平成２８年度

卒業論文

深層学習を用いた
知識獲得予測を最適化する知識分類の抽出

平成２９年３月

指導教員　松尾豊 特任准教授

東京大学工学部システム創成学科知能社会システムコース

03-150984　中川大海

概要

近年，教育と IT の融合が進む中で，個人に最適化された教育を自動で提供するという「アダプティブラーニング」が注目されている．個人に最適化された学習内容の自動提供を実現するもので，アメリカを中心に注目が集まっており，関連するスタートアップや大学での研究に，多額の資金が投入されている．

学習内容を個人に最適化するという考え方は，決して新しいものではないが，一人の教師が複数の生徒に対して同時に教育する形態の現在の学校教育では，全ての生徒に対して最適な学習内容を提供するには，障壁があった．

こうした問題を教育の自動化によって解決したいという意識から，アダプティブラーニングが浸透し始めているが，その躍進には，オンライン教育サービスの普及が背景にある．

オンライン教育サービスは，サービスを利用する生徒の学習行動ログを収集することで，これまで困難であった大規模な学習効果分析を可能にしたことに加え，オンライン上の学習コンテンツを生徒が個人で利用するという形態を活用し，研究成果を元に学習コンテンツを個人に最適化して提供することを容易にした．

一方，近年，教育に限らない多くの研究領域で，深層学習が注目されている．

従来の機械学習では，人間が問題の特徴を捉えて素性を設計する必要があったが，深層学習では，目的に応じた素性を，データから自動で学習することが可能になった．既存の機械学習手法を上回る性能を得られることに加え，人間が認識できないような，データの複雑な特徴を捉えることが可能になったため，これまで人間が作り上げてきた概念を，大きく塗り替える可能性を秘めている．

こうした深層学習の技術は，オンライン教育サービスのデータを元にした学習効果の分析にも活用が期待されており，生徒の知識状態を正しく把握することで，最適な学習コンテンツを特定することを目的とする知識獲得予測の研究に，深層学習を適用した例も報告されている．

しかし，こうした知識獲得予測の研究においては，予測アルゴリズムの部分に深層学習を適用しているものの，素性となる「知識」は，事前に人間が作成した知識分類によって定義されており，人間が設計した素性を利用する旧来の状況から脱していないのが現状で

ある．

データから特徴を自動で学習できる深層学習を活用すれば，人間が認識できないような，知識獲得の過程を反映した知識分類を学習できる可能性は高く，生徒の学習効率を最適化するという最終的な目標を，真に達成するには，知識分類自体も深層学習によって最適化される必要があるといえる．

本研究では，現在の知識獲得予測に用いられている，人間が作成した知識分類は，人間の複雑な知識獲得過程を表現する上では最適化されていない，という仮定に立ち，知識獲得予測を行う上で最適な知識分類を，深層学習に自動的に抽出させることを目的とする．

実験の結果，深層学習が抽出した知識分類を用いることで，人間が作成した既存の知識分類を用いる場合よりも，高い精度で知識獲得を予測できることが検証され，深層学習によって，知識獲得を予測する上で最適化された知識分類を抽出できることが示された．

この結果は，人間が認識できない，知識獲得過程に潜む複雑な知識構造を，深層学習が獲得したことを示しており，抽出された知識分類を活用することで，より最適化された学習内容を生徒に提供できる可能性を示唆している．

さらに，研究の拡張として，本研究で用いた分析手法の教育学での適用可能性や，教育学以外の分野への応用可能性を考察した．

本研究は．オンライン教育サービスの普及や，教育分野における大規模分析の活発化，深層学習の躍進など，ここ数年の多様な領域の進展によって初めて可能になったものである．本研究が，既存の学問体系の再構築，そして人間の学習や知識の解明につながると信じている．

目 次

図 目 次

表 目 次

第1章　序論

本章では，本論文の背景，研究目的および本論文の構成について述べる．

1.1　研究の背景

1.1.1　教育の個人最適化の重要性

近年，教育と IT の融合が進む中で,「アダプティブラーニング」という言葉が注目されている．個人に最適化された学習内容の自動提供を実現するもので，そのシャキ的影響の大きさからアメリカを中心に注目が集まっており，関連するスタートアップや大学での研究に，多額の資金が投入されている [Piccioli, 2014].

学習内容を個人に最適化するという考え方は，決して新しいものではない．現在の学校教育では，一人の教師が，複数の生徒に対して同時に教育する形態が一般的であるが，学習の速度や教科による得手不得手は人それぞれであり，同じ教育を施しても，十分な理解ができず，つまずいてしまう生徒もいる．そのため，習熟が遅れている生徒に補習を行い，つまずいている原因を解明して克服する手助けをするように，生徒の学習状態を考慮して学習内容を設計する試みは，常に行われてきており，そうした指導の調整が巧みな教師は「腕のいい」教師として評価されてきた．

しかしこうした方法では，習熟の遅い生徒を援助することに重きが置かれるため，習熟が周りより早い生徒への対応は後回しにされることが多く，発展的な学習機会や知的好奇心の向上を妨げることに加え，現実的な時間と労力を考慮すると，全ての生徒に個別に対応することは困難である．

より個別化された教育を受ける手段として，個別指導形式の塾や家庭教師，通信教育なども利用される．生徒一人一人に教師がつき，生徒の習熟度合いを考慮して教育を設計できるため，習熟の早い生徒も発展的な内容を学習することが可能で，また，誰かが優先された誰かが後回しにされるということもなく，学習内容を最適化するという目的の上では，より望ましい．

しかし，このように，教育の粒度を細かくし，個人最適化を図ろうとするほど，教師一人あたりが担当できる生徒の数が減ることによる人材的・金銭的負担や，教師ごとの指導能力の違いなどの問題に直面する．結局，このような教師のマンパワーに依存した方法では，誰もが平等に最適な教育を受けるという目的を達成するには，障壁が残る．

1.1.2　オンライン教育サービスの普及と学習効果分析の発展

こうした問題意識から，最適な学習内容を自動で提供することを目的としてアダプティブラーニングの考え方が登場したが，その原動力となっているのが，オンライン教育サービスの普及である．

オンライン教育サービスとは，従来の学校の教室で，一人の教師が複数の生徒に対して同時に教育する形態と異なり，PC やモバイル端末を通じて，オンライン上で提供される学習コンテンツを，生徒が各自で利用するサービスを指す．

オンライン教育サービスの一つである Massive Open Online Courses(MOOCs) [McAuley et al., 2010, Pappano, 2012, Siemens, 2013] は，多様な分野や難易度の講義から，時間や場所を問わずに，生徒が自分のペースで学習したいものを選択して学習できるというもので，従来の教育が抱える，全ての生徒が，自身の習熟度合いに沿った教育を受けられないという問題を解決するものとして，活用が期待されている．例えば，世界最大級の MOOCs の一つである Coursera[1]は，2017 年 1 月の時点で，29 の国にまたがる 148 の教育機関とパートナーシップを結び，コンピュータサイエンス，数学や論理，社会科学などに関する 1600 以上の講座を，2200 万人以上に提供している．日本では 2013 年 2 月に東京大学が Coursera に，2013 年 5 月に京都大学が edX[2]に参加を表明したことから普及し，2013 年 11 月には日本版の MOOCs として JMOOC[3]が設立されるなど，国内外で MOOCs の利用が拡大している．

多様な講座を多くの人に提供する MOOCs 以外にも，より個人の学習過程をサポートすることを目的として設計された，Intelligent Tutoring System(ITS) と呼ばれるオンライン自動学習支援システムの利用も拡大している．世界最大級の ITS である Knewton[4]では，生徒の学力や理解度と，学ぶべき対象をマッピングすることで、生徒に最適な学習過程を設計し，かつ生徒の学習の進捗に応じてその過程を動的に変化させる仕組みを有している [Upbin, 2012].

[1] https://www.coursera.org/
[2] https://www.edx.org/
[3] https://www.jmooc.jp/
[4] https://www.knewton.com/

近年では ITS と MOOCs の融合も進んでおり [Aleven et al., 2015]，オンライン教育サービスの利用が世界中で拡大している．

日本でも，生徒が自宅でオンライン教育サービスを用いて知識を学び，学校ではより参加型のディスカッションを行うという「反転学習」の試みが提唱されており，オンライン教育サービスが社会に与える影響は今後さらに大きくなっていくといえる [Lage et al., 2000, 重田勝介, 2014]．

さらに，オンライン教育サービスは，新たな学習形態を提供するのみにとどまらず，これまで困難であった，大規模な学習効果分析を可能にするプラットフォームとして期待されている．

オンライン上で提供された講義を生徒が学習する際に，その学習行動ログをデータとして蓄積することが可能なため，そうして蓄積された多様な学習者の大規模な学習行動ログから，多様な学習効果の分析が可能になった．特に，演習問題の回答ログは，その問題が問う知識を学習者が獲得しているか否かを表すため，知識獲得の分析に利用できる [Corbett and Anderson, 1994]．例えば，生徒の問題回答ログを利用して知識獲得の予測を行った研究 [MacHardy and Pardos, 2015] は，有名な MOOCs の一つである Khan Academy[5]に蓄積された 100 万件以上の問題回答ログを使用しており，教育の分野における大規模データを適用した分析の一つである．

生徒が個人で利用するというオンライン教育サービスの特性によって，このような研究成果を元に，学習コンテンツを個人に最適化して提供することが容易になったため，学習の個人最適化を進める運動は，急速に活発化している．

1.1.3　深層学習の躍進

一方，近年，教育に限らない多くの研究領域で，深層学習が注目されている．

従来の機械学習では，人間が問題の特徴を捉えて素性を設計する必要があったが，深層学習では，目的に応じた素性を，データから自動で学習することが可能になり，画像認識 [Schroff et al., 2015, Szegedy et al., 2014]，音声認識 [Hinton et al., 2012, Bahdanau et al., 2015]，機械翻訳 [Sutskever et al., 2014, Dong et al., 2015] 等，多様な研究領域で飛躍的な進展が報告がされている．

特に，直近の一年間だけでも画像から動画を生成する研究 [Vondrick et al., 2016] や，会話を人間と同程度に認識できるとする音声認識の研究 [Xiong et al., 2016a]，一部の欧米

[5] https://www.khanacademy.org/

言語間の文レベルで，ほぼ人間と同等に正確な翻訳を実現したとする機械翻訳の研究 [Wu et al., 2016] なども報告されており，深層学習によって，日々驚異的な成果が生み出されている．

また，2016 年 3 月に人間のプロを倒したことで一躍有名になった，Google Deep Mind が開発したコンピュータ囲碁プログラムの「AlphaGo」[Silver et al., 2016] は，過去の人間が打った大量の棋譜に深層学習を適用した後，コンピュータ同士の対局による強化学習を通して，今後１０年は不可能と言われていた，人間のプロを打ち負かすほどの棋力を獲得した．AlphaGo は，過去の対極である棋譜の分析によって人間を真似ただけでなく，それまで人間が考えつかなかったような手を学習しており，囲碁界に衝撃を与えている．このように，深層学習は，人間が認識できないようなデータの複雑な特徴を捉えることで，これまで人間が作り上げてきた概念を，大きく塗り替える可能性を秘めている．

こうした深層学習の技術は，オンライン教育サービスに蓄積されたデータを元にした，学習効果の分析にも活用が期待されている．

特に，生徒の知識状態をモデリングし，知識獲得を予測する Knowledge Tracing の研究は，生徒の知識状態を正しく把握することで，最適な学習内容を特定することにつながるため，以前から研究が盛んだったが，深層学習の適用により大きく進展した．Piech らが発表した，Knowledge Tracing に深層学習を活用する Deep Knowledge Tracing という手法では，時系列分析でよく用いられる深層学習モデルである Recurrent Neural Networks [Williams and Zipser, 1989] を活用することで，高い性能で知識獲得を予測できること，また，予測モデルを分析することで知識間関係をネットワークとして抽出できることが報告されている [Piech et al., 2015].

しかし，こうした知識獲得予測に深層学習を適用する研究においては，ある問題が存在する．知識獲得を予測するアルゴリズムの部分には深層学習を適用しているものの，素性となる「知識」は，事前に人間が作成した知識分類によって定義されており，人間が設計した素性を利用する旧来の状況から脱していない．人間が作成した分類というのは,「知識の体系はこうなっているはずだ」ないしは「この体系に基づいて教えられることが望ましい」という専門家の仮説や理論に基づいて作られたものである．そのため，人間にとっての可読性は高いとしても，実際の生徒の知識獲得の過程を定量的に分析した上で，知識獲得の予測性が最適化されているという根拠はない．

今日多様な分野で成果を生んでいる，データから特徴を自動で学習できる深層学習を活用すれば，人間が認識できないような，知識獲得の過程を反映し，予測を最適化するような知識分類を学習できる可能性は高く，生徒の学習効率を最適化するという最終的な目標

を真に達成するには，知識分類自体も深層学習によって最適化される必要があるといえる．

以上の問題意識に基づいた，既存研究との本研究の差分のイメージを図 1.1 に示す．

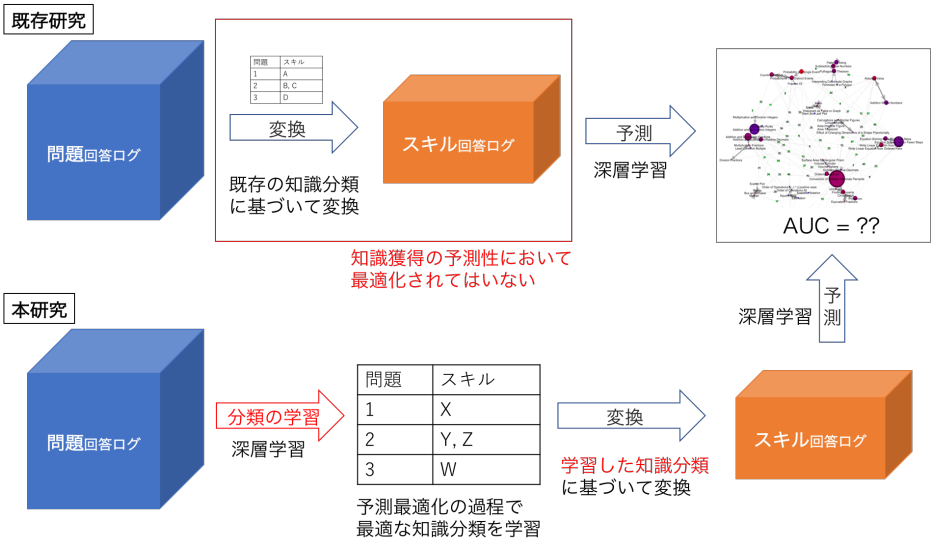

図 1.1: 既存研究と本研究の差分のイメージ

以上，本研究の背景について述べた．次に，上記の背景を踏まえた研究目的について説明する．

1.2　研究目的

本研究では，現在の知識獲得予測に用いられている，専門家が作成した知識分類は，人間の複雑な知識獲得過程を表現する上では最適化されていない，という仮定に立ち，下記を検証することを目的とする．

- 深層学習によって抽出した知識分類を用いることで，専門家が作成した知識分類を用いる場合よりも，高い精度で知識獲得予測を行うことができる．

本論文では，専門家による事前の知識分類を所与のものとせずに，問題の回答ログデータのみを深層学習に適用し，回答正誤予測を行う過程で，その予測性を最適化する知識分類を抽出することを目的としている．

従来，専門家が事前に分類することが必要であった生のデータから，予測性において最適な分類を深層学習によって自動で抽出できることを明らかにすることは，既存の学問体

系やカリキュラム設計の再構築につながるだけでなく，人間の知識状態の推移するメカニズムを解明することにもつながり，学術的な意義が大きいと考える．

以上，本研究の目的について述べた．次に，背景と目的を踏まえて本論文の構成について述べる．

1.3　本論文の構成

以降の本論文の構成について述べる．

2 章では，関連研究について述べる．

知識獲得予測の関連研究を俯瞰し，周辺概念を整理することで，本研究の学術的位置づけを明確にする．

3 章では，分析手法について述べる．生徒の知識獲得を予測する過程で知識分類を抽出し，また，抽出された知識分類を分析する手法について説明する．まず，分析手法全体の流れを概説し，手法全体が 3 つの要素から構成されることを述べる．

4 章では，実験で用いるデータセットについて述べる．3 章で述べたデータセットとしての要件を満たす，オンライン教育サービスにおける生徒の学習回答ログである，2 つのデータセットを紹介し，本研究に適用するための事前の処理を説明する．

5 章では，実験について述べる．3 つのブロックに分けられる実験について，各実験の設定を述べた後，実験結果について述べる．

実験結果においては，提案手法によって抽出された知識分類を用いることで，既存の知識分類を用いた場合よりも高い精度で知識獲得を予測できることを確認し，予測性能が最適化された知識分類が，Deep Knoeldge Tracing の過程で抽出されたことを定量的に示す．

さらに，抽出された知識分類について，既存の知識分類との比較や，デートセットごとの比較をすることで，定量的・定性的に解釈する．

6 章では，実験結果を踏まえた考察を述べる．

まず，本研究で用いた手法や抽出された知識分類について，実験結果を踏まえて，その性質を考察し，実際の教育現場への適用について述べる．

また，本研究で用いた分析手法の，多様なデータへの適用可能性について議論し，教科によらず知識構造を分析できる可能性があること，教科によって抽出される知識分類の性質が異なる可能性があること，一方で，複合的な学問や専門性の高い学問については，適用可能性の検証実験が必要であることを述べる．さらに，本研究で用いた分析手法が，教育学以外の分野にも応用できる可能性を持つことを論じる．

