 実装

5.1 実装環境

本システムの実装環境を表 5.1 に示す。

表 5.1: 実装環境

要素	環境
OS	Ubuntu 12.04[25]
Web サーバ	Apache 2.2.22[26]
言語	PHP 5.3.5[27] JavaScript[28] HTML[29]
実装ソフトウェア	PhpStorm 5.0.3[30]
ライブラリ等	PHP Simple HTML DOM Parser 1.5[31] Bootstrap 2.2.2[32] jQuery 1.8.3[33] jqCloud 1.0.2[34]
データベース	MySQL 5.1.63[35]
ユーザインターフェース	Web[36]
対応ブラウザ	Google Chrome[37]

5.2 実装したシステムの構成要素

本節では、実装したシステムの構成について詳しく述べる。本システムは認証部分、評価入力部分、推薦部分、アイテム提示部分の4つの部分からなる。それらを詳しく解説する際により明瞭に述べるために、本システムの構成要素を表 5.2 にまとめた。

表 5.2: システムの構成要素

ウェブインターフェース	データ処理	データベース
認証 評価入力 アイテム提示	認証処理 評価受付処理 評価予測	認証用テーブル 科目情報テーブル 履修済み科目評価テーブル 擬似科目評価テーブル 特徴量定義テーブル

5.2.1 認証部分

本項ではシステムの認証部分について述べる。本システムは各ユーザの評価を基に推薦を行うシステムであるため、システムを利用しているユーザが誰であるのかを識別する必要がある。本システムは慶應義塾大学湘南藤沢キャンパス（以下 SFC）の学生を対象として実験評価を行うため、SFC で用いられている学生用の ID（以下ログイン名）を学生の識別に用いることにした。ログイン名は部分的に書き加えることで本人のメールアドレスにもなるため、そのメールアドレスを用いて認証を行う。まず、ユーザはシステムにログイン名を入力する。すると、システムはトークンを発行し、そのトークンを付したメールをログイン名に部分的に書き加えたメールアドレス宛に送り、同時にシステムのデータベースの認証用テーブルにログイン名とトークンのセットを追加する。ここで認証用テーブルの構造を表 5.3 に示す。ユーザが送られてきたメールのトークンのついた URL を開

表 5.3: 認証用テーブル

カラム	データタイプ	デフォルト値
ログイン名	varchar(16)	NULL
トークン	varchar(32)	NULL
登録日時	timestamp	CURRENT_TIMESTAMP

くことで、システムはユーザの開いた URL のトークンとデータベースに登録されているトークンを照合し、一致するログイン名を現在利用しているユーザと判断し、セッションに登録する。なお、認証用テーブルの登録日時カラムは一定時間経過した際にテーブルから削除するために用意した。本システムにログインが完了するとシステムのメイン画面に遷移する。図 5.1 にログイン後のメイン画面を示す。メイン画面にはシステムの簡単な利用方法と、履修履歴アップロード用のフォーム、評価入力ページ、推薦提示ページなどのハイパーリンクを表示した。

