

卒業論文

教学データの学習エビデンスに基づく GPA向上のための 情報推薦・学習支援システムの開発

Optimization in Information
Recommendation Using Clustering
in Teaching data

富山県立大学 工学部 電子・情報工学科

1815043 滝沢 光介

指導教員 奥原 浩之 教授

提出年月: 令和3年(2021年)2月

目次

| | |
|----------------------------|-----|
| 図一覧 | ii |
| 表一覧 | iii |
| 記号一覧 | iv |
| 第1章 はじめに | 1 |
| § 1.1 本研究の背景 | 1 |
| § 1.2 本研究の目的 | 2 |
| § 1.3 本論文の概要 | 2 |
| 第2章 教学データ分析と情報推薦 | 4 |
| § 2.1 教学におけるビッグデータ・アナリティクス | 4 |
| § 2.2 eポートフォリオによる学習エビデンス | 5 |
| § 2.3 情報推薦と協調フィルタリング | 5 |
| 第3章 取得科目の推薦と教材の最適化 | 8 |
| § 3.1 協調フィルタリングからの科目の推薦 | 8 |
| § 3.2 シラバスからの教材作成 | 9 |
| § 3.3 GPA 向上のための教材の提示 | 10 |
| 第4章 提案手法 | 11 |
| § 4.1 予測評価値からの適切な授業科目の推薦 | 11 |
| § 4.2 関連資料のアップデート | 11 |
| § 4.3 提案手法のアルゴリズム | 11 |
| 第5章 数値実験並びに考察 | 13 |
| § 5.1 数値実験の概要 | 13 |
| § 5.2 実験結果と考察 | 13 |
| 第6章 おわりに | 14 |
| 謝辞 | 15 |
| 参考文献 | 16 |

図一覽

表一覽

記号一覧

以下に本論文において用いられる用語と記号の対応表を示す.

| 用語 | 記号 |
|--------------------------|-----------------|
| 特定の利用者 | x |
| 特定のアイテム | y |
| 利用者数 | n |
| アイテム数 | m |
| 利用者集合 $\{1, \dots, n\}$ | \mathcal{X} |
| アイテム集合 $\{1, \dots, m\}$ | \mathcal{Y} |
| アイテム y を評価した利用者の集合 | \mathcal{X}_y |
| 利用者 x が評価したアイテムの集合 | \mathcal{Y}_x |
| 対象ユーザー | a |
| 利用者 x のアイテム y への評価値 | r_{xy} |
| 利用者 x による評価値の平均 | \bar{r}_x |
| アイテム y への評価値の平均 | \bar{r}_y |

はじめに

§ 1.1 本研究の背景

近年、ICT 技術の発展により、大規模なデータを容易かつリアルタイムに蓄積できるようになった。これらのデータは広い範囲で「ビッグデータ」と呼ばれ、これらのデータに対する分析はデータサイエンスやアナリティクスと呼ばれその需要を高めている。これらのことは教育機関においても例外ではなく、多くの学校、特に大学においては、生徒の学習内容ログや修学状況などの教学データが日々蓄積されている。

教学データにおける分析は 2000 年前後から盛んになり、最近では九州大学のラーニングアナリティクスセンターにおいて、蓄積された学生の様々な情報を整理、分析し学生に対してフィードバックを行うシステム、「M2B(みつば)」が実装されている [1]。

上記の M2B を筆頭に多くの大学で、生徒に対するアナリティクスシステムを提案するといった事例が多くの大学で行われている。

このように生徒の情報が蓄積された中で、大学における成績評価方法の一つとして Grand Point Average(GPA) が存在する。GPA とは大学における学生の各科目における成績を S, A, B, C, D のレターグレード (Letter Grade) で表し、それらを ($S=4$, $A=3$, $B=2$, $C=1$, $D=0$) のグランドポイント (Grand Point) に置き換え、それらの平均を算出したものである。GPA はその人の大学における成績の総合評価といえる。

大学での成績、特に GPA が就職活動において重要視されているかということについてはしばしば、議題に挙げられている。その理由は GPA の算出方法とその特徴が要因となっている。GPA は 60 点から 100 点までの 1 点刻みを点から 4 点までで刻み、0 点から 59 点までを 0 点で表現しているのでもとの素点と比べるとその精度は $\frac{1}{10}$ になってしまう。また、GPA は授業の難易度、教授の成績のつけ方などの条件で大きく変化し、各大学においてかならずしも同じ尺度で決められた数字ではないといった特徴がある。