最後に，7 章で結論を述べる．

以上，序論について述べた．次に，先行研究について述べる．

第2章　関連研究

本研究が問題提起を行った知識獲得予測の研究は，今日の教育を取り巻く環境や，教育における大規模分析の活発化，深層学習技術やその他の多様な分析技術の進展により発展してきた研究分野である．

本章では，知識獲得予測の関連研究を俯瞰し，現状の環境や周辺概念を整理することで，本研究の学術的位置づけを明確にする．

まず，教育の個人最適化に関する現状や研究について整理した後，教育の個人最適化を解決するプラットフォームとして期待されているオンライン教育サービスについて，具体的な事例を挙げながら，その効果や問題点，関連する研究について述べる．次に，深層学習について概説し，本論文との関わりが深い Recurrent Neural Netoworks について詳細に述べる．さらに，知識獲得の予測手法である Knowledge Tracing について，その有益性や，伝統的な手法，深層学習を用いた最先端の手法について整理した後，大規模データから次元削減を行う関連手法について整理する．

最後に，以上の関連研究を踏まえて，本論文で使用する類似の用語について定義を明確にする．

2.1　教育の個人最適化の現状

教育の個人最適化の重要性と問題について，現状の教育システムや関連する研究を踏まえて述べる．(個人最適化のこれまでのアナログな研究・教育学的アプローチ，つまずきのサポート・習熟の早い生徒のモチベーションアップ，マンパワー依存の限界等について執筆予定)

2.2　オンライン教育サービスと大規模分析

オンライン教育サービスの代表的な例として Massive Open Online Courses(MOOCs) と Intelligent Tutoring System(ITS) を取り上げ，具体的な事例を挙げながら，関連する

研究について述べる．また，こうしたオンライン教育サービスが大規模分析に活用されている状況や研究について整理する．

2.2.1 Massive Open Online Courses

MOOCs は Massive Open Online Courses [McAuley et al., 2010, Pappano, 2012, Siemens, 2013] の略称で，特に日本語で表記する場合は大規模公開オンライン講座と記述することがある．MOOCs は，オンライン上で公開された，大学を始めとする様々な教育機関などの講座を，誰もが無償で受講でき，また修了時には修了証も取得できる教育サービスのことを指す．

学びたい人が，いつでもどこでも学習リソースにアクセスできるという MOOCs の概念自体は古くから提唱されていたが，実現化したのは，2008 年にカナダのマニトバ大学で学生向けのオンライン講座を開設した際に，25 人の受講者だけでなく 2000 人以上の人がその講座に参加したことがきっかけだと言われている [Yuan et al., 2013]．

以前から，大学などの高等教育機関は，オープンコースウェア [Abelson, 2008] という形で講義の動画や資料を公開していたが，MOOCs は，参加人数が非常に大規模で，また，高等教育水準の内容だけでなく，初等中等教育水準の内容の講座も含まれている点で異なる．また，これまでもオンラインの講座というものは存在していたが，MOOCs は，参加人数が非常に大規模である点や公開している講座の数が大規模である点，また，その内容が多様であるという点，利用が無料，あるいは無料に近いという点において，これまでのオンライン講座とは異なる．

MOOCs は，従来の，学校の教室で一斉授業形式で提供される教育形態と異なり，オンライン上の多様な講座に，生徒がアクセスし，講座ごとに提供される講義の動画や演習システムなどを通じて，いつでもどこでも，自身の習熟度合いやペースに合わせて，自分の学習したいものを選択肢して学習できる．従来の教育の，生徒が自身の習熟度合いに見合った学習ができないという問題を解決するものとして注目されていることに加え，産業や社会への影響も注目されている．例えば，大学生だけでなく社会人も，自身の専門領域に関する講座を受講することでより理解を深めたり，あるいは専門領域とは異なる幅広い講座を受講することで，教養を養うことができる．また，公教育の整備が追いついていないような発展途上国においては，MOOCs が教育に与える影響は大きく，その影響や可能性を分析する報告は多い [Trucano et al., 2013, Liyanagunawardena et al., 2013]．

このように，MOOCs は社会の多様な場面で，これまでにない学習機会を提供しており，

教育や学習といったもののあり方に大きな影響を与えている．

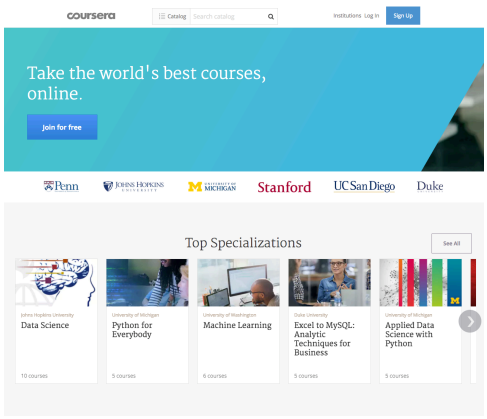

図 2.1: Coursera のイメージ

図 2.2: JMOOC のイメージ

MOOCs の有名な事例として，世界的に有名な Coursera や，日本発の MOOCs である JMOOC が挙げられる．Coursera と JMOOC のイメージを図 2.1，2.2 に示す．Coursera は，2017 年 1 月の時点で，29 の国にまたがる 148 の教育機関とパートナーシップを結び，コンピュータサイエンス，数学や論理，社会科学などに関する 1600 以上の講座を，2200 万人以上に提供している[1]．JMOOC は，2013 年 11 月に日本版の MOOCs として設立され，10 代から 80 代までと幅広い年代に，アートや医療，自然科学や資格試験対策などの講座を提供しており，2017 年 1 月の時点で，140 の講座を 50 万人以上が受講している[2]．

2.2.2　Intelligent Tutoring System

多様な講座を多くの人に提供する MOOCs 以外にも，より個人の学習過程をサポートすることを目的として設計された，Intelligent Tutoring System(ITS) と呼ばれるオンライン自動学習支援システムの利用も拡大している．

ITS の有名な事例として，世界最大級の ITS である Knewton[3]のイメージを図 2.3 に示す．Knewton では，生徒の学力や理解度と，学ぶべき対象をマッピングすることで、生徒に最適な学習過程を設計し，かつ生徒の学習の進捗に応じてその過程を動的に変化させる仕組みを有している [Upbin, 2012]．

また，近年では，これまで難しいと言われていた ITS の MOOCs への埋め込みを達成したとする研究 [Aleven et al., 2015] も報告されており，今後より ITS の利用は拡大して

[1]講座数と利用者数はトップページの記載より引用．
[2]講座数と利用者数はトップページの記載より引用．
[3]https://www.knewton.com/

図 2.3: Knewton のイメージ

いくといえる.

2.2.3　学習行動ログの蓄積と大規模分析の活発化

こうした MOOCs や ITS を始めとするオンライン教育サービスは，人々に新たな学習の機会を提供するという側面だけでなく，これまで難しかった大規模な学習効果分析の可能性を高めるという側面もある.

生徒はオンライン上で提供された講義動画や演習問題を通して学習するが，オンライン上で実施されているため学習行動ログをデータとして蓄積することができ，さらに，そのデータを分析に活用することができる．多様な生徒が利用するため，多様な生徒の大規模な学習行動ログから多様な講座の学習効果の分析が可能となりつつある.

特に，演習問題の回答ログはその演習問題により評価される知識を生徒が獲得しているか否かを表現しているため，知識獲得の分析に利用できる．例えば，MOOCs の演習問題の回答ログを利用して知識獲得の予測を行う研究 [MacHardy and Pardos, 2015] では，世界的に有名な MOOCs である Khan Academy から収集したデータを利用していたが，その問題回答ログ数は 100 万件以上であり，これまでにないほど大規模なデータを対象に分析が実施されたといえる.

2.2.4　個人が利用するプラットフォームとしての性質

さらに，こうしたオンライン教育サービスが学習効果分析を行う価値を大きく高めている要因として，オンライン上のコンテンツを，多様な生徒が，個人で利用するというプラットフォームとしての性質がある．

現在の学校教育の形態では，生徒の学習効果に関する分析を行い，なんらかの知見を得たとしても，それを多様な生徒に適用して効果を検証したり，各個人に提供できるような環境が整備されておらず，学習効果分析が社会に与える影響が限定的であった．

また，従来の e ラーニングによる学習支援システムも，大学のような各教育機関が個別に設定し，学内の生徒が利用者の中心であったため，システムの利用者が限定されており，データの多様性や研究成果の活用可能性も狭い範囲に留まっていた．

一方，MOOCs や ITS のようなオンライン教育サービスは，大学のような教育機関の垣根にとらわれず，多様な背景，適性，能力を持つ生徒が利用していることに加え，学習コンテンツを個人が利用する形態のため，多様なデータを元に得られた一般性のある知見を，多様な生徒に対して，生徒個人の粒度で提供することが可能である．そのため，知識獲得予測を始めとする学習効果分析に基づいた，学習の効率化や継続を促進する教材推薦システムの開発が持つ社会的影響が大きなものとなっている．

以上，オンライン学習サービスを取り巻く環境と，知識獲得予測研究との関連性について述べた．

次に，深層学習について述べる．

2.3　深層学習

本研究において用いる技術の核となっている，深層学習について述べる．まず，深層学習を理解する必要となる周辺知識について説明した後，深層学習の概要について述べ，さらに本研究で用いる深層学習モデルである RNN について詳述する．

2.3.1　深層学習にまつわる周辺知識

深層学習について述べる前に，深層学習を理解する上で必要となる前提知識を，以下の 2 つの項目に分けて述べる．

1. 機械学習

2. ニューラルネットワーク

機械学習

（機械学習の概念，特徴量の設計，教師あり学習と教師なし学習などについて執筆予定）

ニューラルネットワーク

（ニューラルネットワークの概念，構造，活性化関数，最適化，誤差逆伝搬勾配法などについて執筆予定）

2.3.2　深層学習の概要

深層学習は多層のニューラルネットワークによる機械学習のことで，従来の機械学習では，人間が問題の特徴を捉えて素性を設計する必要があったが，深層学習では，目的に応じた素性を，最適化の過程でデータから自動で学習することが可能である．

深層学習の活用により，画像認識 [Schroff et al., 2015, Szegedy et al., 2014]，音声認識 [Hinton et al., 2012, Bahdanau et al., 2015]，会話認識 [Sak et al., 2015]，機械翻訳 [Sutskever et al., 2014, Dong et al., 2015]，質問応答文生成 [Yin et al., 2015]，画像説明文生成 [Xu et al., 2015, Vinyals et al., 2014] 等，多様な研究領域で飛躍的な進展が報告がされている．

特に，直近の一年間だけでも画像から動画を生成する研究 [Vondrick et al., 2016] や，会話を人間と同程度に認識できるとする音声認識の研究 [Xiong et al., 2016a]，一部の欧米言語間の文レベルで，ほぼ人間と同等に正確な翻訳を実現したとする機械翻訳の研究 [Wu et al., 2016] などを始めとする数々の報告がされており，深層学習によって，日々驚異的な成果が生み出されている．

また，2016 年 3 月に人間のプロを倒したことで一躍有名になった，Google Deep Mind が開発したコンピュータ囲碁プログラムの「AlphaGo」[Silver et al., 2016] は，過去の人間が打った大量の棋譜に深層学習を適用した後，コンピュータ同士の対局による強化学習を通して，今後１０年は不可能と言われていた，人間のプロを打ち負かすほどの棋力を獲得した．AlphaGo は，過去の対局の情報である棋譜の分析によって人間を真似ただけでなく，それまで人間が考えつかなかったような手を学習しており，囲碁界に衝撃を与えている．このように，深層学習は，人間が認識できないようなデータの複雑な特徴を捉えることで，これまで人間が作り上げてきた概念を，大きく塗り替える可能性を秘めている．

一般に，深層学習モデルを学習させる際には，大規模な訓練データが必要となる．深層学習モデルが，人の手で素性を設計していない生の訓練データから，特徴的な表現を学習し，最適化するには，膨大な数の内部パラメータを設定して学習することが必要で，ときには数十万から数百万以上の内部パラメータが設定されることもあり，こうした膨大な数のパラメータを学習するには，大規模な訓練データが必要となる．データ数が不足すると，データの潜在的な特徴を十分に学習できないことに加え，汎用性の低い特徴まで過剰に学習してしまう「過学習」に陥りやすくなる [Tetko et al., 1995]．

実際に大規模データを利用した研究の例を挙げると，人間より高い精度で人の顔を見分けらると報告する顔認識の研究 [Schroff et al., 2015] では，数百万人の 2 億枚以上の顔画像を訓練データに利用している．英語からフランス語に翻訳する機械翻訳の研究 [Xu et al., 2015] では，1200 万もの文章を訓練データとして利用している．

2.3.3 Recurreut Neural Networks

深層学習のネットワークには，目的に応じたいくつかの種類があるが，ここでは，知識獲得の予測に深層学習を用いた手法 [Piech et al., 2015] に用いられていたニューラルネットワークである，Recurreut Neural Networks [Williams and Zipser, 1989]（以下，RNN）について説明する．

RNN は深層ニューラルネットワークの一種で，主に系列データの解析に利用される．系列データとは，同質のデータを直列に並べて表現することにより，特定の意味を持ったデータのことで，例えば，時系列に沿って変化する株価のようなデータや，一定の長さで順序を持って並ぶ単語からなる文章などのデータが系列データにあたる．

近年，RNN はデータの大規模化や計算機性能の向上などにより，幅広い領域の系列データに対して適用されるようになった．具体的には，機械翻訳 [Sutskever et al., 2014, Dong et al., 2015]，手書き文字認識 [Graves and Schmidhuber, 2009, Louradour and Kermorvant, 2014]，音声認識 [Hinton et al., 2012, Bahdanau et al., 2015]，ユーザログ解析 [Hidasi et al., 2015]，画像説明文生成 [Xu et al., 2015, Vinyals et al., 2014]，医療診断 [Choi et al., 2015, Lipton et al., 2015] 等の領域で高い性能を発揮することが報告されている．

伝統的な RNN の構造は図 2.4 のように，入力層，隠れ層，出力層の 3 層から構成されている．系列方向を時刻とすれば，時刻 t の隠れ層 $\mathbf{h}_t$ の計算に時刻 $t-1$ の隠れ層の情報を入力する $\mathbf{h}_t = f(\mathbf{x}_t, \mathbf{h}_{t-1})$ の式ように，一つ前の情報を繰り返し（recurrent）入力すると

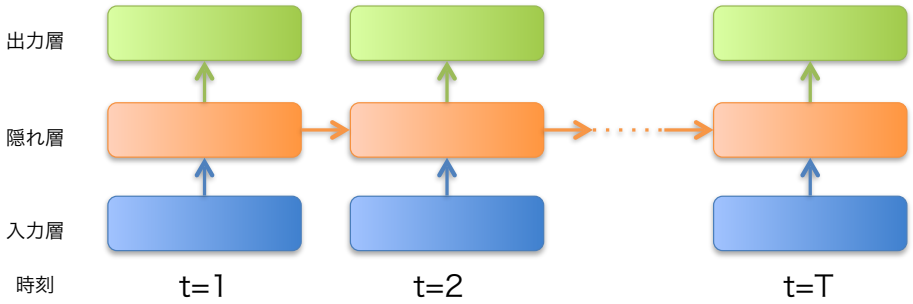

図 2.4: RNN の構造のイメージ

いう構造である．関数 f は，入力である $\mathbf{x}_t$ や $\mathbf{h}_{t-1}$ をアフィン変換[4]して足しあわせた後，活性化関数にかけるというものがよく利用される．活性化関数はシグモイド関数や tanh（Hyperbolic Tangent 関数），Relu [Nair and Hinton, 2010]，ELUs [Clevert et al., 2015] など多く提案されており，通常，非線形関数である．

このように，データの系列に沿った情報を反映して学習できる RNN だが，課題の一つとして，長期的な表現になるほど学習が難しくなるということが挙げられる [Bengio et al., 1994]．RNN の学習には，勾配法に基づいた確率的勾配降下法 [Robbins and Monro, 1951, Kushner and Yin, 2003] や Adam [Kingma and Ba, 2014]，AdaDelta [Zeiler, 2012] など，さまざまな手法が利用可能である．しかし，いずれの勾配法を用いるにせよ，勾配が爆発して学習モデルが壊れてしまうという勾配爆発 [Bengio et al., 1994, Pascanu et al., 2013] という問題や，勾配が消滅して対象データの長期的な特徴量を捉えることができないという勾配消滅 [Pascanu et al., 2013, Hochreiter, 1998] という問題がしばしば発生する．これは，$\mathbf{h}_t = f(\mathbf{x}_t, \mathbf{h}_{t-1})$ の式に表れるように同じ変換を繰り返し行うためであり，このため，特に長い系列データを RNN で学習する場合，効果的に長期的な表現を学習させることが難しい．

こうした問題を解決もしくは緩和するため，学習時の勾配に制約を加える方法やゲート付き活性化関数の利用が提案されている．まず，勾配爆発の緩和に対しては，学習時の勾配に制約を加える方法が有効である．具体的には，[Mikolov, 2012] では学習させるパラメタの勾配の絶対値の最大値を予め決めておき，最大値以上の場合には，勾配の最大値になるように勾配の値を置き換えることで勾配爆発の影響を緩和する方法が報告された．また，[Pascanu et al., 2013] では学習させるパラメタの勾配のノルムの最大値を予め決めておき，最大値以上の場合には，ノルムが最大値以下になるように疑似コード 1 に従いノ

[4]平行移動と線形変換を組み合わせた変換のこと．

ルムを抑制することで勾配爆発の影響を緩和する方法が報告された．

Algorithm 1 勾配爆発を防ぐための勾配ノルム抑制の疑似コード

$\hat{\mathbf{g}} \leftarrow \frac{\delta\varepsilon}{\delta\theta}$
if $\|\hat{\mathbf{g}}\| \geq threshold$ **then**
  $\hat{\mathbf{g}} \leftarrow \frac{threshold}{\|\hat{\mathbf{g}}\|}\hat{\mathbf{g}}$
end if

