




Time- and Learner-Dependent Hidden Markov Model for Writing Process Analysis Using Keystroke Log Data

Masaki Uto¹  · Yoshimitsu Miyazawa² · Yoshihiro Kato³ · Koji Nakajima³ · Hajime Kuwata³

Published online: 12 March 2020

© International Artificial Intelligence in Education Society 2020

Abstract

Teaching writing strategies based on writing processes has attracted wide attention as a method for developing writing skills. The writing process can be generally defined as a sequence of subtasks, such as planning, formulation, and revision. Therefore, instructor feedback is often given based on sequence patterns of those subtasks. For such feedback, instructors need to analyze sequence patterns for all learners, which becomes problematic as the number of learners increases. To resolve this problem, this study proposes a new machine-learning method that estimates sequence patterns from keystroke log data. Specifically, we propose an extension of the Gaussian hidden Markov model that incorporates parameters representing temporal change in a subtask appearance distribution for each learner. Furthermore, we propose a collapsed Gibbs sampling algorithm as the parameter estimation method for the proposed model. We demonstrate effectiveness of the proposed model by applying it to actual keystroke log datasets.

Keywords Writing skills · Writing process · Keystroke log · Hidden Markov model · Markov chain Monte Carlo method

✉ Masaki Uto
uto@ai.lab.uec.ac.jp

¹ The University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo 182-8585, Japan

² The National Center for University Entrance Examinations, 2-19-23 Komaba, Meguro, Tokyo 153-8501, Japan

³ Benesse Educational Research and Development Institute, 1-34 Ochiai, Tama, Tokyo 206-8686, Japan

Introduction

Recently, the importance of nurturing writing skills in higher education has been widely acknowledged (Uto and Ueno 2015). A typical instruction method for writing is an instructor providing feedback on a completed text. As another approach, instruction methods that focus on the writing process have attracted attention in recent years (Deane and Zhang 2015; Leijten and Waes 2013; Seow 2002; Zhang et al. 2016; Bayat 2014; Conijn et al. 2018).

The writing process can be generally regarded as a sequence of subtasks, such as *planning*, *formulation*, and *revision* (Flower and Hayes 1981; Seow 2002; Bayat 2014; Southavilay et al. 2010b). These processes are known to be dependent on writing skills (Stevenson et al. 2006; Hayes and Flower 1980; Sasaki 2000, 2002; Larios et al. 2008; Chan 2017). For example, learners with advanced skills tend to formulate faster but spend more time on revisions, as compared with those with lower skills (Sasaki 2000, 2002; Larios et al. 2008). In addition, learners with advanced skills tend to make major edits in logical structures and main arguments, while those with lesser skills primarily perform superficial corrections such as expressions and typographical errors (Sasaki 2000, 2002; Lester and Witte 1981; Barkaoui 2016). Because there is a relation between writing skills and writing processes, instruction based on the writing process can be an effective approach toward improving writing skills (Bayat 2014; Conijn et al. 2018).

Instructions focused on the writing process are often based on the appearance pattern of the above-described subtasks (Bayat 2014; Conijn et al. 2018). As a method for analyzing appearance patterns of these subtasks, the *think-aloud technique* and *video playback stimulation method* have been long used (Stevenson et al. 2006; de Larios et al. 2008). In the think-aloud technique, learners sequentially utter their thoughts during the writing process. The playback stimulation method presents learners with videos of their writing process and has learners discuss their thoughts. However, these methods require considerable time for analysis, so they are impractical when there are many learners.

To address this issue, writing process analysis methods using keystroke log data recorded during composition on a computer have been recently proposed (Deane and Zhang 2015; Leijten and Waes 2013; Zhang et al. 2016; Chan 2017; Conijn et al. 2018). For example, Zhang et al. (2016) proposed a method in which the writing process is categorized based on the distribution of intervals between word inputs. However, while existing methods can categorize stages in the writing processes, there is no method for estimating subtask appearance patterns by learner.

We therefore propose a method for estimating subtask appearance patterns by learner from keystroke log data. Specifically, we propose a method for converting keystroke log data of each learner to time-series data with multiple features that express writing characteristics for applying an unsupervised machine learning method. Feature extraction is based on a sliding window approach, which divides keystroke log data into analytical frames with a small time-width and extracts features from each frame. The Gaussian hidden Markov model (GHMM) is well-known as a typical unsupervised machine learning method for such time-series data. GHMM assumes that observational data at an arbitrary point in time arise depending on a

latent variable called the *state*, and that the state sequence can be estimated from the data. Therefore, by applying GHMM under an assumption of latent state for each analytical frame as a subtask, a sequence of subtasks for each learner can be estimated from keystroke log data. However, because this approach estimates a subtask for each analytical frame with a small time-width, the variety of subtask sequences becomes extremely large, hindering interpretation of the writing process for each learner. For educational application, knowing the subtask to be performed each moment is not necessarily important. What we need, instead, is to know the subtask appearance patterns by learner, as discussed earlier. The information required to understand the patterns is the appearance ratios for subtasks in a time interval with a certain time span and the temporal changes in the ratios. For example, we wish to know the extent to which each learner performs subtasks such as formulation, major edits, and superficial corrections in every quarter of writing time. Information on temporal changes in the subtask appearance ratios will be helpful by allowing learners and instructors to characterize the writing patterns of each learner quantitatively. Furthermore, it will also be beneficial for instructors by allowing them to give appropriate feedback and instruction toward improving learners' writing activities.

For the above reasons, this paper proposes an extension of GHMM that incorporates parameters representing temporal change in the subtask appearance distribution for each learner. For the model, we first divide feature vector sequences obtained from each learner's keystroke data into a few time intervals. We then incorporate parameters that express state appearance probabilities for each learner in each time interval in the GHMM. The characteristics of the proposed model are as follows:

1. Because the incorporated parameters represent temporal change in subtask appearance patterns for each learner, by interpreting the parameters we can understand the writing process of each learner.
2. By comparing differences in state appearance distributions between learners, we can quantitatively analyze between-learner differences in the writing process.
3. The writing processes of learners can be categorized by applying typical cluster analyses, taking differences in state appearance distributions between learners as the distance function.

As a method for estimating parameters for the proposed model, we propose a collapsed Gibbs sampling algorithm, which is a type of Markov-chain Monte Carlo method. This paper demonstrates the effectiveness of the proposed method through evaluation testing applied to actual keystroke log data.

Related Works

This section describes related works on keystroke data applications.

The most common application of keystroke data is user authentication in the security domain. Many user authentication methods that use keystroke data have been proposed (Karnan et al. 2011; Teh et al. 2013; Quraishi and Bedi 2018). In these methods, users are identified by using supervised classifiers trained on keystroke features. Various statistical and probabilistic models and machine learning methods

have been used for classification (Karnan et al. 2011; Teh et al. 2013; Quraishi and Bedi 2018). Hidden Markov models (HMMs), which are used in this study, have also been used for user authentication (Chen and Chang 2004; Rodrigues et al. 2005; Ali et al. 2016). In HMM-based authentication methods, an HMM is trained for each user from sequences of keystroke timing features such as the durations of key presses and the time elapsed between key presses. User authentication is performed by checking how well a set of keystroke data from an authentication attempt fits the pre-trained HMMs.