しかし、多くの研究で少なからず大学における成績と就職活動には関係性があり、良い成績を納めている生徒ほど就職活動が優位に働いているとされている [2] [3]。さらには多くの大学において、成績優秀者に関して大学院進学における何かしらの優遇があることは事実である。このことから、大学時代における最大の目標である就職または進学の双方において、大学での成績が重要となってくる。

§ 1.2 本研究の目的

大学における GPA を上げる方法として、第一に履修した単位において良い成績を修めることである。しかし、大学生は一つの学期で多くの単位を取得する必要があるすべての履修単位について上位の成績を修めるのは困難である。また、自分が良い成績を修めることができるかどうかはその科目を履修してみないとわからない。さらには、大学における卒業要件単位を満たすために自分の不得意とする科目もある程度取得する必要がある。

それにも関わらず、大学における単位選択は生徒個人で行う必要があるその選択肢は膨大なものになる [4]。その結果、多くの学生、特に入学したての 1 年生に対して単位選択という行動がかなりの負担になる可能性がある。

そのため本研究では、過去の卒業生の教学データに対して分析を行い、システムを使用する学生がまだ取得していない科目において生成器評価の予測を行い、高い GPA を取得できるように単位選択の推薦を行う。それに加え、すべての科目について良い成績がとれるように予測成績が低い科目については Web 上から関連情報を推薦するシステムの開発を行う。

まず、教学データにおけるデモデータの作成を行い、そのデータを蓄積する。蓄積したデータに対して、システムのユーザーを対象とし協調フィルタリングを行い、対象ユーザーのまだ取得していない科目について、どれくらいの成績をとれるかといった予測成績値を算出する。

協調フィルタリングによって得られた予測評価値をもとに、ユーザーに対して推薦する科目を決定する。科目を選択する際には、予測成績が高い科目から推薦を行う。富山県立大学における卒業要件単位を満たすように科目の選択を行う。卒業要件単位を満たすように単位を選択すると予測評価値が低い科目も選択してしまう可能性がある。そのように、予測評価値が低いにもかかわらず選択されてしまった科目については、web 上からその科目に関する資料をスクレイピングし、スクレイピングした情報をユーザーに提供を行う。

提案手法によって構築されたシステムの有効性を検証する必要がある。そのために、科目の推薦を行う際にしっかりとパーソナライゼーションできているかの確認を行う。

§ 1.3 本論文の概要

本論文は次のように構成される。

第 1 章 本研究の背景と目的について説明する。背景では大学におけるビッグデータアナリティクスについてと大学における成績評価法の信頼性について述べる。目的は教学データからの最適な単位選択とそれにおける情報の提供することを述べる。

第 2 章 大学におけるビッグデータアナリティクスの概要とその発展についてまとめる。また、教学データに対する分析手法である、協調フィルタリングとそれにおける情報推薦について述べる。

第 3 章 システムの基礎となる協調フィルタリングについて述べる。またシラバスからの教材作製の手順を述べる。

第4章 学生にとって最適となる科目の選択, それに関する教材のアップデートについて述べる. また, システム全体の流れを述べる.

第5章 本研究の情報推薦の個人化度合いについて述べる. また教材のアップデートによるユーザーの使用感を検証する.

第6章 本研究で述べている提案手法をまとめて説明する. また, 今後の課題について述べる.

教学データ分析と情報推薦

§ 2.1 教学におけるビッグデータ・アナリティクス

教学におけるビッグデータの内容は学生の成績や学習ログのみに限らず、学生の ICT 機器の操作履歴 (どのページを参照したか) や学生へのアンケート、学生の作業物、授業の学習過程の記録、自己評価や他者からの評価など多岐に渡る。教学におけるビッグデータは一般的なビッグデータと比べて以下のような特徴があるとされている。

1. データ量は大きくない

一大学に限れば、テラバイトの処理で十分であり、ペタバイトクラスのデータベースを構築している通信会社の分析とは良の規模が違う。

2. 対象人数は短いが、データの種類の急速に増えている

国内最大規模の大学であっても、在籍学生数は 10 万以下である。一方、個々の学生に関して収集されるデータは、成績だけでなく、授業の出欠・図書館の利用・e ラーニングコンテンツへのアクセスなども含め、日々増加していく。

