

電気通信大学大学院博士前期課程
2022年度 情報理工学研究科 修士論文

同一項目への過去反応を考慮する
Knowledge Tracing

電気通信大学大学院 情報理工学研究科
情報・ネットワーク工学専攻 情報数理工学プログラム

学籍番号 2131091

関口 昌平

主任指導教員 植野 真臣 教授

副指導教員 宇都 雅輝 准教授

2023年1月30日

目次

1	はじめに	2
2	確率モデルを用いた既存モデル	5
2.1	Bayesian Knowledge Tracing(BKT)	5
2.2	項目反応理論	5
3	深層学習を用いた既存モデル	7
3.1	A Self-Attentive model for Knowledge Tracing	7
3.2	Attentive Knowledge Tracing	8
3.3	Deep Knowledge Tracing	10
3.4	Dynamic Key-Value Memory Network	11
3.5	Deep-IRT	13
3.6	DeepIRT-HN	15
4	提案手法	18
4.1	提案手法概要	18
4.2	パラメータ学習	19
5	評価実験	21
5.1	実験データ	21
5.2	反応予測精度比較	22
5.3	パラメータ δ の分析	24
6	むすび	26
7	謝辞	27

1 はじめに

近年、オンライン教育の普及に伴い、大量の学習履歴データが入手できるようになった。教育現場では学習者が効率的に学習できるように個々の学習者に適切な支援を行うアダプティブラーニングが注目されている。人工知能分野では機械学習手法を用いて過去の学習履歴データから学習者の能力を推定し、未知の項目への反応予測を行う Knowledge Tracing(KT) が注目を集めている。[1, 2, 3, 4, 5, 6, 13, 14, 15, 16, 17, 18, 19] 学習者の項目への反応を予測することにより、教師は学習者ごとに適した項目の出題や学習支援を行うことができる。

KT 手法には確率モデルを用いた確率的アプローチと深層学習を用いたディープラーニングアプローチがある。確率アプローチの代表的な手法には Bayesian Knowledge Tracing[20, 21, 22, 23, 24] や Item Response Theory(IRT)[25, 26, 27, 28, 29, 30] があり、学習履歴データからあるスキルに対する学習者の能力値を推定することで未知の項目への正答確率を予測することができる。確率的アプローチは学習者の能力値の他に項目の難易度を示すパラメータをもつため解釈性が高く、多くの学習支援システムで用いられている。しかし確率的アプローチで推定される能力値はスキルごとに独立であり、多次元のスキルの関係性を考慮した能力推定ができない。また IRT 手法では学習者の各項目への反応は独立であることが仮定されているが、アダプティブラーニングにおける学習では一つの項目に複数回取り組む場合が多く、過去の反応が次の反応に影響を与える場合がある。そのため、KT においては正確な能力推定が行われない可能性がある。

近年、多次元スキルにおける学習者の能力変化を考慮し、未知の項目への反応予測精度を向上させるために、多くのディープラーニングアプローチが開発されている [6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 39, 40]。ディープラーニングアプローチの代表的な手法として Deep Knowledge Tracing(DKT)[6] が提案されている。DKT は RNN を用いて学習者の能力変化を表現し、学習者の項目への反応を推定するモデルである。DKT では RNN の隠れ層に全ての多次元スキルに対する学習者の能力値を圧縮して反応予測を行うため、BKT 手法と比較して反応予測精度が高いことが

報告されている。

さらに DKT の予測精度を向上させるために、Dynamic Key-Value Memory Network(DKVMN) が提案されている [14]。DKVMN は Memory Network をに学習者の多次元の潜在能力値を表す潜在変数を持ち、時点ごとに忘却パラメータを用いて更新することで能力の時系列変化を表現する。DKVMN は高い予測精度を示すが、DKT と同様に学習者の能力パラメータをもたないためモデルの解釈性が低いという問題があった。そこで、DKT や DKVMN のパラメータ解釈性を向上させるための手法として、DKVMN と IRT を組み合わせた手法として Deep-IRT[13] が提案されている。Deep-IRT は DKVMN と同様に高い予測精度を保ちながら、学習者の能力パラメータや項目の難易度パラメータをもつことで解釈可能性を向上させたことで注目を集めている。

しかし Deep-IRT では能力値を解答した項目の情報と学習者の潜在能力値から推定していたため、能力値は項目の特性に依存しており、能力値の解釈が困難であった。そこで Tsutsumi ら [19] の研究では、能力値と項目の特性を区別するために、ネットワークを独立させることでパラメータの解釈精度を向上させた。また従来の DKVMN、DeepIRT 手法 [14, 13, 19] では学習者の潜在能力値を更新する際、最新の反応データのみを用いて忘却パラメータを推定していた。そのため、過去の潜在能力値の情報を考慮した忘却パラメータの推定が行われず、学習者の反応予測精度が低下している可能性があった。そこで、Tsutsumi ら [39, 40] では忘却パラメータを最適化するために Deep-IRT[13] にハイパーネットワーク [41] を組み合わせ、最新の項目への反応データだけでなく学習者の過去の潜在能力値を考慮することで、潜在能力値を適切に更新することが可能になり高い解釈性と反応予測精度を示した。

さらに近年、新たなディープラーニングアプローチでは主に自然言語処理の分野で高い汎化性能が示されている Self-Attention を用いた Transformer が利用されている [38]。Transformer は入力データ間に対して Attention を計算することで長期間において強い依存関係のあるデータの予測に有効であることが知られている。そこで KT のために、Self-Attention を用いた Self Attentive Knowledge Tracing(SAKT)[37] が開発されている。SAKT は現在の学習者の反応予測に過去の学習データが大きく関係することに注目し、学習者の過去の全ての学習データに対して Attention を計算

することで反応予測を行う。これに対し、Ghosh らは学習者の現在の反応は全ての学習データに依存せず、直近の短い期間の学習結果に特に依存することを主張した [18]。そこで、彼らは過去の学習履歴を徐々に忘却することで直近の学習に大きく関係する学習履歴をより考慮するように Attention を計算する新たな Attention モデル、Attentive Knowledge Tracing(AKT) を提案している。この手法により従来のディープラーニングアプローチと比較して学習者の反応予測精度が向上することが示された。

アダプティブラーニングでは学習者が同じ項目に繰り返し解答する場合が存在する。そのため学習者の学習履歴によって項目の難易度が変わる可能性がある。例えば過去にある項目に解答し、正答している場合には、次に同一項目に取り組む際の難易度は低下すると考えられる。しかし従来の DKVMN や DeepIRT 手法 [14, 13, 19, 39, 40] では全期間において共通のネットワークを用いて難易度を推定するため、同一項目の難易度は常に一定である。そのため学習者の学習履歴が及ぼす項目の難易度変化を考慮することが出来ない。AKT では類似項目の出題を考慮するために入力データに位置情報を追加する Positional Encoding を行うが、パラメータ学習に用いるデータ数が不十分であるために予測精度が低下する問題がある。

本研究では過去に同一項目に解答することによる項目難易度の変化を考慮するため、学習履歴による従属性の影響を示すパラメータ δ を導入した新たな DeepIRT 手法を提案する。提案手法では時系列モデルである RNN に学習者の学習履歴データを入力することで過去に解いた項目を考慮し、ある時点に出題された項目の難易度の変化を表すパラメータを推定することで従属性による難易度の影響を表現する。 δ によって項目の難易度を変化させることで、学習者の過去の反応を反映した反応予測が可能となる。最後に提案手法と既存手法 (AKT、DeepIRT-HN)[18, 39] との学習者の反応予測精度の比較を行い、提案手法の有効性を示す。

2 確率モデルを用いた既存モデル

2.1 Bayesian Knowledge Tracing(BKT)

BKT[1] は学習者の学習過程における知識状態の変化を隠れマルコフモデルで表現し、未知の課題への正答確率を予測するモデルであり、反応データから学習者の能力の習熟度を推定する。