Another application of keystroke data is emotion recognition (Epp et al. 2011; Salmeron-Majadas et al. 2018). Emotion-estimation methods classify user emotion by applying a supervised machine learning classifier trained from keystroke features for each emotion class.

As approaches for general pattern recognition tasks including writing-pattern recognition, temporal interval Bayesian networks (Zhang et al. 2013) and a generative probabilistic model with Allen's interval-based relations (Liu et al. 2016, 2018) have recently been proposed. These methods can capture complex temporal relations among the occurrences of observable features (so-called atomic/primitive actions) by using Allen's interval algebra (Allen 1983) and Bayesian networks. Such approaches have been used for classification of sequential data, achieving state-of-the-art accuracy for applications such as classifications of sports videos (Zhang et al. 2013; Liu et al. 2016), human actions (Liu et al. 2018), and facial expression (Wang et al. 2013).

It is important to note that the above-mentioned approaches focus on classification. The approaches are not suitable for our research objective because the purpose of this study is to estimate the hidden processes underlying observable keystroke activities, not to classify the datasets.

For a method focused on estimation of the writing process, Southavilay et al. (2010a) proposed a method for estimating the process of collaborative writing activities by using an HMM. The proposed method collects the versions of a document produced during collaborative writing and estimates changes in semantic meaning for the next two versions according to predefined heuristics (Southavilay et al. 2010b). The process of collaborative writing is then analyzed by an HMM trained on the sequences of semantic meanings. The purpose and approaches are similar to those of our study, but that method does not use keystroke data. Moreover, that study analyzes the writing process by interpreting the parameters of an HMM trained for each collaborative group. In this approach, we cannot compare writing processes among groups because the means of the latent states will differ among the groups. Although this approach might be extendable to writing process analysis for each learner, the interpretation of each process would be infeasible.

Keystroke Logging System and Log Data

This study assumes that writing tasks are presented to learners, and that keystroke log data are collected as learners compose their responses. To collect keystroke log data, we developed a keystroke logging system similar to those in previous studies (Deane and Zhang 2015; Leijten and Waes 2013; Salmeron-Majadas et al. 2018). Figure 1



Fig. 1 Interface of keystroke logging system (the writing task on the left side is hidden for copyright reasons)

shows the interface of the developed system. The system presents learners with a writing task in the left panel and a text input area in the right panel. The system records information on typed characters, cursor position, and timestamps for each learner input made to the text area using the keyboard or mouse. The time at which learners access the system is recorded as the response start time. Therefore, keystroke log datasets for each learner consist of a sequence of tuples in the format (written text, cursor position, timestamp), with the number of tuples being the number of keyboard operations plus one. The keystroke logging function is implemented in Javascript and works on our e-testing platform, developed in Java. The system stores obtained keystroke log data in an SQL database.

In this study, we use keystroke log data obtained in this manner to analyze the writing process.

Feature Extraction

Previous studies on analyses of keystroke log data analyzed the writing process based on multiple features, such as the number of characters and keystroke interval times as extracted from each learner's keystroke series (Deane and Zhang 2015; Leijten and Waes 2013; Zhang et al. 2016; Chan 2017). As described in “[Introduction](#)”, previous studies extracted one set of feature values for each learner, and used those features to categorize writing patterns. In contrast, the present study aims to estimate temporal change in the subtasks of each learner. Thus, keystroke log data for each learner must be defined as time-series feature data.

To extract feature sequences, we use a sliding window approach like that widely used for image processing and speech recognition (Leijten and Waes 2013). The sliding window approach extracts features with analytical frame units by building

frames with a small time-width from time-series data. The width of each analytical frame W is called the frame width, and the movement width of adjacent frames H is called the step width. If $H < W$, overlapping of adjacent frames is permitted. Figure 2 shows the sliding window approach.

In this study, we apply the sliding window approach to keystroke log data for each learner. We extract the seven features listed in Table 1 from each analytic frame (those features are designated as *writing features*). These features are commonly used in similar studies (Deane and Zhang 2015; Leijten and Waes 2013; Zhang et al. 2016).

By extracting these features for each analytical frame, we can define keystroke log data as series data for seven dimensions of writing features. Specifically, letting $X_{ijf} \in \mathbb{R}$ be feature $f \in \mathcal{F} = \{1, \dots, F = 7\}$ for the $j \in \mathcal{J} = \{1, \dots, J\}$ -th analytical frame of learner $i \in \mathcal{I} = \{1, \dots, I\}$, series data for writing features of learner i are defined as $X_i = \{X_{i1}, \dots, X_{iJ}\}$ (where $X_{ij} = \{X_{ij1}, \dots, X_{ijF}\}$).

We expect that by applying a machine learning method to dataset $X = \{X_1, \dots, X_I\}$, the subtask for each analytical frame of each learner can be estimated. We use an unsupervised machine learning method for this estimation, because it is unclear what subtask types exist within the analytical frame.

Gaussian Hidden Markov Model-based Writing Process Analysis

As discussed in “Feature Extraction”, data X_i are time-series data, and there likely is a dependency of subtasks between adjacent analytical frames. A typical unsupervised classifier for such time-series data is the hidden Markov model (HMM). Especially in the case where observed data have continuous values, as in the present study, GHMM

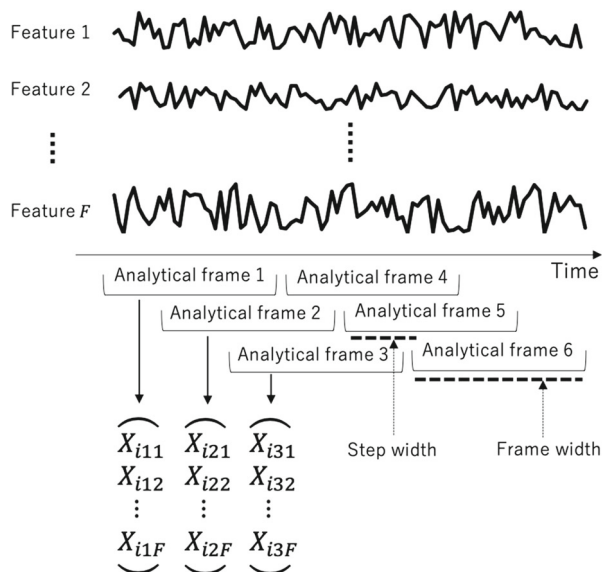


Fig. 2 Feature extraction based on the sliding window approach

Table 1 Writing features

| Index | Feature |
|-------|---|
| 1. | Average number of characters |
| 2. | Number of bursts (continuous inputs within a one-second interval) |
| 3. | Number of stops (no input for at least five seconds) |
| 4. | Number of character-adding operations |
| 5. | Number of character-subtracting operations |
| 6. | Mean relative cursor position (cursor position \div number of characters) |
| 7. | Number of times the cursor is moved |

is generally used. The following provides details of GHMM assuming application to our dataset X .