3. 匿名性が低い

教学データは個人情報といっても過言ではないほど学生個人の人となりを表しておりその扱いは慎重に行う必要がある。

4. データの意味解釈が容易

クレジットカードである商品を買ったとしても、本人が使用するのか、いつ使用するのか、なぜそのカードを使ったのかは不明であるが、ある授業に出席したり、オンラインで特定のコンテンツにアクセスしたりする活動の意味は明白であり、学生がどんな考えをもって行動しているかといった情報が筒抜けになる。

5. 因果関係は複雑

個々のデータの意味は明白であるが、各データ同士の関係は不明であることが多く、相関程度しかわからないケースがある。

6. 多様化・細分化が進行している

教育の ICT 化に伴って、個人レベルのマイクロデータが集積している。多くの大学で、教学におけるビッグデータはいくつかのデータベースにわけられており、それらを扱う際は、データの集積と情報の取捨選択が必要となる。

これらの特徴からいえることは、教学におけるビッグデータ・アナリティクスにおける課題は大量のデータの実行時間が問題となるわけではなく、さまざまな意味を持つデータを分析したところでデータ同士の関係性をはっきりさせることが難しいことである。それに加え、3, 4の特徴からデータにおける個人の把握が容易であり、個人情報として慎重に扱う必要がある。

また6の特徴に伴い、MERCERON らは種類が多いにも関わらず分析する対象が小さいデータを分析してもあまりインパクトのない結果になってしまうと指摘している。このことからラーニングアナリティクスに対する全盛期は終わったと述べて、分析結果におけるEDM・人工知能・学習化学などのコミュニティと協働する必要性があると述べている。

では、上記で述べたような小さいサイズに対する分析における研究はまったく意味がないものなのか。実はそうではなく、小さいサイズに対する分析を行い教育現場の課題解決を行うシステムを開発し、その有用性を示したシステム開発研究は数多く存在する。以下では小さいサイズに対するシステム開発研究の例をいくつか紹介し、システム実装におけるさらなる注意点を示す。

スモールデータに対するシステム開発研究

稲葉らは授業中の生徒同市または生徒と教師の円滑なコミュニケーションを目的とし、レスポンスアナライザシステムの導入を行っている。システムの機能としては質問投稿、質問に対して返信を行える投票機能、表示機能、投稿・投票を確認できるログ分析機能がある。扱っているデータとしては学生の質問・投票データに加え、学生がどの程度アクティビティに参加したかがわかるログデータだけである。データだけを見ると、一つの授業に限ればかなりサイズの小さなデータとなる。しかしながら、これらのシステムを運用することで、授業中の生徒同氏の行動が活発になり授業全体の活性化が見られたと報告している。また、教師側も学生がどこで躓いているかを把握することができ、円滑な授業の組み立てができるようになったと報告されている。

上記で挙げた特徴における課題を解決したとしても実際に大規模な教学データに対してラーニングアナリティクスのようなシステムを実装しようとしたときに

§ 2.2 eポートフォリオによる学習エビデンス

§ 2.3 情報推薦と協調フィルタリング

近年、パーソナライゼーション(個人化)といった言葉がよく使われている。パーソナライゼーションとは無数にある情報の中から、個人にあう情報を抜き出し、それらの情報を個人に適した形で表示することである。そういったパーソナライゼーションの核となる技術が情報推薦である。情報推薦の目的は、利用者にとって重要と思われる、事象や対象、情報を利用者の目的に沿う形で提示することである。有名なものでは、Amazonで行われている「この商品を購入した人はこちらの商品も購入しています。」といったものである。

そもそもなぜこのようなパーソナライゼーションといった考えが必要になったのか。その背景はICT技術の発展に伴う情勢の中で大きく分けて二つ存在する。

1. 大量の情報がインターネットを通して発信されるようになったこと
2. 大量の情報が誰でも、容易に手に入るようになったこと

こういった要因により、情報を受け取る側は、大量に存在する情報の中から自分の望む情報を取捨選択せざるを得なくなってしまった。それにもかかわらず、日々インターネットにはどこの誰かとも知らない人が発信した玉石混合な情報が更新されている。こうした情報があるにもかかわらず、その情報を利用できない状態を情報過多 (information overload), あるいは情報爆発 (information explosion) という。こうした現象に対処するために生み出されたのが、情報推薦技術を利用したパーソナライゼーションである。