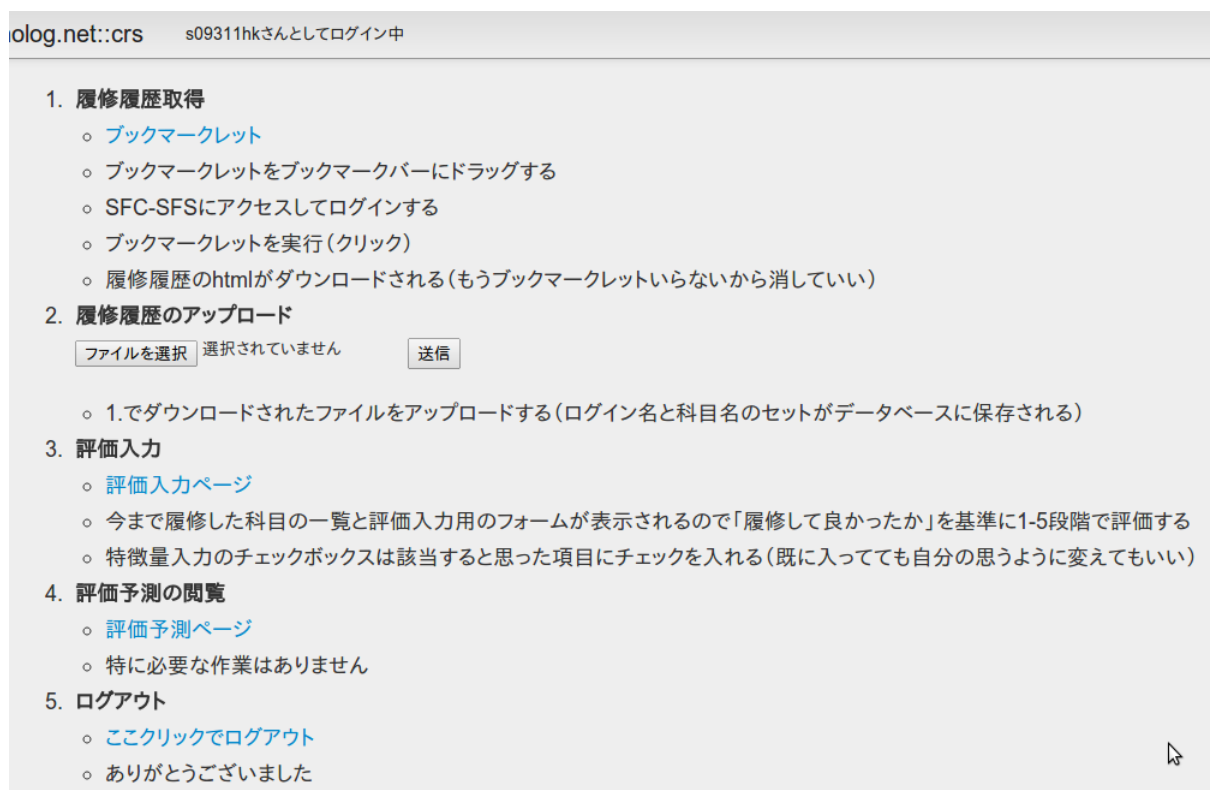


図 5.1: ログイン後の画面

5.2.2 評価入力部分

本項では推薦をする際に用いるユーザデータを入力する部分について述べる。ログインしたユーザはまず自身の今まで履修してきた科目の登録を行う。本システムではユーザに SFC の提供する学生・教員・職員のためのコミュニケーション支援システム（以降 SFC-SFS）[38] にログインしてもらい、本システムの提供するブックマークレットを用いて SFC-SFS から履修履歴のファイルをダウンロードし、本システムにそのファイルをアップロードしてもらうことで、少ない手間です履修履歴の登録を行えるようにした。ここで、SFC-SFS のパスワードを本システムに入力してしまえばシステム側から SFC-SFS にログインすることでユーザにはパスワード入力の手間のみで履修履歴の登録を行うことが可能だが、セキュリティの観点からパスワードは入力させず、本人にアップロードしてもらう形式とした。履修履歴はアップロードされたファイルを解析し、データベースにまだ登録されていない履歴情報があればデータベースに保存する。ここで本システムにアップロードした履修履歴を保存するデータベースのテーブルを表 5.4 に示す。履修履歴を登録したら各科目に対する評価を行う。この評価は「その科目を履修して良かったか」を基準に 1～5 の 5 段階で評価してもらう。

本システムは各科目の特徴量をユーザに入力してもらう手法を用いた。この手法を用いることでシステム開発者が各科目の特徴量を取得・入力する必要がなくなり、開発コストを

表 5.4: 履修済み科目評価テーブル

カラム	データタイプ	デフォルト値
ログイン名	varchar(16)	NULL
科目番号	varchar(6)	NULL
評価値	int(11)	3
評価済み	int(11)	0

減らすことができる．特徴量の種類を定義するデータベースのテーブルを表 5.5 に示す．本システムでは特徴量として本学の授業形態を試験的に用いた．すなわち，“実習”，“発

表 5.5: 特徴量定義テーブル

カラム	データタイプ	デフォルト値
ID	varchar(4)	NULL
意味	varchar(32)	NULL

表”，“試験”，“グループワーク”，“ゲストスピーカー”の 5 つを用いた．最適な特徴量の種類の設定は，各大学によって用意されている科目や，在学する学生によっても変化すると考えられる．ユーザは，それぞれの科目がどのような授業形態であったかを入力する．ただし，科目毎の特徴量はユーザ間で共有されるため，活動ユーザの前に同じ科目の特徴量を設定したユーザがいた場合，それが反映されるため，再度設定する必要はない．ここで，評価入力画面を図 5.2 に示す．入力用フォームに入力され，送信されたデータはデー

The screenshot shows a web interface for entering evaluation data. It consists of a list of subjects on the left and a set of radio buttons for each subject to select an evaluation type. The subjects listed are: ゲームプログラミング, システムプログラミング, 記号処理プログラミング, データベース概論, データベース構築法, Webテキスト処理法, 数学と論理, and 情報数学1. The evaluation options are: 実習 (checked for システムプログラミング, データベース概論, データベース構築法), 発表 (checked for 記号処理プログラミング, データベース構築法), 試験 (checked for システムプログラミング, Webテキスト処理法), グループワーク (checked for 情報数学1), and ゲストスピーカー (unchecked for all).

図 5.2: 評価入力画面

タベースの科目情報テーブルと履修済み科目評価テーブルに保存される．ここで科目情報テーブルの構造を表 5.6 に示す．各科目の特徴量は科目情報テーブルに保存され，科目の

表 5.6: 科目情報テーブル

カラム	データタイプ	デフォルト値
科目番号	varchar(6)	NULL
科目名	varchar(256)	NULL
特徴量定義済み	int(11)	0
特徴量 01	int(11)	0
特徴量 02	int(11)	0
:	:	:

特徴量が一度以上定義されている科目については特徴量が定義済みか否かを示すカラムを 1 にする。科目の特徴量はその時特徴量を定義したユーザとは別のユーザが定義しようとしたとき、前のユーザが定義した情報が引き継がれ表示される。ユーザは前のユーザが定義した特徴量の設定に間違いがあればそれを修正することができる。

ユーザの各科目に対する評価は履修済み科目評価テーブルに保存される。履修履歴をアップロードした際に保存されたデータに対して評価値を設定する。

5.2.3 推薦部分

本項ではシステムの中核となる嗜好の予測を行う推薦部分について述べる。前項においてユーザが履修済み科目について評価値を設定した際に、システム側では擬似的な評価値テーブルを作成する。これはユーザの履修済み科目が全体に対して少なすぎるため、そのまま協調フィルタリングをした場合に精度が上がらないために行う。既にユーザが評価した科目に加えて、特徴量が定義済みの科目のユーザ評価予測値を内容ベースフィルタリングを行って求め、それを擬似的な評価値テーブルに保存する。その後、作成された擬似的な評価値テーブルを用いて協調フィルタリングを行う。以下でそれぞれの工程について詳しく述べる。またそれぞれの工程について述べる際の記号を一部表 2.3 の再掲になるが、表 5.7 のように定義し、行列 F について式 5.1 に示す。

$$F = \begin{pmatrix} F_{1,1} & \cdots & F_{1,m} \\ \vdots & \ddots & \vdots \\ F_{o,1} & \cdots & F_{o,m} \end{pmatrix} \quad (5.1)$$

行列 F について、 $F_{i,s}$ と表記するときに科目 i が特徴量 s を有するか否かを示し、その値は有する時は 1、有さない時は -1 を指し示す。

(1) 擬似的な評価値テーブルの作成

擬似的な評価値テーブルの作成には科目情報テーブルの科目特徴量とユーザの履修

表 5.7: 記号の定義

記号	定義
a	活動ユーザを意味する添字
k	評価予測値を計算されるアイテムを意味する添字
I	全アイテムの集合
U	全ユーザの集合
F	各科目の特徴量をそれぞれ示した行列
F'	特徴量の集合
o	特徴量の種数
I_x	ユーザ x が評価したアイテムの集合
I^f	特徴量の設定をされたアイテムの集合
I_s^f	特徴 s のあるアイテムの集合
U_i	アイテム i を評価したユーザの集合
$F_{i,s}$	科目 i が特徴量 s を有するか否かを示す値
$v_{x,i}$	ユーザ x のアイテム i への評価値
$v'_{x,i}$	ユーザ x のアイテム i への評価値の予測値
$v''_{x,i}$	ユーザ x のアイテム i への擬似的な評価値
\bar{v}_x''	ユーザ x の擬似評価値の平均値