BKT では学習者 i の項目 j に対する知識の習得状態を $Z_{i,j}$ で表す。

$$Z_{i,j} = \begin{cases} 0 & \text{学習者 } i \text{ が項目 } j \text{ に必要な能力を習得していない} \\ 1 & \text{学習者 } i \text{ が項目 } j \text{ に必要な能力を取得している} \end{cases}$$

知識状態の遷移確率パラメータとして $p(L_0)$: 学習者が事前に能力を習得している確率、 $p(T)$: 能力を習得していない状態から習得する確率、 $p(S)$: 能力を習得している状態で項目に誤答する確率、 $p(G)$: 能力を習得していない状態で項目に正答する確率の 4 つのパラメータをもち、学習者が能力を習得する確率を推定することで、学習者 i が項目 j に正答する確率を求める。BKT では学習者の知識の習得状態を 0 か 1 の離散値で表現しているので、スキルごとの習熟度を表現することができない。また標準的な BKT では学習者の知識の忘却が起こらないと仮定しているため、学習者の能力の習熟度の推移を表現することができないという問題がある。

BKT では学習者ごとに $p(L_0)$ 、 $p(T)$ を最適化したモデル [5] や習得状態に段階的な離散値を用いることで学習者の習得状態を詳細に表現したモデル [21] といった拡張モデルが存在している。しかし BKT は能力を一次元で表現しており、独立した多次元のスキルを考慮することができない。

2.2 項目反応理論

項目反応理論 (IRT)[25] には多くのモデルが提案されている。ここでは IRT モデルの中でも特に有名な 2 パラメータロジスティックモデル (2PLM) を紹介する。2PLM では学習者 i の項目 j への反応 $u_{i,j}$ を次のように表現する

$$u_{i,j} = \begin{cases} 0 & \text{学習者 } i \text{ が項目 } j \text{ に誤答した場合} \\ 1 & \text{学習者 } i \text{ が項目 } j \text{ に正答した場合} \end{cases}$$

学習者 i の能力を示すパラメータを θ_i とし、与えられた項目 i に正解する確率を次の式で表現する。

$$P_j(\theta_i) = P(u_{i,j} = 1|\theta_i) = \frac{1}{1 + \exp(-1.7 \times a_j(\theta_i - b_j))} \quad (1)$$

a_j は項目 j の識別パラメータで、項目における生徒の能力に対する識別力を表し、 b_j は項目 j の難易度を表す。

しかし、標準的な IRT モデルでは学習過程を通して学習者の能力は一定であると仮定されている。近年では、学習者の能力値の時系列変化を推定するために時系列 IRT モデルが多数提案されている [31, 32, 34, 35, 36]。また IRT では能力が一次元であることを仮定しているため、独立した多次元スキルを考慮することができない。そのため異なる能力を必要とする項目に対応できない問題がある。

3 深層学習を用いた既存モデル

本章では深層学習を用いた KT 手法を紹介する

3.1 A Self-Attentive model for Knowledge Tracing

深層学習の分野では Attention のみを用いた Transformer[38] が注目されており、自然言語処理の分野で高い汎化性能が示されている。Transformer は入力データ間の内積を計算することで入力データ間の関連性を計算する。そのため従来の再帰処理を用いた RNN の手法では捉えにくい長期間にわたるデータ間の依存関係が捉えやすいことが示されている。教育工学の分野においても学習者の学習データは長期間にわたり依存関係のあるデータであるため、Transformer を活用したモデルが提案されている。Pandey[37] らは学習過程において、学習者の反応には過去の学習履歴が大きく関係することに注目し、Transformer を用いて学習者の過去のすべての学習データに対して Attention を計算することで反応予測を行う A Self-Attentive model for Knowledge Tracing(SAKT) 手法を提案した。

SAKT では学習者の項目 q_j に対する正答 $u_{i,j}$ を項目数 J を用いて $v_t = q_j + u_{i,j} \times J$ と表現する。 $V = (v_1, v_2, \dots, v_t), Q = (q_1, q_2, \dots, q_t)$ をエンベディングし、行列 \mathbf{V}, \mathbf{Q} を得る。

次に出題順序を示す行列 \mathbf{P} を用いて $\hat{\mathbf{V}}, \hat{\mathbf{Q}}$ を計算する。この時、 \mathbf{P}_i は i 番目に出題された項目の位置を表すベクトルである。

$$\hat{\mathbf{V}} = \begin{bmatrix} \mathbf{V}_{v_1} + \mathbf{P}_1 \\ \mathbf{V}_{v_2} + \mathbf{P}_2 \\ \dots \\ \mathbf{V}_{v_n} + \mathbf{P}_n \end{bmatrix}, \hat{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q}_{q_1} \\ \mathbf{Q}_{q_2} \\ \dots \\ \mathbf{Q}_{q_n} \end{bmatrix} \quad (2)$$

求められた行列を用いて次の計算により時点 t の項目と過去の項目の関連性を計算する。Attention を用いることで過去の学習履歴の中から現在の反応予測に関連性のある情報を用いることができる。また d は Π_Q の次元数を示す。

$$\Pi_Q = \hat{Q}W_{Q1}, \Pi_K = \hat{V}W_{K1}, \Pi_V = \hat{V}W_{V1}, \quad (3)$$

$$Attention(\Pi_Q, \Pi_K, \Pi_V) = Softmax\left(\frac{\Pi_Q \Pi_K^T}{\sqrt{d}}\right) \Pi_V. \quad (4)$$

また、SAKT では Transformer で用いられている Multihead Attention[38] を用いることで予測精度を高めている。Multihead Attention は同一構造の Attention を複数個並列に計算し、それぞれの出力を結合し重みを書けている。Multihead Attention は n 番目の Attention を $head_n = Attention(\hat{Q}W_{Q_n}, \hat{V}W_{K_n}, \hat{V}W_{V_n})$ とし、次の式で表す。

$$Multihead(\hat{V}, \hat{Q}) = [head_1, \dots, head_n]W_M, \quad (5)$$

$$(6)$$

次に $d_0 = Multihead(\hat{V}, \hat{Q})$ を用いて次のように予測反応確率を推定する。

$$d_1 = ReLU(W_{d0}d_0 + b_{d0}), \quad (7)$$

$$d_2 = W_{d1}d_1 + b_{d1}, \quad (8)$$

$$p_t = Singmoid(W_{d2}d_2 + b_{d2}). \quad (9)$$

しかし、一般に Transformer が用いられる自然言語分野に比べ KT のデータセットは小さくスパースであるため、学習を十分に行えないという課題がある。

3.2 Attentive Knowledge Tracing

SAKT では現在の学習者の反応は過去の全ての学習データに依存していると仮定して反応予測が行われていた。一方 Ghosh[18] らは学習過程における学習者の反応は、過去の全てのデータに依存するのではなく、直近の短い期間の学習に特に依存すると仮定し、直前のデータに注目する新たな手法 Attentive Knowledge Tracing(AKT) を提案した。AKT は過去の学習データを徐々に忘却することで、反応予測に大きく関係する直近の学習データを重視するように Attention の値を計算する。

AKT には入力データから特徴量を計算する Question Encoder、Knowledge Encoder と計算された特徴量から反応予測を行う Knowledge Retriever が存在する。

AKT では入力データ $\{\phi_1, \dots, \phi_t\}$ を用いて以下の式で特徴量 $\{\hat{\phi}_1, \dots, \hat{\phi}_t\}$ を計算する。