このような情報化社会と呼ばれる現代では必須といっても過言ではないパーソナライゼーションを行うための情報推薦技術では主に利用者の嗜好を予測し、嗜好則した情報推薦を行う。情報推薦における嗜好の予測方法として内容ベースフィルタリング (content-based filtering) と協調フィルタリング (collaborative filtering) の大きく二つに分類される。以下ではそれぞれの方法について述べたのちに、本研究で使用する協調フィルタリングについてさらに詳しく説明を加えていく。

内容ベースフィルタリング

内容ベースフィルタリングは、アイテム (利用者に推薦する事象や対象) ごとの特徴量をベクトルで表し、利用者の嗜好に近いアイテムの推薦を行うものである。例えば、外食先を推薦するシステムについて考える。この際に利用者は自分の今日食べたいものや自身の住んでいる地域などを検索ボックスに入力する。利用者が「原宿 イタリアン」と入力すると、この「原宿」と「イタリアン」という特徴量についてお店がソートされ、類似度の高い店をおすすめの店として紹介する。このように利用者の入力に基づいて情報を推薦するのが内容ベースフィルタリングである。

協調フィルタリング

協調フィルタリングは、利用者がシステムを利用する以前から利用者のアイテムに対する嗜好データをデータベースに蓄積した利用者データベースを保持している。協調フィルタリングではそれらの情報をもとに利用者がどの嗜好パターン (どのようなアイテムを好み、どのようなアイテムを嫌うかといった傾向) にあるのかを分析し、嗜好が似ている利用者は似たようなアイテムを好み、似たようなアイテムを嫌うといった仮定の下、嗜好パターンが類似している利用者を見つけ出し、利用者が好みそうなアイテムを推薦するものである。

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二つに分類される。メモリベース法はその中でもさらに利用者間型メモリベース法とアイテム間型メモリベース法の二つに分類される。情報推薦の全体像は図 n のようになっている。

以下では協調フィルタリングにおけるメモリベース法とモデルベース法の二つの分類について説明し、さらにメモリベース法の中の利用者間型メモリベース法とアイテム間型メモリベース法について説明を記述する。

メモリベース法とモデルベース法

メモリベースは利用者がシステムを使用する以前は特に何もせず、ただ利用者データ

ベースを保持しているだけである。利用者がシステムを使用する際に、利用者データベースを参照し嗜好パターンを読み取り、それに併せた情報を推薦する。メモリベースはシステムが利用されるごとにデータベースを参照するのでデータベースの削除や追加といった変化について柔軟に対応できるメリットがある一方で、データベースを逐次的に参照しているのでモデルベース法に比べて推薦時間を有することになる。

モデルベース法は利用者がシステムを使用する以前にあらかじめ「Aさんの好むものはBさんも好む」といった嗜好パターンをモデルとして構築する。そして、システムを利用するときに利用者データベースではなく、構築しておいたモデルに基づいて情報を推薦する。モデルベース法はシステムが利用される前からモデルを構築するので推薦時間を短くできるメリットがある一方で、データベースに変更を加えると一からモデルの構築を行う必要があるためデータベースの変化に柔軟に対応できないことがある。

利用者間型メモリベース法とアイテム間型メモリベース法

取得科目の推薦と教材の最適化

§ 3.1 協調フィルタリングからの科目の推薦

過去の教学データに対して協調フィルタリング, その中でもユーザーベース協調フィルタリング (UBCF) を適用し, 学生がまだ履修していない科目についての予測評価値を算出する.

ユーザーベース協調フィルタリング

UBCF ではユーザー×アイテムの評価行列から対象となるユーザーと他のユーザーとの類似度を計算し, 対象のユーザーのまだ知らない (評価していない) アイテムについてどのくらいそのアイテムを好むかの嗜好の予測を行う. ユーザー間の類似度は共通して評価しているアイテムについての Pearson 相関で算出し, 式 (3.1) で定義する.

$$\rho_{ax} = \frac{\sum_{y \in \mathcal{Y}_{ax}} (r_{ay} - \bar{r}_a')(r_{xy} - \bar{r}_x')}{\sqrt{\sum_{y \in \mathcal{Y}_{ax}} (r_{ay} - \bar{r}_a')^2} \sqrt{\sum_{y \in \mathcal{Y}_{ax}} (r_{xy} - \bar{r}_x')^2}} \quad (3.1)$$