次に，勾配消滅の緩和に対しては，ゲート付き活性化関数の利用が有効である．先に，言及したが，RNN には異なる活性化関数を利用するという形でいくつかの種類がある．うまく設計された活性化関数を利用することで，勾配消滅を緩和してデータの長期的な特徴をよく捉えられたり，計算コストを削減することができたりする．以降では，よく研究報告で取り上げられる Simple RNN（以下，SRNN）[Williams and Zipser, 1989]，Long Short Term Momory RNN（以下，LSTM-RNN）[Hochreiter and Schmidhuber, 1997]，Gated Recurrent Neural Networks（以下，GRNN）[Cho et al., 2014] の 3 つについて詳細に説明する．

SRNN

SRNN はゲート付き活性化関数を用いない簡単な構造の RNN である．[Le et al., 2015, Krueger and Memisevic, 2015] で報告される工夫を取り入れることで，データの長期的な特徴を効果的に捉えることができるようになるが，多くの場合で，LSTM-RNN や GRNN のようにゲート付き活性化関数を用いる RNN の方がモデルの性能という点で優れている．

SRNN によるモデルの定式はいくつか種類が存在するが，シンプルなものは例えば下記の式で定義される．

$$\mathbf{h}_t = \tanh(\mathbf{W}_{xh}\mathbf{x}_t + \mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h) \tag{2.1}$$

$$\mathbf{y}_t = \sigma(\mathbf{W}_{hy}\mathbf{h}_t + \mathbf{b}_y) \tag{2.2}$$

ここでは，t は時刻を指し，$\mathbf{x}_t$ は時刻 t の入力ベクトルを指し，$\mathbf{h}_t$ は時刻 t の隠れ層を指し，$\mathbf{y}_t$ は時刻 $t+1$ の各問題の正誤確率の予測値を指し，$\mathbf{W}_{xh}$，$\mathbf{W}_{hh}$ はそれぞれ重み行列を指し，$\mathbf{b}_h$，$\mathbf{b}_y$ はそれぞれバイアス項を指し，tanh は $(e^x - e^{-x})/(e^x + e^{-x})$ で定義される Hyperbolic Tangent 関数を指し，σ は $1/(1+e^{-x})$ で定義されるシグモイド関数を指す．訓練時には，重み行列 $\mathbf{W}_{xh}$，$\mathbf{W}_{hh}$ とバイアス項 $\mathbf{b}_h$，$\mathbf{b}_y$ を学習する．

LSTM-RNN

LSTM-RNN は Long Short Term Memory という活性化関数を用いる RNN で，その名前の通り，SRNN では捉えることが難しかったデータの長期的表現と短期的表現の両方の獲得を目的に開発されたものである [Hochreiter and Schmidhuber, 1997]．LSTM-RNN は SRNN と比較すると，モデルの性能という点で優れているが，内部のパラメタの数が非常に大きく学習コストは大きい．最先端の成果を報告する研究でしばしば利用されているが，LSTM-RNN 自体が開発されたのは 1997 年であり LSTN-RNN が新しいというわけではない．

LSTM-RNN によるモデルの定式にはいくつか種類が存在するが，特に，後述する Deep Knowledge Tracing [Piech et al., 2015] で用いられる LSTM-RNN は下記の式で定義される．

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_i) \tag{2.3}$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_{xg}\mathbf{x}_t + \mathbf{W}_{hg}\mathbf{h}_{t-1} + \mathbf{b}_g) \tag{2.4}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_f) \tag{2.5}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_o) \tag{2.6}$$

$$\mathbf{m}_t = \mathbf{f}_t \odot \mathbf{m}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t \tag{2.7}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \mathbf{m}_t \tag{2.8}$$

$$\mathbf{y}_t = \sigma(\mathbf{W}_{my}\mathbf{m}_t + \mathbf{b}_y) \tag{2.9}$$

ここでは，$\mathbf{i}_t$ は Input Gate を指し，$\mathbf{f}_t$ は Forget Gate を指し，$\mathbf{g}_t$ はメモリセルへの入力を指し，$\mathbf{o}_t$ は Output Gate を指し，$\mathbf{m}_t$ はメモリセルを指し，$\mathbf{W}_{xi}$，$\mathbf{W}_{hi}$，$\mathbf{W}_{xg}$，$\mathbf{W}_{hg}$，$\mathbf{W}_{xf}$，$\mathbf{W}_{hf}$，$\mathbf{W}_{xo}$，$\mathbf{W}_{ho}$，$\mathbf{W}_{my}$ はそれぞれ重み行列を指し，$\mathbf{b}_i$，$\mathbf{b}_g$，$\mathbf{b}_f$，$\mathbf{b}_o$，$\mathbf{b}_y$ はそれぞれバイアス項を指し，$\odot$ は要素積を指す．

式 2.7 にあるように，メモリセルへの入力は 1 つ前のメモリセルの状態 $\mathbf{m}_{t-1}$ と入力 $\mathbf{g}_t$ であり，それぞれの入力に対して，過去のメモリセルからの情報を捨てる Forget Gate と現在からの情報を調整する Input Gate を作用させ，$\mathbf{m}_t$ をえる．新しい隠れ層 $\mathbf{h}_t$ は式 2.8 のようにメモリセルからの出力を Output Gate で調整したものを入力として受け取る．これらのゲートにより，長期的な特徴の短期的な特徴が捉えられるとされている．

GRNN

GRNN は Gated Recurrent Unit [Cho et al., 2014] というゲート付き活性化関数を用いる RNN のことで，GRU は LSTM のように，長期的な表現と短期的な表現を捉えるために提案された活性化関数である．Cho ら [Cho et al., 2014] が 2014 年に発表して以来，GRNN 自体や GRNN の活用に関する研究が多く報告されている [Chung et al., 2014, Zaremba, 2015, Chung et al., 2015, Karpathy et al., 2015, Biswas et al., 2015, Pezeshki, 2015]．LSTM よりもゲートの数が少なく学習コストが小さい傾向にあるが，LSTM-RNN，GRNN の性能を比較した研究 [Chung et al., 2014, Zaremba, 2015] において LSTM-RNN と GRNN が同程度の性能であることが報告されている．

GRNN は下記の式により定義される．

$$\mathbf{r}_t = \sigma(\mathbf{W}_{xr}\mathbf{x}_t + \mathbf{W}_{hr}\mathbf{h}_{t-1} + \mathbf{b}_r) \tag{2.10}$$

$$\mathbf{z}_t = \sigma(\mathbf{W}_{xz}\mathbf{x}_t + \mathbf{W}_{hz}\mathbf{h}_{t-1} + \mathbf{b}_z) \tag{2.11}$$

$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W}_{xh}\mathbf{x}_t + \mathbf{W}_{hh}(\mathbf{r}_t \odot \mathbf{h}_{t-1} + \mathbf{b}_h)) \tag{2.12}$$

$$\mathbf{h}_t = \mathbf{z}_t \odot \mathbf{h}_{t-1} + (1 - \mathbf{z}_t) \odot \tilde{\mathbf{h}}_t \tag{2.13}$$

$$\mathbf{y}_t = \sigma(\mathbf{W}_{hy}\mathbf{h}_t + \mathbf{b}_y) \tag{2.14}$$

ここでは，$\mathbf{W}_{xr}, \mathbf{W}_{hr}, \mathbf{W}_{xz}, \mathbf{W}_{hz}, \mathbf{W}_{xh}, \mathbf{W}_{hh}$ は重み行列で，$\mathbf{b}_r, \mathbf{b}_z, \mathbf{b}_h$ はバイアス項である．$\mathbf{r}_t$ が Reset Gate(LSTM における Forget Gate に相当する機構) で，$\mathbf{z}_t$ が Update Gate(LSTM におけるメモリセルに相当する機構) である．$\mathbf{r}_t$ が 0 に近いほど前の隠れ層からの入力よりも現在の入力をより強く考慮するようになり，$\mathbf{z}_t$ が 0 に近いとほど前の隠れ層をより大きく更新するようになる．

以上，周辺知識を整理した上で深層学習について述べ，本研究と関連の深い RNN について詳述することで，技術的な前提知識を確認した．

次に，知識獲得の予測について述べる．

2.4　知識獲得の予測

知識獲得の予測は，生徒が対象の知識を獲得しているか否かを予測する問題である．通常，知識を獲得しているか否かは問題回答の正誤を基に評価されるため，知識獲得の予測のタスクは過去の生徒の問題回答履歴から次に解く問題の正誤を予測するというもので

ある．

最初の定式化の事例は，1994 年に Corbett らによって報告された Knowledge Tracing [Corbett and Anderson, 1994] である．スキルの習熟学習において，領域知識をよく分析し階層的に知識間関係を構築し，階層構造においてより水準の高い知識に着手する前に予め獲得するべき知識が確実に獲得されるように学習体験を設計することで，ほとんどの生徒がスキルを十分に習熟できるとする仮説 [Keller, 1968, Bloom, 1968] や，コンピュータサイエンスの発展を受けて，予め獲得すべき知識が確実に獲得されるように生徒の知識の獲得有無を予測するというのが主な目的であった．Knowledge Tracing における生徒とモデルは，生徒が勉強し知識を獲得したら，モデルが生徒が獲得した知識を予測することで生徒の獲得している知識の変化を追跡する（*Knowledge Tracing*），という関係になっている．

伝統的に，知識獲得の予測には知識獲得の時系列性を重視するものと，知識間の関係性を重視するものがある．[Corbett and Anderson, 1994] で報告された Bayesian Knowledge Tracing という手法は知識獲得の時系列性を重視するもので，問題に予めスキルを割り当て個々のスキルに習熟過程に関する 4 つの確率変数を定義しモデル化するというもので，スキル間の関係性は考慮しないが，個々のスキルの習熟，つまり時系列性を考慮する手法である．[Pavlik Jr et al., 2009] で報告された Performance Factor Analysis という手法は知識間の関係性を重視するもので，個々の知識（あるいは，スキル）に関する過去の回答の正誤を重み付けして，次の問題の正誤を予測しようというもので，Performance Factor Analysis は知識獲得の時系列性より知識間の関係性を重視する手法である．いずれの手法も本論文と関連が深い．

以降では，まず，Knowledge Tracing の定式化について述べ，Bayesian Knowledge Tracing と Performance Factor Analysis の 2 つの手法を説明し，最後に，深層学習を活用した Deep Knowledge Tracing について説明する．

2.4.1　Knowledge Tracing の定式化

Knowledge Tracing は過去の生徒の問題回答履歴から生徒が次に解く問題の正誤を予測するというものである．生徒の時刻 t において観測された問題回答結果を q_t とすれば，$q_1, q_2, \ldots, q_t$ から時刻 $t+1$ において観測される問題回答結果 q_{t+1} を予測するタスクと表現できる．特に，過去の観測された問題の正誤から将来の正誤確率を算出する場合は，$q_1, q_2, \ldots, q_t$ が観測された場合の時刻 $t+1$ に着手する問題において当該生徒の回答正解

となる事後確率 $p(q_{t+1} = correct|q_1, q_2, \ldots, q_t)$ を求めるタスクであるといえる．予測性能の評価は [Yudelson et al., 2013, FALAKMASIR et al., 2015] では Accuracy[5]で，[Piech et al., 2015] では AUC[6]で行っており，目的に応じてさまざまである．本研究では，AUC によって予測性能を評価する．

なお，モデルの入力次元である「問題」の粒度はさまざまである．問題はその問題を回答するのに必要な知識を生徒が獲得しているか否かを評価するという点で，問題は知識集合を表現しており，また，その粒度もさまざまである．個々の問題をそのままモデルの入力次元とするものや，問題に予めタグを割り当て問題により評価される知識の粒度をある程度整え，そのタグをモデルの入力次元とすることもある．例えば，[Piech et al., 2015] ではモデルの入力次元は演習タグもしくはスキルタグと呼ばれるものであり，演習問題に割り当てられ，それぞれの演習問題で扱われる学習要素を説明するものである．通常，こうしたタグは専門家によって設計され，利用される．本論文では個々の問題をそのままモデルの入力次元として用いる．

2.4.2　Bayesian Knowledge Tracing

Bayesian Knowledge Tracing [Corbett and Anderson, 1994]（以下，BKT）はベイズの定理の事前確率と事後確率の関係に基づいて正解確率 $p(q_{t+1} = correct|q_1, q_2, \ldots, q_t)$ をモデリングする手法である．BKT には下記の 4 つの確率変数がある．

- 初めから当該スキル理解している確率 $p(L_0)$（もしくは $p\text{-}init$）
- 生徒が当該スキルを理解していない状態から理解している状態へ遷移する確率 $p(T)$（もしくは $p\text{-}transit$）
- 生徒が当該スキルを理解しているが誤答する確率 $p(S)$（もしくは $p\text{-}slip$）
- 生徒が当該スキルを理解していないが推測で正解する確率 $p(G)$（もしくは $p\text{-}guess$）

これらの 4 つの確率変数がすべてのスキルに定義されている．つまり，スキル数を N とすれば，確率変数の合計数は $4N$ である．生徒 u がスキル k の問題を時刻 t に解いた場合に正解する確率は下記の式に基づいて更新される．

[5]正解率．予測結果全体と、答えがどれぐらい一致しているかを判断する指標．0〜1 で表され，完全な予測時に 1 となる．

[6]正例を正しく分類した割合を縦軸に，負例を正しく分類した割合を横軸に取る ROC 曲線における，曲線より下の面積．0〜1 で表され，完全な予測時に 1 となり，ランダムな予測で 0.5 付近を示す．

$$p(L_1)_u^k = p(L_0)^k \tag{2.15}$$

$$p(L_t|obs = correct)_u^k = \frac{p(L_{t-1})_u^k \cdot (1 - p(S)^k)}{p(L_{t-1})_u^k \cdot (1 - p(S)^k) + (1 - p(L_{t-1})_u^k) \cdot p(G)^k} \tag{2.16}$$

$$p(L_t|obs = wrong)_u^k = \frac{p(L_{t-1})_u^k \cdot p(S)^k}{p(L_{t-1})_u^k \cdot p(S)^k + (1 - p(L_{t-1})_u^k) \cdot (1 - p(G)^k)} \tag{2.17}$$

$$p(L_t)_u^k = p(L_t|obs)_u^k + (1 - p(L_t|obs)_u^k) \cdot p(T)^k \tag{2.18}$$

$$p(C_t)_u^k = p(L_{t-1})_u^k \cdot (1 - p(S)^k) + (1 - p(L_{t-1})_u^k) \cdot p(G)^k \tag{2.19}$$

右上の k はスキル番号を示し，右下の u はユーザ番号を示すことに注意されたい．まず，生徒 u が初めから当該スキル k を身につけている確率は式 2.15 の通り定義する．正解が観測され，正しく当該スキルを身につけている確率は，式 2.16 で与えられ，不正解が観測されたが，正しく当該スキルを身につけている確率は，式 2.17 で与えられ，それらを合わせて，次の時刻に当該スキルを身につけている確率は，式 2.18 で与えられる．このように定めることで，理解しているがうっかり間違ってしまう場合や， 理解していないがあてずっぽうで正解してしまう場合を考慮できる．なお当該モデルでは，身につけたスキルの忘却は無視している．最後に，生徒 u がスキル k の問題を時刻 t に解いた場合に正解する確率 $p(C_t)_u^k$ は式 2.19 のように算出され，この値を次の問題の正誤予測に利用する．

上記に説明したモデルの学習にはいくつかの方法が適用され報告されている．1 つは [Corbett and Anderson, 1994] にあるように HMM を用いて生成モデルとして学習させる方法であり，1 つは [Yudelson et al., 2013] にあるように勾配法を用いて識別モデルとして学習させる方法である．それぞれ長所と短所があるが，特に，大規模データへの適用という観点では HMM に基づいた生成モデルの手法では計算量が大きく学習に非常に多くの時間がかかってしまうということもあり，[Yudelson et al., 2013] では勾配法に基づいた識別モデルとして学習させている．具体的には，[Yudelson et al., 2013] では，目的関数に負の対数尤度（Negative Log Likelihood）を利用し，勾配降下法（Gradient Descent）で学習させている．

2.4.3 Performance Factor Analysis

Performance Factor Analysis [Pavlik Jr et al., 2009]（以下，PFA）も過去の生徒の問題回答履歴から生徒が次に解く問題の正誤を予測するための手法である．しかし，知識獲得の時系列性を考慮する BKT と異なり，知識獲得の順番を考慮せず知識間の関係性を考慮して予測する手法である．PFA は下記のように定義される．

$$p(i, j \in KCs, s, f) \quad = \quad \sigma(\beta_j + \sum_{k \in KCs} (\gamma_k s_{i,k} + \rho_k f_{i,k})) \tag{2.20}$$

ここでは，s は事前に正答した問題回答，f は事前に誤答した問題回答，p はユーザ i が知識 j に正答する確率，β_j は知識 j の簡単さ，γ_k と ρ_k はそれぞれ知識 k の正答と誤答の重み，$s_{i,k}$ と $f_{i,k}$ はそれぞれユーザ i が知識 k に事前に正答した問題回答，事前に誤答した問題回答である．σ はシグモイド関数，過去の各知識の正誤を重み付けしシグモイド関数にかけ，別の問題の正誤を予測するというものである．

PFA は知識間の関係性を考慮できない場合に複数の知識がないと獲得できない知識のモデルが難しいという Bayesian Knowledge Tracing やその拡張手法の問題を解決するために提案された．PFA は知識間の関係性を重み付けして考慮しているが，問題回答の順番は考慮しない．