まず ϕ_t と ϕ_τ から $Query_t, Key_\tau, Value_\tau$ の値を推定する

$$Query_t = W_{Q2}\phi_t, \quad (10)$$

$$Key_\tau = W_{K2}\phi_\tau, \quad (11)$$

$$Value_\tau = W_{V2}\phi_\tau. \quad (12)$$

これらの値を用いて $\alpha_{t,\tau}$ を計算し、特徴量ベクトル \hat{x}_t を計算する。この時 $\tau \leq t$ である。また $\sqrt{D_k}$ は Key_τ ベクトルの次元数を表す。

$$\alpha_{t,\tau} = \text{Softmax} \left(\frac{Query_t^\top Key_\tau}{\sqrt{D_k}} \right) \in [0, 1], \quad (13)$$

$$\hat{\phi}_t = \sum_{\tau} \alpha_{t,\tau} Value_\tau \quad (14)$$

これらの式を用いて Question Encoder では $\{q_1, \dots, q_t\}$ から $\{\hat{q}_1, \dots, \hat{q}_t\}$ を推定し、Knowledge Encoder では $\{v_1, \dots, v_{t-1}\}$ から $\{\hat{v}_1, \dots, \hat{v}_{t-1}\}$ を推定している。

次に計算された特徴量を用いて反応予測を行う。Knowledge Retriever では直近のデータを重視するために、過去のデータを徐々に忘却する Monotonic Attention を計算する。Monotonic Attention は以下の式で計算する。

$$\mu_{t,\tau} = \frac{\exp(-\eta \cdot d(t, \tau)) \cdot W_{Q2}\hat{q}_t^\top W_{K2}\hat{q}_\tau}{\sqrt{D_k}}, \quad (15)$$

$$\alpha_{t,\tau} = \frac{\exp(\mu_{t,\tau})}{\sum_{\tau'} \exp(\mu_{t,\tau'})}. \quad (16)$$

$$(17)$$

ここで $\eta > 0$ は減衰の割合を決めるパラメータである。また $d(t, \tau)$ は時点 t と τ の間のデータによって計算される値で次の式で計算される。

$$d(t, \tau) = |t - \tau| \cdot \sum_{t'=\tau+1}^t \psi_{t,t'}, \quad (18)$$

$$\psi_{t,t'} = \frac{\exp\left(\frac{W_{Q_2} \hat{q}_t^\top W_{K_2} \hat{q}_{t'}}{\sqrt{D_k}}\right)}{\sum_{1 \leq \tau' \leq t} \exp\left(\frac{W_{Q_2} \hat{q}_t^\top W_{K_2} \hat{q}_{\tau'}}{\sqrt{D_k}}\right)} \quad (19)$$

このように Monotonic Attention では反応予測に強い影響を与える直近のデータを重視することができる。

また、従来の KT 手法では同じスキルをもつ項目同士は等価であるために項目の特性が考慮されていないことを指摘しており、この問題を解決するために AKT ではスキル s_j と項目 q_j の両方を入力する。その結果 AKT は従来の KT 手法に比べ優れた予測精度を達成した。しかし学習者の各スキルにおける能力値の推移を表すことができないため、パラメータの解釈性が低いという問題がある。

3.3 Deep Knowledge Tracing

Deep Knowledge Tracing(DKT)[6] は学習者の未知の項目に対する反応を予測する深層学習モデルである。DKT は時点 t に解答した項目 q_t に対応する反応を one-hot ベクトルを用いて時点 $t + 1$ に解答する未知の項目への反応予測を行う。項目への予測正答確率 p_{t+1} は次の式で求める。

$$h_t = \tanh(W_{hx}q_t + W_{hh}h_{t-1} + b_h), \quad (20)$$

$$p_{t+1} = \sigma(W_{yh}h_t + b_y). \quad (21)$$

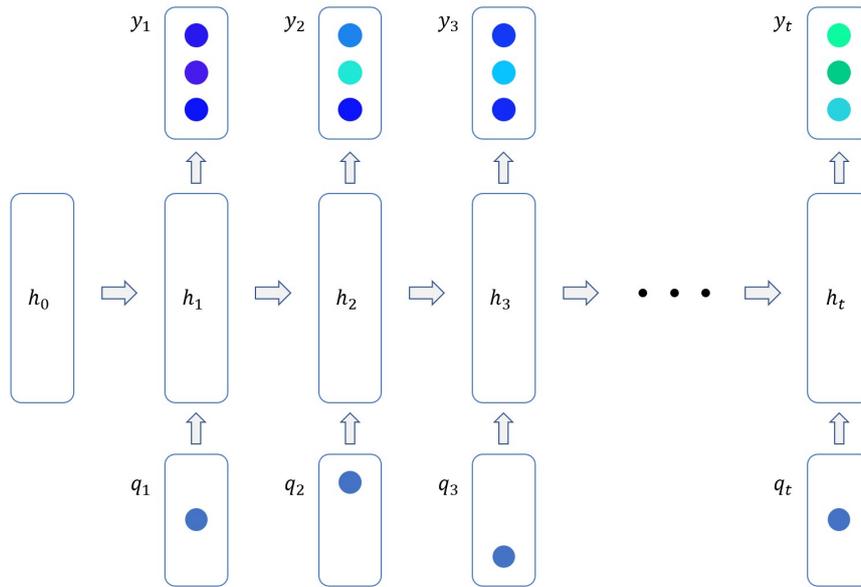


図1 DKT のモデル図

DKT では学習者の複数のスキル能力を h_t という単一の隠れ状態ベクトルに保持するため、学習者におけるスキルごとの能力変化を表現できないという問題がある。

3.4 Dynamic Key-Value Memory Network

DKT の反応予測精度を向上させるために多くの深層学習手法が提案されている [7, 8, 9, 10, 11, 12]。Dynamic Key-Value Memory Network(DKVMMN)[14] は学習者の多次元スキルに対する能力値を保持するための Memory Network を用いたモデルである。DKVMMN では、学習過程において N 個の潜在的なスキルに対応する能力が存在すると仮定し、スキルごとの独立したベクトルに保持する。さらに各時点で出題された項目 q_j と潜在スキルの関係を推定することで、項目と学習者の能力の関係性を考慮しながら反応予測を行う。

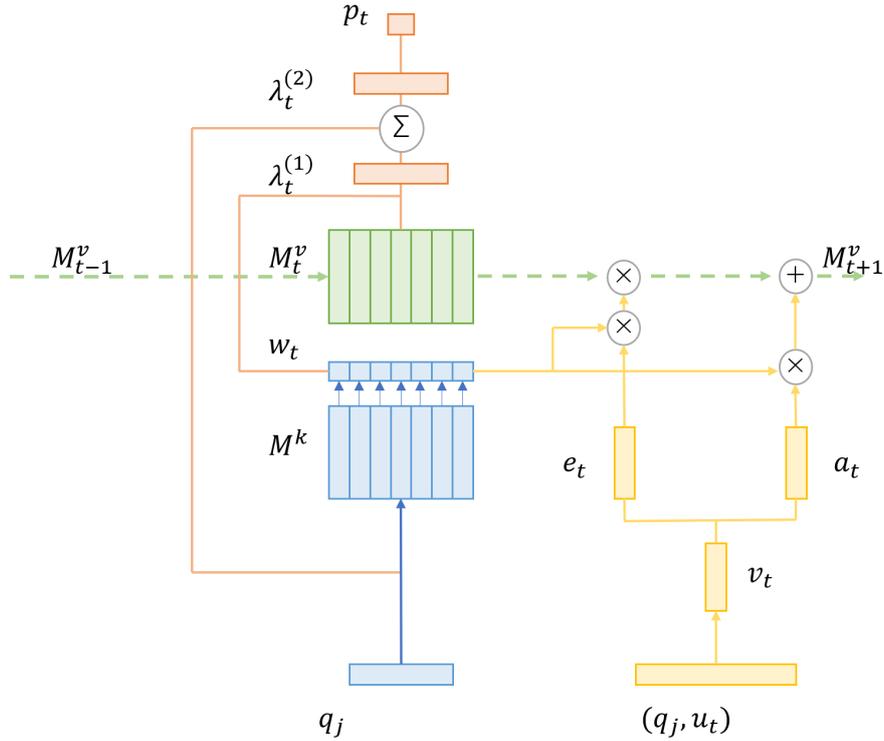


図2 DKVMN のモデル図

DKVMN では、各時点で出題された項目 q_j と多次元の潜在スキルとの関係性を保持する Key memory M^k を用いて、項目と l 番目の潜在スキルとの関連性を次の式で計算する。