ただし, $\mathcal{Y}_{ax} = \mathcal{Y}_a \cap \mathcal{Y}_x$ であり, $\bar{r}_x' = \frac{r_{xy}}{|\mathcal{Y}|}$ である. しかしこの時に対象ユーザー a と他のユーザー x が互いに共通して評価したアイテムが一つ以下である場合, Pearson 相関は計算できない. そのためこのような場合は $\rho_{ax} = 0$ として互いの類似度がまったくない状態にして Pearson 相関を算出する. 互いに共通して評価したアイテムが一つ以下ということは, 互いにほとんど異なったアイテムについてのみ興味を示しているということなので, ここで互いの類似度を 0 としてしまってもあまり大きな問題ではない. 未評価のアイテムに対する予測評価値は式 (3.1) の類似度で重み付けした対象ではない他のユーザーのあいてむ y への評価値の加重平均で予測を行い, 式 (3.2) で算出される.

$$\hat{r}_{ay} = \bar{r}_a + \frac{\sum_{x \in \mathcal{X}_y} \rho_{ax} (r_{xy} - \bar{r}_x')}{\sum_{x \in \mathcal{X}_y} |\rho_{ax}|} \quad (3.2)$$

この時, 対象ユーザーが既にアイテム y を評価済みである場合は \hat{r}_{ay} の予測評価値は算出する必要はない. 式 (3.2) の第一項は, 第二項が中間的な評価を取る特徴を補正するためのバイアス項であり, 第二項は \mathcal{X}_y の集合が大きくなると式 (3.2) の分子が大きくなってしまい加重平均全体が大きくなりやすい問題があるのでそれを補正するための正規化項である.

本研究では、UBCF におけるユーザー = 学生、アイテム = 学生が履修することのできる科目、評価値 = 各科目に対する学生の成績と置き換えて UBCF を実装する。UBCF の「同じアイテムに対して高い評価をしているユーザー同士は未評価のアイテムに関してもどちらかが良い評価を下していればもう片方は良い評価を下す」という考えを教学データに対して適用して、「同じ科目で高い成績を修めている学生同士は、互いにまだ履修していない科目でも片方が良い成績を修めていれば、もう片方も良い成績を修めることができる」という考えをして UBCF を実装する。

§ 3.2 シラバスからの教材作成

シラバスとは大学における講義内容や授業のスケジュールをまとめた資料である。シラバスには各教科ごとに開講学期、単位数、単位区分、教育目標、授業概要、学生の到達目標、授業計画授業に関連するキーワード、成績評価基準、授業の際の参考資料、履修条件が細かく記載されているので、学生がその授業の単位履修を考える際は、必ずと言っていいほど参考にする資料である。このシラバスは入学時に冊子として配布されるほかにも web 上での閲覧が可能な web シラバスが存在する。web シラバスには検索機能が設けられており、開講年度、授業科目コード、授業科目名、学則科目名、担当教員、科目区分、配当学年、キーワードで検索が可能である。

本研究では上記で述べた web シラバスからの情報をスクレイピングし、シラバスから得られた情報について再びスクレイピングを行い、その結果をユーザーである学生に提示する。このシラバスからユーザーに提示する情報を教材と呼び、学生はこの教材を利用し自身の成績の向上を図ってもらう。以下では教材作成までの流れの説明を行う。

まず、シラバスから各授業ごとの「授業計画」についてスクレイピングを行う。この授業計画にはそれぞれの授業の第一回目から第十五回目までの講義内容がどのようなになっているかが書かれている。それらをスクレイピングし CSV ファイルに出力を行う。次に出力された csv ファイルを読み込み、第一回目から第十五回目までの授業計画を検索ボックスに入力し、スクレイピングを行う。スクレイピングが完了すると授業回数ごとの関連したホームページの URL とホームページのタイトルが出力されるのでそれを教材として保存する。シラバスから出力された講義内容を図に示す。また、その講義内容からスクレイピングした内容の一部を示す。

シラバスからの教材作成の際には、Python での Web スクレイピングを行う。Python でスクレイピングを行うには Google Chrome や Firefox などのブラウザの操作を自動化できる Selenium と HTML や XML 内を解析して目的としているコンテンツを抽出できる BeautifulSoup4 を使用する。

Selenium

Selenium は元々、Web アプリケーションの UI テストや JavaScript のテストの目的で開発されていたが、現在では、テスト以外にもタスクの自動化や Web サイトのクロールなど様々な用途で利用されている。

