THE IEICE TRANSACTIONS ON INFORMATION AND SYSTEMS (JAPANESE EDITION)

電子情報通信学会 論文誌 D 情報・システム

VOL. J106-D NO. 2
FEBRUARY 2023

情報・システムソサイエティ

一般社団法人 電子情報通信学会
THE INFORMATION AND SYSTEMS SOCIETY
THE INSTITUTE OF ELECTRONICS, INFORMATION AND COMMUNICATION ENGINEERS

学習データの忘却を最適化するHypernetworkを組み込んだDeepIRT

堤　瑛美子[†a)]　　郭　　亦鳴[†b)]　　植野　真臣[†c)]

DeepIRT with a Hypernetwork to Optimize the Degree of Forgetting of Past Data

Emiko TSUTSUMI[†a)], Yiming GUO[†b)], and Maomi UENO[†c)]

あらまし　近年，教育現場ではオンラインラーニングシステムで収集された教育ビッグデータをいかに有効に活用するかが課題となっている．人工知能分野では，これらの教育ビッグデータに機械学習手法を適用し，学習者の課題への反応を予測することにより，学習者への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．Tsutsumi ら (2021) はアダプティブラーニングのために深層学習手法と項目反応理論を組み合わせ，パラメータの解釈性をもちながら高精度な反応予測を可能とする DeepIRT を開発し，高い予測精度とパラメータの解釈性を実現している．しかし，DeepIRT では学習者の潜在的な能力値を推定する際に最新の学習データのみを用いるために，過去の学習データを十分に反映できていない可能性がある．本研究では，DeepIRT に新たな Hypernetwork を組み合わせ，学習者の過去の学習データと最新の学習データの重要性を推定することで両者のバランスを最適化しながら能力値推定を行う．評価実験では，提案手法が最先端の反応予測手法を上回る反応予測精度を示した．

キーワード　パフォーマンス予測，Deep Learning, Hypernetwork, 項目反応理論

1. まえがき

近年，学校現場へのコンピュータやタブレット端末の普及に伴ってオンライン学習システムを用いた学習が広まり，大量の教育ビックデータが蓄積されるようになり，それらをいかに有効活用するかが課題になっている．学習支援システム分野や人工知能分野では，機械学習を用いて教育ビッグデータを分析することにより学習者の特性や成長に合わせて適切な問題提供，学習支援を行うアダプティブ・ラーニングが注目されている [1]〜[13]．具体的には，学習履歴データから学習過程における学習者の習熟度（理解度）の変化と未知の項目への反応（正答・誤答）を予測することで得意分野・苦手分野を把握し，個人に適切な学習支援を行う．学習者の未知の項目への反応予測を行う手法には主に確率的アプローチ [1], [2], [7], [8], [10], [12]〜[15] と深層学習アプローチ [6]〜[9], [11], [16]〜[19] があるが，近年では深層学習アプローチが盛んに開発されている．

深層学習を用いて学習者の反応予測を行う代表的な手法に Deep Knowledge Tracing (DKT) がある [6]．DKT は Long-short term memory (LSTM) [20] を用いて時系列変化する学習者のスキルの習得状態を表現し，学習者の各項目への反応を予測するモデルである．ただし，DKT では LSTM の隠れ層に全てのスキルの習得状態が圧縮されているため，パラメータの解釈性が低いという問題があった．そこで，DKT の反応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 [11]．DKVMN は項目の特性や学習者の潜在的な能力の情報を保存するための memory network をモデルに組み込むことにより，DKT と比較して反応予測精度が高いことが報告されている．また，DKVMN は学習者の過去の反応データと潜在的な能力値を保存する value memory パラメータをもち，memory updating component と呼ばれる機構で前時点の value memory をある程度忘却し，最新の反応データを用いて value memory を更新する．しかし，DKVMN においても学習者の能力が隠れ変数行列に圧縮されており，項目の難易度パラメータや学習者の能力パラメータの解釈が難しいといった問題が

† 電気通信大学大学院情報理工学研究科，調布市
Graduate School of Informatics and Engineering, The University of Electro-Communications, 1–5–1 Chofugaoka, Chofu-shi, 182–8585 Japan
a) E-mail: tsutsumi@ai.lab.uec.ac.jp
b) E-mail: guo_yzmzng@ai.lab.uec.ac.jp
c) E-mail: ueno@ai.lab.uec.ac.jp
DOI:10.14923/transinfj.2022LEP0003

ある．

一方，近年では，一般に自然言語処理の分野で多く利用される Transformer を用いた Self-Attentive Knowledge Tracing (SAKT) [21] が開発されている．SAKT は過去の学習履歴からスキルと項目の関係性を推定しながら学習者の反応予測を行う．更に，最先端の手法として SAKT の構造に過去の反応データの忘却機能を組み込んだ Attentive Knowledge Tracing (AKT) [16] が提案されている．AKT は過去の反応データから反応予測に必要なデータのみを抽出してモデル内の重みを最適化するため，現在，最も学習者の反応予測精度の高い手法として知られている．ただし，AKT においても解釈性のあるパラメータをもたないため，学習過程での学習者の能力変化を表現することができず，教育応用には限界があった．

DKVMN と AKT は共に学習者の過去の反応データを忘却しながら反応予測を行うが，パラメータの解釈性を保つためには DKVMN のように直前の潜在能力値を用いて能力値を更新する仕組み (memory updating component) が重要であることが分かってきた．深層学習手法におけるパラメータの解釈性を向上させるための手法として，DKVMN と項目反応理論 (Iterm Respose Theory; IRT) を組み合わせた DeepIRT が提案されている [9]．Yeung の DeepIRT [9] は IRT と同様に学習者の能力値と項目の難易度を表すパラメータを推定することで DKVMN にパラメータ解釈性をもたせた．しかし，学習者の能力値が解答項目の特性に依存しており，依然としてパラメータの解釈性には課題があった．

そこで，Tsutsumi らは学習者の能力値と項目の難易度パラメータをそれぞれ独立したネットワークを用いて推定することでパラメータの解釈性を向上させた新たな DeepIRT を提案した [18], [19]．Tsutsumi らの DeepIRT では学習者の能力値が項目の特性に依存せず，多次元のスキルに対する能力変化を表現することができる．この手法を用いることにより，解釈性の高いパラメータ推定と高精度な反応予測の両立が可能となった．しかし，Tsutsumi らの DeepIRT [18], [19] では，DKVMN と同様の memory updating component を用いており，学習者の能力推定の際に過去の反応データの忘却度を調整する忘却パラメータが最適化されていないために，学習者の反応予測精度が低下している可能性がある．DKVMN や DeepIRT 手法 [9], [18], [19] の忘却パラメータは学習者の過去の反応データと潜在的な能力値を保存する value memory を更新する際に，最新の学習者の反応データのみを用いて最適化されている．そのため，Tsutsumi ら (2021) [19] では value memory に保存されている過去の反応データの情報を十分に利用できずに反応予測を行っていると考えられる．