GHMM assumes a latent variable $S_{ij} \in \mathcal{S} = \{1, \dots, S\}$, called the *state*, for each feature X_{ij} . Each state S_{ij} is obtained according to a transition probability that is dependent on the state immediately before $S_{i,j-1}$. Specifically, letting the transition probability from state s to state s' be $A_{ss'}$ (where $0 \leq A_{ss'} \leq 1$, $\sum_{s'=1}^S A_{ss'} = 1$) and letting A_s be S -dimensional multinomial distribution $\{A_{s1}, \dots, A_{sS}\}$, the probability of state S_{ij} being dependent on state $S_{i,j-1} = s$ can be written as $P(S_{ij}|S_{i,j-1} = s, A_s) = A_{s,S_{ij}}$. The initial state S_{i1} is obtained following $p(S_{i1}|\pi) = \pi_{S_{i1}}$ in accordance with the initial probabilities, which are defined as the S -dimensional multinomial distribution $\pi = \{\pi_1, \dots, \pi_S\}$ (where $0 \leq \pi_s \leq 1$, $\sum_{s=1}^S \pi_s = 1$).

The features for each analytic frame X_{ij} follow a normal distribution dependent on state S_{ij} . In this study, given state $S_{ij} = s$, we assume that the f -th feature X_{ijf} follows a normal distribution with values of mean μ_{sf} and variance σ_{sf}^2 : $p(X_{ijf}|\mu_{sf}, \sigma_{sf}^2) = N(\mu_{sf}, \sigma_{sf}^2)$. Therefore, the emission probability for features X_{ij} can be obtained from the following equation when state $S_{ij} = s$:

$$p(X_{ij}|S_{ij} = s, \mu, \sigma) = \prod_{f=1}^F p(X_{ijf}|\mu_{sf}, \sigma_{sf}^2), \quad (1)$$

where $\mu = \{\mu_{11}, \dots, \mu_{SF}\}$ and $\sigma = \{\sigma_{11}, \dots, \sigma_{SF}\}$.

By applying GHMM under the assumption of latent states for each analytical frame as subtasks, a sequence of subtasks for each learner is expected to be estimated. In this approach, however, a subtask is estimated for each analytical frame, which is defined with a small time-width and which overlaps adjacent multiple frames. Therefore, subtask sequence patterns become significantly large, making it difficult to grasp temporal changes in subtasks for each learner.

Proposed Model

To address this issue, this study proposes an expanded GHMM model that incorporates parameters representing temporal change in subtasks for each learner. In this

study, we divided series data for each learner into a small number of time intervals, and incorporated into the GHMM a parameter representing the subtask appearance probability for each time interval for each learner. Specifically, time-series data are divided into T time intervals $\mathcal{T} = \{1, \dots, T\}$ with a constant time span. We then incorporate a state appearance probability ϕ_{its} that learner i is in state (subtask) s in time interval $t \in \mathcal{T}$, as shown in Fig. 3. In this figure, ϕ_{it} represents a state appearance distribution of learner i in time interval t that is defined as S -dimensional multinomial distribution $\{\phi_{it1}, \dots, \phi_{itS}\}$.

In the proposed model, state probabilities for individual analytical frames are assumed to follow the product of the transition probability and the state appearance probability, as

$$P(S_{ij}|S_{i,j-1} = s, \mathbf{A}_s, \boldsymbol{\phi}_i) \propto A_{s,S_{ij}} \cdot \phi_{i,t_{ij},S_{ij}}, \quad (2)$$

where $\boldsymbol{\phi}_i = \{\phi_{i11}, \dots, \phi_{iT S}\}$ and $t_{ij} \in \mathcal{T}$ is the time interval to which data X_{ij} belong.

Similarly, we assume that the initial probability is defined as

$$P(S_{i1}|\boldsymbol{\pi}, \boldsymbol{\phi}_i) \propto \pi_{S_{i1}} \cdot \phi_{i,1,S_{i1}}. \quad (3)$$

The features are obtained using (1) in the same manner as in GHMM.

In the proposed model, the subtask appearance distribution ϕ_{it} can be estimated in an arbitrary time interval for each learner. Thus, by interpreting temporal change in the distribution, writing pattern trends of each learner can be quantitatively grasped. Furthermore, by measuring differences in the subtask appearance distributions $\boldsymbol{\phi}_i$ between learners, using for example the Kullback–Leibler divergence or the Jensen–Shannon divergence, differences in the writing process between learners can be quantitatively evaluated. In addition, categorizing writing processes becomes possible using typical cluster methods with those divergence functions.

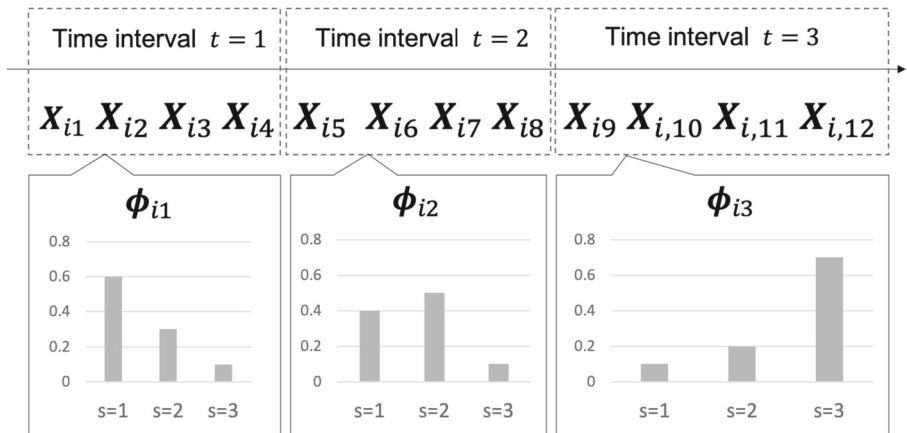


Fig. 3 State appearance distributions for each time interval

Parameter Estimation Algorithm

Representative parameter estimation methods for GHMM are maximum likelihood estimation using the Baum–Welch algorithm and Bayesian estimation using Markov chain Monte Carlo (MCMC) (Bishop 2006). For complex models, Bayesian estimation by MCMC would provide higher accuracy (Bishop 2006; Brooks et al. 2011). This method estimates the posterior distribution of each parameter and uses expected or maximum values as a point estimate for parameters. MCMC approximates posterior distributions via sampling. This study proposes a collapsed Gibbs sampling (CGS) algorithm for the proposed model. CGS has been widely used for learning-topic modeling and HMM as a method to improve the efficiency of MCMC by marginalizing out a part of the parameter set (Griffiths and Steyvers 2004; Griffiths et al. 2004; Paisley and Carin 2009).

CGS repeatedly samples parameter values from the conditional posterior distribution for each parameter, approximating the posterior distribution of parameters using the obtained samples. Conditional posterior distribution is defined as a distribution in which all parameters other than those of interest are given, after marginalizing a specific parameter set from the joint distribution. In this study, we marginalize the initial probability π , transition probability A , and state appearance probability ϕ , and sample the latent state $S = \{S_{11}, \dots, S_{IL}\}$ and emission distribution parameters $\xi = \{\mu, \sigma\}$.

The remainder of this subsection presents the details of this algorithm. Figure 4 shows a graphical representation of the proposed model for the following derivation. In the figure, α , β , and γ represent hyperparameters for the distributions of A , ϕ , and π , respectively, while μ_0 , n_0 , g_1 , and g_2 are hyperparameters for the emission distribution.