2.4.4 Deep Knowledge Tracing

Deep Knowledge Tracing [Piech et al., 2015]（以下，DKT）は RNN を利用し Knowledge Tracing を行う手法である．2015 年 6 月に発表された．数学の問題回答ログのデータセットで実験され，高い性能で将来の知識獲得を予測できること，予測モデルを分析することで知識間関係をネットワークとして抽出できることが報告された．生徒が獲得している知識から，ある知識の獲得されやすさをそのまま予測しており，得られた知識間関係から抽出されたネットワークは知識獲得における知識構造を表現しているといえる．DKT の構造と最適化，および知識間関係の抽出手法について順に説明していく．

構造

まず，DKT の構造について述べる．DKT の構造は伝統的な RNN の構造に基づいている．伝統的な RNN は入力のベクトル系列 $\mathbf{x}_1, \ldots, \mathbf{x}_T$ を出力のベクトル系列 $\mathbf{y}_1, \ldots, \mathbf{y}_T$ に写像する．この写像は，隠れ状態 $\mathbf{h}_1, \ldots, \mathbf{h}_T$ を計算することで達成されるが，一連の写像

の過程で過去観測から得られる関連情報を将来予測のために連続的に符号化している，とみなせる．確率変数は下記の式で定義されるネットワークにより関連付けられる．

$$\mathbf{h}_t = f(\mathbf{x}_t, \mathbf{h}_{t-1}) \tag{2.21}$$

$$\mathbf{y}_t = g(\mathbf{h}_t) \tag{2.22}$$

モデルは関数 f と g によって定義されており，これらの関数 f, g には SRNN の式 2.1，2.2 や LSTM-RNN の式 2.3–2.9，GRNN の式 2.10–2.14 を利用できる．

RNN で生徒の学習行動の観測結果をモデリングするため観測結果を固定長の入力ベクトル $\mathbf{x}_t$ の系列に変換する必要があるが，DKT ではシンプルな変換を行っている．具体的には，生徒の学習行動の観測結果を one-hot ベクトルに符号化し $\mathbf{x}_t$ とする，というものである．観測結果は演習問題と正誤の組み合わせで表現できるため，演習問題の数を M とすれば，$\mathbf{x}_t$ の長さは $2M$ となる．

表 2.1: Deep Knowledge Tracing における回答ログデータと対応する入力ベクトルの例

回答ログ				入力ベクトル	
ユーザ ID	ログの順番	問題番号	正誤	変数名	値
A	1	1	0	$\mathbf{x}_1$	[0000⋮1000]
A	2	1	1	$\mathbf{x}_2$	[1000⋮0000]
A	3	2	1	$\mathbf{x}_3$	[0100⋮0000]
A	4	3	0	$\mathbf{x}_4$	[0000⋮0010]
A	5	3	1	$\mathbf{x}_5$	[0010⋮0000]
A	6	4	1	$\mathbf{x}_6$	[0001⋮0000]

具体例を交えて説明する．例えば，演習問題の数が 4 つで，問題回答は 1 つずつしかできないと仮定する ．$M = 4$ であり，$\mathbf{x}_t$ の長さは 8 である．ある生徒が，表 2.1 の回答ログように問題を回答し正誤が観測されたとする．この時に，例えば，表 2.1 に記載のような入力ベクトルの系列となる．このようにして，回答行動の観測結果を符号化することで，どの演習問題をいつ正解もしくは不正解したのかを RNN に入力できる．

出力 $\mathbf{y}_t$ は問題と同じ長さのベクトルで，それぞれの要素が当該生徒がそれぞれの問題に正しく回答する確率の予測値となっている．したがって，$t+1$ の回答 q_{t+1} の正誤予測は $t+1$ に回答される問題 q_{t+1} に対応する $\mathbf{y}_t$ の要素から読み取れる．

最適化

次に，DKT の最適化について述べる．訓練時に用いられる目的関数は，モデルにおいて生徒の回答行動の観測系列の負の対数有度（Negative Log Likelihood）である．$\delta(q_{t+1})$ を時刻 $t+1$ にどの問題が回答されたかの one-hot ベクトルとし，a_{t+1} を時刻 $t+1$ に当該問題で正答したか否か（1 か 0）とし，l をクロスエントロピーとすれば，当該予測結果に対するロス関数は $l(\mathbf{y}^T\delta(q_{t+1}), a_{t+1})$ であり，生徒一人のロスは下記の式で与えられる．

$$L = \sum_t l(\mathbf{y}_t^T\delta(q_{t+1}), a_{t+1}) \tag{2.23}$$

学習時はミニバッチごとに確率的勾配降下法で目的関数を最小化する．[Piech et al., 2015] では，モデル学習時には過学習を防ぐため $\mathbf{y}_t$ への入力としての $\mathbf{h}_t$ には dropout [Srivastava et al., 2014] を適用している（$\mathbf{h}_{t+1}$ の方向には dropout を適用しない）．また，系列方向の誤差逆伝搬 [Werbos, 1990] において勾配が爆発するのを防ぐため，閾値以上のノルムの勾配は [Pascanu et al., 2013] にしたがい，制約を設けている．

知識間関係抽出法

次に，DKT のモデルを利用した知識間関係（あるいは，問題間関係）抽出法について述べる．DKT のモデルは，従来ではよく人間の専門家が行っていたデータの潜在的な構造や概念を発見するタスクに応用できる．問題 i と j のすべての有向ペアのうち下記の条件を満たすものに対して下記の影響度 J_{ij} を割り当てる．

条件　有効ペア (i, j) について，問題 i が出現した後に残りの問題系列の中で問題 j が出現する系列数が問題 i が出現する問題系列数全体の $V\%$ 以上であること．

影響度

$$J_{ij} = \frac{y(j|i)}{\sum_k y(j|k)}$$

ここでは，$y(j|i)$ は，ある生徒が最初に問題 i に正答した場合に，RNN によって割り当てられる次の時刻に問題 j に正答する確率である．[Piech et al., 2015] では，問題間影響行

列からのネットワーク抽出には，$V = 1$ を用いた．また，ネットワークの可視化に際しては，影響度が 0.1 以上であればエッジを引くというようにしてネットワークを構築した．

さらに，[Piech et al., 2015] は，得られたネットワークは，単に生徒の問題 (i, j) 間の遷移率から構築したネットワークや問題 i の正解が観測された後に問題 j の正解が観測される条件付き確率から構築したネットワークよりよく知識間関係を捉えていることを指摘している．

こうして得られた行列 J は，問題 i で評価される知識が既に獲得されている場合に，問題 j で評価される知識の獲得されやすさを表現しており，J は知識間関係行列であるといえ，この知識間関係行列から構築したネットワークは知識獲得における知識構造を表現していると考えられる．

2.4.5　知識獲得予測の手法における DKT の最適性

ここまで知識獲得予測の様々な手法について述べたが，DKT を拡張する手法が，本研究の目的を達成する上で最適な手法であることを，以下の二点に基づいて説明する．1) 知識間の関係性は，知識獲得予測の文脈において，定量的に検証されて抽出されるべきこと，2) 知識獲得予測は，複数の知識間の影響関係や，知識獲得の時系列性を考慮して行われるべきこと，

まず，1 について述べる．知識間の関係性は，知識獲得予測の過程で抽出されるものと，そうでないものがある．後者については，専門家が作成するという手法や，テキスト解析により概念関係ネットワークを構築するという手法 [Chen et al., 2008] がある．しかし，これらの手法は，専門家や研究者が立てた仮説に基づいた定性的なものであり，実際の生徒の学習過程をよく説明するものであるという定量的な根拠はない．

問題回答正誤の分析により知識の構造化を行う方法 [参考文献引用] も，2 つの問題 i と j の間で問題 i が正解後と不正解後の問題 j の正解率の差と着手順序を基に知識を構造化するが，この手法は 2 つの問題 i と j の関係性のみを考慮しており，他の問題との関係性は独立だと見なされている．得られた知識間関係は 2 つの知識の間についてのものを線形に合算したものであり，複雑で密接に関係している複数の知識の獲得順序や影響関係を捉えているものではない．

一方で，知識間の関係性の抽出を，知識獲得を予測する過程で行うものは，生徒の知識状態と行動を元に，知識の獲得を予測しているため，生徒の学習過程を反映した知識間関係を表現している可能性が高い．

したがって，知識間関係を定量的に抽出する手法としては，知識獲得を予測する過程で知識間関係を抽出する手法に絞る．

次に，２について述べる．知識獲得予測には，複数の知識の影響関係や知識獲得の時系列性を考慮するものと，そうでないものがある．後者については，複数の知識を独立なものとして，それらの状態の遷移を定義する Bayesian Knowledge Tracing(BKT) の手法や，過去の回答の結果を 1 つにまとめて定義する Performance Factor Analysis(PFA) の手法などがある．しかし，BKT の手法は，複数の知識を独立なものとして捉えるため，複数の知識からなる複雑な知識状態を捉えきれず，また，PFA の手法は，直前の回答も十分な時間が立った後の回答も，一つの過去の回答として捉えるため，生徒の時間に沿った知識獲得の状態を，現実的に捉えきれていない．

一方，DKT は，時系列に沿って RNN の隠れ層を更新することにより，複数の知識間の影響関係や，知識獲得の時系列性を考慮して知識獲得を予測しているので，より現実に沿った知識間関係を抽出できる可能性が高い．

現に [Piech et al., 2015] で，既に DKT によって知識間関係を抽出できることが報告されており，複雑で密接に関係している複数の知識の獲得順序や影響関係を捉えている可能性が高く，DKT を利用することが最適であると考えられる．

また，PFA と DKT のいずれの手法も知識間関係を考慮して予測に利用しているが，DKT の方が有効性が高いと考える．なぜなら，[Piech et al., 2015] では言及されていなかったが，DKT は PFA の拡張になっているためである．DKT は RNN を利用しており，

$$\mathbf{h}_t = f(\mathbf{x}_t, \mathbf{h}_{t-1}) \tag{2.24}$$

$$\mathbf{p}_t = \sigma(\mathbf{h}_{t-1} \cdot \mathbf{W}_{hp} + \mathbf{b}_p) \tag{2.25}$$

で与えられる．一方で PFA は

$$p(i, j \in KCs, s, f) = \sigma(\beta_j + \sum_{k \in KCs} (\gamma_k s_{i,k} + \rho_k f_{i,k})) \tag{2.26}$$

で与えられる．したがって，

$$f(\mathbf{x}_t, \mathbf{h}_{t-1}) = \mathbf{x}_t + \mathbf{h}_{t-1} \tag{2.27}$$

$$\mathbf{h}_0 = [0, 0, \cdots, 0] \tag{2.28}$$

とすると，$\mathbf{h_t}$ がこれまでの各問題についての正答回数を表現するベクトルと各問題につ

いての誤答回数を表現するベクトルを結合したベクトルになるが，これは，ベクトル $\mathbf{s}$ と $\mathbf{f}$ を結合したものと同じである．したがって，PFA は DKT 内部の RNN の繰り返しの部分を表現する関数 f を式 2.27 にした特殊なケースであり，DKT は PFA の拡張になっている．

以上，知識獲得の予測研究について述べ，本研究の素地となっている研究について整理した．

次に，次元削減手法について述べる．

2.5　次元削減手法

高次元のデータから低次元の特徴表現を抽出する，次元削減手法について述べる．本研究が目的とする知識分類表現の抽出は，一般的な次元削減の手法を拡張したものであり，一般的な次元削減手法について説明することで，基礎となる知識や本研究との差分を明確にすることを目的とする．

一般に，機械学習や統計においては，扱うデータの次元が大きい場合に，次元削減を行うことが多い．これは，データの次元が大きすぎることにより，データのサンプル数に対してモデルが複雑化してしまい，認識精度が悪くなる「次元の呪い」[Bellman and Corporation, 1957, Friedman, 1997] という現象を回避する目的の他，可読性を高めることにより，人間が解釈しやくすることなどを目的としている．知識獲得予測において用いられる知識分類も，分類のなされていない生の問題は次元数が大きく，そのままでは人間が解釈したり教育に用いることが困難なため，内容や難易度の類似度など，一定の尺度に従って，人間の手により次元削減が行われた例である．本研究ではこうした次元削減を，人間の手ではなく深層学習によって行うことで，最適化することを目的としている．

本節では，機械学習が現れる以前から一般的に使用されていた次元削減手法の代表である Principal Component Analysis や，ニューラルネットワークを活用した次元削減手法である Autoencoder，そして，深層学習の過程で行われる Embedding と呼ばれる埋め込み手法を取り上げ，次元削減の手法について概観する．

2.5.1　Principal Component Analysis

Principal Component Analysis は，日本語では主成分分析と訳される．相関のある多数の変数の中から，分散の大きい変数をデータ全体を説明する上で重要な「主成分」と見な

し，順にそれまでの主成分と直行するように主成分を定める変換を繰り返し行うことで，変数間の相関をなくし，重要度の高い主成分のみを採用し，次元削減を行う手法である．

その由来は古く，1901 年に力学の分野において初めて導入された [Pearson, 1901] ことをきっかけに，以後，経済学や統計学，機械学習などの分野で幅広く利用されてきた [Wold et al., 1987, Ku et al., 1995]．

一般的なアルゴリズムを以下に示す．

元のデータを $\mathbf{X}$，データの次元を M，変換後の次元を N，とすると、

1. データの共分散行列を求める．
2. 共分散行列の固有値と固有ベクトルを求める．
3. 固有値の大きい順に，対応する固有ベクトルを並べ替え，N 個の固有ベクトルを並べた行列 $\mathbf{P}$ を作る．
4. データからその平均ベクトルを引いたデータを $\mathbf{X}_{bar}$ とし，以下の式に基づいてデータを変換する．

$$\mathbf{X}_{pca} = \mathbf{X}_{bar}\mathbf{P} \tag{2.29}$$

主成分分析には様々な拡張がある [Schölkopf et al., 1997, Tipping and Bishop, 1999] が，その根底にある数学的な意味は，固有値問題を解くことにある．固有値問題とは、ある行列 $\mathbf{A}$ について、

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \tag{2.30}$$

となるような固有値 λ と固有ベクトル $\mathbf{x}$ を求めることである．

2.5.2 Autoencoder

Autoencoder は，日本語では自己符号化器と訳される．1980 年代から考案されていたが，2006 年の [Hinton and Salakhutdinov, 2006] らの研究によって広く普及した．様々な種類のものが考案されているが，基本的な概念は，ニューラルネットワークに入力されたデータを一旦低次元で表現し，その低次元表現から再び入力である自身を再現するように学習させることで，データの特徴を適切に表す低次元表現を獲得することを目的としている．教師データが存在しないが，自身を教師データとして行う教師学習のような形を取っており，教師あり学習と教師なし学習の中間に位置するような概念として理解されることもある．

具体的な定式化について述べる．まず，入力層 $x \in \mathbb{R}_d$ に対して，以下の式によって隠れ層 h と復元層 x^r を定義する．

$$h = f_\theta(x) = s(\mathbf{W}x + b) \in \mathbb{R}_{d_h} \tag{2.31}$$

$$x^r = g_{\theta'}(h) = s_r(\mathbf{W'}\ h + b') \in \mathbb{R}^d \tag{2.32}$$

ここで，$\theta = (\mathbf{W}, b), \theta' = (\mathbf{W}', b')$ は学習されるパラメータであり，s, s_r は活性化関数である．こうして定義された復元層 x_r が入力層 x にできるだけ近づくように，訓練データ $\mathbf{D} = x_1, \cdots, x_n$ に対する損失関数の平均値を最小化する過程で，パラメータ θ, θ' を学習する．

$$\min_{\theta,\theta'} \frac{1}{n} \sum_{i=1}^{n} L(x_i, g(f(x_i))) \tag{2.33}$$

式 2.33 で表される損失関数は，一般に再構成誤差 (Reconstruction Error) と呼ばれる．損失関数 L は，入力がバイナリ値ならば交差エントロピー誤差，実数値ならば二乗誤差を用いるのが一般的であり，本研究においては，入力は問題回答の正誤を意味するバイナリ値なので，交差エントロピー誤差を用いる．

活性化関数が恒等写像，つまり活性化関数が存在せず，損失関数が二乗誤差の場合は PCA と等価になることが知られており [参考文献]，Autoencoder は，ニューラルネットワークの構造と非線形の活性化関数を用いることにより，非線形の PCA を行っていると見なすことができる．

このように，一般的な次元圧縮，特徴表現抽出を非線形に行うことができる Autoencoder だが，深層学習が発達した今日では，深層学習モデルの各ユニットに良い初期値を与えるための事前学習として利用されることが多い [Erhan et al., 2010]．より頑健な特徴表現を獲得させるためにデータにノイズを加える Denoising Autoencoder [Vincent et al., 2008] や，潜在的な特徴表現の分布に特定の確率分布が存在すると仮定する Variational Autoencoder [Kingma et al., 2014] など，様々な工夫が考案されている．

2.5.3　Embedding

Embedding は，日本語では埋め込みと訳される．一般に，ニューラルネットワークにおいて入力データの次元数が大きい場合，そのままデータを入力すると，データの特徴的な表現について学習がしにくくなることがある．そのため，一度次元数の少ない層へ埋め込みを行い，データのより特徴的な表現を抽出した上で学習を進めることで，精度が向上す

る場合があることが知られている [参考文献？].

学習後のモデルにおける埋め込み層は，入力データを低次元で表現する上で最適な特徴表現となっている可能性が高く，この表現を抽出することでデータの特徴を保ったまま次元削減を行うことができる.

最後に，以上の関連研究を踏まえて，本論文で使用する類似の用語について定義を明確にする.