済み科目評価値を用いる．擬似的な評価値を求める手法を式 5.2，及び式 5.3a，5.3b に示す．

$$f_{x,s} = \frac{\sum_{i \in I_x \cap I_s^f} v_{x,i}}{|I_x \cap I_s^f|} \quad (5.2)$$

$$v''_{x,i} = \begin{cases} v_{x,i} & (i \in I_x) \\ \frac{\sum_{s \in F'} f_{x,s} F_{i,s}}{|F'|} & (i \notin I_x) \end{cases} \quad (5.3a)$$

$$(5.3b)$$

ここで $v_{x,i}$ の定義域は予め 0 を中心とする値に正規化しておくこととする．例えば，評価値の定義域を 1 から 5 と設定していた場合，それら全てに対して 3 だけ減算する等の手法をとる． $f_{x,s}$ はユーザ x についての特徴量 s の嗜好度合いを示している．システムは各ユーザについて各特徴量の嗜好度合いを求める．次に各特徴量についての嗜好度合いを用いて擬似的な評価値テーブルを作成するが，既にユーザが評価済みのアイテムについてはその値を直接用いる．もし未評価，すなわち履修したことがない場合は式 5.3b にあるように，特徴量の嗜好度合いを係数として加重平均した結果を擬似的な評価値とする．この一連の計算を登録された全ユーザと I^f のアイ

テム全てについて行い，擬似評価値テーブルとする．

擬似的な評価値とは内容ベースフィルタリングによる評価予測値に他ならない．したがって，擬似的な評価値を求めた時点で既にユーザに推薦されるアイテムは決まっているかのように思えるが，本項で述べた内容ベースフィルタリングのみでは十分な精度の推薦ができないため，内容ベースフィルタリングで求めた評価値を協調フィルタリングに用いるためのデータとすることで精度を高めている．

(2) 擬似的な評価値を用いた協調フィルタリング

本システムで用いる協調フィルタリングは第 2 章で解説したピアソン積率相関係数を元に加重平均を用いて求める手法を用いる．ピアソン積率相関係数をそのまま用いた場合，相関係数を求める際に用いたアイテム数が少ないものはその値の信頼度は低く，多ければ信頼度は高くなるが，それらは無視してしまっている．しかし，本システムでは擬似的な評価値テーブルを作成しているため，相関係数を求める際に用いるアイテム数は全て等しくなるため，ピアソン積率相関係数をそのまま用いる手法で問題がない．協調フィルタリングに用いる手法を式 5.4，及び式 5.5 に示す．

$$p_{x,y} = \frac{\sum_{i \in I^f} (v''_{x,i} - \bar{v}''_x) (v''_{y,i} - \bar{v}''_y)}{\sqrt{\sum_{i \in I^f} (v''_{x,i} - \bar{v}''_x)^2} \sqrt{\sum_{i \in I^f} (v''_{y,i} - \bar{v}''_y)^2}} \quad (5.4)$$

$$v'_{a,k} = \bar{v}''_a + \frac{\sum_{\substack{z \in U \\ z \neq a}} p_{a,z} (v''_{z,k} - \bar{v}''_z)}{\sum_{\substack{i \in U \\ i \neq a}} |p_{a,i}|} \quad (5.5)$$

ここで第 2 章において述べた式 2.3 では総和の範囲が $I_x \cap I_y$ であったのに対し，式 5.4 では I^f としている．これは擬似的な評価値テーブルを作成する際に，登録したユーザは特徴量の設定されたアイテム全て，すなわち I^f のアイテム全てについての擬似評価値を生成しているためである．

常に最新の推薦結果を提示するには，ユーザが特徴量未評価の科目に対して新たに評価する度に擬似的な評価値テーブルを全て再計算する必要がある．しかし運用しているホストのリソースを考慮し，実際にはユーザが評価入力した際にはそのユーザの分のみ擬似的な評価値を計算し，全ユーザに対するテーブルの更新は一定時間置きに行うこととした．

5.2.4 アイテム提示部分

本項では推薦された科目の提示を行う部分について述べる．推薦部分によって各科目に対するスコアがついたところで，ユーザに対して推薦されるアイテムの提示を行う．4.2 節で述べたように推薦結果の表示を行う際には推薦アイテム数を絞るか，それぞれのアイ

テムの推薦度合いを一見して伝わるようにする必要がある。また，2.7 節で述べたように，推薦されたアイテムについての情報を得る手段を明確にすることが推奨されている。本システムではタグクラウドを用いて，文字の大きさによって提示されたアイテムがどの程度活動ユーザに推薦されているかを表現した。また，科目の情報を素早く閲覧できるように，推薦された各科目名には講義概要が記述されたページへのハイパーリンクを付した。推薦結果表示画面を図 5.3 に示す。