Conditional Posterior Distribution for Sampling State S_{ij} for $j > 1$

We first derive the conditional posterior distribution of state S_{ij} for $j > 1$.

Letting $X^{\setminus ij} = X \setminus \{X_{ij}\}$ and $S^{\setminus ij} = S \setminus \{S_{ij}\}$, the conditional posterior distribution where $S_{ij} = s$ is obtained for $j > 1$ can be written as

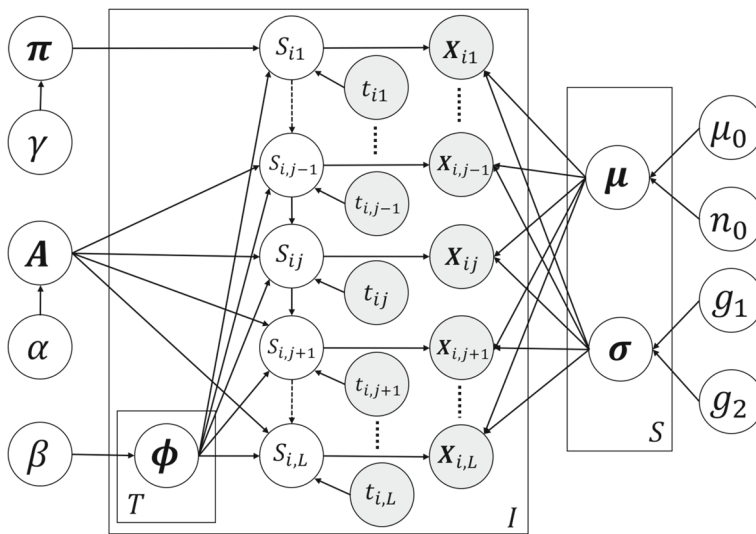
$$p(S_{ij} = s | X_{ij}, X^{\setminus ij}, S^{\setminus ij}, \xi) \propto p(X_{ij} | S_{ij} = s, \xi) \cdot p(S_{ij} = s | S^{\setminus ij}). \quad (4)$$

The first term on the right side of (4) is obtained using (1). Furthermore, by omitting constant terms, the second term can be written as

$$\begin{aligned} p(S_{ij} = s | S^{\setminus ij}) &\propto p(S_{ij} = s, S_{i,j+1} | S^{\setminus i,j,j+1}) \\ &\propto p(S_{i,j+1} | S_{ij} = s, S^{\setminus i,j,j+1}) \cdot p(S_{ij} = s | S_{i,j-1}, S^{\setminus i,j-1,j,j+1}), \end{aligned} \quad (5)$$

where $S^{\setminus i,j,j+1} = S \setminus \{S_{ij}, S_{i,j+1}\}$ and $S^{\setminus i,j-1,j,j+1} = S \setminus \{S_{i,j-1}, S_{ij}, S_{i,j+1}\}$.

If Dirichlet priors with hyperparameters α and β are respectively used for the transition probabilities A_s and the state appearance probabilities ϕ_{it} , then by omitting



constant terms, the first term on the right side of (5) can be reorganized as

$$\begin{aligned}
p(S_{i,j+1}|S_{ij} = s, \mathbf{S}^{\setminus i,j,j+1}) \\
&\propto \int p(S_{i,j+1}|\mathbf{A}_s) \cdot p(\mathbf{A}_s|\mathbf{S}^{\setminus i,j,j+1})d\mathbf{A}_s \\
&\quad \cdot \int p(S_{i,j+1}|\boldsymbol{\phi}_{i,t_{i,j+1}}) \cdot p(\boldsymbol{\phi}_{i,t_{i,j+1}}|\mathbf{S}^{\setminus i,j,j+1})d\boldsymbol{\phi}_{i,t_{i,j+1}} \\
&= \frac{n_{s,S_{i,j+1}}^{\setminus i,j,j+1} + \alpha}{\sum_{s'=1}^S \left(n_{s,s'}^{\setminus i,j,j+1} + \alpha \right)}, \tag{6}
\end{aligned}$$

where $n_{s,s'}^{i,j,j+1}$ represents the frequency at which state s transitioned to state s' among $\mathbf{S}^{i,j,j+1}$.

The second term on the right side of (5) is similarly rewritten as

$$\begin{aligned}
p(S_{ij} = s | S_{i,j-1}, \mathbf{S}^{\setminus i,j-1,j,j+1}) \\
&\propto \int p(S_{ij} = s | \mathbf{A}_{S_{i,j-1}}) \cdot p(\mathbf{A}_{S_{i,j-1}} | \mathbf{S}^{\setminus i,j-1,j,j+1}) d\mathbf{A}_{S_{i,j-1}} \\
&\quad \cdot \int p(S_{ij} = s | \boldsymbol{\phi}_{i,t_{ij}}) \cdot p(\boldsymbol{\phi}_{i,t_{ij}} | \mathbf{S}^{\setminus i,j-1,j,j+1}) d\boldsymbol{\phi}_{i,t_{ij}} \\
&= \frac{n_{S_{i,j-1},s}^{\setminus i,j-1,j,j+1} + \alpha}{\sum_{s'=1}^S \left(n_{S_{i,j-1},s'}^{\setminus i,j-1,j,j+1} + \alpha \right)} \cdot \frac{n_{i,t_{ij},s}^{\setminus i,j-1,j,j+1} + \beta}{\sum_{s'=1}^S \left(n_{i,t_{ij},s'}^{\setminus i,j-1,j,j+1} + \beta \right)} \\
&\propto (n_{S_{i,j-1},s}^{\setminus i,j-1,j,j+1} + \alpha) \cdot (n_{i,t_{ij},s}^{\setminus i,j-1,j,j+1} + \beta). \tag{7}
\end{aligned}$$

In these equations, $n_{s,s'}^{\setminus i,j-1,j,j+1}$ represents the appearance frequency at which state s transitioned to state s' among $\mathcal{S}^{\setminus i,j-1,j,j+1}$. Furthermore, $n_{i,t,s}^{\setminus i,j-1,j,j+1}$ represents the appearance frequency of state s in a state set $\{S_{ij} \in \mathcal{S}^{\setminus i,j-1,j,j+1} | t_{ij} = t\}$ for learner i .