2.6　用語の定義

本節では，本論文で用いられる類似した用語の定義を行い，意味上の違いを明確にすることで，以降の手法の説明を含めた本論文の展開を明確にすることを目的とする. (TODO: 全てこれに統一)

2.6.1　知識獲得予測と回答正誤予測

一般に Knowledge Tracing と呼ばれ，本研究が問題提起を行っているのは「知識獲得予測」である．しかし，本研究は従来の「知識獲得予測」が「知識」の定義として用いている知識分類を所与のものとせず，問題の回答ログのみを入力に用いて，その回答の正誤予測を最適化する過程で適切な知識分類を学習することを目的としている．よって，知識分類を学習する段階では，まだ「知識」の定義はなされていないため，この段階では正確性を期すために「回答正誤予測」という単語を用いる.

一方，学習によって得た知識分類を用いて既存の知識分類と比較を行う際には，既に「知識」の定義がなされているため，一般的な「知識獲得予測」の単語を用いる.

2.6.2　知識分類と知識タグ

本研究で扱うデータセットには，既存の「知識分類」が存在する．問題に対して事前に分割されている知識ごとの分類や，そのような状況を「知識分類」と呼び，生徒が回答する各問題に紐づき，その内容を指し示す 1 つ 1 つの具体的なタグのことを「知識タグ」と呼ぶ．指し示している状況は似ているが，問題全体が事前に分類されている状況を意図するときは「知識分類」の単語を，より個別の問題に紐づく具体的なタグを意図するときは「知識タグ」の単語を使用する.

以上，関連研究について述べ，本研究の学術的位置付けや周辺概念を俯瞰した．

次に，分析手法について述べる．

第3章　分析手法

本章では，分析手法について説明する．

まず，分析手法全体の流れを概説し，手法全体が3つの要素から構成されることを述べたのち，各要素について詳述する．

3.1　分析手法全体の流れ

まず，分析手法全体の流れを説明する．本研究の手法は，データセットの作成という事前処理と，提案手法による知識分類の学習，学習された知識分類の知識獲得予測性能に関する検証，学習された知識分類の性質に関する比較分析という3つの分析から構成される．

まず，データセットの作成について述べる．本論文が扱う，生徒の知識獲得予測において，生徒が知識を獲得しているか否かの評価には，生徒の問題回答ログデータを対象データセットとして用いる．その際，比較検証に用いるため，知識獲得の予測に利用できる複数のデータセットを利用し，また，本研究に適用するために，幾つかの条件に基づいて対象データを抽出する．

次に，3つの分析の手法について述べる．

まず，知識分類の学習について述べる．知識獲得の予測性を最適化するような知識分類を抽出するには，問題の空間と，抽出目的の知識タグ空間の最適な関係性を深層学習によって学習する必要がある．そのために，問題を知識分類に変換する写像関数をパラメータ化し，回答正誤予測の最適化の過程で同時に学習する．このモデル構造は，Deep Knowledge Tracing のモデルを拡張して設計する．学習された写像関数は連続値表現の行列として表されており，これを適切に離散化することで，知識分類を抽出する．

次に，知識分類の性能検証について述べる．学習された知識分類に基づき，問題を知識タグ変換し，知識獲得予測を行う．抽出された知識タグを用いることにより，既存の知識タグや一般的な次元圧縮手法によって得た知識タグを用いる場合より，高い精度で知識獲得が予測できることを示すことで，本手法により知識獲得の予測性を最適化する知識分類が得られたことを定量的に示す．

最後に，知識分類の性質分析について述べる．学習された知識分類に基づく知識タグの性質や構造を，既存の知識タグの構造と比較することにより，その性質を定性的に検証する．

以上の分析手法全体の流れを，図 3.1 にまとめた．

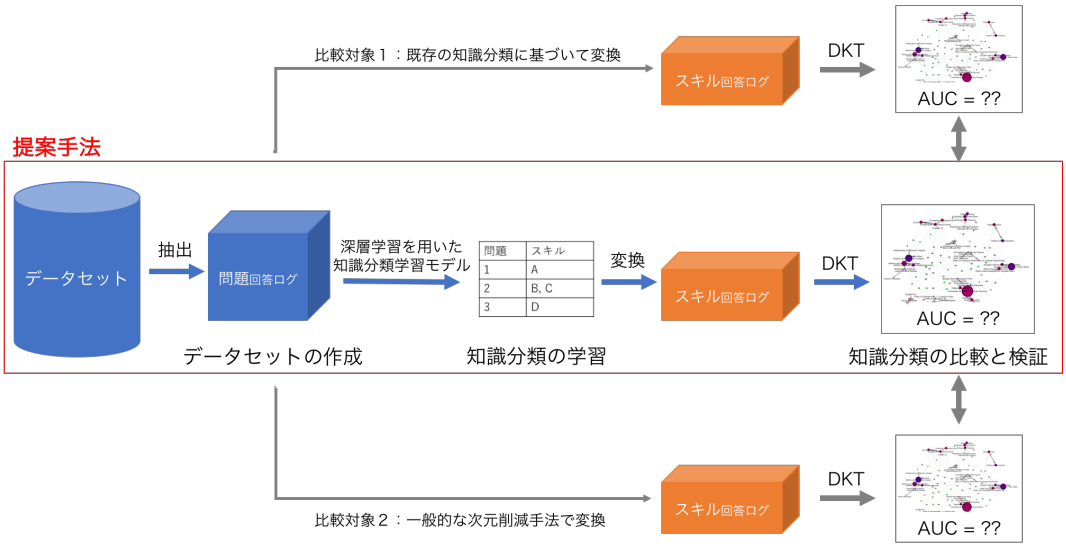

図 3.1: 分析手法全体の流れ

以降では，データセットの作成，提案手法による知識分類の学習，学習された知識分類の知識獲得予測性能に関する検証，学習された知識分類の性質に関する比較分析について順に詳述する．

3.2　データセットの作成

本論文が扱う生徒の知識獲得予測においては，生徒の問題回答ログデータをデータセットとして用いる．生徒の問題への回答結果は，その問題が問う知識を，生徒が既に獲得しているか否かを表現していると捉えることができるため，回答結果が正解であれば，該当の知識を既に獲得しており，回答結果が不正解であれば，該当の知識を未だ獲得していないと捉えることができるからである．

問題回答ログデータから作成するデータセットは，下記の要件を満たす必要がある．

1. データセットが大規模であること．

2. 比較検証できるデータセットが複数存在すること．

3. 問題に既存の知識タグが割り当てられていること．

まず，データセットが大規模である必要について説明する．一般に，深層学習は大量のデータを元に特徴的な表現を抽出するため，深層学習モデルを十分に学習させるには，大規模なデータが必要である．これは，深層学習モデルの一つである Recurrent Neural Network(RNN) を活用する Deep Knowledge Tracing についても同様である [Piech et al., 2015]．したがって，大規模なデータを有することがデータセットの要件の一つとなる．

また，深層学習によって知識獲得の予測を行う本研究においては，単に全体のデータ数が多いだけでなく，分析対象となる個別の問題や生徒について，十分なデータ数を確保できることが重要である．例えば，一度しか回答されていない問題については，その問題の正答や誤答によって，生徒の知識状態がどのように変化するかが観察できないため，分析に適していない．また，十分な数の生徒のデータがないと，特定の生徒の学習傾向が強く反映され，知識獲得過程の一般性が損なわれる可能性がある．

次に，比較検証できるデータセットが複数存在する必要について説明する．本研究で用いる，問題回答ログからなるデータセットは，そのデータセットが提供されるプラットフォームにより，問題を回答している集団や，扱っている教科，内容のレベルなどが異なる．特定のデータセットのみに対して得られた結果は，そのデータセットの環境においてのみ有効である可能性があり，一般性のある結果や知見が得られたとは言いにくい．そのため，本研究では，教科を数学に絞った上で，複数のプラットフォームにおける問題回答ログから複数のデータセットを作成することで，手法の一般性を検証する．なお，数学に限らない，他教科への適用可能性については，第 6 章で考察する．

最後に，回答された問題に，既存の知識タグが割り当てられている必要について説明する．本研究では．現在の一般的な知識獲得予測に用いられている知識タグは，人間の複雑な知識獲得過程を表現する上では最適化されていない，という仮定に立ち，問題回答ログのみを利用して，より最適化された知識タグを作成することを目的としている．抽出された知識タグの妥当性を検証するには，既存の知識タグと比較することが必要であり，そのため，分析対象となるデータセットは，問題に既存の知識タグが割り当てられている必要がある．

3.3　提案手法による知識分類の学習

本節では，知識分類学習モデルによる知識分類の学習について述べる．

本研究では，知識獲得予測を最適化する知識分類を学習するために，問題を知識分類に変換する関数をパラメータ化し，Knoeledge Tracing の最適化の過程で同時に学習する．なお，以降では，この提案手法のモデル構造を「知識分類学習モデル」と呼ぶ．この知識分類学習モデルの構造は，Deep Knowledge Tracing（以下，DKT）を元に設計されているため，まず，DKT を拡張する方法について述べる．

その後，抽出された問題空間を知識タグ空間に写像する関数を離散化し，実際の知識分類を作成する方法について述べる．

3.3.1　DKT の拡張による写像関数の学習

問題を知識分類に変換する写像関数をパラメータ化し，Knoeledge Tracing の最適化の過程で同時に学習するために，既存の DKT モデルを拡張する方法を，以下の 3 つの要素に分けて説明する．

1. 入力データの粒度
2. モデル全体の構造
3. 最適化手法

入力データの粒度

まず，既存の DKT と大きく異なる点として，モデルに入力されるデータの粒度の違いについて述べる．DKT のモデルにおいては，使用するデータの元は生徒の問題回答ログデータであるが，モデルへの入力は，事前に定義された知識分類に基づいて，知識タグに落とし込まれ，どの知識タグに回答したかが入力される．これは，既存の知識タグの中で，生徒がどのように知識獲得をしていくかを予測することを前提にしているためであるが，本研究においては，そもそもの問題と知識タグの関係性を最適化することを目的とするため，この入力は適さない．

よって本研究では，モデルへの入力は問題に対する回答のままにとどめ，生徒が次にどの問題に正解するかを予測する過程において，深層学習モデル自身に最適な知識の分類方法を判断させることで，最終的に，知識獲得予測を最適化するような知識分類を学習させる．

こうして学習された知識分類に基づく知識タグは，結果的に知識獲得の文脈で最適化されていると考えることができ，既存の知識タグと比較することで，その性質を解釈することができる．

モデル全体の構造

次に，具体的なモデル全体の構造の拡張について述べる．まず，DKT では入力が RNN の隠れ層へ直接伝達されるのに対し，知識分類学習モデルは，まず入力層の問題空間 X から，抽出目的の知識タグ空間 U への写像を行う．ここで言う知識タグ空間とは，既存の知識タグと同じ次元数に設定された空間で，問題回答の正誤の情報を低次元の空間で表すことを目的としている．

具体的には，まず，問題数を M とした場合，モデルへの入力ベクトル $\mathbf{x}_t$ の長さは 2 M である．抽出目的の知識タグの次元数は，既存の知識タグと同じ次元数に揃え，事前に N と定義する．そして，M 次元の問題空間 X から N 次元の知識タグ空間 U へ変換する写像関数 P を，以下の式により定義する．

$$\mathbf{P} = \sigma(\mathbf{W}_{xu}\mathbf{x} + \mathbf{b}_u) \tag{3.1}$$

ここで，$\mathbf{x}$ は長さ M の問題空間ベクトルを指し，$\mathbf{W}_{xu}$ は MxN の大きさの重み行列を指し，$\mathbf{b}_u$ は長さ N のバイアス項を指し，σ は $1/(1+e^{-x})$ で定義されるシグモイド関数を指す．訓練時には，$\mathbf{W}_{xu}$，$\mathbf{b}_u$ を学習する．

$$\mathbf{P} = \sigma(\mathbf{W}_{xu}\mathbf{x} + \mathbf{b}_u) \tag{3.2}$$

時刻 t における問題の回答の正誤の情報は，DKT 同様，それぞれ長さ M のベクトルで別々に表現し，連結して長さ $2M$ のベクトル $\mathbf{x}_t$ として表現されている．写像関数 P を，$\mathbf{x}_t$ の前半の正答部分と後半の誤答部分に別々に適用し，連結することによって，回答の正誤の情報を保った，長さ 2 N の知識タグ空間ベクトル $\mathbf{u}_t$ が生成できる．

$$\mathbf{u}_t = [\mathbf{P}(\mathbf{x}_{t_positive}), \mathbf{P}(\mathbf{x}_{t_negative})] \tag{3.3}$$

ここで，$\mathbf{x}_{t_positive)}$，$\mathbf{x}_{t_negative)}$ はそれぞれ問題回答の正答と誤答を表す長さ M のベクトルである．

こうして得られた知識タグ空間ベクトル $\mathbf{u}_t$ は，一般的な RNN 同様，隠れ層を経由して

時系列情報を反映した後，再び長さ 2 N の知識タグ空間ベクトル $\mathbf{v}_t$ となる．

$$\mathbf{h}_t = \tanh(\mathbf{W}_{uh}\mathbf{u}_t + \mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h) \tag{3.4}$$

$$\mathbf{v}_t = \sigma(\mathbf{W}_{hv}\mathbf{h}_t + \mathbf{b}_v) \tag{3.5}$$

ここで，$\mathbf{h}_t$ は時刻 t の隠れ層を指し，$\mathbf{W}_{uh}$，$\mathbf{W}_{hh}$ はそれぞれ重み行列を指し，$\mathbf{b}_h$，$\mathbf{b}_v$ はそれぞれバイアス項を指し，tanh は $(e^x - e^{-x})/(e^x + e^{-x})$ で定義される Hyperbolic Tangent 関数を指す．この知識タグ空間ベクトル $\mathbf{v}_t$ は，時刻 t までの回答情報を反映した，生徒の知識タグ空間における知識状態を表していると言える．

最終的に，この知識タグ空間ベクトル $\mathbf{v}_t$ から，予測結果として M 次元の問題回答予測ベクトル $\mathbf{y}_t$ を算出する．

$$\mathbf{y}_t = \sigma(\mathbf{W}_{vy}\mathbf{v}_t + \mathbf{b}_y) \tag{3.6}$$

ここで，$\mathbf{W}_{vy}$ は重み行列を指し，$\mathbf{b}_y$ はバイアス項を指す．

$\mathbf{y}_t$ は 0 から 1 の間の値を取り，次の時刻 $t+1$ において各問題に正答する確率を表しており，既存の DKT と同じ予測表現となっている．

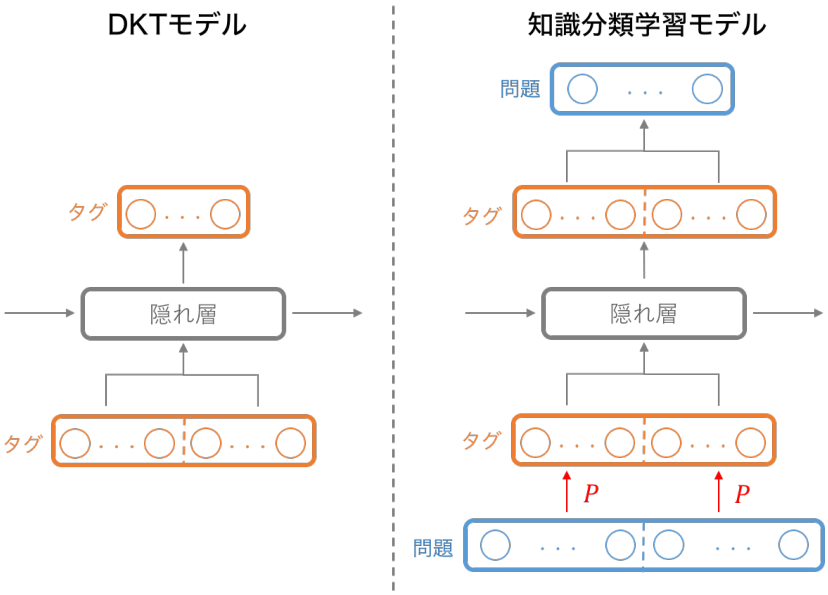

図 3.2: モデル構造上の拡張

以上のようなモデル構造上の拡張をまとめた図を図 3.2 に表す．橙色の層はタグ空間を，

青色の層は問題空間を表し，層が二つに区切られている部分は前半・後半がそれぞれ正答・誤答の情報を表現している．問題空間からタグ空間への写像関数 P は $P = \sigma(\mathbf{W}x + b)$ で表され．正答部分と誤答部分に同じ関数が適用される．

この拡張の目的は，知識獲得予測の最適化を行う過程で，M 次元の問題空間 X から N 次元の知識タグ空間 U へ変換する写像関数 P をパラメータ化し，学習することにある．

最適化手法

既存の DKT における最適化手法は，次の時刻の問題回答予測ベクトルと，実際の次の時刻の問題回答ベクトルの誤差を最適化手法として，これを最小化するものである．

$$\log(p_1 \times p_2 \times \cdots \times p_{m_t}) = \sum_{k}^{m_t} \log(p_k) \tag{3.7}$$

であるため，$\tilde{\delta}(\mathbf{q}_{t+1})$ を時刻 $t+1$ にどの問題が回答されたかの m_t-hot ベクトルとし，$\mathbf{a}_{t+1}$ を時刻 $t+1$ に対応する問題で正答したか否か（1 か 0）のベクトルとすれば，損失関数は

$$L_p = \sum_{t} l(\mathbf{y}_t^T \tilde{\delta}(\mathbf{q}_{t+1}), \mathbf{a}_{t+1}) \tag{3.8}$$