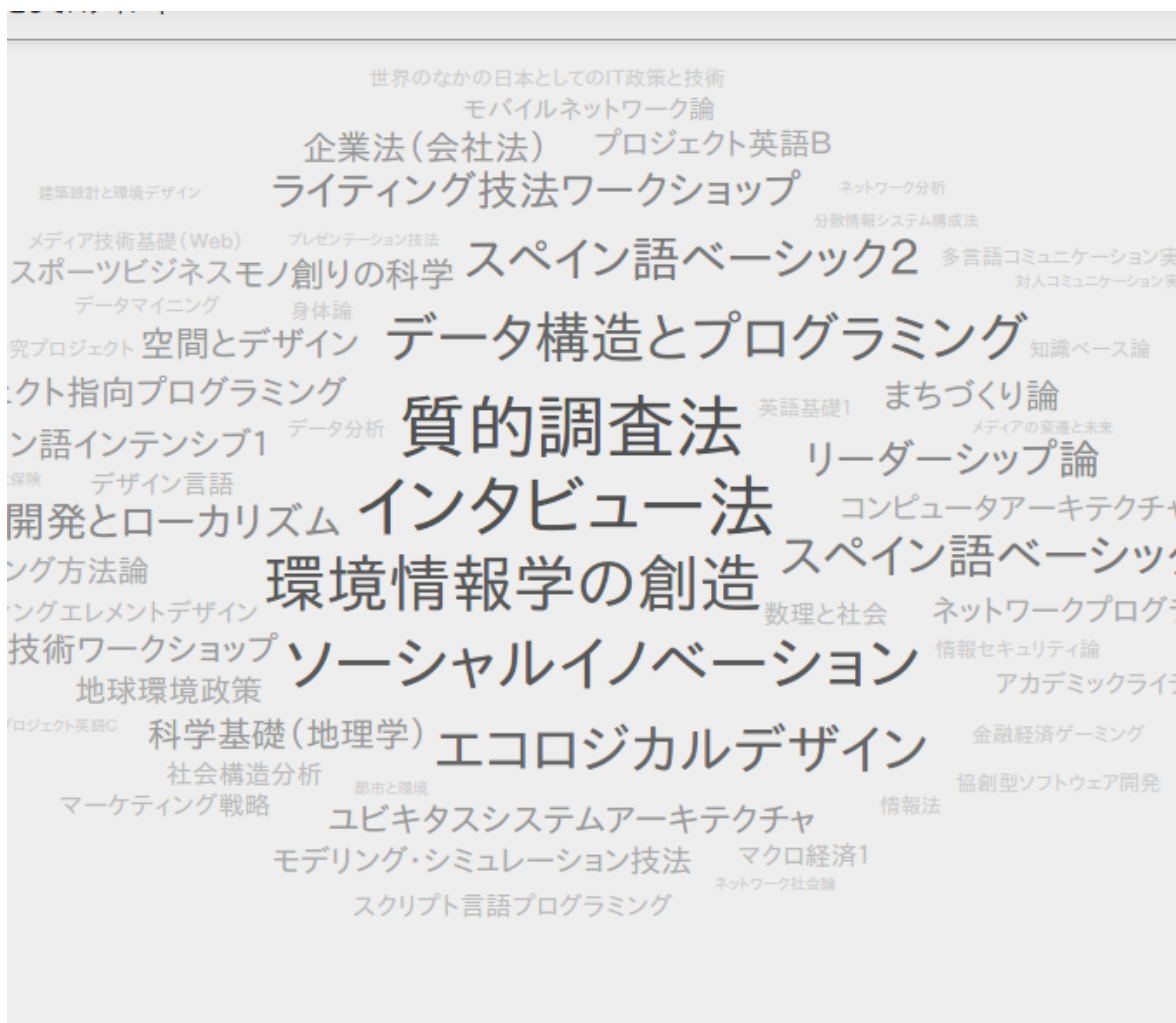


図 5.3: 推薦結果表示画面

5.3 まとめ

本章では，第 4 章で提案した手法の実装方法について述べる。まず実装環境とシステムの構成要素を示し，続いて推薦システムを構成する 3 つのフェーズ，すなわちユーザプロフィール獲得フェーズ，嗜好の予測フェーズ，推薦アイテム提示フェーズに加えて，本システムには必須であった認証機能について，本システムが用いた実装手法を詳しく述べた。

第6章 評価

本章では、第5章で実装したシステムについての評価を行う。まずシステムの評価を行うための評価指標について述べ、その評価指標を求めるためのデータをどのように準備するかについて述べる。その後実際にシステムの評価を行い、その結果についての考察を述べる。

6.1 評価手法

本研究におけるシステムの評価指標には2.8.2項で述べた $RMSE$ を用いる。本節では $RMSE$ と、 $RMSE$ に似た評価指標である MAE について、実際にサンプルデータを用いて計算し、その違いを明らかにしつつ述べる。

6.1.1 $RMSE$ と MAE

$RMSE$ は、アイテムについてのユーザ評価値がある場合に使用可能で、実際のユーザ評価値にマスキングし、その評価値をその他のユーザ評価値から予測した結果、実際の値と予測した結果でどの程度振れがあるかを測定することによって求まる評価指標である。 $RMSE$ は値が小さいほど振れが少なく0で完全一致を表す。測定は実際のユーザ評価値と予測の結果の差の二乗値を求める作業を評価用データの集合に対してそれぞれ行い、その差の二乗値の平均に対する平方根で求まる。 $RMSE$ と似た評価指標に MAE がある。 MAE は差の二乗値ではなく差の絶対値の平均で求まる。本研究では、推薦結果をタグクラウドを用いて提示する際、推薦システムから浮動小数点数で与えられた評価値を10段階になるように端数処理し、フォントサイズや色を決めている。そのため、評価値の小さな振れは数値を10段階に縮めた際に消滅するため、問題にならない。しかし、より大きな振れ、例えばユーザ評価値が1であるものを推薦システムが5と評価した時は、推薦されるべきでない科目が大きなフォントで表示されるため、ユーザのニーズに合った推薦とは言えない。そのため、より大きな振れに対してより大きく影響する $RMSE$ を評価指標として選択した。 $RMSE$ は MAE と比べ、実際の値からの外れ方が大きい値が評価に大きく影響する評価指標である。例として表6.1にあるユーザデータに対する2種類の推薦結果に対して $RMSE$ と MAE をそれぞれ求める。推薦結果1についての $RMSE$

表 6.1: サンプルデータ

	D							
	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8
ユーザデータ U	3	5	2	4	5	1	4	2
推薦結果 P^1	2	4	1	3	5	1	3	3
推薦結果 P^2	2	50	1	3	5	1	3	3

と MAE はそれぞれ式 6.1, 式 6.2 のように計算される .

$$RMSE = \sqrt{\frac{\sum_{d \in D} (P_d^1 - U_d)^2}{|D|}} = 0.87 \quad (6.1)$$

$$MAE = \frac{\sum_{d \in D} |P_d^1 - U_d|}{|D|} = 0.75 \quad (6.2)$$

推薦結果 2 についての $RMSE$ と MAE はそれぞれ式 6.3, 式 6.4 のように計算される .

$$RMSE = \sqrt{\frac{\sum_{d \in D} (P_d^2 - U_d)^2}{|D|}} = 15.93 \quad (6.3)$$

$$MAE = \frac{\sum_{d \in D} |P_d^2 - U_d|}{|D|} = 6.25 \quad (6.4)$$

表 6.1 の推薦結果 2 は推薦結果 1 の振れを 1 つの値だけ大きくしたデータ集合である . それぞれの $RMSE$ と MAE の計算結果から , 推薦結果 1 の $RMSE$ と MAE の差は 0.12 であるのに対し , 振れを一部大きくした推薦結果 2 の $RMSE$ と MAE の差は 9.68 であるため , $RMSE$ は大きな振れに対しての感度が高いことがわかる .