From the above, the conditional posterior distribution of S_{ij} for $j > 1$ can be described as

$$p(S_{ij} = s | \mathbf{X}_{ij}, \mathbf{X}^{\setminus ij}, \mathbf{S}^{\setminus ij}, \boldsymbol{\xi}) \propto \left[\prod_{f=1}^F p(X_{ijf} | \mu_{sf}, \sigma_{sf}^2) \right] \cdot \frac{n_{s,S_{i,j+1}}^{\setminus i,j,j+1} + \alpha}{\sum_{s'=1}^S (n_{s,s'}^{\setminus i,j,j+1} + \alpha)} \cdot (n_{S_{i,j-1},s}^{\setminus i,j-1,j,j+1} + \alpha) \cdot (n_{i,t_{ij},s}^{\setminus i,j-1,j,j+1} + \beta). \quad (8)$$

Conditional Posterior Distribution for Sampling Initial States

The conditional posterior distribution of initial state S_{i1} can be written as

$$p(S_{i1} = s | \mathbf{X}_{i1}, \mathbf{X}^{\setminus i1}, \mathbf{S}^{\setminus i1}, \boldsymbol{\xi}) \propto p(\mathbf{X}_{i1} | S_{i1} = s, \boldsymbol{\xi}) \cdot p(S_{i1} = s | \mathbf{S}^{\setminus i1}). \quad (9)$$

Here, the first term on the right side of (9) is obtained using (1), while the second term can be written as

$$p(S_{i1} = s | \mathbf{S}^{\setminus i1}) \propto p(S_{i2} | S_{i1} = s, \mathbf{S}^{\setminus i,1,2}) \cdot p(S_{i1} = s | \mathbf{S}^{\setminus i,1,2}). \quad (10)$$

The first term on the left side of the above equation is calculable from (6). When the Dirichlet prior with hyperparameter γ is used for the initial distribution $\boldsymbol{\pi}$, the second term can be expressed by omitting constant terms as

$$\begin{aligned} p(S_{i1} = s | \mathbf{S}^{\setminus i,1,2}) &\propto \int p(S_{i1} = s | \boldsymbol{\pi}) \cdot p(\boldsymbol{\pi} | \mathbf{S}^{\setminus i,1,2}) d\boldsymbol{\pi} \cdot \int p(S_{i1} = s | \boldsymbol{\phi}_i) \cdot p(\boldsymbol{\phi}_i | \mathbf{S}^{\setminus i,1,2}) d\boldsymbol{\phi} \\ &= \frac{n_s^{\setminus i,1,2} + \gamma}{\sum_{s'=1}^S (n_{s'}^{\setminus i,1,2} + \gamma)} \cdot \frac{n_{i,1,s}^{\setminus i,1,2} + \beta}{\sum_{s'=1}^S (n_{i,1,s'}^{\setminus i,1,2} + \beta)} \\ &\propto (n_s^{\setminus i,1,2} + \gamma) \cdot (n_{i,1,s}^{\setminus i,1,2} + \beta), \end{aligned} \quad (11)$$

where $n_s^{\setminus i,1,2}$ represents the appearance frequency of S_{i1} among $\mathbf{S}^{\setminus i,1,2}$ becoming state s , while $n_{i,1,s}^{\setminus i,1,2}$ represents the appearance frequency of state s in the state set $\{S_{ij} \in \mathbf{S}^{\setminus i,1,2} | t_{ij} = 1\}$ for learner i .

Thus, the sampling distribution of S_{i1} is obtained as

$$p(S_{i1} = s | \mathbf{X}_{i1}, \mathbf{X}^{\setminus i1}, \mathbf{S}^{\setminus i1}, \boldsymbol{\xi}) \propto \left[\prod_{f=1}^F p(X_{ijf} | \mu_{sf}, \sigma_{sf}^2) \right] \cdot \frac{n_{s,S_{i2}}^{\setminus i,1,2} + \alpha}{\sum_{s'=1}^S (n_{s,s'}^{\setminus i,1,2} + \alpha)} \cdot (n_s^{\setminus i,1,2} + \gamma) \cdot (n_{i,1,s}^{\setminus i,1,2} + \beta). \quad (12)$$

Conditional Posterior Distribution of Emission Probability Parameters

The conditional posterior distributions of emission probability parameters μ_{sf} and σ_{sf}^2 for feature f can be expressed as

$$p(\mu_{sf}|X, S, \xi^{\setminus s, f}, \sigma_{sf}^2) \propto p(\mu_{sf}|X_f^s, \sigma_{sf}^2) \quad (13)$$

$$p(\sigma_{sf}^2|X, S, \xi^{\setminus s, f}, \mu_{sf}) \propto p(\sigma_{sf}^2|X_f^s, \mu_{sf}), \quad (14)$$

where $\xi^{\setminus s, f} = \xi \setminus \{\mu_{sf}, \sigma_{sf}^2\}$ and $X_f^s = \{X_{ijf} \in X | S_{ij} = s, i \in \mathcal{I}, j \in \mathcal{J}\}$. These equations are consistent with the conditional posterior distributions of a typical normal distribution for a sample set X_f^s . The normal distribution $N(\mu_0, \sigma_{sf}^2/n_0)$ is generally used as the conjugate prior of the mean parameter μ_{sf} , where μ_0 and n_0 are hyperparameters. Concretely, the conditional posterior probability of μ_{sf} is written as Uto and Ueno (2016) and Fox (2010)

$$p(\mu_{sf}|X_f^s, \sigma_{sf}^2) = N\left(\frac{n_0\mu_0 + |X_f^s| \cdot \bar{X}_f^s}{n_0 + |X_f^s|}, \frac{\sigma_{sf}^2}{n_0 + |X_f^s|}\right), \quad (15)$$

where $\bar{X}_f^s = \sum_{x \in X_f^s} \frac{x}{|X_f^s|}$ and $|X_f^s|$ indicates the number of data points in X_f^s .

The inverse gamma distribution $IG(g_1/2, g_2/2)$, a type of conjugate prior, is often used as the prior distribution of normal distribution variance σ_{sf}^2 (Gelman 2006), where g_1 and g_2 are hyperparameters and are generally small positive real numbers, such as $g_1 = g_2 = 0.01$. Specifically, the conditional posterior distribution of variance σ_{sf}^2 can be expressed as Uto and Ueno (2016) and Fox (2010)

$$p(\sigma_{sf}^2|X_f^s, \mu_{sf}) = IG\left(\frac{g_1 + |X_f^s|}{2}, \frac{\sigma_0^2}{2}\right), \quad (16)$$

where

$$\sigma_0 = g_2 + \sum_{x \in X_f^s} (x - \bar{X}_f^s)^2 + \frac{|X_f^s| \cdot n_0}{|X_f^s| + n_0} \cdot (\bar{X}_f^s - \mu_0). \quad (17)$$

The algorithm proposed by Tanizaki (2008) is useful for obtaining random samples from an inverse gamma distribution.

Estimation of Marginalized Parameters

Given the obtained state samples, we can estimate the initial probabilities π , transition probabilities A , and state appearance probabilities ϕ as follows:

$$\pi_s = \frac{n_s + \gamma}{\sum_{s'=1}^S (n_{s'} + \gamma)} \quad (18)$$

$$A_{ss'} = \frac{n_{ss'} + \alpha}{\sum_{s'=1}^S (n_{ss'} + \alpha)} \quad (19)$$

$$\phi_{its} = \frac{n_{its} + \beta}{\sum_{s'=1}^S (n_{its'} + \beta)} \quad (20)$$

In these equations, n_s is the frequency at which initial state S_{i1} becomes state s , $n_{ss'}$ is the frequency at which the state transitioned from s to s' , and n_{its} is the frequency of the state becoming s among a state set with time interval t for learner i .

Algorithm

CGS of the proposed model repeatedly samples states S and the parameters of the emission probability distribution $\xi = \{\mu, \sigma\}$ according to the equations introduced in the previous subsection. Specifically, (8) is used to sample $\{S_{ij} \in S \mid j > 1, i \in \mathcal{I}\}$, and (12) is used for $\{S_{i1} \in S \mid i \in \mathcal{I}\}$. Equations (15) and (16) are used to sample μ and σ , respectively. Additionally, in the CGS algorithm, the initial probability π , transition probability A , and state appearance probability ϕ are calculated from the obtained state samples using (18), (19), and (20), respectively. Finally, the expected values for the obtained parameters are calculated. Algorithm 1 shows pseudocode for the algorithm. A burn-in period is required to remove the effect of initial values.