である．この損失関数を，問題回答予測に関する損失関数 L_p とする．

本研究では，この問題回答予測に関する損失関数に加え，式 2.31–2.33 で表される再構成誤差も損失関数として導入する．ここで，式**??**におけるパラメータについては，本研究では，$\mathbf{W}$，$\mathbf{b}$ が，入力層で問題空間から知識タグ空間へ写像する際に用いる $\mathbf{W_{xu}}$，$\mathbf{b_u}$ であり，$\mathbf{W}'$，$\mathbf{b}'$ が，出力層で知識タグ空間から問題空間へ復元する際に用いる $\mathbf{W_{vy}}$，$\mathbf{b_y}$ である．

この入力の再構成誤差の損失関数を L_r とすると，結果的に，モデル全体の損失関数 L は以下の式によって定められ，この損失関数を最小化するようにモデルが学習される．

$$L = L_p + L_r \tag{3.9}$$

ここで，再構成誤差を損失関数に導入する理由について述べる．まず，一般的に再構成誤差を用いる Autoencoder は，深層学習モデルに良い初期値を与えるための事前学習のための仕組みとして用いられるが，本研究では，この Autoencoder の構造を，DKT と同時に学習させるモデル構造になっている．

Autoencoder の構造を，事前学習ではなく普通の学習モデルに適用することは，必ずし

も精度向上につながるわけではないため，一般的ではなく，単純に低次元のベクトルへ埋め込むのみに留まることが多い．

本研究では，問題から知識分類を学習するには十分とはいえないデータ量が，一部の問題について存在するため，データの特徴を把握しきれない underfitting の状況に陥る可能性がある．そうした学習不足への対策として，学習を矯正する正則化項として再構成誤差を導入している．このように，データ数が不足する場合に，モデルがより適切に学習を進めるように正則化項を設ける手法は一般的であり，有効な手法とされている．

また，こうした問題と知識タグに Autoencoder の関係を定義することは，問題の正答・誤答と知識タグの理解状態は，相互に変換できるはずだという，教育学的な文脈との整合性と合致し，学習される知識分類の質を損ねないと判断した．

3.3.2　写像関数の離散化による知識分類の作成

3.3.1 の知識分類学習モデルで得られた写像関数 P から，実際に知識分類を作成して問題をタグ付けし，既存の知識タグと比較する手法について説明する．

知識獲得予測の最適化の過程で得られた写像関数 P は，M 次元の問題空間を N 次元の知識タグ空間に写像する，MxN の大きさの行列として表現されている．この行列は 0 から 1 の値を取る連続値表現であるため，そのままでは問題がどの知識タグに紐づくかを特定することはできない．そのため，何らかの方法で，この行列を 0 か 1 の 2 値を取る離散表現に改める必要がある．

離散化の方法としては，各問題に対して最も関係性の強い知識タグのみをタグとする方法や，行列全体で特定の閾値を定め，その閾値を超えたものをタグとする方法，両者を組み合わせる方法など，様々な方法が考えられる．各方法によって，抽出された知識タグの性質の異なる解釈をすることが可能だが，本研究では，知識獲得予測に適用した際に最も良い精度で予測できる分類を，知識獲得の遷移を最もよく説明する分類として見なして利用し，その性質も解釈する．

3.4　学習された知識分類の知識獲得予測性能に関する検証

次に，抽出された知識タグ (以下，抽出タグ) の知識獲得予測における性能を，比較検証する手法について述べる．まず，抽出タグが，知識獲得の予測性を最適化する分類になっていることを示すために，抽出タグに基づいた回答ベクトルを一般的な DKT に入力することで，既存の知識タグ (以下，既存タグ) を用いた場合よりも精度が向上することを確認

する．また，この精度向上が，知識獲得予測の最適化の過程で知識分類を作成したことに起因することを示すため，PCA や Auto Encoder といった，知識獲得予測を用いない一般的な次元圧縮手法で作成した知識タグを用いた場合とも比較を行う．

これは，本手法における知識分類の作成が，

- 人間の可読性を目的として専門家によって設計された分類か否か
- 知識獲得予測の文脈で最適化された分類か否か

という二つの観点において，既存の知識分類と異なることを受け，性能の変化が何に起因してもたらされたのかという，差分をより明確にするために行う検証である．

3.5　学習された知識分類の性質に関する比較分析

最後に，こうした知識獲得予測の精度向上という定量的な検証に加え，抽出タグの性質について，既存の知識タグとの比較を通して，定性的な分析も行う．これは，知識獲得予測において最適化されている知識分類の性質を解釈し，新たな知見を得ることで，教育における実用性を考察することにつなげることが目的である．

まず，抽出タグと，既存の知識タグについて，回答される頻度や問題との紐付き方の分布に着目し，知識獲得予測の精度を向上させる要因を，データの構造から分析する．

また，抽出タグと既存の知識タグが，内容の面においてどのような関係性があり，既存の知識タグがどのように再配置されたのかを検証する．

抽出タグと既存タグの内容を比較する方法として，まず，抽出タグが紐づく問題と，既存タグが紐づく問題から，抽出タグと既存タグの共起行列を作成する．この行列は，各抽出タグを，各既存タグとの関係性の近さを表現している行列と捉えることができるが，各抽出タグの特徴をより明確に捉えるために，TF-IDF 法 [参考文献引用 or 前提知識] を用いて，特徴の重み付けを行う．TF-IDF 法は，元々文書の分類に用いられる手法で，複数の文書がある時に，各文書を特徴づける単語を特定することを目的にしている．具体的には，まず，各文書内での，各単語の出現頻度 (Term Frequency，以下 TF) を求める．TF は，文書内に多く出現する単語ほど重要だ，という考えに基づいた指標であるが，TF のみだと，例えば助詞や助動詞と言った，いくつもの文書で横断的に使われている単語の重要度が高くなってしまうため，各単語が全文書の内いくつの文書にあらわれているかという制約 (Inverse Document Frequency) を設けて，一般的な単語の影響を除外する．単語

i, j の TF，IDF と，それらを用いた TF-IDF 値は以下の式で表される．

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{3.10}$$

$$IDF_i = \log \frac{|D|}{|\{d : d \ni t_i\}|} \tag{3.11}$$

$$TFIDF_{i,j} = TF_{i,j} \cdot IDF_i \tag{3.12}$$

$n_{i,j}$ は単語 i の文書 j における出現回数を指し，$\sum_k n_{k,j}$ は文書 j におけるすべての単語の出現回数の和を指し，$|D|$ は総文書数を指し，$|\{d : d \ni t_i\}|$ は単語 i を含む文書数を指す．

さて，今回の，抽出タグの特徴付けにおいては，各既存タグの成分が単語にあたり，抽出タグ 1 つ 1 つが文書にあたる．複数の抽出タグに共通して現れる既存タグの成分ほど，特徴づけにおける重みが小さくなり，特定の抽出タグのみに現れる既存タグの成分ほど，特徴づけにおける重みが大きくなる．

この手法により各抽出タグの特徴付けを行い，各抽出タグにおいて特徴的な既存タグを特定する．

そして，既存の DKT のネットワーク分析手法に則り，既存タグをノードとする知識間ネットワークを作成した上で，各抽出タグのノードを追加で作成し，先程特定した特徴度に応じて既存タグのノードに対してエッジを引くことで，抽出タグが既存タグをどのように再構築しているかを分析する．

以上，分析手法について述べた．次章では，実験で利用するデータセットについて述べる．

第4章　データセット

本章では，実験で用いるデータセットについて述べる．

本研究では，オンライン学習サイトにおける，生徒の問題回答ログをデータとして用いる．

その際，比較検証のため，教科を数学に絞った上で，２つのデータセットを用意する．いずれのデータセットも，前章で述べたデータセットの要件の一つである，既存の知識タグを有するという要件を満たしている．

以下では，各データセットについて概説した後，本研究に適用するためのデータの抽出方法について述べる．

4.1　ASSISTments 2009-2010

本データセットは，オンライン学習サービスの「ASSISTments[1]」における，生徒の問題回答ログから生成されている．まず，ASSISTments のサービスについて概説した後，本研究で用いるデータセットについて説明する．その際，本研究に適用する際に問題となるデータの性質に言及した上で，その問題点を解消するためのデータの抽出方法を述べる．

4.1.1　ASSISTments のサービス

ASSISTments は Intelligent Tutoring System(ITS) の一つで，2017 年 1 月現在で，14 の国と 42 の州において利用されている．数学や科学，英語や社会と言った科目をカバーしており，レベルは日本における小学生から高校生まで様々である．基本的な仕組みは，システムが生徒に課した問題を生徒が回答し，その結果をシステムが自動的に採点，間違えた場合はヒントを出すなどして，生徒の知識獲得を促すもので，教師や親がその回答結果や統計情報かた生徒の習熟度を確認できること，また，必要があれば，システム上で提供されている教材を自由に編集して，新たな問題を作成できる柔軟性の高さなどから，様々な教育機関や，オンライン教育サービス上で活用されている．

[1] https://www.assIstments.org/

4.1.2　対象データセット

本研究で用いるデータセットは,「ASSISTments 2009-2010」と呼ばれる，ASSISTments における，生徒の 2009 年から 2010 年の間の数学の問題回答ログの内,「skill_builder」[2] と呼ばれるデータセットである.

元々,「ASSISTments 2009-2010」には「skill_builder」と「non_skill_builder」という，系統の異なる 2 つのデータセットが含まれている.「skill_builder」は，生徒に知識を段階的に身につかせることを目的にした系統で，ある知識を問う問題に生徒が連続で正答できた場合に，該当の知識を習得したものとみなし，次に進ませるというものである．日本の教育現場で言えば，授業ごとの小テストに近いものといえる．一方,「non_skill_builder」は，生徒がそれまで学んできたことを正しく身につけられているかを確認することを目的にした系統で，さまざまな知識を問う問題を，まとめて生徒に課すものである．日本の教育現場で言えば，期末テストに近いものといえる.

このような性質から,「skill_builder」のデータセットの方が，生徒の知識獲得過程の細かな推移を観察する上で適しているため，Deep Knowledge Tracing [Piech et al., 2015] を始めとする，Knowledge Tracing に関する多くの研究で利用されるデータセットであり，本研究でも同様に,「skill_builder」のデータセットを用いる．生徒が解く問題 (problem) には，1 つ以上の知識タグ (skill) が紐付いている.「skill_builder」のデータセットには，4,217 人の生徒の，124 の知識タグが紐づく 26,688 の問題に対する，401,756 の回答ログが含まれている．なお,「skill_builder」のデータセットは，行の重複などによって大幅な不備が指摘されたため訂正版が提供されており，それ以前の研究結果は信憑性が低い．本研究では，訂正後のデータセットを用いている.

4.1.3　データの抽出

本データセットは，本研究に適用する上で以下の 3 つの問題を抱えている．順に，各問題と対策について述べる.

まず，問題 (problem) の中には複数の知識タグ (skill) が紐付いているものがあるが，そうした問題が回答された場合には，知識タグの数だけ，ログが別々に作成されている．これは，見かけ上のログ数 (401,756) が，実際に回答された回数より多くなっているだけでなく，同時に回答された問題や知識タグが，別々に回答されたとみなされる危険性があり,

[2] https://sites.google.com/site/assistmentsdata/home/assistment-2009-2010-data/skill-builder-data-2009-2010

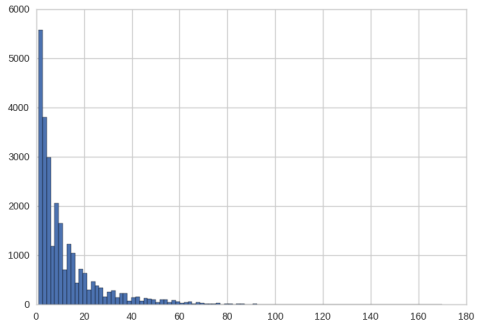

図 4.1:「ASSISTments 2009-2010」における問題ごとの回答数の分布

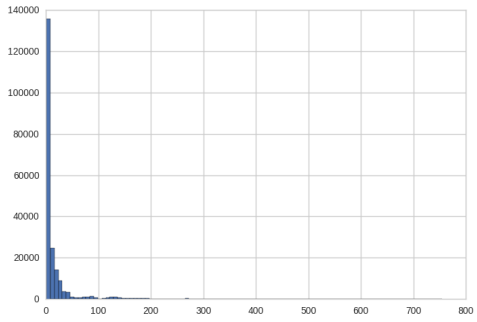

図 4.2:「bridge to algebra 2006-2007」における問題ごとの回答数の分布

この前提を考慮しない Knowledge Tracing は不適切であることが指摘されている [Xiong et al., 2016b]．このため，重複している行を一つにまとめる作業が必要である．

次に，既存の知識タグ (skill) についても，存在はするものの，名前が割り当てられていないものが存在し，これらは抽出された知識タグと比較することが不可能なため，除外する必要がある．

また，本データセットは，全体で見ると十分大規模であるといえるが，個別の問題に関して言えば，ほとんど回答されていない問題が，全体に対して大きな割合を占めている．(図**??**を参照) 十分なログ数を保有しない問題は，大規模データから深層学習によって知識タグを抽出するという本研究の目的を満たさないため，ログ数について一定の閾値を設けてデータセットを切り分けることによって，適切なデータを抽出する必要がある．

以上より，本研究では，元のデータセットから，以下の方法で分析対象とするデータを抽出している．1) 同時回答を意味する重複行を一つにまとめる．2) 名前が割り当てられている知識タグを持つ問題に関するログのみを抽出する．3)2) のうち，最低 30 回以上回答されている問題に関するログのみを抽出する．4)3) に含まれる問題を，最低 2 回以上回答している生徒に関するログを抽出する．

なお，3) の 30 回以上という具体的な数字は，深層学習として有意な結果が得られるログ数として，実験的に得たものであり，網羅的に検証されたものではない．4) の 2 回以上という数字は，知識獲得の推移を観察する上で最低限必要なログ数である．結果的に，3,410 人の，56 の知識タグが紐づく 2,635 の問題に対する，129,317 の回答ログが分析対象である．

4.2　bridge to algebra 2006-2007

本データセットが利用された「KDDCup」について概説した後，データセット自体について説明する．その際，ASSISTments 同様，本研究に適用する際に問題となるデータの性質に言及した上で，その問題点を解消するためのデータの抽出方法を述べる．

4.2.1　KDDCup

「KDDCup（Knowledge Discovery and Data Mining Cup）[3]」は、100 以上の国にまたがり、10 万人を超える会員を持つコンピューターサイエンス分野の学会である「ACM(the Association for Computing Machinery)」の分科会である「SIGKDD（Special Interest Group on Knowledge Discovery and Data Mining）」が毎年開催する競技会であり、この分野で最も古く権威のある競技会の一つである．

4.2.2　対象データセット

本研究では，2010 年に開催された KDDCup の内の一つの，教育分野の競技会である「Educational Data Mining Challenge」で使用された,「Bridge to Algebra 2006-2007」[Stamper et al., 2010] というデータセットを用いる．これは，オンライン教育サービスの「Carnegie Learning[4]」が提供するオンライン学習支援システム「Cognitive Tutor」における，2006 年から 2007 年の間の，数学の問題に対する生徒の問題回答ログである．Codnitive Tutor は ASSISTments と同じ ITS だが，やや性質の異なるものになっている．ASSISTments は，生徒が毎日の宿題を解く過程をサポートするような，比較的単純な設計となっている一方で，Cognitive Tutor は，より，生徒が個別の知識を獲得する過程を緻密にサポートする設計となっている．具体的には，各問題 (problem) が，複数のステップ (step) に分解されており，ステップ 1 つ 1 つに知識タグ (knowledge component) が紐付いている．

このデータセットには，1,146 人の生徒の，494 の知識タグが紐づく 19,186 の問題と 19,766 のステップに対する，3,679,199 の回答ログが含まれている．

4.2.3　データの抽出

本データセットも,「ASSISTments 2009-2010」同様に，本研究に適用する上で 3 つの問題を抱えている．以下に各問題と対策を述べる．

[3] http://www.kdd.org/kdd-cup
[4] https://www.carnegielearning.com/

まず，問題 (problem) やステップ (step) という粒度が存在する本データセットにおいて，何を一回の問題回答と見なしてデータを作成するかが問題となる．「Cognitive Tutor」では，一つの問題内に複数のステップが用意されており，各ステップについて逐次回答し，正答できるまで取り組み，正答できた場合に次のステップに進むように設計されている．そのため，生徒の知識獲得の推移を観察する上では，一つ一つのステップに注目することが適切だといえる．よって，本データセットでは，問題内のステップに対する回答を一回の問題回答と見なし，データを抽出する．

次に，既存の知識タグ (knowledge concept) についても，存在はするものの，名前が割り当てられていないものが存在し，これらは抽出された知識タグと比較することが不可能なため，除外する必要がある．

また，本データセットは，全体で見ると十分大規模であるといえるが，個別の問題に関して言えば，ほとんど回答されていない問題が，全体に対して大きな割合を占めている．(図 4.2 を参照) 十分なログ数を保有しない問題は，大規模データから深層学習によって知識タグを抽出するという本研究の目的を満たさないため，ログ数について一定の閾値を設けてデータセットを切り分けることによって，適切なデータを抽出する必要がある．

以上より，本研究では，元のデータセットから，以下の方法で分析対象とするデータを抽出している．1) 問題 (problem) とステップ (step) の組み合わせを一回の問題回答とみなす，2) 名前が割り当てられている知識タグを持つ問題に関するログのみを抽出する．3)2) のうち，最低 100 回以上回答されている問題に関するログのみを抽出する．4)3) に含まれる問題を，最低 2 回以上回答している生徒に関するログを抽出する．

なお，3) の 100 回以上という具体的な数字は，深層学習として有意な結果が得られるログ数として，実験的に得たものであり，網羅的に検証したものではない．4) の 2 回以上という数字は，知識獲得の推移を観察する上で最低限必要なログ数である．結果的に，1,136 人の，193 の知識タグが紐づく 3,439 のステップに対する，605,683 の回答ログが分析対象である．