6.1.2 $RMSE$ 値の評価

推薦システムの評価指標として $RMSE$ を用いている例として Robert 氏らの論文 [39] では , Netflix Prize[40] で与えられたデータセットにおいて用いるデータによって 1.1296 から 0.9657 の $RMSE$ 値を得られたとしている . ここで Netflix Prize について以下に詳しく述べる .

- Netflix Prize
Netflix Prize とは Netflix 社 [41] で 2006 年から主催された推薦アルゴリズムコンテ

ストで、同社の推薦システム Cinematch の精度を 10%向上させると賞金 100 万ドルが貰えるというものである。Cinematch の精度とは $RMSE$ 値で 0.9525 であったため、 $RMSE$ 値を 0.85725 以下にできた際に賞金を貰えるチャンスが来る。与えられる教師用データには映画の特徴量などは一切含まれないため、協調フィルタリングの精度を競う。教師用データは Netflix 社から表 6.2 にある形式のデータがダウンロード可能である。表 6.2 の形式のデータは合計 17770 本映画に対する約 48 万人の

表 6.2: Netflix dataset

	要素	フォーマット
1	user ID	integer
2	movie ID	integer
3	date of grade	YYYY-MM-DD
4	grade	integer($1 \leq \text{grade} \leq 5$)

評価値 1 億件以上であり、そのうちの約 9900 万件が教師用データとして使用可能であり、残りの約 300 万件がテスト用として grade の値がマスキングされたデータ、すなわち予測値を求めコンテストに送信するためのデータである。

Netflix Prize では $RMSE$ 値として 0.9525 から 0.8567 までの範囲で 44014 件の投稿があった。Netflix Prize では精度の高い推薦アルゴリズムを求めるためのコンテストであったため、これらに近い値であれば十分に高い精度の推薦ができていると考えられる。また、参考としてランダムに生成した 1 から 5 の値で埋められた配列と、そのおよその平均値である 3 で埋められた配列の共に長さ 10000 の 2 つの配列について $RMSE$ 値を求めたところ、結果は 1.414 であった。そのため、 $RMSE$ 値が 1.414 以上の場合は有意な推薦はできていないと判断することができる。

6.1.3 実際の測定手法

$RMSE$ 値の測定を実際に行う手法について述べる。評価に用いるデータはそれぞれのユーザが今までに履修したことのある科目、すなわちユーザ評価値のつけられた科目の、ユーザ評価値を用いる。表 6.3 を例に述べる。データベースのユーザ評価値テーブルが表 6.3 となっていたと想定したとき、 $RMSE$ 値の測定に用いるデータであるユーザ X の科目 B, C 、ユーザ Y の科目 A, B, C, E 、ユーザ Z の科目 A, C, D を集合 Δ の要素とする。この集合 Δ に含まれる要素のうち 1 件のユーザ評価値を ϕ と仮定し、本システムによって ϕ と仮定した該当ユーザの該当科目について評価予測値を得る。この作業を集合 Δ の要素全てについて行い、集合 Δ の要素全てについてユーザ評価値と評価予測値のセットを得る。ここで得られたユーザ評価値と評価予測値の差の二乗平均平方根が $RMSE$ 値の測定手法である。

表 6.3: 実際の評価手法の解説に用いる表

	科目 A	科目 B	科目 C	科目 D	科目 E
ユーザ X	ϕ	2	5	ϕ	ϕ
ユーザ Y	3	4	1	ϕ	1
ユーザ Z	5	ϕ	4	1	ϕ

6.2 評価用データの準備

本節では、評価に用いるデータの準備方法について述べる。本研究の実装ではユーザの入力した評価値はデータベースのユーザ評価値テーブルに保存され、そのテーブルから擬似評価値テーブルを作成し、協調フィルタリングを行う。そのため、ユーザの評価値をマスキングした時の推薦結果は、ユーザ評価値をユーザ評価値テーブルから一時的に削除し、一連の推薦プログラムを実行することで得られる。この作業を全てのユーザ評価値に対して行って得られた推薦結果と実際のユーザ評価値について $RMSE$ を計算することで評価結果を得る。

6.3 評価結果

本節では、これまで述べてきた手法を用いて本研究で実装したシステムについての評価を行い、その結果について述べる。

6.3.1 全体データ対する評価

本システムの登録ユーザ 16 名から合計 435 件の評価用データを取得し、評価実験を行った。その結果、 $RMSE$ 値は 1.140 となった。この結果について、実際にシステムを利用する際にどのように表示されるかをあるユーザを例に実験した。実験はあるユーザについて最新の学期の履修履歴の評価値を全て ϕ 、すなわち未履修状態とし、その状態でシステムを利用してもらい、実際には履修し評価した科目がどのように表示されるかを実験した。その結果を図 6.1 に示す。また、 ϕ と見なした科目名とそのユーザ評価値を表 6.4 にまとめた。

図 6.1 に示した画面の下線部分が最新学期履修科目を表す。図 6.1 と表 6.4 から、どの程度一致しているかを把握することができる。例えばユーザ評価値の高い「Web デザインとマネジメント」はタグクラウド上で実際に大きく表示され、システムも高く評価することに成功している。一方で、「Web テキスト処理法」はそれよりも低いユーザ評価値の「記号処理プログラミング」よりタグクラウド上で小さく表示されている。このような誤差は $RMSE$ 値に示された程度にある。また、「データ構造とプログラミング」は最も大き