Algorithm 1 CGS for the proposed model.

```

Initialize  $S, \mu, \sigma$ .
for  $loop = 1$  to  $M$  do
  for  $i = 1$  to  $I$  do
    Sample  $S_{i1}$  from (12)
    for  $j = 2$  to  $J$  do
      Sample  $S_{ij}$  from (8)
    end for
  end for
  for  $f = 1$  to  $F$  do
    for  $s = 1$  to  $S$  do
      Sample  $\mu_{sf}$  from (15)
      Sample  $\sigma_{sf}^2$  from (16)
    end for
  end for
  if  $loop > \text{burn-in period}$  then
    Calculate  $\pi, A$ , and  $\phi$  using (18),(19),(20)
    Store  $\pi, A, \phi, \mu$  and  $\sigma$ 
  end if
end for
return Average values of  $\pi, A, \phi, \mu$  and  $\sigma$ 

```

Application and Evaluation

In this section, we evaluate the effectiveness of the proposed model using actual keystroke log data collected using the keystroke logging system introduced in

“**Keystroke Logging System and Log Data**”. In this experiment, we collected actual keystroke log data as follows.

We assigned a writing task to 72 subjects and collected keystroke log data while the subjects composed their responses. The task was a reading-to-write task in which the subjects read a short text and related material, then wrote their opinion. This task required no prior knowledge. The subjects were 37 boys/men and 35 girls/women. The range of ages was 16–23, and the median age was 19 years. Of the subjects, 34 were high school students and 38 were university students. Among the university students, 18 were studying arts/humanities and 20 were studying science of some kind. All subjects had experience using a keyboard to create documents. We provided 45 min to respond, and subjects were not allowed to finish before 45 min had elapsed. The total number of keystroke operations obtained in the experiment was 184,916. The mean and standard deviation of the number of keystroke operations by learners were 2568.28 and 852.53, respectively. The mean and standard deviation values for the final number of characters by learner were 608.92 and 168.30.

Keystroke log data were transformed to writing feature vectors using the sliding window approach, with frame width $W = 30$ s and step width $H = 10$ s. W and H are the hyperparameters, as described in “**Feature Extraction**”. W controls the granularity of subtask estimation. A small W value enables the capture of more detailed subtasks, although extremely small values of W increase the number of frames with no or only a few keystroke operations, which makes the subtask estimation unstable. A small H value increases the smoothness of the temporal change in the state appearance probabilities, although overlap among the frames is increased by using a small H . Excessive overlap is not desirable because the computational cost increases rapidly with overlap. Based on the above-mentioned factors, $W = 30$ and $H = 10$ were selected by empirical observation. As a result of this feature extraction, features X_i for learner i were obtained as the seven previously described dimensional features at 268 timepoints. In the remainder of this section, we evaluate the proposed model through application to this dataset X .

Model Selection Using Information Criteria

Writing process analysis using the proposed model depends on the number of states S and the number of time intervals T . To select the state number in GHMM, the Akaike information criterion (AIC) (Akaike 1974) and the Bayesian information criterion (BIC) (Schwarz 1978) have been widely used. AIC and BIC assume asymptotic normality of maximum likelihood estimates (Watanabe 2010, 2013). However, GHMM does not satisfy this assumption, so these information criteria are not theoretically appropriate. When MCMC is used, a log-marginal likelihood (log-ML) that does not assume asymptotic normality is approximately calculable (Newton and Raftery 1994). In recent years, various studies have used the log-ML calculated using MCMC for model selection (Uto et al. 2017; Griffiths and Steyvers 2004; Wallach et al. 2009; Taddy 2012; Uto and Ueno 2018). Therefore, in this study, we use the log-ML to select S and T . Specifically, we calculate the log-ML while changing the number of states $S = \{2, \dots, 10\}$ and the number of time intervals $T = \{1, \dots, 12\}$. Note that $S = 1$ is meaningless for writing process analysis because the same writing

process will be estimated for all learners. Thus, $S = 1$ was ignored in this experiment. Furthermore, the upper limit values of $S = 10$ and $T = 12$ were selected because extremely large values make the interpretation of the estimated writing process difficult. We discuss the appropriateness of the upper limit values later, using data for justification. For comparison, we also calculated the log-ML for each state number with GHMM.

Table 2 shows the experimental results, where a larger value for log-ML indicates increased appropriateness of the model. Table 2 shows that the proposed model tends to produce higher values than does the GHMM when the state number increases. This suggests that trends in subtask appearance differ among learners and among time intervals, and that the proposed model represents them appropriately. Here, to confirm the appropriateness of the upper limits for S and T , Figs. 5 and 6 show, respectively, the average log-ML for each $S \in \{1, \dots, 10\}$ and for each $T \in \{1, \dots, 12\}$. Figure 5 shows that the log-ML values rapidly increase until around $S = 7$, and the increase rate is slow for $S > 7$. Furthermore, the value with $S = 10$ is smaller than that with $S = 9$. Figure 6 shows that the log-ML values tend to increase until $T = 11$. When $T = 12$, however, the value is sharply reduced. These results suggest that the optimal values probably lie within $S = \{2, \dots, 10\}$ and $T = \{1, \dots, 12\}$. In these ranges, the proposed model with $S = 9$ and $T = 10$ had the highest indicator value, so we used those values for S and T in the following experiments.

Table 2 Log-marginal likelihood values for each number of states and time interval

| | Number of states S | | | | | | | | |
|------|----------------------|---------|---------|---------|---------|---------|---------|----------------|---------|
| | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| T=1 | −366651 | −344725 | −328978 | −336416 | −317773 | −309699 | −309927 | −307868 | −308437 |
| T=2 | −366722 | −352936 | −332503 | −328909 | −319634 | −310745 | −309262 | −304256 | −308507 |
| T=3 | −366260 | −352868 | −331685 | −329332 | −318167 | −311805 | −310828 | −304567 | −307783 |
| T=4 | −366236 | −352538 | −331038 | −325940 | −317770 | −308680 | −311242 | −304477 | −301883 |
| T=5 | −366318 | −357123 | −330320 | −326599 | −320426 | −309603 | −307340 | −300504 | −307858 |
| T=6 | −366019 | −352082 | −329183 | −323887 | −320131 | −309700 | −306757 | −302471 | −309461 |
| T=7 | −365937 | −352103 | −329131 | −327876 | −317000 | −309733 | −309023 | −302517 | −305505 |
| T=8 | −366089 | −352572 | −329391 | −319752 | −316254 | −307003 | −308255 | −300596 | −301741 |
| T=9 | −366373 | −357258 | −329702 | −320510 | −313955 | −307637 | −304634 | −302341 | −301564 |
| T=10 | −365377 | −351409 | −332032 | −320756 | −324515 | −309295 | −311565 | −297933 | −303207 |
| T=11 | −365554 | −351868 | −327435 | −319277 | −315560 | −309276 | −304823 | −299203 | −300418 |
| T=12 | −412429 | −351211 | −329115 | −324260 | −313688 | −305830 | −303868 | −301641 | −302244 |
| GHMM | −364967 | −347916 | −328963 | −320028 | −315718 | −311850 | −312755 | −310044 | −310187 |