この問題を解決するために，本研究では Tsutsumi らの DeepIRT に新たな Hypernetwork を組み込み，最新の学習者の反応データと直前の学習者の潜在能力値の両方を用いて忘却パラメータを最適化する．Hypernetwork は，近年，自然言語処理の分野で LSTM を拡張するために用いられている方法である [22], [23]．一般的な LSTM では潜在変数を更新するための重みパラメータの値が全時点で共有されているが，Ha ら (2016) は Hypernetwork を用いて潜在変数を更新する前に重みを時点ごとに最適化することで，LSTM のパフォーマンスが向上することを示した [22]．また，Melis ら (2020) では各時点ごとの重みだけでなく，直前の潜在変数も Hypernetwork で最適化する手法を提案している [24]．本研究では，LSTM への適用ではなく，DeepIRT における value memory を更新する前に，Hypernetwork 内で最新の学習者の反応データと前時点での value memory のバランスを調整しながら忘却パラメータを推定する．更に，memory updating component 内で最適化した忘却パラメータを用いて value memory を更新することで，過去の反応データの適切に反映した能力推定と反応予測が可能となる．最後に，提案手法と既存手法（DKVMN, DeepIRT 手法 [9], [19]，AKT）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．更に，提案手法の能力パラメータ推定精度が既存の DeepIRT 手法 [9], [19] より向上したことを示す．

2. 先行研究

2.1 Attentive Knowledge Tracing (AKT)

学習者の反応予測を行う最先端の手法として，Attentive Knowledge Tracing (AKT) が開発されている [16]．AKT は自然言語処理の分野で多く利用される attention 機構を用いた Transformer と 1 パラメータロジスティックモデルの IRT であるラッシュモデル [25] を組み合わせた手法である．AKT における attention は忘却パラメータであり，学習者の最新の反応データと過去の反応データの関連性を保持しながら，指数関数的に過去の反応データを忘却する．更に，AKT では過去の全反応データから反応予測に必要なデータのみを抽出し

て attention を最適化することで反応予測精度を向上させている．

AKT では，attention $\boldsymbol{\alpha}$ を以下の式で求める．

$$\alpha_{t,\lambda} = \frac{\exp(f_{t,\lambda})}{\sum_{\lambda'} \exp(f_{t,\lambda})}, \tag{1}$$

$$f_{t,\lambda} = \frac{\exp(-\eta d(t,\lambda)) \cdot \boldsymbol{q_t}^\top \boldsymbol{k}_\lambda}{\sqrt{D_k}}. \tag{2}$$

ここで，$\boldsymbol{q_t} \in \mathbb{R}^{D_k}$ は時点 $t = 1$ から t までの学習者の反応データを表し，$\boldsymbol{k_\lambda} \in \mathbb{R}^{Dk}$ は時点 λ でのキー行列，D_k はキー行列 $\boldsymbol{k_\lambda}$ の次元を表す．$\eta > 0$ は過去データの忘却度を決めるパラメータである．$d(t,\lambda)$ は時点 λ に入力された過去の反応データと時点 t に入力された最新の反応データの関連性を表し，次式で求められる．

$$d(t,\lambda) = |t-\lambda| \sum_{t'=\lambda+1}^{t} \frac{\frac{\boldsymbol{q_t}^\top \boldsymbol{k}_{t'}}{\sqrt{D_k}}}{\sum_{1 \le \lambda' \le t'} \frac{\boldsymbol{q_t}^\top \boldsymbol{k}'_{\lambda'}}{\sqrt{D_k}}}, \quad \forall t' \le t. \tag{3}$$

したがって，AKT では学習者の最新の反応データと過去の反応データの関連性を考慮しながら反応予測に必要なデータのみを抽出することができ，同時に，経過時間 $|t-\lambda|$ が大きくなるほど attention $\alpha_{t,\lambda}$ が減少する（データを忘却する）ように推定される．これらの仕組みにより，AKT は現在，最も高い反応予測精度を示す手法として知られている．ただし，AKT は解釈性のあるパラメータをもたないため，学習過程での学習者の能力変化を表現することができない．

2.2 DKVMN と DeepIRT 手法

近年，DKT の反応予測精度・解釈性を向上させるために，多くの手法が開発されている [26]〜[32]．特に，Dynamic Key-Value Memory Network (DKVMN) は情報を保存するための memory network と attention 機構を組み込むことにより，多次元の潜在スキルに対する能力変化を推定することができる [11]．DKVMN では N 個の潜在スキルを仮定しており，各項目と潜在スキルの関係を key memory $\boldsymbol{M}^k \in \mathbb{R}^{N \times d_k}$ に保存し，時点 t の各潜在スキルに対する能力を value memory $\boldsymbol{M}^v_t \in \mathbb{R}^{N \times d_v}$ に保存する．d_k，d_v はチューニングパラメータである．先行研究では DKVMN を用いることで，反応予測精度を大幅に向上したことが示されている．しかし，DKVMN においても学習者の能力値が隠れ変数行列に圧縮されており，項目の難易度パラメータや学習者の能力パラメータの解釈が難しいといった問題があった．

そこで，Yeung は 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 と項目反応理論 (Iterm Respose Theory; IRT) を組み合わせた DeepIRT を提案した [9]．Yeung の DeepIRT [9] は IRT と同様に学習者の能力値と項目の難易度を表すパラメータを推定することで DKVMN にパラメータ解釈性をもたせた．しかし，学習者の能力値が解答項目の特性に依存しており，依然としてパラメータの解釈性には課題があった．

高精度な反応予測とパラメータ解釈性を両立させるために，Tsutsumi らはそれぞれ独立した学習者ネットワークと項目ネットワークを用いて学習者の能力値と項目の難易度パラメータを推定する新たな DeepIRT を提案している [19]．Tsutsumi らの DeepIRT では学習者の能力値が項目の特性に依存せず，多次元のスキルに対する能力変化を表現することができる．また，学習者ネットワークと項目ネットワークのそれぞれの出力値を用いて学習者が項目に正答する確率を算出し，反応予測を行う（図 1 左側）．

しかし，DKVMN と DeepIRT 手法 [9], [11], [19] は時系列変化する潜在能力値を更新・忘却するための仕組みとして同様の memory updating component をもつ．memory updating component では，最新の学習者の反応データが入力された際に潜在能力値 value memory $\boldsymbol{M}^v_t$ を以下のように更新する．はじめに，時点 t において学習者が解答した項目 j のスキルと項目 j への反応データ u_{jt} から埋め込み処理を行った行列 $\boldsymbol{v_t} \in \mathbb{R}^{d_v}$ を求める．次に，$\boldsymbol{v_t}$ をもとに value memory $\boldsymbol{M}^v_t$ を更新する．

$$\boldsymbol{e_t} = \sigma(\boldsymbol{W}^e \boldsymbol{v_t} + \boldsymbol{\tau}^e), \tag{4}$$

$$\boldsymbol{a_t} = \tanh(\boldsymbol{W}^a \boldsymbol{v_t} + \boldsymbol{\tau}^a), \tag{5}$$

$$\tilde{\boldsymbol{M}}^v_{(t+1)} = \boldsymbol{M}^v_t \otimes (1 - \boldsymbol{w_t e_t})^\top, \tag{6}$$

$$\boldsymbol{M}^v_{(t+1)} = \tilde{\boldsymbol{M}}^v_{(t+1)} + \boldsymbol{w_t a_t}^\top. \tag{7}$$