$$w_t(l) = \text{Softmax}(q_j^\top M^k(l)) \quad (22)$$

$w_t(l)$ は時点 t に出題された項目が l 番目のスキルとどの程度関連があるかを示す重みベクトルである。この重みベクトルと各スキルの潜在能力値が保持された $M_t^v(l)$ を用いることで、 q_j に解答した際の能力を示すベクトル λ_t を次式で計算する。計算された $\lambda_t^{(1)}$ と q_t を用いて次のように学習者が q_j に正解する確率を計算する。

$$\lambda_t^{(1)} = \sum_{l=1}^N w_t(l) M_t^v(l) \quad (23)$$

$$\lambda_t^{(2)} = \tanh(W_1^\top [\lambda_t^{(1)}, q_j] + b_1), \quad (24)$$

$$p_t = \text{Sigmoid}(W_2^\top \lambda_t^{(2)} + b_2). \quad (25)$$

また、DKVMN は学習者の反応 $u_{i,j}$ を用いて潜在的な多次元の能力値 M_t^v を更新する。 q_j と項目への反応 $u_{i,j}$ をエンベディングした v_t を用いて次のように M_{t+1}^v を計算する。

$$e_t = \text{Sigmoid}(W_e^\top v_t + b_e), \quad (26)$$

$$\tilde{M}_t^v(l) = M_{t-1}^v(l)[1 - w_t(i)e_t], \quad (27)$$

$$a_t = \tanh(W_a^\top v_t + b_a)^\top, \quad (28)$$

$$M_t^v(l) = \tilde{M}_{t-1}^v(l) + w_t(i)a_t. \quad (29)$$

ここで e_t, a_t は忘却パラメータを表す。 e_t によって $M_{t-1}^v(l)$ をどの程度忘却し、 a_t によって $\tilde{M}_{t-1}^v(l)$ にどの程度時点 t の情報を追加するかを決める。

DKVMN は高い反応予測精度を示すことが報告されているが、DKT と同様に学習者の能力値を示すパラメータをもたないため、解釈性が低いという問題があった。

3.5 Deep-IRT

DKT や DKVMN における能力値パラメータの解釈性が低い問題を解決するために深層学習手法と IRT を組み合わせた Deep-IRT が考案されている [26]。Deep-IRT では DKVMN の出力層に IRT を導入することで高い解釈性と反応予測精度を示すことが報告されている。Deep-IRT では DKVMN における式 (24) で推定された $\lambda_t^{(2)}$ を用いて学習者の能力 $\theta_{t,j}$ を推定する。

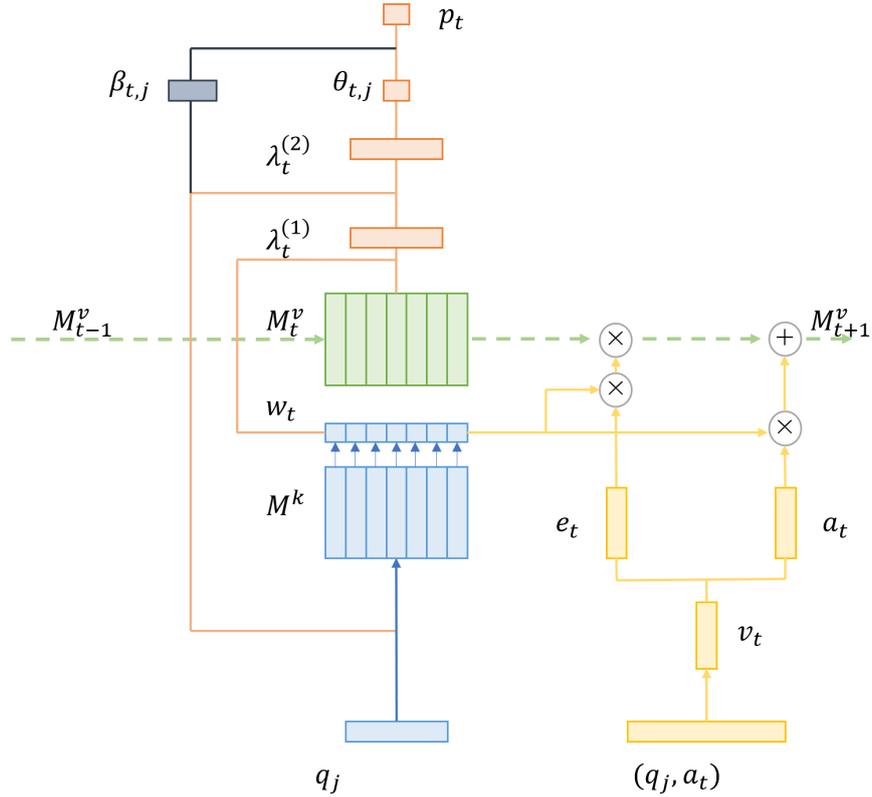


図3 Deep-IRT のモデル図

$$\theta_{t,j} = \tanh(W_\theta \lambda_t^{(2)} + b_\theta) \quad (30)$$

また解答項目 q_j から下記式を用いて項目の難易度 $\beta_{t,j}$ を推定する。

$$\beta_{t,j} = \tanh(W_\beta q_j + b_\beta) \quad (31)$$

次に推定された $\theta_{t,j}$ と $\beta_{t,j}$ を使い、下記式で項目への反応確率を計算する。

$$p_t = \sigma(3.0 \times \theta_{t,j} - \beta_j) \quad (32)$$

Deep-IRT では能力値を推定する際に式 (30) で推定するが、学習者の潜在能力値だけでなく、項目に依存する。そのため、能力値を解釈するのが困難である問題がある。

3.6 DeepIRT-HN

Tsutsumi らの研究では、能力値と項目の特性を区別するために、能力値推定ネットワークと難易度推定ネットワークを独立したネットワーク構造に変更した。これにより能力値が項目の特性に依存することなく、解釈性の高い推定値を得ることができる。