Beautifulsoup4

Beautifulsoup に HTML を渡すと、その HTML 内をツリー構造で表現したオブジェクト

トを生成する．そのオブジェクトに対して自分が必要とする情報を検索して情報の抽出を行う．Beautifulsoup4 以前のバージョンは Python3 系には対応していないので今回は Python3 系に対応している BeautifulSoup4 を使用する．

§ 3.3 GPA 向上のための教材の提示

協調フィルタリングにより予測評価値が算出された科目はシステムを使用するユーザーに対して推薦される．しかし，推薦された科目だけを履修していても GPA 向上にはあまりおおきな影響は及ぼさない．そこで本研究では 3.2 章において Web シラバスをもとに作成した教材を科目と一緒に推薦し，学生にそれらの教材で学習してもらうことによってさらなる GPA の向上を図る．甲南大学で行われた，a. 教員に質問，b. ディスカッション，c. 予習・復習，d. 発表の準備，e. 卒業論文，f. 期末テスト，g. ノートに書き写す，h. 授業中の私語，i. 遅刻・欠席九つの要因のうちどの変数が GPA に影響を与えているかという実験がある．その結果，ダミーデータのモデル 3 において c と g の要因が GPA にプラスの影響をもたらしていることがわかった．以上の結果から授業の予習・復習は GPA 向上につながるのので，科目を推薦すると同時に教材を推薦することは大いに GPA に影響があるといえる．

提案手法

§ 4.1 予測評価値からの適切な授業科目の推薦

3.1 章での UBCF の予測評価値をもとに学生に対して科目の推薦を行う。本研究の目的は学生になるべく高い GPA を取得してもらい、就職・進学に有利を進めてもらうというもので、予測評価値 (予測成績の値) が高い科目を優先して学生に推薦を行う。具体的には、予測成績の値が 3 以上、つまりレターグレードが A 以上と予測される科目を優先する。

大学において卒業要件単位というものが存在する。卒業要件単位とはその名の通り卒業するのに必要な単位数のことである。これらの卒業要件単位を満たさないとたとえどれだけ多くの単位を取得していても卒業することはできない。富山県立大学もその例外はなく、卒業要件単位が存在し卒業要件単位を満たさなければ卒業できない。しかし、卒業要件単位を満たしていても卒業できない場合がある。それは必修科目を取得できていないときである。必修科目とは必ず単位を取得しなければいけない科目のことである。卒業要件単位を満たしていてもこの必修科目を取得できていなければ卒業ということにはならない。卒業するための取得単位の制約は他にもあり、選択必修科目と呼ばれるものがある。選択必修科目は、その単位自他を必ず取得する必要はないが、指定された単位のうちいくつかの単位を取得する必要がある単位である。もちろんこの選択必修単位が足りなければ卒業には届かない。

このように大学には、卒業を考えた際に注意すべき履修条件が膨大であり、予測成績が高い単位だけを取得しているだけでは卒業できない。そこで本研究ではこれらの卒業要件単位、必修科目、選択必修科目のすべてを考慮した上で予測評価値が最も高くなるように学生に科目を推薦する。つまりはここでいう適切とは、「予測成績が高く、卒業要件単位、必修科目、選択必修科目のすべてを満たす」ということである。

§ 4.2 関連資料のアップデート

3.3 章で述べたように科目の推薦と同時に 3.2 章で作成した教材を提供する。しかし、3.2 章で作成した教材では Web ページで検索された上位の結果がそのまま、おすすめの教材として学生に表示されてしまう。そのままの状態が表示されるとたとえ全く科目と関係の無い内容が混ざっていてもそれがおすすめの科目として推薦されてしまう。そこで学生の評価によってためになる教材とそうでない教材がわかるようにするシステムを実装する。

§ 4.3 提案手法のアルゴリズム

提案手法のアルゴリズムを示す.

[Step1] 教学データと教材データの蓄積

学生がシステムを使用する前に、学生×科目の行列を数年分、あらかじめ蓄積しておきこれらをデータベースとして扱う. しかし、2.1 章でも述べたように学生の教学データは個人情報として扱われるので、デモデータを作成し CSV ファイルに保存したものをデータベースとする. また、3.2 章で述べた教材もあらかじめ作成しておき CSV ファイルで保存しておく.