ここで，$\boldsymbol{W}^{\cdot}$ は重みパラメータ，$\boldsymbol{\tau}^{\cdot}$ はバイアスパラメータ，$\boldsymbol{w_t}$ は項目のスキルと潜在スキルの関係性の強さを表す attention である．σ はシグモイド関数，$\otimes$ はアダマール積を表す．また，$\boldsymbol{e_t}$ はそれまでの value memory の値をどの程度保存しておくか調整し，$\boldsymbol{a_t}$ は時点 t の反応データをどの程度反映するか調整するパラメータである．以下，本論文では $\boldsymbol{e_t}$ と $\boldsymbol{a_t}$ を忘却パラメータと呼ぶ．

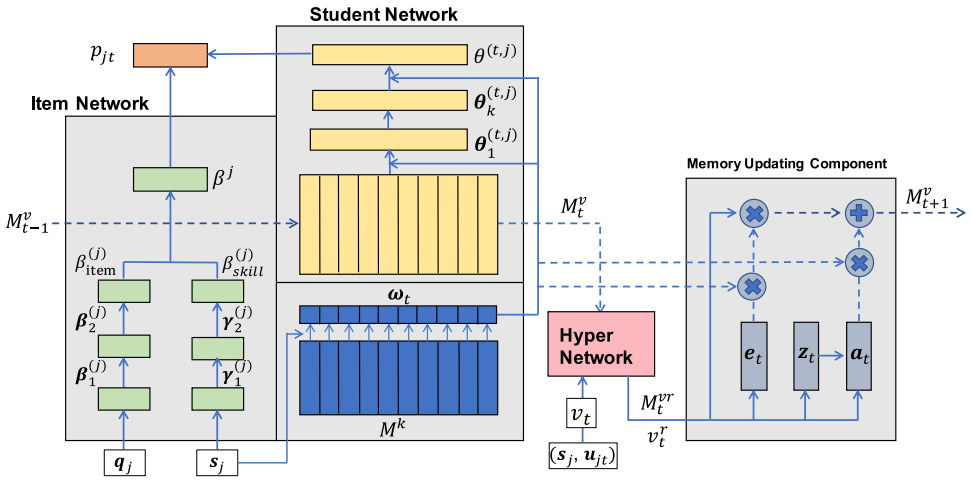

図 1　提案手法の概要

上記の memory updating component は直前の value memory $\boldsymbol{M}_t^v$ を用いて現在の $\boldsymbol{M}_{t+1}^v$ を更新するため，パラメータの解釈性を保つ上では AKT の忘却機能より効果的である．しかし，忘却パラメータ $\boldsymbol{e}_t, \boldsymbol{a}_t$ が学習者の最新の反応データを表す $\boldsymbol{v}_t$ のみを用いて推定されており，過去のデータを用いて最適化されていないため，長期の学習になるほど学習者の過去の学習データが $\boldsymbol{M}_t^v$ に反映されず，反応予測精度を低下させている可能性がある．

3. 提 案 手 法

本研究では，高精度な反応予測精度とパラメータ解釈性を両立させるために，Tsutsumi らの DeepIRT [19] に新たな Hypernetwork を組み込み，最新の学習者の反応データと直前の学習者の潜在能力値の両方を用いて忘却パラメータを最適化する．Hypernetwork は，近年，自然言語処理の分野で LSTM を拡張するために用いられている方法である [22], [23]．Ha ら (2016) は LSTM 潜在変数を更新する前に Hypernetwork 内で重みを時点ごとに最適化することで，モデルのパフォーマンスが向上することを示した [22]．また，Melis ら (2020) では各時点ごとの重みだけでなく，直前の潜在変数も Hypernetwork で最適化する手法を提案している [24]．本研究では，これらの先行研究を参考にして，過去の反応データと潜在能力値を保存する value memory を更新する前に，Hypernetwork 内で最新の学習者の反応データと前時点での value memory のバランスを調整しながら忘却パラメータを推定する．その

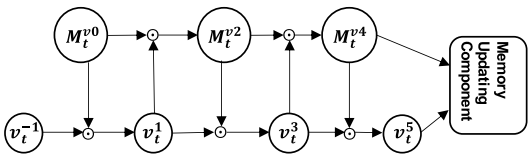

図 2　Hypernetwork の構造

後，memory updating component で最適化した忘却パラメータを用いて value memory を更新することで，過去の反応データの適切に反映した能力推定と反応予測が可能となる．提案手法のモデル図を図 1 に示す．図 1 右側は Hypernetwork と value memory の更新を行う memory updating component を表し，左側は学習者の能力値と項目の困難度を推定する独立ネットワークを表す．

3.1 Hypernetwork

Hypernetwork では最新の学習者の反応データと直前の学習者の潜在能力値 $\boldsymbol{M}_t^v$ を用いて両者の重要性を推定しながら最適化する．Hypernetwork の構造を図 2 に示す．Hypernetwork では前時点での value memory $\boldsymbol{M}_t^v$ と DKVMN と同様に最新の学習者の反応データから埋め込み処理を行った行列 $\boldsymbol{v}_t \in \mathbb{R}^{d_v}$ を入力として，$\boldsymbol{v}_t$ と $\boldsymbol{M}_t^v$ を以下の式で最適化する．

$$\boldsymbol{v}_t^r = \delta_1 \cdot \sigma(\boldsymbol{W}^v \boldsymbol{M}_t^{vr-1}) \odot \boldsymbol{v}_t^{r-2}, \tag{8}$$

$$\boldsymbol{M}_t^{vr} = \delta_2 \cdot \sigma(\boldsymbol{W}^m \boldsymbol{v}_t^{r-1}) \odot \boldsymbol{M}_t^{v\,r-2}. \tag{9}$$

ここで，$\delta_1 \in \mathbb{R}$，$\delta_2 \in \mathbb{R}$ と $r = \{1, \cdots, R\}$ はチューニングパラメータであり，r は繰り返し回数を表す．た

だし，$r = 1$ のとき，$\boldsymbol{v}_t^{-1} = \boldsymbol{v}_t$，$\boldsymbol{M}_t^{v0} = \boldsymbol{M}_t^v$ である．式 (8) と式 (9) を繰り返し行うことにより，Hypernetwork 内で最新の反応データ $\boldsymbol{v}_t$ と過去の潜在能力値 $\boldsymbol{M}_t^v$ のバランスを最適化できる．本研究では，チューニングパラメータ δ_1, δ_2 と繰り返し回数 r はデータセットごとに最適値を求める．

3.2 Memory Updating Component

次に，Hypernetwork で最適化した $\boldsymbol{v}_t$ と $\boldsymbol{M}_t^v$ を用いて memory updating component 内で忘却パラメータ $\boldsymbol{e}_t, \boldsymbol{a}_t$ を推定する．提案手法では $\boldsymbol{a}_t$ の最適化のため，以下のように新たなパラメータ $\boldsymbol{z}_t$ を追加する．

$$\boldsymbol{e}_t = \sigma(\boldsymbol{W}^{e1}\boldsymbol{v}_t^r + \boldsymbol{W}^{e2}\boldsymbol{M}_t^{vr} + \boldsymbol{\tau}^e), \tag{10}$$

$$\boldsymbol{z}_t = \sigma(\boldsymbol{W}^{z1}\boldsymbol{v}_t^r + \boldsymbol{W}^{z2}\boldsymbol{M}_t^{vr} + \boldsymbol{\tau}^z), \tag{11}$$

$$\boldsymbol{a}_t = tanh(\boldsymbol{W}^{a1}\boldsymbol{z}_t + \boldsymbol{W}^{a2}\boldsymbol{M}_t^{vn} + \boldsymbol{\tau}^a). \tag{12}$$