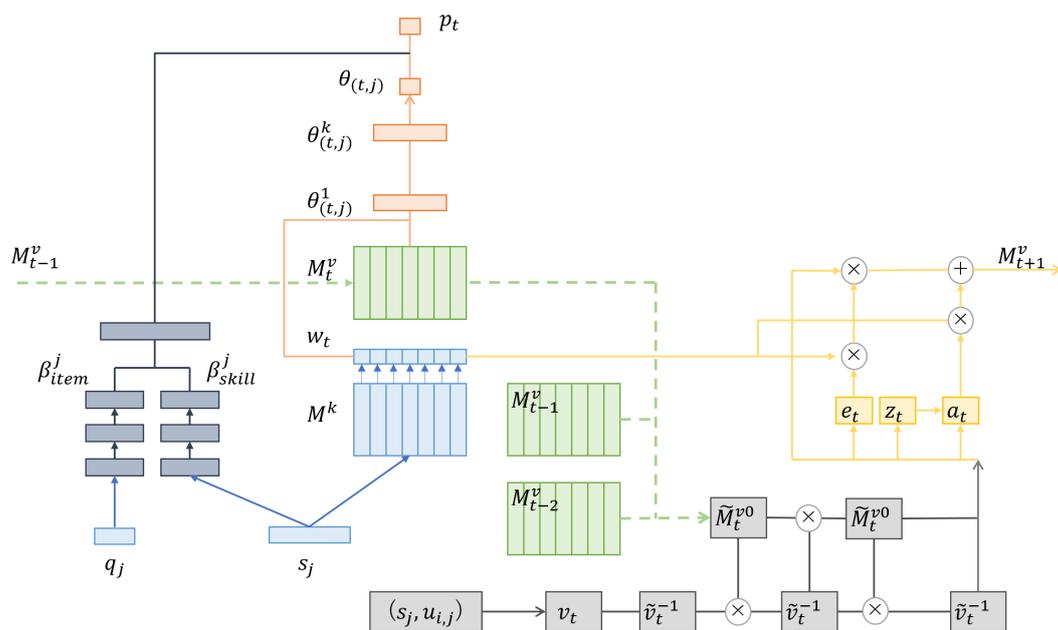


図4 DeepIRT-HN のモデル図

Tsutsumi らの DeepIRT では潜在能力値 M_k^v から以下の式で $\theta_{(t,j)}$ を推定する。

$$\theta_{(t,j)}^1 = \sum_{l=1}^N M_t^v(l), \quad (33)$$

$$\theta_{(t,j)}^k = \tanh(W_{(\theta_k)} \theta_{(t,j)}^{k-1} + b_{(\theta_k)}), \quad (34)$$

$$\theta_{(t,j)} = w_t^\top \theta_{(t,j)}^k. \quad (35)$$

また、Deep-IRT では同じスキルをもつ項目は等価であると仮定しているため、推定される項目の難易度は出題される項目のスキルに依存する。この仮定は項目が同じ

スキルを有していても大きく異なる場合は成り立たない。Tsutsumi らの DeepIRT では、項目の難易度 β_{item} と β_{skill} をそれぞれ以下の式で推定する。

$$\beta_1^j = \tanh(W^{(\beta_1)} q_j + \tau^{(\beta_1)}), \quad (36)$$

$$\beta_m^j = \tanh(W^{(\beta_m)} \beta_{(m-1)}^j + \tau^{(\beta_m)}), \quad (37)$$

$$\beta_{item}^j = W^{(\beta_{item})} \beta_m^j + \tau^{(\beta_{item})}. \quad (38)$$

$$\gamma_1^j = \tanh(W^{(\gamma_1)} s_j + \tau^{(\gamma_1)}), \quad (39)$$

$$\gamma_m^j = \tanh(W^{(\gamma_m)} \gamma_{(m-1)}^j + \tau^{(\gamma_m)}), \quad (40)$$

$$\beta_{skill}^j = W^{(\beta_{skill})} \gamma_m^j + \tau^{(\beta_{skill})}. \quad (41)$$

ここで、 m は隠れ層の数を示すハイパーパラメータであり、実データにおける予測精度によって最適な値を決めている。

推定されたパラメータを用いて学習者の項目 j への予測正答率を下記式で推定する。

$$p_t = \sigma(3.0 \times \theta_{(t,j)} - (\beta_{item}^j + \beta_{skill}^j)) \quad (42)$$

また Deep-IRT や DKVMN では学習者の潜在能力値を更新する際、忘却パラメータは時点 t の反応 v_t のみを用いて推定する。そのため学習者の過去の能力値を考慮した潜在能力値の更新が行われず、長期の学習過程では反応予測精度が低下する問題があった。この問題を解決するために Tsutsumi らは潜在能力値を更新する際、Hypernetwork を用いて潜在能力値に応じて忘却パラメータ最適化している [39]。

DeepIRT-HN は以下のようにして M_{t+1}^v を推定し、学習者の潜在能力値を更新する。

$$\tilde{v}_t^r = \xi_1 * \sigma(W^v \tilde{M}_t^{vr-1}) \otimes v_t^{r-2}, \quad (43)$$

$$\tilde{M}_t^{vr} = \xi_2 * \sigma(W^M \tilde{v}_t^{r-1}) \otimes \tilde{M}_t^{vr-2}. \quad (44)$$

この時 $\xi_1 \in \mathbb{R}, \xi_2 \in \mathbb{R}, r = 1, \dots, R$ はハイパーパラメータである。また r は処理を行う回数を表しており、 $r = 1$ の時、 $\tilde{v}_t^{-1} = v_t, \tilde{M}_t^{v0} = \tilde{M}_t^v$ である。

次に求められたパラメータを用いて M_{t+1}^v を計算する。

$$e_t = \sigma(W_{e1}\tilde{v}_t^r + W_{e2}\tilde{M}_t^{vr} + \tau_e), \quad (45)$$

$$z_t = \sigma(W_{z1}\tilde{v}_t^r + W_{z2}\tilde{M}_t^{vr} + \tau_z), \quad (46)$$

$$a_t = \tanh(W_{a1}z_t + W_{a2}\tilde{M}_t^{vr} + \tau_a), \quad (47)$$

$$M_{t+1}^v = \tilde{M}_t^{vr} \otimes (1 - w_t e_t)^\top + w_t a_t^\top. \quad (48)$$

これらの構造によって忘却パラメータを最適化することで、適切な能力値の更新が行えるようになり、能力値の推定精度と反応予測精度が向上したことが示されている。

アダプティブラーニングでは、学習者ごとに項目の出題順序が異なり、一つの項目に複数回取り組む場面がある。しかし、従来の KT 手法は項目の難易度を全期間において固定されたネットワークを用いて推定している。そのため項目間の従属性による難易度の変化を捉えられず、同一項目に複数回解答した場合に変化する難易度を表現することができない。

4 提案手法

4.1 提案手法概要

Tsutsumi らの DeepIRT-HN[39] では項目の難易度は式 (38) で表される全結合層によって推定するため、項目 j の難易度は全時点で一定である。しかし、オンライン学習システムにおいては類似項目に複数回取り組む場合が多いため、過去に解答した項目の難易度が変化する可能性がある。そこで本研究では項目の出題順序が難易度を与える影響を考慮するため、難易度の変化を表現するパラメータ $\delta_{t,j}$ を導入する。

提案手法では学習者の過去の反応データから $\delta_{t,j}$ を推定するために、時系列モデルである RNN を用いる。RNN は各時点における入力と、1 つ前の時点で計算された隠れ状態ベクトルを用いる事で再帰的に計算するモデルである。提案手法では各時点での項目と反応を表現するベクトル ψ_t を入力することで学習者の過去の学習履歴データから $\delta_{t,j}$ を推定する。

RNN の詳細を図 5 に示す。提案手法では時点 t までの学習履歴を表現する隠れ状態ベクトル h_{t-1} と項目 q_t を用い c_t を計算する。式 (50) では学習者の過去の学習履歴と時点 t で出題された項目を考慮するため、 c_t を用いて $\delta_{t,j}$ を推定する。学習者が項目に解答後 ψ_t を用いることで隠れ状態ベクトル h_t を計算する。 ψ_t は次の式で表現する。