4.3　データセットの概観

以上の条件から抽出された，本実験に用いるデータセットの統計量を表 4.1 に示す．

なお,「bridge to algebra 2006-2007」は,「ASSISTments 2009-2010」に比べて，問題数に対する知識タグの数の割合が大きく異なっているが，これは,「bridge to algebra 2006-200」では，問題がさらにステップに分割されており，知識タグの粒度がより細かくなっている

表 4.1: 各データセットの統計量

データセット名	生徒数	問題数	知識タグ数	ログ数
ASSISTments 2009-2010	3,410	2,635	56	129,317
bridge to algebra 2006-2007	1,136	3,439	193	605,683

ためである．

以上，データセットについて述べた．次章では，実験について述べる．

第5章　実験

本章では，実験について述べる．
まず，実験設定について述べ，その後，実験結果について述べる．

5.1　実験設定

本研究の実験は，大きく以下の３つのブロックに分けられる．

1. 知識分類学習モデルによる知識分類の学習
2. 学習された知識分類の知識獲得予測性能に関する検証
3. 学習された知識分類の性質に関する比較分析

以下では，順に，実験設定について述べる．

5.1.1　知識分類学習モデルによる知識分類の学習

生徒の問題回答ログに対し，図 3.2 に表される知識分類学習モデルを適用し，回答正誤の予測性において最適な知識分類を抽出する．知識タグ空間の次元数は，既存の知識タグの次元数と統一し,「ASSISTments 2009-2010」では 56,「bridge to algebra 2006-2007」では 193 とした．実際のモデルのユニットにおいては，正答ベクトルと誤答ベクトルを分けてユニットを作るため，それぞれ２倍のユニット数で表現されている．

ハイパーパラメタについては，学習率の初期値を 200，モーメントを 0.98，1 エポックごとに，減衰率 0.99 として学習率を最小学習率 10 まで減衰させる．また，勾配のノルムの最大値を 0.00001 として [Pascanu et al., 2013] に従い勾配に制約を設けた．dropout は [Piech et al., 2015] と同様に $\mathbf{y}_t$ の方向にのみかけ，dropout 率は 0.5 とした．隠れ層のユニット数は 400 として，各重み行列の初期化は [Glorot and Bengio, 2010] にしたがった．時系列方向の誤差逆伝搬は最長で 200 まで伝搬するように制約を設けた．

これらのハイパーパラメタは実験的に高い予測性能を発揮したため設定しており，網羅的に探索したわけではない．通常，深層学習の手法はハイパーパラメタの数が非常に大きく，また，計算コストが大きいため大規模な探索は行えない．Grid Search や Random Search [Bergstra and Bengio, 2012] といった探索手法が提案されているが，専門家が手で調整した方が優れていることが報告されている [Larochelle et al., 2007, Bergstra and Bengio, 2012].

最適化手法は，3.8 で表される問題回答予測に関する誤差関数 L_p と，**??**で問題空間と知識タグ空間の Reconstruction Error を表す誤差関数を L_r の和である L3.9 を目的関数として最小化するものである．

学習時は [Piech et al., 2015] と同様にミニバッチごとに確率的勾配降下法で目的関数を最小化する．評価指標は AUC スコアを採用する．

２つのデータセットいずれにおいても，訓練：検証：テスト ＝ 8：1：1 となるようにユーザを分け，訓練ユーザのデータでモデルを構築し，検証ユーザのデータでハイパーパラメタを調整し，検証ユーザのデータで精度が最も高かったモデルをテストユーザのデータに適用し当該モデルの最終的な精度とした．

実装には Theano を用いた [Bergstra et al., 2010, Bastien et al., 2012]．Theano は多次元行列を含む数学的表現の定義や計算，最適化を効率的に行える Python のライブラリで，深層学習の研究ではよく利用される．

5.1.2　学習された知識分類の知識獲得予測性能に関する検証

5.1.1 で得られた問題空間から知識タグ空間への写像行列を離散化し，抽出された知識分類を用いて一般的な DKT を行い，既存の知識分類を用いた場合の精度の比較を行う．また，本手法で抽出される知識分類と既存の知識分類の差分を明確にするため，以下の方法で作成された知識分類を用いた場合とも比較を行う．

- DKT を用いず，一般的な次元圧縮手法である PCA や Auto Encoder によって作成された知識分類
- DKT を用いるが，Reconstruction Error を誤差関数に入れない，一般的な埋め込みによって作成された知識分類

まず，写像行列の離散化は，いずれのデータセットにおいても，各問題に対して最も関係性の強い知識タグのみを 1 とし，他を 0 にする方法が，最も高い精度で知識獲得を予測できることを確認したため，この手法を用いる．

ハイパーパラメタについては，学習率の初期値を 200，モーメントを 0.98，1 エポックごとに，減衰率 0.99 として学習率を最小学習率 10 まで減衰させる．また，勾配のノルムの最大値を 0.00001 として [Pascanu et al., 2013] に従い勾配に制約を設けた．dropout は [Piech et al., 2015] と同様に $\mathbf{y}_t$ の方向にのみかけ，dropout 率は 0.5 とした．隠れ層のユニット数は 400 として，各重み行列の初期化は [Glorot and Bengio, 2010] にしたがった．時系列方向の誤差逆伝搬は最長で 200 まで伝搬するように制約を設けた．

最適化手法は，一般的な DKT と同じく，3.8 で表される問題回答予測に関する誤差関数 L_p を目的関数として最小化するものである．

学習時は [Piech et al., 2015] と同様にミニバッチごとに確率的勾配降下法で目的関数を最小化する．評価指標は AUC スコアを採用する．

5.1.1 と同様に，２つのデータセットいずれにおいても，訓練：検証：テスト $= 8:1:1$ となるようにユーザを分け，実装には Theano を用いた．

5.1.3　学習された知識分類の性質に関する比較分析

5.1.1 で抽出された知識分類を既存の知識分類を比較分析することで，その性質を検証する．

まず，各知識タグが回答ログに出現する頻度の分布や，紐づく問題数の分布に着目し，知識獲得予測の精度を向上させる要因を，データの構造から分析する．また，抽出された知識タグと既存の知識タグを，同じネットワークに配置して可視化することで，既存の知識タグがどのように再配置され，どのようなネットワーク構造となったのかを分析する．

以上，実験設定について述べた．

5.2　実験結果

実験結果について述べる．まず，各手法によって作成された知識分類についての知識獲得予測における予測性能を比較し，いずれのデータセットいずれにおいても，提案手法によって学習された知識分類から作成された知識タグを利用することで，最も良い精度で予測が可能になっていることを定量的に確認する．

さらに，学習された知識分類を，既存の知識分類と比較することにより，その性質を定性的に分析する．

表 5.1: 各知識分類の知識獲得予測における予測性能

データセット	AUC				
	既存タグ (marginal)	事前圧縮タグ		深層学習タグ	
		PCA	AutoEncoder	Embedding	AutoEncoder
ASSISTments 2009-2010	0.76 (0.61)	0.??	0.??	0.??	**0.78**
bridge to algebra 2006-2007	0.80 (0.70)	0.??	0.??	0.??	**0.85**

5.2.1　各知識分類の知識獲得予測における予測性能

ベースラインとなる既存の知識タグ (既存タグ) と，知識分類学習モデルから作成された知識タグ (深層学習タグ)，さらに，差分検証のための，深層学習モデルを用いずに通常の次元削減によって作成された知識タグ (事前圧縮タグ) を，各データセットから作成し，Deep Knowledge Tracing に適用した結果を表 5.1 に示す．marginal は各問題についてそれぞれ正解の周辺確率を予測結果とするものである．[Piech et al., 2015] にも記載されていたため，本稿でも同様にベースラインとして記載した．また，値が大きい箇所は太字で記載した．

いずれのデータセットにおいても，提案手法である「深層学習タグ (AutoEncoder)」が，最も高い AUC を記録した．

5.2.2　学習された知識分類の可視化と概観

最も良い予測性能を発揮した「深層学習タグ (AutoEncoder)」を可視化し，その概要を明らかにする．(ネットワークや対応関係と絡めて執筆予定)

5.2.3　学習された知識分類の比較分析

抽出された知識タグを既存の知識タグと比較することにより，抽出された知識タグの性質を定性的に確認する．

まず，各知識タグが回答ログに出現する頻度の分布や，紐づく問題数の分布に着目し，知識獲得予測の精度を向上させる要因を，データの構造から分析した．(図 xx) 図より，既存タグは各指標について分散が大きい一方で，抽出タグは各指標について分散が小さく，特定の値の周辺に集中している．この分布の違いと予測精度の関係性についても，次章で考察する．

次に，既存の DKT の手法に基づき作成された既存タグの知識間ネットワークに対し，既存タグと抽出タグの共起行列から TD-IDF 値を求めて作成した行列を元にノードとエッジを追加したネットワーク図を図 xx に示す．図より，表より，複数の抽出タグにおいて特徴的と判断されている既存タグが存在する一方，特定の抽出タグ内にまとめられ，他のタグでは特徴的でないと判断されている既存タグも存在する．この特徴付けにより，どのような性質のタグが抽出されているかについては，次章で考察する．

以上，実験について述べた．次章では，考察について述べる．

第6章　考察

本章では，実験結果を踏まえた考察を述べる．

まず，知識獲得予測の性能の比較実験の結果から，本研究で用いた知識分類学習モデルの有効性について考察する．

次に，同モデルにより抽出された知識タグと既存の知識タグを比較し，それぞれの性質や，知識獲得予測に与える影響について考察する．

また，本研究や関連研究が対象としたデータの範囲から，本研究の手法の教育学における他のデータへの適用可能性について考察する．

以上の考察を受けて，本研究の成果を実際の教育現場に適用し，活用する方法について考察する．

最後に，今後の展望として，本研究で用いた手法の教育学での適用の拡大について述べた後，最後に，教育学以外の分野への応用可能性について述べる．

6.1　知識分類学習モデルの有効性

知識獲得予測の性能の比較実験の結果から，本研究で用いた知識分類学習モデルの有効性について考察する．

まず，既存の知識タグを利用した場合と，各手法によって抽出したタグを利用した場合の，知識獲得予測の精度に関して考察する．

実験結果より，一般的な次元削減手法である，PCA や AutoEncoder のみを用いた場合は，既存タグを用いた場合よりも精度が悪かった一方で，知識分類学習モデルにより，知識獲得予測を行う過程で最適化したタグを用いた場合は，既存タグよりも精度の良いものがあった．

換言すれば，高次元の問題空間を圧縮する手法において，制約を設けずに単純に次元数の削減を行うだけでは知識獲得予測において有効な低次元表現は得られないが，知識獲得予測の精度も担保させるという制約を課した環境で次元数の削減を行うと，知識獲得予測において有効な低次元表現は得られることを意味している．

これは，直接的に解釈すると，目的のタスクに応じた環境情報を制約に加えることで，その環境と矛盾しない範囲で最適化が行われたということだが，より教育学的な解釈を試みれば，問題と知識分類は自明な関係ではなく，それを回答する生徒の正誤や知識獲得の推移という状態の観測を通して定義されることで，適切な分類が可能になると見なすことも可能である．

また，知識分類学習モデルを用いた手法においても，単純な低次元空間への埋め込みでは精度が向上しないものの，問題空間とタグ空間の再構成誤差を導入することにより，精度が向上した．これは，データの分量の不足に対する正則化項の導入という，ニューラルネットワークの文脈における，データの量的側面と，問題の正答・誤答と知識タグの理解状態は相互に変換できるはずだという，教育学の文脈における，データの質的側面と，双方の性質を活かす最適化の要素として，再構成誤差が効果を発揮したと考えられる．

6.2　各知識分類の性質と知識獲得予測に与える影響

次に，既存タグと抽出タグのそれぞれの性質を考察し，それがどのように知識獲得予測に影響をあたえるのかを考察する．

まず，既存タグと抽出タグの，問題回答ログにおける出現頻度と紐づく問題数の分布から，既存タグは分散が大きい一方で，抽出タグは分散が小さく，特定の値の周辺に集中していた．

既存タグは，人間の可読性や直感的なわかりやすさを重視して作られており，その知識を問う問題の出現頻度は作成時の評価軸にない．そのため，レベルの低い，基礎的な問題に関しては多くの生徒に回答される一方，よりレベルの高い，専門的であったり難易度の高い問題に関しては，回答される回数が必然的に少なくなるため，タグ間で出現頻度に差が出る．しかし，DKT のモデルに入力される際には，どの知識タグも均等に 1 つのユニットで表現されるため，実際の知識獲得過程において各タグが持つ情報量の偏りを十分に表現できない可能性が高い．

一方，抽出タグは，知識獲得予測を最適化する過程で学習されているため，DKT のモデル構造上，各ユニットが均等に情報量を保つことで，特定のタグに関する情報量が失われることを防いでいると考えられる．

この性質は，TF-IDF 法を用いた各抽出タグの特徴分析にも現れている．図 xx に表されるように，元々出現頻度が低い専門的な既存タグは，複数がまとめて一つの抽出タグに集約され，逆に元々出現頻度の高い基礎的な既存タグは，複数の抽出タグにまたがって表

現されるなど，より効率的に情報を保持できるタグ構造が抽出されていることがわかる．

さらに，こうした情報量の均等な分配構造は，内容と全く無関係に生成されるものではない．図 xx に表されるように，類似した内容の範囲内で情報量の分配を行っているタグがいくつか見受けられる．こうしたタグは，一種の関連知識一般を表している可能性があり，人間が知識を獲得していく過程を，これまでとは別の角度から検証する標となりうる．

6.3　本手法の他データへの適用可能性

本研究の分析手法の，他の科目やオンライン教育サービスへの適用可能性について考察する．

本研究の手法は，データセット作成という事前の処理と，1) 知識分類学習モデルによる知識分類の学習，2) 学習された知識分類の知識獲得予測性能に関する検証，3) 学習された知識分類の性質に関する比較分析という 3 つの分析から構成されていた．

データセット作成については，オンライン教育サービスから収集される問題回答ログデータは，サービスや科目によらず大規模であると考えられる．また，実験の「2) 学習された知識分類の知識獲得予測性能に関する検証」，および「3) 学習された知識分類の性質に関する比較分析」は，知識分類学習モデルによって，適切な知識分類を学習できるかに依存する．よって，本手法の他科目や他サービスへの適用可能性は,「1) 知識分類学習モデルによる知識分類の学習」に依存すると考えられる．知識分類学習モデルは DKT を拡張したモデル構造において学習されるため，DKT 自体の他科目や他サービスへの適用可能性によって，本手法の適用可能性も検証されると考えられる．

そこで，DKT の他科目や他サービスにおける予測性能を考察する．

[Piech et al., 2015] では，本研究同様，数学に関するデータセットにおいてのみ，DKT の有効性が検証されていた．那須野ら [参考文献引用] はリクルートが提供するオンライン教育サービス「学習サプリ」のデータを使って，算数や数学に関するデータセットと地理や歴史に関するデータセットに DKT を適用した場合，Bayesian Knowledge Tracing からの精度向上という点では大きな差はないことを確認しており，DKT の適用可能性は科目に依らない可能性がある．

また，[Piech et al., 2015] を始めとする既存研究では，モデルへの入力次元には問題に割り当てられたタグが利用されており，DKT の有効性はタグを用いた場合のみ，検証されていた．本研究の実験では，抽出されたタグを比較検証するために，既存のタグが存在するデータセットを用いたものの，問題回答のみから知識分類を学習できることは，実験

結果から示されており，このことから，既存のタグが存在しないような他科目や他サービスのデータに対しても，生徒の知識獲得を予測することが可能であることを示している．

一方で，これまで検証されているのは，特定の科目に関する問題回答ログであり，総合的な知識レベルを問うような，複数の科目が含まれている問題回答ログへの適用可能性は示されていない．

また，利用できるデータセットは，生徒が該当のオンライン教育サービスで学習する過程で，段階的に知識を獲得していく前提のデータのみであり，オンライン教育サービス外での学習や，生徒ごとの能力差，事前知識などの情報に関しては，DKT が扱うことは難しい可能性がある．

以上のような考察を踏まえると，本手法は，DKT が分析可能な他サービスや他科目のデータに加え，DKT による分析が困難な，事前の知識分類が存在しないデータに対しても適用できるという側面がある一方，複数科目のデータや生徒に関する事前情報など，現実に即した複雑な情報が多く含まれたデータに対しては，適用可能性が限定的である可能性もあり，検証が必要である．

6.4　教育現場への適用

ここまでの考察を踏まえ，本研究の成果を，実際の教育現場に適用し，活用する方法について考察する．(教材推薦システムと絡めて執筆予定)

6.5　今後の展望

本研究の今後の展望について大きく 2 つの方針を述べる．1 つは教育学における対象データの拡大についてであり，1 つは教育学以外の分野への本手法の応用についてである．

6.5.1　教育学における対象データの拡大

まず，教育学における対象データの拡大について述べる．対象データの拡大とは，科目や難易度の多様化，予測期間の長期化，そして複数科目の統合である．

まず，科目や難易度の多様化について述べる．本研究では．算数や数学の問題回答ログに対して深層学習を適用し，知識獲得予測に適した知識タグを得た．これまでの DKT の研究成果から，算数や数学以外の教科に対しても適用できる可能性は高いが，実際にどのような知識タグが抽出されるかは分析していない．また，今回扱ったデータセットは，小