更に，推定された忘却パラメータを用いて $\boldsymbol{M}_{t+1}^v$ を以下で更新する．

$$\boldsymbol{M}_{t+1}^v = \boldsymbol{M}_t^{vr} \otimes (1 - \boldsymbol{w}_t\boldsymbol{e}_t)^\top + \boldsymbol{w}_t\boldsymbol{a}_t^\top. \tag{13}$$

ここで，$\boldsymbol{w}_t$ は項目のスキルと潜在スキルの関係性の強さを表す attention であり，項目 j のスキルから計算された埋め込みベクトル $\boldsymbol{s}_j \in \mathbb{R}^{d_k}$ を用いて以下で求める．

$$\boldsymbol{w}_t = Softmax\left(\boldsymbol{M}^k\boldsymbol{s}_j\right). \tag{14}$$

$\boldsymbol{v}_t, \boldsymbol{M}_t^v$ を Hypernetwork で最適化することにより，忘却パラメータ $\boldsymbol{e}_t, \boldsymbol{a}_t$ が学習者の過去の反応データ（潜在能力値）と最新の反応データの反映の度合いを最適化するように推定される．これにより，$\boldsymbol{M}_t^v$ が学習者の過去の学習データの情報を十分に保持できるため，長期の学習においても潜在能力の変化を正確に捉えることができる．

3.3 学習者ネットワーク・項目ネットワーク

学習者の能力パラメータと項目の難易度パラメータは二つの独立なネットワークを用いて DeepIRT [19] と同様に次の手順で求める．なお，以下では可読性のため学習者 i に固定し，i の表記は省略する．学習者ネットワークでは，以下の n 層の深層ニューラルネットワークから学習者が時点 t で項目 j に解答する際の能力値 $\theta^{(t,j)}$ を算出する．学習者ネットワークの層数 n は実データに基づいて最適値を決定する．

$$\boldsymbol{\theta}_1^{(t,j)} = \sum_{l=1}^{N} w_{tl}\left(\boldsymbol{M}_{tl}^v\right)^\top, \tag{15}$$

$$\boldsymbol{\theta}_k^{(t,j)} = \tanh\left(\boldsymbol{W}^{(\theta_k)}\boldsymbol{\theta}_{k-1}^{(t,j)} + \boldsymbol{\tau}^{(\theta_k)}\right), \tag{16}$$

$$\theta^{(t,j)} = \sum_{l=1}^{N} w_{tl}\theta_{nl}^{(t,j)}. \tag{17}$$

ここで，w_{tl} は潜在スキル l の attention を表し，$k = \{2, 3, \cdots, n\}$，$\boldsymbol{\theta}_n^{(t,j)} = \{\theta_{n1}^{(t,j)}, \theta_{n2}^{(t,j)}, \cdots, \theta_{nN}^{(t,j)}\}$ である．$\boldsymbol{\theta}_n^{(t,j)}$ は学習者の多次元の能力ベクトルであり，測定モデルである多次元 IRT における能力値と同様に解釈可能である [33].

項目ネットワークでは項目 j の難易度 β_{item}^j とその項目に必要なスキルの難易度 β_{skill}^j を求め，これらの和を解答する際の難易度とする．β_{item}^j は学習者が解答した項目 j から計算された埋め込みベクトル $\boldsymbol{q}_j$ を用いて n 層のニューラルネットワークで以下のように推定する．

$$\boldsymbol{\beta}_1^{(j)} = \tanh\left(\boldsymbol{W}^{(\beta_1)}\boldsymbol{q}_j + \boldsymbol{\tau}^{(\beta_1)}\right), \tag{18}$$

$$\boldsymbol{\beta}_k^{(j)} = \tanh\left(\boldsymbol{W}^{(\beta_k)}\boldsymbol{\beta}_{k-1}^{(j)} + \boldsymbol{\tau}^{(\beta_k)}\right), \tag{19}$$

$$\beta_{item}^{(j)} = \boldsymbol{W}^{(\beta_n)}\boldsymbol{\beta}_n^{(j)} + \boldsymbol{\tau}^{(\beta_n)}. \tag{20}$$

同様に，項目 j のスキルから計算された埋め込みベクトル $\boldsymbol{s}_j \in \mathbb{R}^{d_k}$ を用いて β_{skill}^j を計算する．

$$\boldsymbol{\gamma}_1^{(j)} = \tanh\left(\boldsymbol{W}^{(\gamma_1)}\boldsymbol{s}_j + \boldsymbol{\tau}^{(\gamma_1)}\right), \tag{21}$$

$$\boldsymbol{\gamma}_k^{(j)} = \tanh\left(\boldsymbol{W}^{(\gamma_k)}\boldsymbol{\gamma}_{k-1}^{(j)} + \boldsymbol{\tau}^{(\gamma_k)}\right), \tag{22}$$

$$\beta_{skill}^{(j)} = \boldsymbol{W}^{(\gamma_n)}\boldsymbol{\gamma}_n^{(j)} + \boldsymbol{\tau}^{(\gamma_n)}. \tag{23}$$

項目ネットワークの層数は $k = \{2, 3, \cdots, n\}$ から実データに基づいて最適値を決定する．このように学習者の能力パラメータと難易度パラメータを独立に計算することにより学習者の能力推定値が項目の特性に依存せず，能力値と難易度の解釈性が向上する．

最後に，独立なネットワークによりそれぞれ算出した能力値と難易度を用いて時点 t での項目 j への正答確率 p_{jt} を予測する．

$$p_{jt} = \sigma\left(\theta^{(t,j)} - (\beta_{item}^{(j)} + \beta_{skill}^{(j)})\right). \tag{24}$$

一般に，深層学習手法は誤差逆伝播法を用いて損失関数 ℓ を最小化することにより，パラメーターを学習する．提案手法の損失関数には Tsutsumi ら (2021) と同様にクロスエントロピーを採用する [19]．予測正答確率 p_{jt} と学習者の真の反応データ $u_{jt} = \{1, 0\}$ から以

表 1　データセットの詳細

Dataset	学習者数	スキル数	項目数	正答率	平均解答項目数
Statics2011	333	1223	N/A	79.8%	180.9
ASSISTments2009	4151	111	26684	63.6%	52.1
ASSISTments2015	19840	100	N/A	73.2%	34.2
ASSISTments2017	1709	102	3162	39.0%	551.0

表 2　δ_1, δ_2 の最適値と反応予測精度 (AUC)

Dataset	ASSISTments2009	ASSISTments2015	ASSISTments2017	Statics2011
δ_1, δ_2,	1.5, 1.5	1.5, 1.5	1.5, 1.5	1.0, 1.7
AUC	81.34 +/- 0.36	72.95 +/- 0.14	85.06+/- 1.17	82.24 +/- 0.54
Acc	76.54 +/- 0.45	75.02 +/- 0.15	79.11+/- 1.06	80.61 +/- 0.83
Loss	0.48 +/- 0.10	0.51 +/- 0.03	0.48 +/- 0.24	0.41 +/- 0.19

下の式でクロスエントロピーを計算する．

$$\ell = -\sum_{t} \left(u_{jt} \log p_{jt} + (1 - u_{jt}) \log(1 - p_{jt}) \right). \quad (25)$$