$$\psi_t = \begin{cases} 0 & (u_t = 0) \\ q_t & (u_t = 1) \end{cases}$$

$$c_t = \tanh(W_q q_t + W_h h_{t-1} + \tau_q), \quad (49)$$

$$\delta_{t,j} = 0.5 \times \tanh(W_o c_t + \tau_o), \quad (50)$$

$$h_t = \tanh(W_{hq} \psi_t + W_{hc} c_t + \tau_h). \quad (51)$$

提案手法では隠れ状態ベクトル h_t に学習者の過去の解答項目を保持することにより項目の出題順序を考慮する。

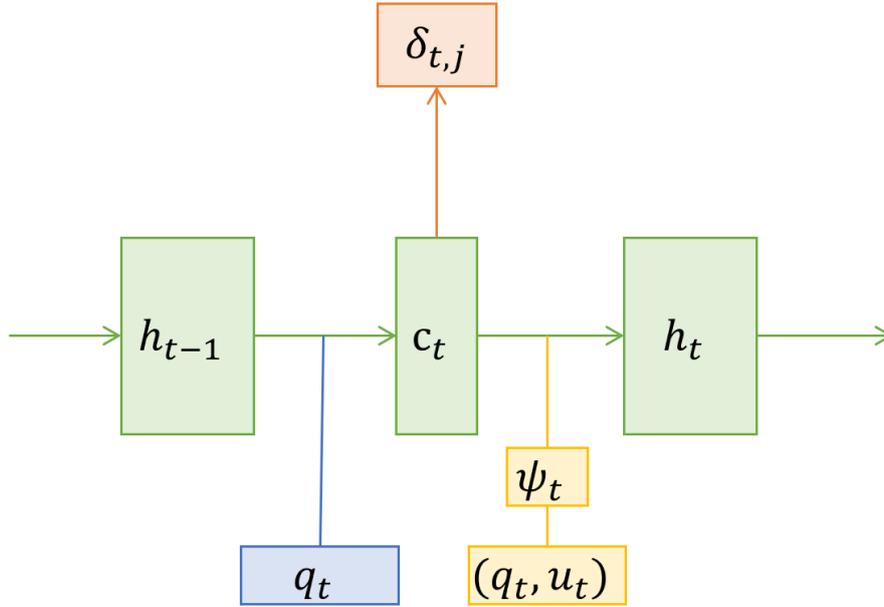


図5 δ 推定ネットワークの構造

次に推定された $\delta_{t,j}$ を式 (42) に導入することで未知の項目への正答確率を予測する。

$$p_t = \sigma(3.0 \times \theta^{(t,j)} - (\beta_{item}^j + \beta_{skill}^j + \delta_{t,j})) \quad (52)$$

$\delta_{t,j}$ は過去の学習履歴によって生じる難易度変化を表しており、 $\delta_{t,j}$ が負の場合には項目の難易度が本来より低く、 $\delta_{t,j}$ が正の場合には高くなると解釈する。つまり、学習者が過去に同一項目に正答していた場合、 $\delta_{t,j}$ の値が負で推定され、項目の難易度が下がることで予測正答確率が上がる。

4.2 パラメータ学習

一般に、深層学習では微分可能な損失関数 L を定義し、誤差逆伝播法によりパラメータを学習する。提案手法では Tsutsumi らと同様にクロスエントロピーを採用する [39]。予測正答確率 p_t と学習者の真の反応データから以下の式でクロスエントロピーを計算する。

$$L = - \sum_t (u_t \log p_t + (1 - u_t) \log(1 - p_t)) \quad (53)$$

ここで u_t は $u_{i,j}$ を表す。

5 評価実験

5.1 実験データ

本研究では、深層学習を用いた KT 手法のうち代表的な手法 (AKT、DeepIRT-HN)[18, 39] と提案手法を用いて学習者の反応予測を行う。具体的には、5 分割交差検証を用いてデータセットを訓練データ、検証データ、評価データに分割し、訓練データ、検証データから推定したパラメータを利用して評価データの反応予測を行う。予測精度の評価指標として Accuracy(一致割合)、AUC スコア、Loss スコアを算出し、各手法の精度比較を行う。本実験ではオンライン学習システムで収集された公開データセット Eedi^{*1}、KDD^{*2}、Statics2011^{*3}、ASSISTments2009^{*4}、ASSISTments2017^{*5} を用いて比較実験を行う。これらのデータセットは 1 つのレスポンスに対して項目の ID と関連付けられているスキルの ID が存在している。

データセットの概要を表 1 に示す。Eedi は 2018 年から 2020 年にオンライン学習プラットフォームから収集された数学の解答、KDD は代数学に関する解答、Statics2011 は大学レベルの工学コースから収集された解答、ASSISTments2009 は 2009 年から 2010 年にオンライン学習プラットフォーム ASSISTments で収集された解答、ASSISTments2017 は 2004 年から 2007 年にオンライン学習プラットフォーム ASSISTments で収集された中学生の数学に関する解答が含まれているデータセットである。

また、学習データは学習者ごとに解答数が大きく異なることが報告されているため、本研究ではデータの偏りを避けるために先行研究 [13] で行われた実験条件と同様に入力する学習データの上限を学習者 1 人につき 200 項目とした。

*1 <https://eedi.com/projects/neurips-education-challenge>

*2 <https://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp>

*3 <https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=507>

*4 <https://sites.google.com/site/assistmentsdata/home/2009-2010-assistments-data?authuser=0>

*5 <https://sites.google.com/view/assistmentsdatamining>

表 1 用いたデータの概要

データセット名	学習者数	項目数	スキル数	解答数	平均正答率	同一項目に 解答する平均数
KDD Cup	821	551	74	10,559	78.43%	2.15
Statics2011	1,050	1,208	98	38,272	79.42%	2.05
ASSISTments2009	4,151	26,684	111	325,637	65.85%	2.13
ASSISTments2017	1,709	3,162	102	942,816	37.27%	4.08
Eedi	80,000	27,613	1,200	2,379,193	63.85%	0.0

5.2 反応予測精度比較

各手法における反応予測精度を表 2 に示す。

提案モデルは KDD、Statics2011、ASSISTments2017 で DeepIRT-HN よりも高い AUC を示し、ASSISTments2009 で DeepIRT-HN よりも高い AUC を示した。表 1 からこれらのデータセットは学習者が一つの項目に複数回取り組む場合が多く、各項目の難易度変化を考慮した反応予測を行うことで予測精度が向上したと考えられる。一方で Eedi は学習者が同一項目に複数回解答しないことから $\delta_{t,j}$ による項目の難易度変化の影響が低く、DeepIRT-HN と同程度の予測精度になったと考えられる。

次にパラメータ $\delta_{t,j}$ によって同一項目に解答した反応データにおける予測精度が向上することを示す。表 3 は、学習履歴の中で同一項目に解答している反応データにおいて DeepIRT-HN と提案手法の反応予測精度 (AUC) を比較したものである。提案手法は、KDD、Statics2011、ASSISTments2009 では同一項目に解答する場面で、従来手法よりも反応予測精度が改善していることがわかる。一方で、ASSISTments2017 では、提案手法と同程度の精度を示している。このことから、提案手法は同一項目に解答する場合以外にも、類似項目におけるパラメータ推定に影響を及ぼしている可能性が考えられる。

表2 各モデルの反応予測精度

データセット名	metrics	AKT	DeepIRT-HN	提案手法
KDD	AUC	59.52 +/- 2.36	60.71 +/- 2.82	61.68 +/- 1.56
	ACC	78.11 +/- 2.11	76.14 +/- 3.11	76.37 +/- 2.92
	Loss	5.16 +/- 0.32	5.49 +/- 0.72	5.44 +/- 0.67
Statics2011	AUC	78.48 +/- 0.47	81.07 +/- 1.03	81.19 +/- 1.11
	ACC	80.41 +/- 0.71	80.64 +/- 0.21	80.68 +/- 0.41
	Loss	4.28 +/- 0.11	4.46 +/- 0.05	4.45 +/- 0.10
ASSISTments2009	AUC	83.01 +/- 0.39	82.37 +/- 0.32	82.47 +/- 0.28
	ACC	77.80 +/- 0.55	77.33 +/- 0.55	77.42 +/- 0.51
	Loss	4.72 +/- 0.07	5.22 +/- 0.13	5.20 +/- 0.12
ASSISTments2017	AUC	75.31 +/- 0.29	77.50 +/- 0.25	77.67 +/- 0.54
	ACC	70.69 +/- 0.29	71.91 +/- 0.28	72.21 +/- 0.58
	Loss	5.60 +/- 0.04	6.47 +/- 0.07	6.40 +/- 0.13
Eedi	AUC	79.31 +/- 0.12	79.14 +/- 0.13	79.14 +/- 0.10
	ACC	73.61 +/- 0.12	73.43 +/- 0.26	73.46 +/- 0.20
	Loss	5.26 +/- 0.02	6.12 +/- 0.06	6.11 +/- 0.05
平均	AUC	75.13	76.16	76.43
	ACC	76.12	75.89	76.03
	Loss	5.00	5.55	5.52