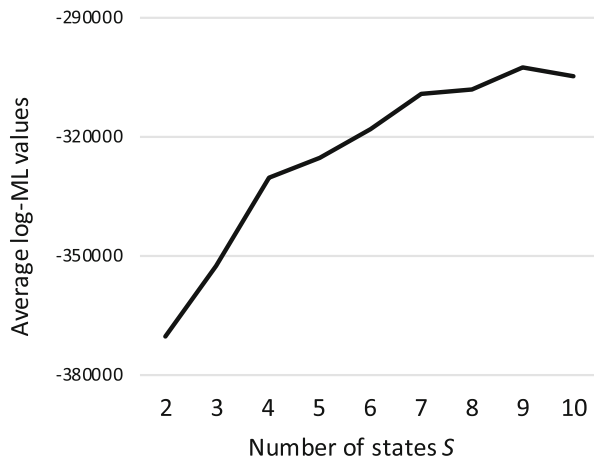


Fig. 5 Average log-ML value for each S

States Interpretation

To analyze the writing patterns of each learner based on the proposed model, we first need to interpret the characteristics of each state. For this interpretation, Table 3 presents the mean and standard deviation parameters of the emission distribution for each feature in each state. Furthermore, for ease of interpretation, Fig. 7 shows the mean feature values for each state, which are normalized so that the maximum is 1 and the minimum is 0. From these results, the characteristics of each state can be interpreted as follows:

State 1 can be interpreted as a waiting stage, because the number of stops is the highest and there are no addition or subtraction operations.

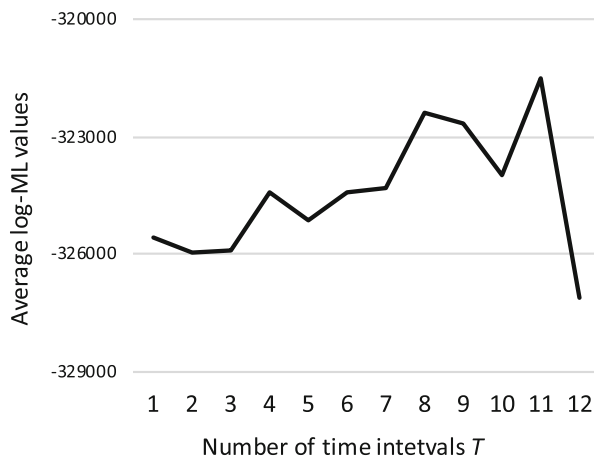
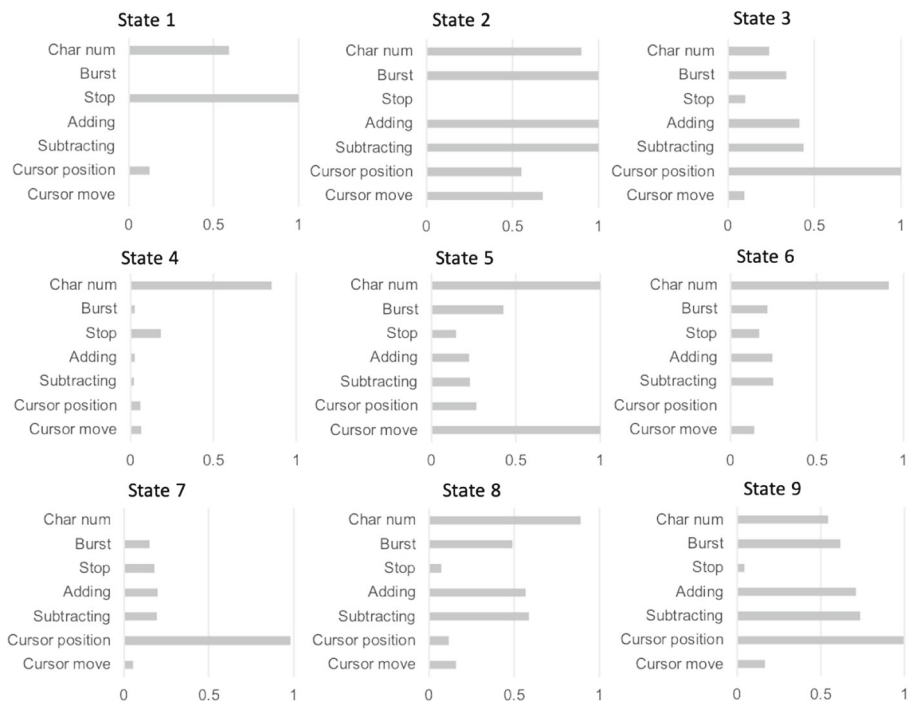


Fig. 6 Average log-ML value for each T

Table 3 Mean and standard deviation parameters of emission distributions for each state

| State | Char num | Burst | Stop | Adding | Subtracting | Cursor position | Cursor move |
|-------|----------------|---------------|------------|--------------|--------------|-----------------|--------------|
| 1 | 335.62(307.17) | 0.00(0.58) | 6.00(0.09) | 0.00(0.27) | 0.00(0.24) | 0.62(0.45) | 0.00(0.14) |
| 2 | 395.52(230.90) | 102.55(23.04) | 0.14(0.40) | 41.96(13.71) | 35.53(11.71) | 0.79(0.25) | 16.71(23.00) |
| 3 | 267.00(175.32) | 34.44(7.64) | 0.72(0.62) | 17.26(3.92) | 15.50(3.78) | 0.96(0.04) | 2.37(1.68) |
| 4 | 385.81(228.50) | 2.54(3.30) | 1.23(0.67) | 1.17(1.51) | 0.69(1.25) | 0.60(0.34) | 1.59(1.53) |
| 5 | 415.18(234.54) | 43.65(21.82) | 1.02(0.75) | 9.37(7.37) | 7.97(6.53) | 0.68(0.26) | 24.67(16.46) |
| 6 | 398.94(194.47) | 21.82(9.00) | 1.12(0.66) | 10.09(4.66) | 8.83(4.46) | 0.57(0.25) | 3.37(2.61) |
| 7 | 221.21(154.32) | 15.20(6.44) | 1.18(0.56) | 8.26(3.42) | 6.89(3.20) | 0.96(0.05) | 1.36(1.27) |
| 8 | 393.34(223.06) | 49.91(11.24) | 0.57(0.63) | 23.84(5.74) | 20.86(5.09) | 0.62(0.22) | 3.90(3.29) |
| 9 | 326.69(213.41) | 63.42(13.01) | 0.42(0.56) | 29.71(6.72) | 26.09(5.64) | 0.96(0.04) | 4.15(3.15) |

States 2, 3, 7, and 9 can be interpreted as formulation stages, because the cursor is at the end of the text, and some number of addition and subtraction operations can be seen. Here, the numbers of bursts, addition operations, and subtraction operations exhibit the following relation: $state\ 7 < state\ 3 < state\ 9 < state\ 2$. In other words, these four states can be differentiated by keystroke speed.