4. 予測精度評価

本章では，深層学習を用いた代表的な KT 手法 (DKVMN, Yeung, Tsutsumi et al., AKT) [9], [11], [16], [19] 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 Statics2011(注1)，ASSISTments2009，ASSISTments2015，ASSISTments2017(注2) を用いる．データセットの概要を表 1 に示す．表中の平均解答項目数は学習者が解答した全項目数の平均値を示す．各学習データには学習者の反応 $u_{jt} = \{1, 0\}$，解答した項目番号とスキルタグが付与されている．比較実験は学習者が解答した項目に付与されたスキルタグのみを入力とする場合と項目とスキルタグの両方を入力する場合で行う [16], [19]．ただし，Statics2011 と ASSISTments2015 にはスキルの情報しか含まれておらず，項目が区別できないため表 1 の項目数は N/A と表記する．本研究で用いる Statics2011 データセットは先行研究 [9], [19] とスキル数と項目数が異なっているが，現在は Ghosh ら (2020) [16] で公開されたデータセットを用いることが一般的となっている．そのため，本研究ではスキル情報のみをもつ Statics2011 データセット [16] を用いて比較実験を行う．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 [9], [19] 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．

4.1 Hypernetwork の設定

4.1.1 チューニングパラメータ δ_1, δ_2 の推定

本研究では，Hypernetwork を設定するために，データセットごとに最適なチューニングパラメータ δ_1, δ_2 を求める．全てのデータセットにおいて繰り返し回数を $r = 3$ で固定し，$\{\delta_1, \delta_2\}$ の値を変更して実験を行った．結果を表 2 に示す．表 2 より，最適な $\{\delta_1, \delta_2\}$ は ASSISTments2009・ASSISTments2015・ASSISTments2017 では $\{1.5, 1.5\}$ となり，Statics2011 では $\{1.0, 1.7\}$ となった．以下の実験ではこれらの $\{\delta_1, \delta_2\}$ の値を用いる．

4.1.2 繰り返し回数 r の推定

提案手法の反応予測精度は繰り返し回数 r を増加させると凸関数に従う特徴がある．したがって，提案手法では予測精度を最大化するために，各データセットに対して **4.1.1** の結果をもとに $r = 2$ から値を増加させて最適値を決定した．結果を表 3 に示す．繰り返し回数 r はスキルタグ入力の Statics2011 と ASSISTments2017 では $r = 2$，スキルタグ入力の ASSISTments2019 と ASSISTments2015 では $r = 3$，スキルタグと項目タグを入力とした ASSISTments2019 と ASSISTments2017 では $r = 6$ が最適値となった．

4.2 実験結果

学習者の解答した項目のスキルのみを入力データとした場合の予測精度の実験結果を表 4 に示す．表

（注1）：https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=507
（注2）：https://sites.google.com/view/assistmentsdatamining

表 3　r を変化させた場合の反応予測精度 (AUC)

Dataset	Number of rounds r					
	2	3	4	5	6	7
Statics2011 (skill)	**82.25**	82.24	82.20	82.20	82.16	82.11
ASSISTments2009 (skill)	81.19	**81.34**	81.25	81.23	81.2	80.96
ASSISTments2015 (skill)	72.91	**72.95**	72.90	72.89	72.81	72.73
ASSISTments2017 (skill)	**85.06**	82.73	81.64	80.17	73.23	72.64
ASSISTments2017 (item & skill)	75.94	76.17	76.74	76.70	**76.85**	76.74
ASSISTments2009 (item & skill)	81.30	81.14	81.38	81.49	**81.57**	81.20

表 4　スキルタグ入力の場合の反応予測精度

Dataset	metrics	DKVMN	Yeung	Tsutsumi et al.	AKT	Proposed
Statics2011	AUC	81.20 +/- 0.42	81.38 +/- 0.42	81.45 +/- 0.45	82.15 +/- 0.35	**82.25 +/- 0.55**
	Acc	79.24 +/- 0.84	80.33 +/- 0.78	79.18 +/- 0.67	80.41 +/- 0.67	80.63 +/- 0.85
	Loss	0.42 +/- 0.14	0.42 +/- 0.18	0.42 +/- 0.12	0.42 +/- 0.13	0.41 +/- 0.20
ASSISTments2009	AUC	81.21 +/- 0.31	**81.34 +/- 0.39**	**81.34+/- 0.24**	80.81 +/- 0.41	**81.34 +/- 0.36**
	Acc	75.11 +/- 0.66	76.55 +/- 0.45	**76.91+/- 0.24**	76.57 +/- 0.55	76.54 +/- 0.45
	Loss	0.47 +/- 0.05	0.48 +/- 0.10	**0.47+/- 0.10**	0.49 +/- 0.08	0.48 +/- 0.10
ASSISTments2015	AUC	72.61 +/- 0.16	72.53 +/- 0.23	72.34 +/- 0.13	**72.97 +/- 0.12**	72.95 +/- 0.14
	Acc	75.05 +/- 0.18	74.97 +/- 0.14	74.95 +/- 0.39	**75.25 +/- 0.10**	75.02 +/- 0.15
	Loss	0.51+/- 0.02	0.52 +/- 0.03	0.52 +/- 0.02	**0.51 +/- 0.01**	**0.51 +/- 0.03**
ASSISTments2017	AUC	72.67+/- 0.37	72.08 +/- 0.32	72.32+/- 0.69	73.25+/- 0.41	**85.06 +/- 1.17**
	Acc	68.46+/- 0.24	68.36 +/- 0.30	68.07+/- 0.54	69.17+/- 0.70	**79.11 +/- 1.06**
	Loss	0.58+/- 0.03	0.59 +/- 0.07	0.60+/- 0.08	0.58+/- 0.09	**0.48 +/- 0.24**
Average	AUC	76.92	76.83	76.86	77.29	**80.40**
	Acc	74.46	75.05	74.77	75.35	**77.83**
	Loss	0.50	0.50	0.51	0.50	**0.47**

表 5　項目・スキルタグ入力の場合の反応予測精度

Dataset	metrics	Tsutsumi et al.	AKT	Proposed
ASSISTments2009	AUC	80.70 +/- 0.56	**82.20 +/- 0.25**	81.57 +/- 0.39
	Acc	76.13 +/- 0.58	**77.30 +/- 0.55**	76.85 +/- 0.56
	Loss	0.54 +/- 0.10	**0.49 +/- 0.10**	0.53 +/- 0.13
ASSISTments2017	AUC	74.15+/- 0.27	74.54+/- 0.21	**76.85 +/- 0.39**
	Acc	68.73+/- 0.11	69.83+/- 0.15	**71.08 +/- 0.50**
	Loss	0.57+/- 0.06	0.58+/- 0.06	**0.55 +/- 0.06**
Average	AUC	77.42	78.37	**79.21**
	Acc	72.43	73.56	**74.00**
	Loss	0.56	**0.54**	**0.54**

4 より，全ての指標の平均値で提案手法が最も高い反応予測精度を示した．提案手法は既存手法である Tsutsumi ら [19] の予測精度を平均的に上回っており，Hypernetwork を組み込むことで能力値が過去の学習データを正確に反映するように推定され，反応予測精度が向上したと考えられる．また，平均回答数の長い ASSISTments2017 では最先端の AKT 手法の予測精度を大きく上回った．これは，学習過程が長くなるほど提案手法の忘却パラメータが AKT より最適に推定されていることを示唆している．