表3 同一項目に解答したデータにおける AUC の比較

データセット名	従来手法	提案手法
KDD	64.83	65.36
Statics2011	81.90	82.24
ASSISTments2009	80.54	82.16
ASSISTments2017	77.63	77.62

次に各データセットで推定された時点ごとの項目の難易度変化を分析する。表4は学習者が過去に同一項目に正答した場合と誤答した場合で、それぞれ推定された $\delta_{t,j}$ を全時点で平均した結果である。つまり、過去に同じ項目に解答した場合に、項目の難易度が平均してどの程度変化しているかを表している。ただし、以前に同一項目に複数解答している場合は直近の解答結果の反応のみを用いる。なお、Eedi は学習者が同一項目に解答しないため除外する。

表4 過去の解答ごとの δ の集計結果

データセット名	過去に正答	過去に誤答
KDD	0.04	0.17
Statics2011	-0.21	0.02
ASSISTments2009	-0.09	0.06
ASSISTments2017	-0.09	0.09

Statics2011、ASSISTments2009、2017 では、過去に同一項目に正答した場合、 $\delta_{t,j}$ は項目 j の難易度が低下するように負の値で推定されることがわかる。また過去に同一項目に誤答した場合は $\delta_{t,j}$ は項目 j の難易度が上がるように変化している。一方 KDD では項目に正答した場合、 $\delta_{t,j}$ が正の値を示した。またすべてのデータセットで $\delta_{t,j}$ は過去に項目 j に正答した場合よりも誤答した場合の方が難易度が高くなるように推定されることがわかった。

5.3 パラメータ δ の分析

次に (1) 学習者が同じ項目に繰り返し誤答する場合 と (2) 学習者が同じ項目に繰り返し正答する場合の 2 つの状況において、学習者の学習履歴から推定される $\delta_{t,j}$ の値を分析した。

5.3.1 学習者が同じ項目に繰り返し誤答する場合

同じ項目に繰り返し誤答している例として ASSISTments2017 を用いてある学習者の学習履歴から推定した各時点の $\delta_{t,j}$ を分析する。図6では学習者が時点 $t = 126$ か

ら時点 $t = 137$ まで項目 438 に繰り返し取り組んだ際の学習履歴を示しており、表中には各時点において推定された $\delta_{t,j}$ の値、提案手法反応予測、DeepIRT-HN の反応予測、実際の学習者の反応を示している。ここでは学習者の反応は”1”が正答、”0”が誤答を表している。 $\delta_{t,j}$ は時点 $t = 126$ で学習者が項目に誤答した後に値が増加していることから、時点 $t = 127$ 以降は項目の難易度が高く推定されていることがわかる。これにより、提案手法における正答確率の予測精度が向上し、DeepIRT-HN の反応予測が改善されている。

時点	126	127	128	129	130	131	132	133	134	135	136	137
項目	438	438	438	438	438	438	438	438	438	438	438	438
δ の値	0.17	0.49	0.49	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
提案モデルの予測	1	0	0	0	1	1	0	0	0	0	0	0
従来モデルの予測	1	1	0	1	1	1	1	1	1	1	1	1
実際の反応	0	0	0	0	0	0	0	0	0	0	0	1

図 6 ASSISTments2017 でのある学習者の学習履歴と反応予測の比較

5.3.2 学習者が同じ項目に繰り返し正答する場合

次に同じ項目に繰り返し正答している例として ASSISTments2009 を用いて同様に $\delta_{t,j}$ を推定した学習履歴を図 7 に示す。時点 $t = 195$ で学習者が正答した後、時点 $t = 196$ で $\delta_{t,j}$ の値が減少し項目の難易度が低下している。つまり、提案手法では $\delta_{t,j}$ を推定することによって一度正答した項目の難易度を下げることによって反応予測精度を向上させている。

時点	195	196	197
項目	7052	7052	7052
δ の値	0.5	-0.5	-0.5
提案モデルの予測	1	1	1
従来モデルの予測	1	0	0
実際の反応	1	1	1

図 7 ASSISTments2009 にでのある学習者の学習履歴と反応予測の比較

6 むすび

アダプティブラーニングでは学習者ごとに項目の出題順序が異なるため、学習者が一つの項目に複数回取り組む場面がある。そのような場面では学習者の学習履歴によって項目の難易度が変わる可能性がある。本研究では同一項目に複数回解答することによって生じる項目の難易度変化を考慮するため、学習履歴による従属性の影響を示すパラメータを導入したモデルを提案した。提案手法では学習者の学習履歴と反応を RNN に入力することで、時点 t に出題される項目の難易度を変化させるパラメータ δ を推定した。反応予測に δ を用いることによって学習者の学習履歴を反映した反応予測が可能になった。評価実験では、実データを用いて学習者の反応予測精度の比較を行い、提案手法の反応予測精度が向上したことを示した。また推定されたパラメータの分析では、パラメータが学習者の学習履歴に応じて項目の難易度を変化させ反応予測を改善させたことを示した。

本研究では RNN を用いてパラメータ δ を推定したが、RNN は長期的な依存関係を捉えることが難しい問題がある。そのため、深層学習分野では LSTM や Attention モデルのような長期的なデータ間の依存関係を捉えることができるアルゴリズムが提案されている。そのため、パラメータ δ の推定においても、これらの深層学習アルゴリズムを用いることで提案手法の精度向上に努めたい。また、近年、学習者の習熟度に合ったヒントを提示する足場がけ学習システム [42, 43] が研究されている。足場がけ学習システムでは学習者の予測正答率に応じてヒントを出題するため、提案手法を応用する事で適切な学習支援が行えると考える。

7 謝辞

研究室に配属されてから学業だけでなく様々な面で多大なるご指導をしていただいた指導教員の植野真臣教授へ深謝致します。論文執筆や研究について丁寧に指導していただいた先輩に感謝申し上げます。また本研究の遂行にあたりご協力頂いた研究室の皆様に誠意の意を表します。

最後に本論文の執筆にあたり多くの激励を頂きました友人と両親にお礼と感謝をいたします。ありがとうございました。

参考文献

- [1] A. T. Corbett and J. R. Anderson, “Knowledge tracing: Modeling the acquisition of procedural knowledge,” *User Model. User-Adapt.Interact.*, vol. 4, no. 4, pp. 253–278, Dec 1995.
- [2] Z. Pardos and N. Heffernan, “T.: Modeling individualization in a bayesian networks implementation of knowledge tracing,” in *In Proceedings of the 18th International Conference on User Modeling, Adaption,and Personalization*, 06 2010, pp. 255–266.
- [3] Z. A. Pardos and N. T. Heffernan, “Kt-idem: Introducing item difficulty to the knowledge tracing model,” in *Proceedings of 19th International Conference on User Modeling, Adaptation and Personalization (UMAP2011)*, 01 2011, pp. 243–254.
- [4] J. Lee and E. Brunskill, “The impact on individualizing student models on necessary practice opportunities,” in *Proceedings of the Fifth International Conference on Educational Data Mining*, 01 2012, pp. 118–125.
- [5] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon, “Individualized bayesian knowledge tracing models,” in *Artificial Intelligence in Education*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp.171–180.
- [6] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein, “Deep knowledge tracing,” in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D.Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015,pp. 505–513.
- [7] P. Chen, Y. Lu, V. Zheng, and Y. Pian, “Prerequisite-driven deep knowledge tracing,” in *In IEEE International Conference on Data Mining, ICDM 2018*, 2018, pp. 39–48.
- [8] X. Liangbei and D. Mark, “Dynamic knowledge embedding and tracing,” in *Proceedings of the 13th International Conference on Educational Data Mining*,