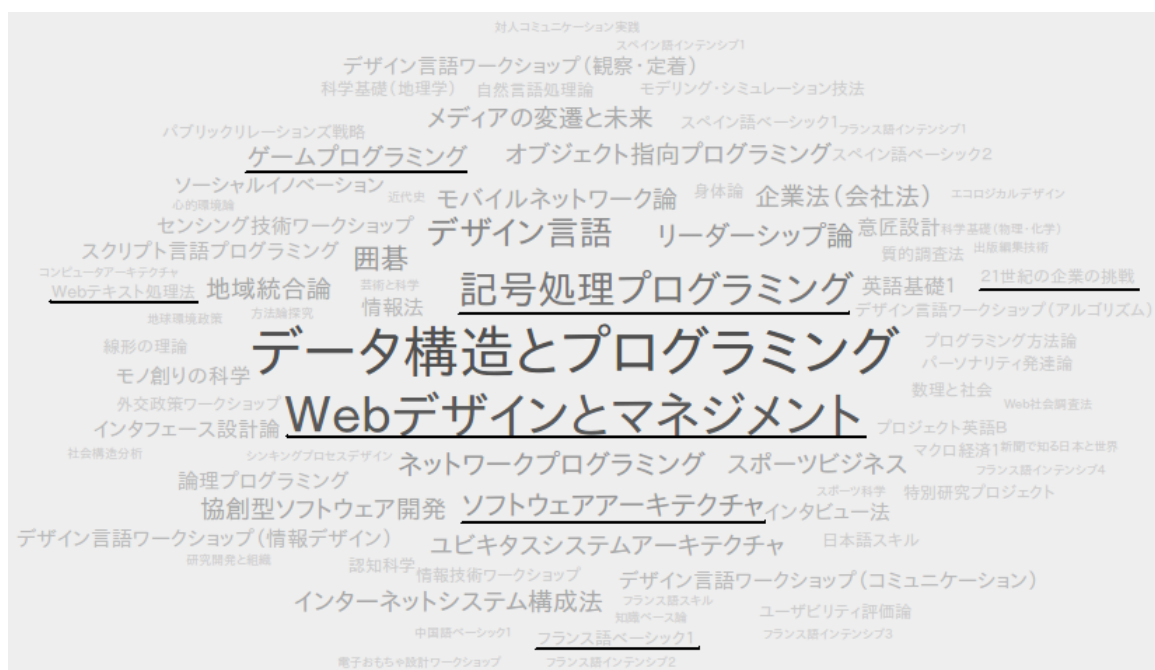


図 6.1: 推薦結果表示

表 6.4: 最新学期履修科目とそのユーザ評価値データ

科目名	ユーザ評価値
Web デザインとマネジメント	4
ソフトウェアアーキテクチャ	4
Web テキスト処理法	4
ゲームプログラミング	3
記号処理プログラミング	3
ネットワーク文化	3
21 世紀の企業の挑戦	2
フランス語ベーシック 1	1

く表示され、他の科目と比較して高いユーザ評価値を得ると予測されが、このケースでは被験者が履修をしていないためにその精度を確認することができなかった。

6.3.2 評価済み科目数と $RMSE$ 値の関係

一般的には、入力した評価の数が多いほどユーザについての情報が増えるため精度の高い推薦が可能である。本システムの対象ユーザは目標や目的がまだ漠然としていて、シラバス検索システムではどのような検索クエリを入力すれば良いか思いつかないような学生

である．そのような学生は高学年より低学年に多いことが考えられるが，低学年の場合，履修システムに評価を入力した科目数は相対的に少なくなる．そのため，評価を入力した科目数の少ない学生に対しての精度が多く入力した学生に対して大幅に低いことは本研究の目的に沿わない．そのため，評価済みの科目数と，そのときの $RMSE$ 値を各ユーザについて求めた．その結果を図 6.2 に示す．図 6.2 に示したグラフから，評価した科目

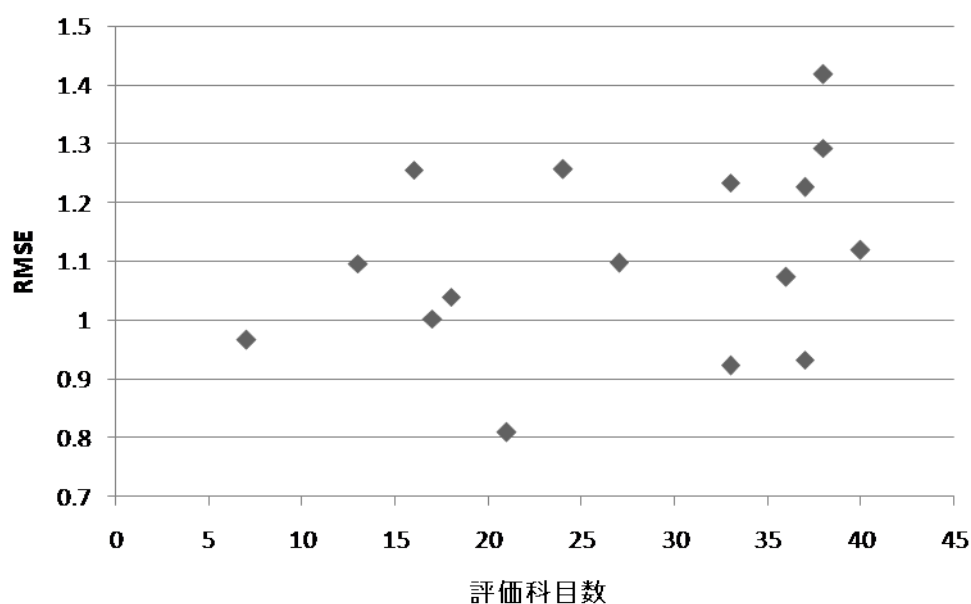


図 6.2: 評価済み科目数と $RMSE$

数の少ない場合に推薦の精度が有意に低くなる傾向は見られない．

6.4 まとめ

本章ではまず，第 4 章で提案し，第 5 章で実装したシステムについての評価手法を述べ，その評価手法に用いる評価指標である $RMSE$ について詳しく述べ，それを実際どのような手順によって測定するかを述べた．その後本システムに対して評価実験を行った．その結果， $RMSE$ 値は 1.140 であった．実際にシステムの利用を想定した際に， $RMSE$ 値に示される精度によってどのように推薦結果が提示されるかを示した．また，各ユーザが評価した科目数と推薦の精度の相関について調べ，評価した科目数の比較的少ないユーザに対して評価した科目数の多いユーザと比べて有意に低い精度の推薦となる傾向は見られないことを述べた．最後に，本研究で実装した推薦システムに対する考察を述べた．