[Step2] 協調フィルタリングによる予測成績の予測

学生にはシステムを使用する際に学籍番号を入力してもらう. そこで指定された学籍番号の科目情報を抽出する. 抽出した学生データを既に卒業した学生データと結合し、指定した学籍番号をターゲットとして Pearson 相関で卒業済みの学生との類似度を計算する. 類似度が計算できたら類似度が高い順にソートを行い、類似度の高い上位 10 名を抽出する. 抽出した 10 名とターゲットにした学生の科目の成績の加重平均を求めることで予測成績を導出する.

[Step3] 適切な科目の選択と推薦

すべての科目について予測成績が導出されたら、予測評価値が高い順にソートを行う. 次に必修科目を推薦科目一覧に追加する. 続いて、選択必修単位数を満たすように選択必修科目の中で予測評価値が高いものから推薦科目一覧に追加していく. 最後に卒業要件単位を満たすようにまだ推薦科目一覧に入っていない科目の中で予測評価値が高いものから推薦科目一覧追加を行い、最後に推薦科目一覧を HTML に表示し学生が閲覧できるようにする. この時、学生には卒業研究に集中してもらうべく、なるべくすべての単位を 3 年の後期までに推薦する. 4 年の後期以降は落単した単位のみを自発的に履修してもらう.

[Step4] 推薦科目の教材提供

Step3 で推薦した科目に関する教材を提供する. 提供する教材は Web ページと Youtube の動画の 2 種類である. 推薦した科目名をクリックするとその科目の教材を見ることが可能である. 教材は、講義の第一回から第十五回まであり、それぞれの回ごとに Web ページと Youtube の動画の 5 つずつを HTML に表示する.

[Step5] 教材の更新

数値実験並びに考察

§ 5.1 数値実験の概要

本研究の協調フィルタリングによる最適な科目の推薦が行えているかの確認を行う。協調フィルタリングで情報推薦を行ってもパーソナライゼーションができておらず、同じ内容ばかりを推薦しては推薦システムとして意味のないものになってしまう

そこで一つ目の実験として、まったく同じ科目を履修している学生二名について実験を行う。学生AとBは1年の後期の講義までをまったく同じ単位を履修している。しかし、二人の成績はそれぞれ異なり、取得した単位は同じであるが成績が異なるといった状況を考える。この時の学生AとBに対する推薦科目を確認し推薦内容がしっかりとパーソナライゼーションできているかを確認する。

実験方法は単純なものであり、取得成績が同じであるが成績の異なる二人の学生のデータを準備する。それらについて協調フィルタリングを行い、推薦科目を決定する。その推薦された科目を確認し二人の推薦結果がどれくらいの割合で被覆しているかを5回測定し、その平均を算出し確認を行う。

§ 5.2 実験結果と考察

結果を図に示す。結果としては2年前期の推薦科目の被覆率は53%、2年後期は68%、3年前期は50%、3年後期は52%という結果になった。

考察としてはほとんどの学期で被覆率は50%となっており、全体で見ても56%ほどの被覆率となっているので、たとえ同じ科目を履修したとしてもしっかりとパーソナライゼーションができている結果となった。

おわりに

謝辞

本研究を遂行するにあたり，多大なご指導と終始懇切丁寧なご鞭撻を賜った富山県立大学電子・情報工学科情報基盤工学講座の奥原浩之教授，António Oliveira Nzinga René 講師に深甚な謝意を表します．最後になりましたが，多大な協力をしていただいた研究室の同輩諸氏に感謝致します．

2022 年 2 月

滝沢 光介

参考文献

- [1] “M2B(みつば) 学習支援システム”, 九州大学 基幹教育院 ラーニングアナリティクスセンター
- [2] 鶴田美保子, “大学生の就職活動を成功させる要因”, 金城学院大学論集 人文科学編, 第 15 巻第 1 号, pp.109-119, Sept. 2018.
- [3] 畔津憲司, “九州市立大学経済学部 2012 年度卒業生の卒業後進路及び就職活動実態等に関する調査報告”, 北九州市立大学『商経論集』, 第 49 巻第 1・2 号, pp75-120, Dec. 2013.
- [4] 由谷真之, 森幹彦, 喜多一, “電子シラバスを用いた大学教養教育における科目選択支援”, 情報処理学会第 68 回全国大会, 4U-3, pp4-469 - 4-470, 2006.