更に，項目とスキルの両方のタグを入力とした場合の反応予測精度を表 5 に示す．表 4 と同様に，ASSISTments2017 では提案手法が最も高い予測精度を示した．一方，ASSISTments2009 においては AKT が最も高い予測精度を示すことから，AKT は学習過程が長くなるほど予測制度が低下していく可能性がある．図 3 に ASSISTments2017 において AKT で推定された，200 項目に対する全ての学習者の attention（忘却パラメータ）の平均値を示す．縦軸は attention の平均値を表し，横軸は学習者が解答した項目数を表す．図 3 より，現在の反応データより以前に解答されている反応データでは attention が減衰していることが分かる．つ

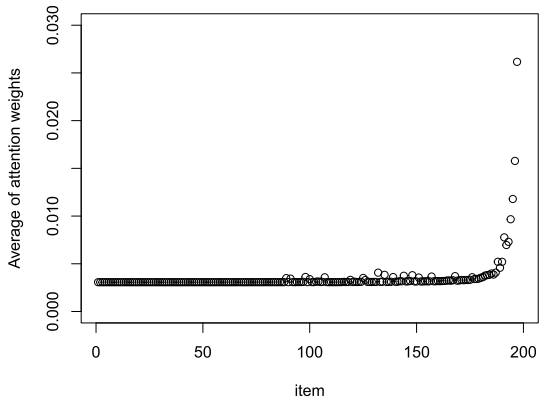

図 3　ASSISTments2017 における AKT の attention 平均値

まり，過去の反応データを徐々に忘却している．しかし，attention の値は特定の値に収束しているため，反応予測に不要な過去の反応データであっても完全に忘却することはできない．したがって，AKT は長期の学習データが入力された場合，過去の反応データを十分に忘却できず，反応予測精度が低下している可能性がある．これらの結果から，Tsutsumi らの DeepIRT [19] に忘却パラメータを最適化する Hypernetwork を組み込むことの有効性が示唆された．

5. 能力パラメータの解釈性評価実験

5.1 能力パラメータの推定精度

本章では，提案手法の能力パラメータの解釈性を評価するために，シミュレーションデータを用いて提案手法と先行研究の DeepIRT 手法 [9], [19] でそれぞれ推定した推定能力値の比較を行う．データセットは Temporal IRT (TIRT) と呼ばれる時系列 IRT を用いて生成する [14], [15]．TIRT では学習者 i が項目 j に時点 t で正答する確率を以下で計算する．

$$P_{ij}(x_{ij}=1 \mid \theta_{it}) = \frac{1}{1+\exp\left(-\tilde{a}_{\Delta_t}(\theta_{it}-b_j)\right)}, \quad (26)$$

$$\tilde{a}_{\Delta_t} = \frac{a_j}{\sqrt{1+\epsilon a_j^2 \Delta_t}}. \quad (27)$$

ここで，$\Delta_t = t-j$ であり，$\tilde{a}_{\Delta_t} \in (0,\infty)$ は時点 t での識別力パラメータ，$b_j \in (-\infty,\infty)$ は項目 j の難易度パラメータである．項目の識別力パラメータ a と難易度パラメータ b の事前分布には，$a \sim LN(0,1), b \sim N(0,1)$ の対数正規分布及び正規分布を用いる．また，$\theta_{it} \in (-\infty,\infty)$ は学習者 i の時点 t の能力値を表す．θ_{it} の事前分布は $\theta_{i0} \sim N(0,1), \theta_{it} \sim N(\theta_{it-1},\epsilon)$ の正規分布とする．ϵ は θ_{it} の分散であると同時に，過去の学習データを忘却度を決める忘却パラメータ（チューニングパラメータ）である．

本実験では，TIRT を用いて学習者 2000 人の $\{50, 100, 200, 300\}$ 項目への反応データをデータセットとして生成した．初めに学習者 1800 人分の反応データを用いて項目の識別力パラメータ a と難易度パラメータ b を推定し，推定した a, b を所与として残りの 200 人の各時点での能力値を推定する．また，各データセットについて ϵ の値を $\epsilon = \{0.1, 0.3, 0.5, 1.0\}$ に変化させて生成したデータセットを用いて実験を行った．ϵ は学習者の能力値の変動を調整するパラメータであり，ϵ の値が大きいほど真の能力値は各時点で大きく変動するように推定される．

本研究では真のモデル (TIRT) で生成された真の能力値と提案手法，先行研究の DeepIRT 手法 [9], [19] で推定された能力値を比較するために，各時点での能力値について Pearson の積率相関係数，Spearman の順位相関係数，Kendall の順位相関係数を求めた．母集団に分布が仮定されている Pearson の積率相関係数に対して，Spearman の順位相関係数は母集団に分布を仮定しないノンパラメトリックな指標である．Kendall の順位相関係数は異常値の頑健な推定について評価する指標である [34]．各相関係数は TIRT と提案手法，先行研究の DeepIRT で推定された各学習者の学習過程での能力値 θ_t，$t \in \{1, 2, \cdots, T\}$ を用いて算出し，その後，全ての学習者の相関係数の平均をとった．一般に，能力値の推定精度評価には 2 乗平均平方根誤差 (Root Mean Squared Error: RMSE) を用いることが多い．しかし，TIRT の学習者の能力パラメータは事前分布が時点ごとに変動し，標準正規分布を仮定していない．提案手法，DeepIRT [9], [19] においても能力パラメータは分布をもたない．したがって，本実験では TIRT 及び提案手法，DeepIRT [9], [19] では能力値の標準化ができず，RMSE を評価することができないため，相関係数で評価する．

表 6 に各条件のデータセットについて算出した相関係数の平均値を示す．結果より，全てのデータセットにおいて提案手法の推定能力値が真の能力パラメータと最も強い相関があることを示した．TIRT では学習過程で学習者の能力値の分布が変化するため，提案手法の Spearman の順位相関係数は，Pearson の相関係数よりも大きくなっている．また，提案手法の Kendall の順位相関係数は ϵ が大きい場合にも高い相関をして

表 6　真の能力値と推定能力値の相関係数

	No. items	50	100	200	300	50	100	200	300	50	100	200	300
ϵ	Method	Pearson				Spearman				Kendall			
0.1	Yeung	0.626	0.667	0.740	0.738	0.626	0.660	0.750	0.745	0.441	0.473	0.550	0.549
	Tsutsumi et al.	0.885	0.907	0.924	0.916	0.892	0.915	0.940	0.938	0.710	0.746	0.785	0.782
	Proposed	**0.902**	**0.916**	**0.930**	**0.927**	**0.910**	**0.923**	**0.943**	**0.941**	**0.736**	**0.761**	**0.790**	**0.792**
0.3	Yeung	0.730	0.799	0.808	0.823	0.751	0.831	0.862	0.873	0.551	0.628	0.659	0.670
	Tsutsumi et al.	0.827	0.891	0.883	0.890	0.863	0.926	0.941	0.945	0.671	0.755	0.778	0.785
	Proposed	**0.840**	**0.905**	**0.900**	**0.907**	**0.877**	**0.932**	**0.947**	**0.954**	**0.689**	**0.767**	**0.791**	**0.804**
0.5	Yeung	0.773	0.800	0.807	0.814	0.812	0.861	0.877	0.890	0.605	0.654	0.676	0.692
	Tsutsumi et al.	0.855	0.870	0.860	0.849	0.893	0.928	0.929	0.930	0.705	0.755	0.758	0.761
	Proposed	**0.874**	**0.871**	**0.869**	**0.859**	**0.901**	**0.928**	**0.934**	**0.940**	**0.720**	**0.755**	**0.768**	**0.779**
1.0	Yeung	0.788	0.809	0.824	0.813	0.834	0.884	0.891	0.888	0.626	0.684	0.695	0.692
	Tsutsumi et al.	0.843	0.830	0.844	0.834	0.886	0.911	0.919	0.918	0.696	0.728	0.740	0.740
	Proposed	**0.854**	**0.840**	**0.854**	**0.836**	**0.894**	**0.920**	**0.930**	**0.919**	**0.708**	**0.744**	**0.762**	**0.743**