第7章 結論

本章では、第 1.3 節で示した目的の中で、達成された部分を述べる。そして、本研究の更なる発展のために今後の展望を述べる。

7.1 本研究における目的の達成

本節では、本論文のまとめを述べ、最後に 1.3 節で述べた本研究の目的を再度確認し、それらの目的をどの程度達成できたかについて述べる。

7.1.1 本研究の目的

本研究の目的は、シラバス検索システムに入力する検索クエリが思いつかないような学生に、快適な履修科目選択が可能とすることが目的である。その目的の達成に必要な項目を以下にまとめる。

- (1) 履修すべき科目が明確であること
対象とするユーザはどの科目を履修すべきか思案中の学生である。そのため、その履修科目選択を支援するためには学生に対して明確に科目の提示をする必要がある。
- (2) 学生のニーズに合っていること
対象とするユーザが提示された科目を実際に履修した場合に、その科目が学生のニーズに合っていなければ学生にとって必要な科目を提示できていなかったことになり、適切に履修科目選択の支援をできたことにはならない。

7.1.2 達成された目的

本研究では 7.1.1 節に挙げた目的を達成するため、現状用いられているシラバス検索システムの他に推薦システムを用いた。目的の達成に必要な項目それぞれについてその達成状況を以下にまとめる。

- (1) 履修すべき科目が明確であること
シラバス検索システムでは検索結果が並列に提示される。しかし、本研究で実装した推薦システムにおいて科目を提示する際にはタグクラウドを用い、各科目のフォ

ントの大きさを以て推薦度合いとした．タグクラウドは一見して重要である部分が見える．すなわち，提示された科目のそれぞれの重要度が明確であるため，学生にとって履修すべき科目が明確に提示された．

(2) 学生のニーズに合っていること

対象とするユーザが提示された科目を実際に履修した場合に，その科目が学生のニーズに合っていないければ学生にとって必要な科目を提示できていなかったことになり，適切に履修科目選択の支援をできたことにはならない．本研究で実装した推薦システムは推薦システムの評価指標として Netflix Prize などで行われている $RMSE$ を用い，その結果として $RMSE$ 値は 1.140 であった．これは Netflix Prize に投稿されたデータには劣るものの，それらに近い値であるため，十分に高い精度で推薦できたと言える．したがって，学生のニーズを高い精度で予測することができた．

7.2 まとめ

本研究の目的は 7.1.1 に挙げた．これらを達成することによって，現状大学から提供されているシラバス検索システムのみでは足りなかった部分を補完することができるようになると思われる．そこで，本研究では履修科目選択の支援システムとして履修科目推薦システムを提案し，それを設計・実装した．そして実際に学生に使ってもらい，本提案の有効性を検証した．結果として得られた $RMSE$ 値は 1.140 であり，これは他分野に適用されている推薦システムから得られる $RMSE$ 値に近い値であり，履修科目選択の分野についての推薦としては高い精度であると考えられる．今後，本研究が発展し，大学側からシラバス検索システムに加えてシラバス推薦システムが提供されることになれば，学生はより快適に履修科目選択を行うことができるようになると思われる．

7.3 今後の展望

本研究の発展のため，主に推薦部分の精度について今後の展望を述べる．本研究の実装では内容ベースフィルタリングと協調フィルタリングを組み合わせた手法を使用した．具体的には，科目に設定された特徴量とユーザの評価値からユーザの各特徴量に対する嗜好度合いを求め，それを基に擬似的な評価値テーブルを作成し，その擬似的評価値テーブルを用いて協調フィルタリングを行うというものである．その結果として $RMSE$ 値は 1.140 であった．本研究では科目の特徴量として授業形態を表す 5 つの単語を用いた．しかし，科目の特徴量として何を選ぶべきかは学生が各科目のどのような特徴について検討しているかによって変わる．そのため，特徴量の設定は実際に推薦システムを利用すると考えられる学生に対してアンケートを実施するなどして決定すると，より有効な特徴量の設定が可能であると考えられる．また，協調フィルタリングのアルゴリズムについて，今回の実装では内容ベースフィルタリングによって擬似的に生成された評価値とユーザが実際に入力した評価値について等しい重み付けで協調フィルタリングを行った．この点につ

いて、ユーザが実際に入力した評価値を擬似的に生成された評価値より高い重み付けにすれば、内容ベースフィルタリングによって生成された評価値が本来あるべき数値から大きく振れていた際にも影響は少なくなると考えられる。

以上に挙げた 2 つの点について改良することにより本システムによる推薦の精度が向上し、より有効な推薦ができるようになると考えられる。

謝辞

本論文の執筆にあたり、ご指導頂いた慶應義塾大学環境情報学部学部長 村井 純博士、同学部教授 徳田 英幸博士、同学部教授 中村 修博士、同学部准教授 楠本 博之博士、同学部准教授 高汐 一紀博士、同学部准教授 三次 仁博士、同学部准教授 植原 啓介博士、同学部専任講師 中澤 仁博士、同学部准教授 Rodney D. Van Meter III 博士、同学部教授 武田 圭史博士、同大学政策・メディア研究科齊藤 賢爾博士、同研究科特任講師 佐藤 雅明博士、同研究科特任講師吉藤 英明氏に感謝致します。特に武田圭史博士は、研究で行き詰まる私に快くご指導していただきました。本当にありがとうございました。