- EDM, 2020, pp. 524–530.
- [9] Y. Lu, D. Wang, Q. Meng, and P. Chen, “Towards interpretable deep learning models for knowledge tracing,” in Proceedings of the 13th International Conference on Educational Data Mining, EDM, 2020, pp.185–190.
- [10] H. Tong, Y. Zhou, and Z. Wang, “Exercise hierarchical feature enhanced knowledge tracing,” in Proceedings of the 13th International Conference on Educational Data Mining, EDM, 2020, pp. 324–328.
- [11] Z. Wang, X. Feng, J. Tang, G. Huang, and Z. Liu, “Deep knowledge tracing with side information,” in The 20th International Conference on Artificial Intelligence in Education (AIED), 2019, pp. 303–308.
- [12] C. K. Yeung and D.-Y. Yeung, “Addressing two problems in deep knowledge tracing via prediction-consistent regularization,” in Proceedings of the Fifth ACM Conference on Learning @ Scale, 2018, pp. 1–10.
- [13] C. Yeung, “Deep-irt: Make deep learning based knowledge tracing explainable using item response theory,” in Proceedings of the 12th International Conference on Educational Data Mining, EDM, 2019.
- [14] J. Zhang, X. Shi, I. King, and D.-Y. Yeung, “Dynamic key-value memory network for knowledge tracing,” in Proceedings of the 26th International Conference on World Wide Web, ser. WWW ’ 17. International World Wide Web Conferences Steering Committee, 2017, pp. 765–774.
- [15] Q. Liu, Z. Huang, Y. Yin, E. Chen, H. Xiong, Y. Su, and G. Hu, “Ekt:Exercise-aware knowledge tracing for student performance prediction,” IEEE Transactions on Knowledge and Data Engineering, vol. PP, pp.1–1, 06 2019.
- [16] H. Nakagawa, Y. Iwasawa, and Y. Matsuo, “Graph-based knowledge tracing: Modeling student proficiency using graph neural network,” in 2019 IEEE/WIC/ACM International Conference on Web Intelligence(WI), 2019, pp. 156–163.
- [17] S. Pandey and J. Srivastava, “Rkt: Relation-aware self-attention for knowl-

- edge tracing,” in Proceedings of the 29th ACM International Conference on Information and Knowledge Management, 2020, pp.1205–1214.
- [18] A. Ghosh, N. Heffernan, and A. S. Lan, “Context-aware attentive knowledge tracing,” in Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020.
- [19] E. Tsutsumi, R. Kinoshita, and M. Ueno, “Deep-irt with independent student and item networks,” in Proceedings of the 14th International Conference on Educational Data Mining (EDM), 2021.
- [20] D. Agarwal, R. Baker, and A. Muraleedharan, “Dynamic knowledge tracing through data driven recency weights,” in Proceedings of the 13th International Conference on Educational Data Mining, EDM, 2020, pp.725–729.
- [21] Pelánek, R. “Conceptual Issues in Mastery Criteria: Differentiating Uncertainty and Degrees of Knowledge”. *Artificial Intelligence in Education*, pp. 450-461 (2018).
- [22] S. Gowda, J. Rowe, R. Baker, M. Chi, and K. Koedinger, “Improving models of slipping, guessing, and moment-by-moment learning with estimates of skill difficulty.” in EDM 2011 – Proceedings of the Fourth International Conference on Educational Data Mining, 01 2011, pp.199–208.
- [23] M. Khajah, Y. Huang, J. Gonzalez-Brenes, M. Mozer, and P. Brusilovsky, “Integrating knowledge tracing and item response theory: A tale of two frameworks,” *Personalization Approaches in Learning Environments*, vol. 1181, pp. 5–17, 2014.
- [24] J. Reye, “Student modelling based on belief networks,” *International Journal of Artificial Intelligence in Education*, vol. 14, pp. 63–96, 2004.
- [25] F. Baker and S. Kim, *Item Response Theory: Parameter Estimation Techniques*, Second Edition, ser. *Statistics: A Series of Textbooks and Monographs*. Taylor & Francis, 2004.
- [26] C. Ekanadham and Y. Karklin, “T-skirt: Online estimation of student proficiency in an adaptive learning system,” *CoRR*, vol. abs/1702.04282, 2017.

- [27] K. H. Wilson, Y. Karklin, B. Han, and C. Ekanadham, “Back to the basics: Bayesian extensions of irt outperform neural networks for proficiency estimation,” in 9th International Conference on Educational Data Mining, vol. 1, 06 2016, pp. 539–544.
- [28] 堤 瑛美子, 宇都 雅輝, 植野 真臣, “ダイナミックアセスメントのための隠れマルコフ IRT モデル,” 電子情報通信学会論文誌 (D), vol.J102, no.2, pp.79–92, 2019.
- [29] 堤 瑛美子, 木下 涼, 植野 真臣, “Knowledge Tracing のための Sliding Window 隠れマルコフ IRT,” 電子情報通信学会論文誌 (D), vol.J103, no.12, pp.894–905, 2020.
- [30] E. Tsutsumi, R. Kinoshita, and M. Ueno. Deep item response theory as a novel test theory based on deep learning. *Electronics*, 10(9), 2021.
- [31] F. Bartolucci, F. Pennoni, and G. Vittadini, “Assessment of school performance through a multilevel latent markov–rasch model,” *Journal of Educational and Behavioral Statistics*, 09 2011.
- [32] J. V. Dylan Molenaar, Daniel Oberski and P. D. Boeck, “Hiddenmarkov item response theory models for responses and response times,” *Multivariate Behavioral Research*, vol. 51, pp. 606–626, 2016.
- [33] J. Gonzalez-Brenes, Y. Huang, and P. Brusilovsky, “General features in knowledge tracing to model multiple sub skills, temporal item response theory, and expert knowledge,” In: *The Seventh International Conference on Educational Data Mining*, 01 2014.
- [34] J. H. Park, “Modeling preference changes via a hidden markov item response theory model,” In *Handbook of Markov Chain Monte Carlo*, pp. 479–491, 2011.
- [35] R. Weng and D. Coad, “Real-time bayesian parameter estimation for item response models,” *Bayesian Analysis*, vol. 13, 12 2016.
- [36] X. Wang, J. Berger, and D. Burdick, “Bayesian analysis of dynamic item response models in educational testing,” *The Annals of Applied Statistics*, vol. 7, no. 1, pp. 126–153, 2013.

- [37] S. Pandey and G. Karypis, “A self-attentive model for knowledge tracing,” in Proceedings of International Conference on Education Data Mining, 2019.
- [38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” In Advances in Neural Information Processing Systems, pp. 5998–6008, 2017.
- [39] E. Tsutsumi, Y. Guo, and M. Ueno, “Deepirt with a hyper network to optimize the degree of forgetting of past data,” in Proceedings of the 15th International Conference on Educational Data Mining (EDM), 2022.
- [40] 堤瑛美子, 郭亦鳴, 植野真臣, “学習データの忘却を最適化する Hypernetwork を組み込んだ DeepIRT” 電子情報通信学会論文誌 (D), vol.J106, no.2, pp.xxx–xxx, 2023.
- [41] G. Melis, Tomáš Kočiský, and B. Phil, “Mogrifier lstm,” in Proceedings of ICLR 2020, 2020
- [42] Maomi Ueno and Yoshimitsu Miyazawa. “Probability based scaffolding system with fading” . In Proceedings of Artificial Intelligence in Education – 17th International Conference, AIED, pp. 237–246, 2015.
- [43] Maomi Ueno and Yoshimitsu Miyazawa. “IRT-based adaptive hints to scaffold learning in programming” . IEEE Transactions on Learning Technologies, Vol. 11, pp. 415–428, 10 2018.