しており，提案手法は真の能力値が異常値をとる場合にも頑健に推定できることを示唆している．更に，提案手法は項目数が増えるほどより強い相関を示す傾向がある．これらの結果から，提案手法が過去のデータの忘却度を最適化することにより，長期の学習過程において真の学習者の能力遷移を正確に推定しているといえる．

5.2　能力パラメータの特性比較

本章では，提案手法と Tsutsumi らの DeepIRT [19] で推定した能力パラメータの特性について考察する．パラメータの特性を推定するために，Tsutsumi ら (2021) [19] に従って以下の二つの指標を算出した．

（1）　個人間分散：時点 t で推定された全学習者の能力値の個人間分散 $V_t^{'}$ と全時点での平均値 $V^{'}$．

$$\bar{\theta^t} = \frac{1}{I}\sum_{i=1}^{I}\theta_i^t, \tag{28}$$

$$V_t^{'} = \frac{1}{I}\sum_{i=1}^{I}(\theta_i^t - \bar{\theta^t})^2, \tag{29}$$

$$V^{'} = \frac{1}{T_i}\sum_{t=1}^{T_i}V_t^{'}. \tag{30}$$

ここで，θ_i^t は学習者 i の各時点 t の能力値 $\theta^{(t,j)}$ を表す．

（2）　個人内分散：学習過程 ($t \in \{1, \cdots, T_i\}$) で推定された学習者 i の能力値の分散 V_i と全学習者の平均値 V．

$$\bar{\theta_i} = \frac{1}{T_i}\sum_{t=1}^{T_i}\theta_i^t, \tag{31}$$

$$V_i = \frac{1}{T_i}\sum_{t=1}^{T_i}(\theta_i^t - \bar{\theta_i})^2, \tag{32}$$

表 7　個人間分散

Inputs	Tsutsumi et al.	Proposed
skill	0.1020	2.2128
item&skill	0.0503	0.0896

表 8　個人内分散

Inputs	Tsutsumi et al.	Proposed
skill	0.0704	1.9631
item&skill	0.0339	0.0724

$$V = \frac{1}{I}\sum_{i=1}^{I}V_i. \tag{33}$$

本実験では，**4.** で用いた公開データセットの中で最も長い平均解答項目数をもつ ASSISTments2017 において，スキルタグのみの入力とスキルと項目の両方のタグを入力とした場合の個人間分散・個人内分散を求め，提案手法が長期の学習過程において有効であることを示す．表 7 に提案手法と Tsutsumi ら (2021) [19] の手法で推定した能力値の個人間分散 $V^{'}$ を示す．表 7 より，提案手法は Tsutsumi ら (2021) より高い個人間分散を示すことが分かった．個人間分散は値が大きいほどモデルが学習者間の能力値をよく識別していることを表しているため，提案手法は Tsutsumi ら (2021) より学習者の識別力が高いといえる．

更に，表 8 に各手法で推定した能力値の個人内分散 V を示す．個人内分散は各学習者の学習過程での能力値分散を表しており，値が大きいほど能力値推移の範囲が広いことを意味する．結果より，提案手法は Tsutsumi ら (2021) より高い個人内分散を示すことが分かった．一方で，Tsutsumi ら (2021) は低い値を示しており，推定能力値がほとんど変化していないと考えられる．提案手法は最新の学習データの入力と過去

図 4　提案手法における多次元の能力パラメータ

の反応データの忘却を最適化しながら能力値推定を行うため，長期の学習においても能力推定値は広範囲で大きく変動するが，Tsutsumi ら (2021) では過去の反応データの忘却が最適化されていないために，反応データを正しく反映した能力推定が行われなかった可能性がある．

5.3　能力パラメータの推定値分析

本章では，提案手法を用いて推定した学習者の能力値推移を可視化し，能力パラメータが解釈性をもつことを示す．多次元のスキルに対する能力値変化を可視化することは，教師が学習者の得意・不得意分野を把握し，学習者へ適切な支援を行うために重要である．本研究では，Tsutsumi ら (2021) と同様にデータセット ASSISTments2009 を用いて，ある学生の各スキルの能力値推移を提案手法で推定した．実データにおける評価実験では，学習者の真の能力値が分からないため，推定能力値が反応データを正しく反映しているかを評価する．

ある学習者の 30 項目に対する反応データを用いて提案手法で推定した能力値推移を図 4 に示す．縦軸左側がスキル，縦軸右側は学習者の能力値を表し，横軸は学習者の解答項目を表す．また，学習者が項目に正答した場合を"•"，誤答した場合を"◦"で表す．学習者は四つのスキルに解答しており，それぞれのスキルは"ordering factions"（オレンジ色），"equation solving more than two steps"（青色），"equation solving two or fewer steps"（灰色），"finding percents"（黄色）で示されている．

図 4 より，提案手法はスキル間の関係性を考慮して能力値推定を行うため，各スキルの能力値は全体的に同様に変化する傾向があることが分かる．つまり，学習者がある項目に正しく解答すると，その項目に対応するスキルの能力が上昇するだけでなく，他のスキルの能力も上昇する．特に，"ordering factions"（オレンジ色）と"finding percents"（黄色）のスキルは学習者の反応データをよく反映していることが分かる．一方，"equation solving more than two steps"（青色）と"equation solving two or fewer steps"（灰色）のスキルにおいては，対応するスキルの項目に正答している場合でも能力値は若干上昇するものの，あまり変化していない．これは，"equation solving more than two steps"（青色）と"equation solving two or fewer steps"（灰色）のスキルにそれぞれのスキルと関係が低い項目が含まれているために，反応データをあまり反映しない能力推定値になっていると考えられる．このような現象が起きる場合には，各項目と対応するスキルの関係を検証する必要がある．

6.　む　す　び

本研究では，高精度な反応予測精度とパラメータ解釈性を両立させるために，Tsutsumi らの DeepIRT [19] に新たな Hypernetwork を組み込み，最新の学習者の反応データと直前の学習者の潜在能力値の両方を用いて忘却パラメータを最適化する新たな DeepIRT を提案した．提案手法では，モデルが学習者の潜在能力値を更新する前に，Hypernetwork 内で最新の反応データと前時点での潜在能力値のバランスを調整しながら忘却パラメータを推定する．最適化した忘却パラメータを用いて潜在能力値を更新することで，過去の反応データの適切に反映した能力推定と反応予測が可能となる．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最先端の反応予測手法に比べて向上したことを示した．特に，提案手法の忘却パラメータ最適化は長期の学習において有効であることが分かった．シミュレーションデータを用いた実験では，提案手法が既存の DeepIRT 手法 [9], [19] よりも時系列 IRT の真のパラメータと強い相関をもち，真の能力値が異常値をとる場合にも頑健に推定できることを示した．