そして、本研究を進めていく上で、様々な励ましと助言、お手伝いをいただきました、村井研究室卒業生である水谷 正慶氏、上原 雄貴氏、重松 邦彦氏、梅田 昇翔氏、福岡 英哲氏、相見 眞男氏、Doan Viet Tung 氏、Vu Xuan Duong 氏、Pham Van Hung 氏、吉原 洋樹氏に感謝致します。慶應義塾大学政策・メディア研究科修士課程、碓井 利宣氏、関根 冬輝氏、山本 知典氏に感謝致します。特に山本 知典氏は、私の研究の相談にいつも素早く応えていただきました。氏なしでは卒論執筆はもちろんのこと、充実した研究室生活を送ることはできませんでした。本当に感謝致します。

研究室で苦楽を共にした有馬 怜文氏、大矢 崇央氏、Nguyen Anh Tien 氏、小松 真氏、中島 明日香氏、由井 卓哉氏、吉原 大道氏に感謝致します。彼らと一緒に研究をすることでお互いを刺激しあい、より質の高い議論や研究をすることができました。特に、ISCの同級生には、研究室内での活動だけでなく、对外発表や对外活動、レクリエーションなど様々な場面でお世話になりました。この場を借りてお礼を述べさせていただきます。

最後に、大学入学からの4年間だけでなく22年間をあらゆる面で支えていただいた父、鴻野 立明、母、鴻野 美奈子に心から感謝致します。

参考文献

- [1] ネットワークプログラミングシラバス. http://vu8.sfc.keio.ac.jp/course2007/summary/syll_view_c.cgi?yc=2012_25476&ks=13070, 2012.
- [2] Sheena S. Iyengar and Mark R. Lepper. When choice is demotivating: Can one desire too much of a good thing? *Journal of Personality and Social Psychology*, 79(6):995–1006, 2000.
- [3] Sheena S. Iyengar, Gur Huberman, and Wei Jiang. How much choice is too much?: Contributions to 401(k) retirement plans. Technical report, Pension Research Council, 2003.
- [4] Inc. Amazon.com. <http://www.amazon.co.jp/>, 2012.
- [5] Google. <https://www.google.co.jp/>, 2012.
- [6] Toshihiro Kamishima. Algorithms for recommender systems (1). *Journal of JSAI*, 22(6), 2007.
- [7] Dan Cosley, Shyong K. Lam, Istvan Albert, Joseph A. Konstan, and John Riedl. Is seeing believing? how recommender interfaces affect users’ opinions, 2003.
- [8] Robin Burke. Hybrid recommender systems: Survey and experiments, 2002.
- [9] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. GroupLens: An open architecture for collaborative filtering of netnews, 1994.
- [10] Badrul Sarwar, George Karypis, Joseph Konstan, , and John Riedl. Item-based collaborative filtering recommendation algorithms, 2001.
- [11] Kamal Ali and Wijnand van Stam. Tivo: Making show recommendations using a distributed collaborative filtering architecture, 2004.
- [12] 千田 智治, 廣安 知之, and 三木 光範. 凝集法と k-means 法. <http://www.is.doshisha.ac.jp/report/2008/7/5/20081014002/index.html>, 7 2008.
- [13] Mark Claypool, Anuja Gokhale, Tim Miranda, Pavel Murnikov, Dmitry Netes, and Matthew Sartin. Combining content-based and collaborative filters in an online newspaper, 1999.

- [14] Beyond Algorithms: An HCI Perspective on Recommender Systems. Kirsten Swearingen and Rashmi Sinha, 2001.
- [15] Cyril W. Cleverdon and Michael Kean. Factors determining the performance of indexing systems, 1968.
- [16] Rob J Hyndman and Anne B Koehler. Another look at measures of forecast accuracy, 2005.
- [17] 神鷲 敏弘. なんとなく協調フィルタリング 順序応答に基づく推薦, 2003.
- [18] Yukinori Koide. Study on recommender system construction and its evaluation method -application to student supporting system for class selection-. Master's thesis, Keio University Graduate School of Media and Governance, 2010.
- [19] 森 幹彦, 由谷 真之, and 喜多 一. 大学教養教育における科目選択支援, 2006.
- [20] 土肥 紳一, 中村 尚五, 島田 尊正, and 川勝 眞喜. ダイナミックの開発ー情報環境学部におけるダイナミックシラバスの開発ー, 2001.
- [21] Kazuki TAKEGAWA, Yoshinori HIJIKATA, and Shogo NISHIDA. Implementation of a content-based music search and recommendation method, 2008.
- [22] Yuji TARUI. Recommendation system of tourist site using collaborative filtering method and contents analysis method, 2011.
- [23] No Free Lunch Theorems for Search. David H. Wolpert and William G. Macready, 1996.
- [24] タグクラウド. <http://e-words.jp/w/E382BFE382B0E382AFE383A9E382A6E38389.html>, 2012.
- [25] Ubuntu 12.04. <http://www.ubuntu.com/>, 2012.
- [26] The Apache HTTP Server Project. <http://httpd.apache.org/>, 2012.
- [27] PHP 5.3.5. <http://php.net/>, 2011.
- [28] JavaScript. <http://www.w3.org/wiki/Javascript>.
- [29] HTML. <http://www.w3.org/wiki/HTML>.
- [30] PHP IDE :: JetBrains PhpStorm. <http://www.jetbrains.com/phpstorm/>, 2012.
- [31] PHP Simple HTML DOM Parser. <http://simplehtmldom.sourceforge.net/>, 2012.
- [32] Bootstrap. <http://twitter.github.com/bootstrap/>, 2012.

- [33] jQuery. <http://jquery.com/>, 2012.
- [34] jQCloud. <https://github.com/lucaong/jQCloud>, 2012.
- [35] MySQL 5.1.63. <http://www.mysql.com/>.
- [36] Web. <http://e-words.jp/w/Web.html>, 12 2012.
- [37] Google Chrome. <http://www.google.co.jp/intl/ja/chrome/browser/>, 2012.
- [38] SFC-SFS. <http://vu8.sfc.keio.ac.jp/sfc-sfs/>, 2012.
- [39] Robert M. Bell and Yehuda Koren. Scalable collaborative filtering with jointly derived neighborhood interpolation weights. *IEEE International Conference on Data Mining*, 2007.
- [40] Netflix Prize. <http://www.netflixprize.com/>, 2006.
- [41] Netflix. <http://www.netflix.com/>.