学校から高校程度の算数や数学に関する問題回答であり，より高度で専門的な大学レベルの学問に適用する場合についても，どのような知識タグが抽出されるかは分析していない．知識獲得の最適化に関する知見は，学校側からの指導や生徒自身の学習設計に活用されており，多様な難易度や科目において知識獲得を最適化する知識構造を明らかにすることは，重要であると考えられる．

次に，予測期間の長期化について述べる．本研究で用いたデータセットの対象期間は，１～２年程度であった．しかし，DKT を用いた生徒の知識獲得の予測は，それまでの生徒の知識獲得の過程に依存しており，できるだけ長い期間の知識獲得を分析するほうが，より高い精度で知識獲得を予測でき，また，より適切な知識タグを抽出できる可能性は高い．

最後に，複数科目の統合について述べる．本研究や既存研究では，特定の科目について独立に知識獲得を予測し，知識構造を分析している．しかし，実際の生徒の学習の成長過程は，科目間で完全に独立であるとはいえず，例えば，歴史と地理や，数学と物理などの科目間では，知識獲得の過程が密接に関係している可能性がある．一人の生徒の，科目を横断した知識獲得過程に関する研究は，これまで報告されていないが，複数科目を統合したデータに対して本手法を適用することで，科目内の分類や学習設計だけでなく，包括的な学習設計や，既存の科目という分類にとらわれない，人間が獲得する知識一般に関する知見も得られる可能性がある．

6.5.2　教育学以外の分野への応用

次に，本手法の教育学以外の分野への応用について述べる．

本手法は，Knowledge Tracing という，教育学の，知識獲得予測タスクにおける手法だが，より手法を一般化することで，教育学以外の分野にも応用できる可能性を秘めている．

本手法は，生徒の時系列問題回答ログから，回答を重ねるごとに遷移していく知識状態をモデリングし，知識獲得の過程を適切に表現する知識タグを抽出するというものだが，

これをより一般化して捉えると，人間の，何らかのコンテンツ集合に対する時系列行動ログから，行動を重ねるごとに遷移していく人間の何らかの状態をモデリングし，行動の遷移を適切に説明する分類表現を抽出しているといえる．

教育学の知識獲得の分野においては，このコンテンツ集合に対する行動が生徒の問題回答であり，問題回答により遷移する生徒の知識状態をモデリングしているが，これと同様のことは，教育学に限らず行える可能性がある．

例えば，消費者が商品を購買するログを分析することで，消費者の嗜好が遷移する過程

をモデリングし，従来の商品分類と異なる，消費者の嗜好の遷移を反映した分類を抽出することが可能になる．教育学では，問題回答の正誤と知識の間の特殊な関係性をモデル設計に反映しているように，手法を適用する領域によって調整は必要であるが，コンテンツに対する行動の時系列性を反映した分類を抽出できる可能性は高く，様々な領域で，学術的にも，実用的にも，価値の高い知見を得られると考えられる．

以上，考察について述べた．次章では，結論を述べる．

第7章　結論

本論文では，既存の知識獲得予測における問題として，人間が作成した知識分類を利用していることを指摘し，深層学習を適用することによって，知識獲得の予測において最適化された知識分類を抽出できることを検証した

また，抽出された知識分類を定性的に分析することにより，xxx という知見を得た．

本研究の成果を実際の教育現場で活用する例の一つとして，教材支援システムへの適用を論じた．

また，本研究の分析手法の他の科目やオンライン教育サービスへの適用可能性について議論し，科目や既存の知識分類の有無によらずに最適な知識分類を学習して知識獲得を予測できる可能性があること，および，複数科目を統合した知識獲得や，大学水準の知識獲得に関する分析については，検証実験を行う必要が有ることを述べた．

さらに，本研究の拡張として，適用対象の拡張という点で対象データの多様化，対象期間の長期化，複数科目の統合，の 3 つの拡張を述べ，また，より一般性を高めることで，教育学以外の領域においても本手法が適用できる可能性に触れ，人間行動に関するの多様な知見を発見できる可能性を論じた．

本研究は．オンライン教育サービスの普及や，教育分野における大規模分析の活発化，深層学習の躍進など，ここ数年の多様な領域の進展によって初めて可能になったものである．本研究が，既存の学問体系の再構築，そして人間の学習や知識の解明につながると信じている．

参考文献

[Abelson, 2008] Abelson, H. (2008). The creation of opencourseware at mit. *Journal of Science Education and Technology*, 17(2):164–174.

[Aleven et al., 2015] Aleven, V., Sewall, J., Popescu, O., Xhakaj, F., Chand, D., Baker, R., Wang, Y., Siemens, G., Rosé, C., and Gasevic, D. (2015). The beginning of a beautiful friendship? intelligent tutoring systems and moocs. In *International Conference on Artificial Intelligence in Education*, pages 525–528. Springer.

[Bahdanau et al., 2015] Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P., and Bengio, Y. (2015). End-to-end attention-based large vocabulary speech recognition. *arXiv preprint arXiv:1508.04395*.

[Bastien et al., 2012] Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I. J., Bergeron, A., Bouchard, N., and Bengio, Y. (2012). Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop.

[Bellman and Corporation, 1957] Bellman, R. and Corporation, R. (1957). *Dynamic Programming*. Rand Corporation research study. Princeton University Press.

[Bengio et al., 1994] Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *Neural Networks, IEEE Transactions on*, 5(2):157–166.

[Bergstra and Bengio, 2012] Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, 13(1):281–305.

[Bergstra et al., 2010] Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., and Bengio, Y. (2010). Theano: a

CPU and GPU math expression compiler. In *Proceedings of the Python for Scientific Computing Conference (SciPy)*. Oral Presentation.

[Biswas et al., 2015] Biswas, S., Chadda, E., and Ahmad, F. (2015). Sentiment analysis with gated recurrent units. *Advances in Computer Science and Information Technology*.

[Bloom, 1968] Bloom, B. S. (1968). Learning for mastery. instruction and curriculum. regional education laboratory for the carolinas and virginia, topical papers and reprints, number 1. *Evaluation comment*, 1(2):n2.

[Chen et al., 2008] Chen, N.-S., Wei, C.-W., Chen, H.-J., et al. (2008). Mining e-learning domain concept map from academic articles. *Computers & Education*, 50(3):1009–1021.

[Cho et al., 2014] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

[Choi et al., 2015] Choi, E., Bahadori, M. T., and Sun, J. (2015). Doctor ai: Predicting clinical events via recurrent neural networks. *arXiv preprint arXiv:1511.05942*.

[Chung et al., 2014] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

[Chung et al., 2015] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2015). Gated feedback recurrent neural networks. *arXiv preprint arXiv:1502.02367*.

[Clevert et al., 2015] Clevert, D.-A., Unterthiner, T., and Hochreiter, S. (2015). Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*.

[Corbett and Anderson, 1994] Corbett, A. T. and Anderson, J. R. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278.

[Dong et al., 2015] Dong, D., Wu, H., He, W., Yu, D., and Wang, H. (2015). Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*. ACL.

[Erhan et al., 2010] Erhan, D., Bengio, Y., Courville, A., Manzagol, P.-A., Vincent, P., and Bengio, S. (2010). Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, 11(Feb):625–660.

[FALAKMASIR et al., 2015] FALAKMASIR, M., Yudelson, M., Ritter, S., and Koedinger, K. (2015). Spectral bayesian knowledge tracing. In *Proceedings of the 8th International Conference on Educational Data Mining., OC Santos, JG Boticario, C. Romero, M. Pechenizkiy, A. Merceron, P. Mitros, JM Luna, C. Mihaescu, P. Moreno, A. Hershkovitz, S. Ventura, and M. Desmarais, Eds. Madrid, Spain*, pages 360–364.

[Friedman, 1997] Friedman, J. H. (1997). On bias, variance, 0/1—loss, and the curse-of-dimensionality. *Data mining and knowledge discovery*, 1(1):55–77.

[Glorot and Bengio, 2010] Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *International conference on artificial intelligence and statistics*, pages 249–256.

[Graves and Schmidhuber, 2009] Graves, A. and Schmidhuber, J. (2009). Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 545–552.

[Hidasi et al., 2015] Hidasi, B., Karatzoglou, A., Baltrunas, L., and Tikk, D. (2015). Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

[Hinton et al., 2012] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97.

[Hinton and Salakhutdinov, 2006] Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507.

[Hochreiter, 1998] Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6(02):107–116.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[Karpathy et al., 2015] Karpathy, A., Johnson, J., and Li, F.-F. (2015). Visualizing and understanding recurrent networks. *arXiv preprint arXiv:1506.02078*.

[Keller, 1968] Keller, F. S. (1968). Good-bye, teacher... *Journal of applied behavior analysis*, 1(1):79.

[Kingma and Ba, 2014] Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Kingma et al., 2014] Kingma, D. P., Mohamed, S., Rezende, D. J., and Welling, M. (2014). Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, pages 3581–3589.

[Krueger and Memisevic, 2015] Krueger, D. and Memisevic, R. (2015). Regularizing rnns by stabilizing activations. *arXiv preprint arXiv:1511.08400*.

[Ku et al., 1995] Ku, W., Storer, R. H., and Georgakis, C. (1995). Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics and intelligent laboratory systems*, 30(1):179–196.

[Kushner and Yin, 2003] Kushner, H. J. and Yin, G. (2003). *Stochastic approximation and recursive algorithms and applications*, volume 35. Springer Science & Business Media.

[Lage et al., 2000] Lage, M. J., Platt, G. J., and Treglia, M. (2000). Inverting the classroom: A gateway to creating an inclusive learning environment. *The Journal of Economic Education*, 31(1):30–43.

[Larochelle et al., 2007] Larochelle, H., Erhan, D., Courville, A., Bergstra, J., and Bengio, Y. (2007). An empirical evaluation of deep architectures on problems with many factors of variation. In *Proceedings of the 24th international conference on Machine learning*, pages 473–480. ACM.

[Le et al., 2015] Le, Q. V., Jaitly, N., and Hinton, G. E. (2015). A simple way to initialize recurrent networks of rectified linear units. *arXiv preprint arXiv:1504.00941*.

[Lipton et al., 2015] Lipton, Z. C., Kale, D. C., Elkan, C., and Wetzell, R. (2015). Learning to diagnose with lstm recurrent neural networks. *arXiv preprint arXiv:1511.03677*.

[Liyanagunawardena et al., 2013] Liyanagunawardena, T., Williams, S., and Adams, A. (2013). The impact and reach of moocs: A developing countries' perspective. *eLearning Papers*, (33).

[Louradour and Kermorvant, 2014] Louradour, J. and Kermorvant, C. (2014). Curriculum learning for handwritten text line recognition. In *Document Analysis Systems (DAS), 2014 11th IAPR International Workshop on*, pages 56–60. IEEE.

[MacHardy and Pardos, 2015] MacHardy, Z. and Pardos, Z. A. (2015). Toward the evaluation of educational videos using bayesian knowledge tracing and big data. In *Proceedings of the Second (2015) ACM Conference on Learning@ Scale*, pages 347–350. ACM.

[McAuley et al., 2010] McAuley, A., Stewart, B., Siemens, G., and Cormier, D. (2010). The mooc model for digital practice.

[Mikolov, 2012] Mikolov, T. (2012). Statistical language models based on neural networks. *Presentation at Google, Mountain View, 2nd April*.

[Nair and Hinton, 2010] Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814.

[Pappano, 2012] Pappano, L. (2012). The year of the mooc. *The New York Times*, 2(12):2012.

[Pascanu et al., 2013] Pascanu, R., Mikolov, T., and Bengio, Y. (2013). On the difficulty of training recurrent neural networks. *arXiv preprint arXiv:1211.5063*.

[Pavlik Jr et al., 2009] Pavlik Jr, P. I., Cen, H., and Koedinger, K. R. (2009). Performance factors analysis–a new alternative to knowledge tracing. *Online Submission*.

[Pearson, 1901] Pearson, K. (1901). Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

[Pezeshki, 2015] Pezeshki, M. (2015). Sequence modeling using gated recurrent neural networks. *arXiv preprint arXiv:1501.00299*.

[Piccioli, 2014] Piccioli, V. (2014). E-learning market trends & forecast 2014-2016 report. *Athens (GA)-USA*.

[Piech et al., 2015] Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L. J., and Sohl-Dickstein, J. (2015). Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, pages 505–513.

[Robbins and Monro, 1951] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407.

[Sak et al., 2015] Sak, H., Senior, A., Rao, K., and Beaufays, F. (2015). Fast and accurate recurrent neural network acoustic models for speech recognition. *arXiv preprint arXiv:1507.06947*.

[Schölkopf et al., 1997] Schölkopf, B., Smola, A., and Müller, K.-R. (1997). Kernel principal component analysis. In *International Conference on Artificial Neural Networks*, pages 583–588. Springer.

[Schroff et al., 2015] Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. *arXiv preprint arXiv:1503.03832*.

[Siemens, 2013] Siemens, G. (2013). Massive open online courses: Innovation in education. *Open educational resources: Innovation, research and practice*, 5.

[Silver et al., 2016] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489.

[Srivastava et al., 2014] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

[Stamper et al., 2010] Stamper, J., Niculescu-Mizil, A., Ritter, S., Gordon, G., and Koedinger, K. (2010). Bridge to algebra 2006-2007. development data set from kdd cup 2010 educational data mining challenge. http://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp.

[Sutskever et al., 2014] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

[Szegedy et al., 2014] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2014). Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*.

[Tetko et al., 1995] Tetko, I. V., Livingstone, D. J., and Luik, A. I. (1995). Neural network studies. 1. comparison of overfitting and overtraining. *Journal of chemical information and computer sciences*, 35(5):826–833.

[Tipping and Bishop, 1999] Tipping, M. E. and Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622.

[Trucano et al., 2013] Trucano, M., Kendrick, C., and Gashurov, I. (2013). More about moocs and developing countries.

[Upbin, 2012] Upbin, B. (2012). Knewton is building the world ' s smartest tutor.

[Vincent et al., 2008] Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In

Proceedings of the 25th international conference on Machine learning, pages 1096–1103. ACM.

[Vinyals et al., 2014] Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2014). Show and tell: A neural image caption generator. *arXiv preprint arXiv:1411.4555*.

[Vondrick et al., 2016] Vondrick, C., Pirsiavash, H., and Torralba, A. (2016). Generating videos with scene dynamics. In *Advances In Neural Information Processing Systems*, pages 613–621.

[Werbos, 1990] Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560.

[Williams and Zipser, 1989] Williams, R. J. and Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280.

[Wold et al., 1987] Wold, S., Esbensen, K., and Geladi, P. (1987). Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52.

[Wu et al., 2016] Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

[Xiong et al., 2016a] Xiong, W., Droppo, J., Huang, X., Seide, F., Seltzer, M., Stolcke, A., Yu, D., and Zweig, G. (2016a). Achieving human parity in conversational speech recognition. *arXiv preprint arXiv:1610.05256*.

[Xiong et al., 2016b] Xiong, X., Zhao, S., Van Inwegen, E. G., and Beck, J. E. (2016b). Going deeper with deep knowledge tracing. In *Proceedings of the 9th International Conference on Educational Data Mining (EDM 2016)*, pages 545–550.

[Xu et al., 2015] Xu, K., Ba, J., Kiros, R., Courville, A., Salakhutdinov, R., Zemel, R., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*.

[Yin et al., 2015] Yin, J., Jiang, X., Lu, Z., Shang, L., Li, H., and Li, X. (2015). Neural generative question answering. *arXiv preprint arXiv:1512.01337*.

[Yuan et al., 2013] Yuan, L., Powell, S., and CETIS, J. (2013). Moocs and open education: Implications for higher education.

[Yudelson et al., 2013] Yudelson, M. V., Koedinger, K. R., and Gordon, G. J. (2013). Individualized bayesian knowledge tracing models. In *Artificial Intelligence in Education*, pages 171–180. Springer.

[Zaremba, 2015] Zaremba, W. (2015). An empirical exploration of recurrent network architectures.

[Zeiler, 2012] Zeiler, M. D. (2012). Adadelta: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

[重田勝介, 2014] 重田勝介 (2014). 反転授業 ict による教育改革の進展. **情報管理**, 56(10):677–684.

謝辞

本研究の遂行や本論文の執筆にあたり，非常に多くの方からご指導，ご支援をいただきました．心より御礼申し上げます．

指導教官である 松尾豊特任准教授 には，研究構想の相談や論文の書き方，本論文の論理構成について，貴重なご指導をいただきました．ここに，深く謝意を表します．

分析サーバや GPU 解析環境の用意等，物理的な研究環境の構築に多大なご協力を下さった研究室の教官である 中山浩太郎先生 に，深く感謝致します．

上野山克也助教授には，研究の方向性や論文の構成について，多大なご協力をいただきました．深く感謝致します．

松尾研究室や GCI の皆様には，多大なご協力，ご支援いただきました．秘書の 中野佐恵子さん，永本登代子さん，浪岡亮子さん，木全やえさんは，日頃から研究室の環境を整えて下さり，研究生活を支えてくださいました．松尾研究室の博士・修士課程の先輩である 岩澤有祐さん，飯塚修平さん，野中尚輝さん，鈴木雅大さん，金子貴輝さん，那須野薫さん，黒滝さん，保住さん，冨山翔司さんには，未熟な自分の研究の相談に幾度も乗っていただき，多大なご助力をいただきました．特に，研究テーマや，研究全体の設計について何度も相談に応じていただいたことに加え，深層学習の技術的な問題に関する相談にも幾度となく乗っていただき，多大なご助力を頂いた那須野薫さんには，深く感謝致します．研究室同期である大野氏，田村浩一郎氏は，卒業論文の構想や執筆に関して率直に意見を交わし，互いに切磋琢磨し合いながら研究を進めさせていただきました．

ここに，松尾研究室の皆様へ謝意を表します．

東京大学工学部

システム創成学科知能社会コース

松尾研究室　学部四年

中川大海

平成 29 年 3 月