謝辞　本研究は JSPS 科研費 JP19H05663, P22K19825, JP22J15279 の助成と 2022 年度岡太彬訓研究助成を受けた．

文　　献

[1]　A.T. Corbett and J.R. Anderson, “Knowledge tracing: Model-

ing the acquisition of procedural knowledge," User Model. User-Adapt. Interact., vol.4, no.4, pp.253–278, Dec. 1995.
[2] 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.
[3] J. Lee and E. Brunskill, "The impact on individualizing student models on necessary practice opportunities," Proc. 5th Int. Conf. Educational Data Mining, pp.118–125, Jan. 2012.
[4] Z. Pardos and N. Heffernan, "T.: Modeling individualization in a bayesian networks implementation of knowledge tracing," Proc. 18th Int. Conf. User Modeling, Adaption, and Personalization, pp.255–266, June 2010.
[5] Z. Pardos and N. Heffernan, "Kt-idem: Introducing item difficulty to the knowledge tracing model," Proc. 19th Int. Conf. User Modeling, Adaptation and Personalization (UMAP 2011), pp.243–254, Jan. 2011.
[6] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L.J. Guibas, and J. Sohl-Dickstein, "Deep knowledge tracing," pp.505–513, Curran Associates, 2015.
[7] 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.
[8] R. Weng and D. Coad, "Real-time bayesian parameter estimation for item response models," Bayesian Analysis, vol.13, Dec. 2016.
[9] C. Yeung, "Deep-irt: Make deep learning based knowledge tracing explainable using item response theory," Proc. 12th Int. Conf. Educational Data Mining, EDM, 2019.
[10] M.V. Yudelson, K.R. Koedinger, and G.J. Gordon, "Individualized bayesian knowledge tracing models," Artificial Intelligence in Education, pp.171–180, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
[11] J. Zhang, X. Shi, I. King, and D.-Y. Yeung, "Dynamic key-value memory network for knowledge tracing," Proc. 26th Int. Conf. World Wide Web, pp.765–774, WWW '17, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 2017.
[12] 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," 信学論（D）, vol.J102-D, no.2, pp.79–92, Feb. 2019.
[13] 堤瑛美子, 植野真臣, "Knowledge tracing のための sliding window 隠れマルコフ irt," 信学論（D）, vol.J103-D, no.12, pp.894–905, Dec. 2020.
[14] K.H. Wilson, Y. Karklin, B. Han, and C. Ekanadham, "Back to the basics: Bayesian extensions of irt outperform neural networks for proficiency estimation," 9th Int. Conf. Educational Data Mining, vol.1, pp.539–544, June 2016.
[15] C. Ekanadham and Y. Karklin, "T-skirt: Online estimation of student proficiency in an adaptive learning system," CoRR, vol.abs/1702.04282, 2017.
[16] A. Ghosh, N. Heffernan, and A.S. Lan, "Context-aware attentive knowledge tracing," Proc. 26th ACM SIGKDD Int. Conf. Knowledge Discovery & Data Mining, 2020.
[17] 木下　涼, 植野真臣, "深層学習によるテスト理論：item deep response theory," 信学論（D）, vol.J103-D, no.4, pp.314–329, April 2020.
[18] 堤瑛美子, 木下　涼, 植野真臣, "独立な学習者・項目ネットワークをもつ deep-irt," 信学論（D）, vol.J104-D, no.7, pp.596–608, July 2021.
[19] E. Tsutsumi, R. Kinoshita, and M. Ueno, "Deep-irt with independent student and item networks," Proc. 14th Int. Conf. Educational Data Mining, EDM, 2021.
[20] H. Sepp and S. Jurgen, "Long short - term memory," Neural Computation, vol.14, pp.1735–1780, 1997.
[21] S. Pandey and G. Karypis, "A self-attentive model for knowledge tracing," Proc. Int. Conf. Education Data Mining, 2019.
[22] D. Ha, A. Dai, and Q.V. Le, "Hypernetworks," arXiv preprint arXiv:1609.09106, 2016.
[23] K. Stanley, D. D'Ambrosio, and J. Gauci, "A hypercube-based encoding for evolving large-scale neural networks," Artificial life, vol.15, pp.185–212, Feb. 2009.
[24] G. Melis, K. Tomáš, and B. Phil, "Mogrifier lstm," Proc. ICLR 2020, 2020.
[25] R. Georg, "Probabilistic models for some intelligence and attainment tests," Studies in mathematical psychology, 1993.
[26] P. Chen, Y. Lu, V. Zheng, and Y. Pian, "Prerequisite-driven deep knowledge tracing," IEEE Int. Conf. Data Mining, ICDM 2018, pp.39–48, 2018.
[27] X. Liangbei and D. Mark, "Dynamic knowledge embedding and tracing," Proc. 13th Int. Conf. Educational Data Mining, EDM, pp.524–530, 2020.
[28] Y. Lu, D. Wang, Q. Meng, and P. Chen, "Towards interpretable deep learning models for knowledge tracing," Proc. 13th Int. Conf. Educational Data Mining, EDM, pp.185–190, 2020.
[29] H. Tong, Y. Zhou, and Z. Wang, "Exercise hierarchical feature enhanced knowledge tracing," Proc. 13th Int. Conf. Educational Data Mining, EDM, pp.324–328, 2020.
[30] Z. Wang, X. Feng, J. Tang, G. Huang, and Z. Liu, "Deep knowledge tracing with side information," 20th Int. Conf. Artificial Intelligence in Education (AIED), pp.303–308, 2019.
[31] X. Xiong, S. Zhao, V. Inwegen, Eric G., and J.E. Beck, "Going deeper with deep knowledge tracing," Proc. Int. Conf. Education Data Mining, 2016.
[32] C.K. Yeung and D.-Y. Yeung, "Addressing two problems in deep knowledge tracing via prediction-consistent regularization," Proc. 5th ACM Conf. Learning @ Scale, pp.1–10, 2018.
[33] T. Ackerman, "Unidimensional irt calibration of compensatory and noncompensatory multidimensional items," Applied Psychological Measurement, vol.13, pp.113–127, June 1989.
[34] E. Tsutsumi, R. Kinoshita, and M. Ueno, "Deep item response theory as a novel test theory based on deep learning," Electronics, vol.10, no.9, 2021. https://www.mdpi.com/2079-9292/10/9/1020

（2022 年 3 月 18 日受付, 8 月 1 日再受付,
10 月 25 日早期公開）

堤　瑛美子

2020 電気通信大学大学院情報理工学研究科博士前期課程了．同年，電気通信大学大学院情報理工学研究科博士後期課程入学，在学中．

郭　　亦鳴

2020 College of Software. Jilin University 卒．2021 電気通信大学大学院情報理工学研究科博士前期課程入学，在学中．

植野　真臣　（正員）

1992 神戸大学大学院教育学研究科了，1994 東京工業大学大学院総合理工学研究科了．博士（工学）．東京工業大学，千葉大学，長岡技術科学大学を経て 2006 より電気通信大学助教授，2013 より教授，現在に至る．