**Fig. 7** Normalized mean values of emission distributions for each state

States 4, 5, 6, and 8 can be interpreted as a revision stage, because the cursor positions are relatively toward the beginning of the text, the numbers of characters are relatively high, and there are a certain number of bursts, addition operations, and subtraction operations. A characteristic of state 5 is that the cursor moves often, while characteristics of state 6 are that the cursor is positioned relatively toward the beginning of the text and there are few cursor moves. Moreover, a characteristic of state 4 is that there are extremely few addition and subtraction operations. Conversely, there are many addition and subtraction operations in state 8. From these analyses, we can interpret state 4 as a revision state involving few edits, state 5 as a revision state involving overall edits, state 6 as a revision state involving edits of specific parts in the beginning and middle parts of the text, and state 8 as a revision state with many edits.

Table 4 summarizes the characteristics based on the above analyses.

It is worth noting that we might be able to evaluate the appropriateness of our state interpretation by comparing the interpretations with the subjects' intentions. Subjects' intentions can be investigated via traditional writing process analysis methods, such as the *video playback stimulation method*. For this analysis, however, we must create the subtask labels summarized in Table 4 in advance by estimating the proposed model parameters using data from all subjects. Due to our experimental constraints, we could not gather the same subjects after all the data had been collected. The evaluation of appropriateness by comparison with subjects' intentions remains for future work.

Interpretation of State Appearance Distribution

This section discusses interpretation of the state appearance distribution for each learner based on the above interpretation of states. To that end, Figs. 8 and 9 show the state appearance distribution ϕ_i for two learners. The horizontal axes in these figures show the time interval, while vertical axes show the appearance probability of each state. Line types show individual states. The figures lead to the following interpretations of the writing process for each learner.

Writer 1 (Fig. 8) has a high ratio of waiting states in the first time interval, which we interpret as the learner reading the task and planning. As time progresses, the

Table 4 Interpretation of each state

| Major division | Subdivision | States |
|----------------|------------------|----------------|
| Waiting | — | State 1 |
| Formulation | Fast writing | States 2 and 9 |
| | Slow writing | States 3 and 7 |
| Revision | Many edits | State 8 |
| | Few edits | State 4 |
| | Overall edits | State 5 |
| | Individual edits | State 6 |

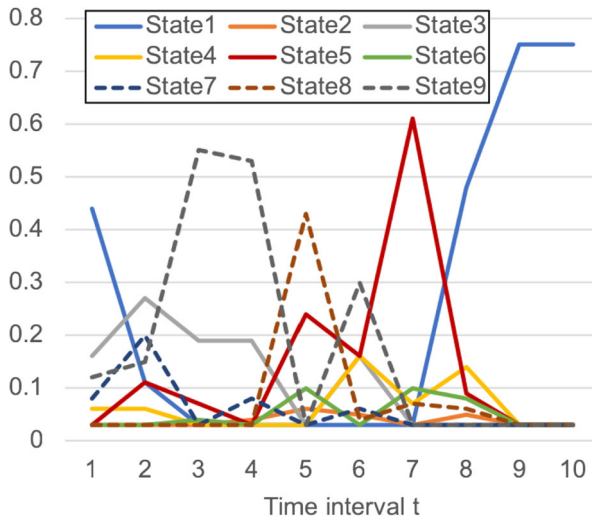


Fig. 8 State appearance distribution of learner 1

appearance ratio increases in the order of state 9 (formulation stage with fast writing), state 8 (revision stage with many edits), and state 5 (revision of overall text). The learner returns to the waiting state in later time intervals, suggesting that this learner has an ideal writing process.

In contrast, learner 2 (Fig. 9) shows high appearance ratios for states 3 and 7, representing slow writing formulation stages across all time intervals. The appearance

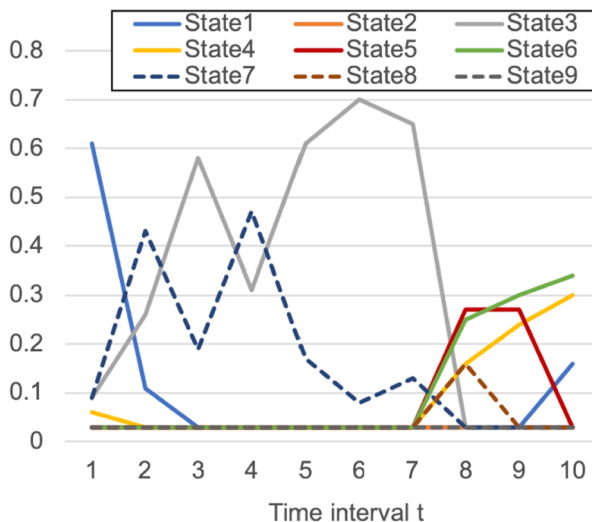


Fig. 9 State appearance distribution of learner 2

ratio of revision states is low. This learner might not have been spending enough time on revisions.

The above analysis shows that the proposed model allows quantitative analysis of temporal changes in subtask appearance patterns for each learner.

Validity Evaluation of State Appearance Distributions

This subsection evaluates the validity of subtask appearance distribution estimations by the proposed method.

For this experiment, we randomly selected ten learners from among the subjects. Then we showed a replay of their keystrokes to two experts (evaluators A and B, below). We asked the evaluators to assess the appearance ratio of nine states for each time interval for each learner. However, it might be difficult for humans to directly differentiate between the nine states. Therefore, we first asked the evaluators to assess ratios of *waiting*, *formulation*, and *revision*, which are presented as major divisions for the nine states for individual time intervals of each learner using five categories: *5. appears very often*, *4. appears often*, *3. appears relatively often*, *2. appears somewhat*, and *1. does not appear*. If *revision* was scored 2 or higher in this assessment, the range and amount of revision were evaluated as *2. wide* or *1. narrow*, and *2. large amount* or *1. small amount*. Furthermore, keystroke speed was scored as *2. fast* or *1. slow* for all learners.

In this experiment, we calculated subtask distributions for each learner from these evaluation data (hereinafter called the *correct distribution*). To create the correct distribution, we calculated scores for each subtask subdivision in Table 4, based on each evaluator's assessment data. Concretely, for each time interval for each learner, the scores for the seven subtask subdivisions were calculated as follows:

Waiting : Use the evaluation score for *waiting*.

Formulation (Fast writing): If *input speed* is *2. fast*, use the evaluation score for *formulation*. If not, use half of the score.

Formulation (Slow writing): If *input speed* is *1. slow*, use the evaluation score for *formulation*. If not, use half of the score.

Revision (Many edits): If *the number of edits* is *2. large amount*, use the evaluation score for *revision*. If not, use half of the score.

Revision (Few edits): If *the number of edits* is *1. small amount*, use the evaluation score for *revision*. If not, use half of the score.

Revision (Overall edits): When *edit range* is *2. wide*, use the evaluation score for *revision*. If not, use half of the score.

Revision (Individual edits): When *edit range* is *1. narrow*, use the evaluation score for *revision*. If not, use half of the score.

The correct distribution was created by normalizing those scores for each evaluator. Note that this experiment did not distinguish between states 2 and 9 or between states 3 and 7, because those pairs are difficult for humans to differentiate.

We evaluated the validity of the proposed model by comparing the correct distribution with state appearance distributions from the proposed model. To evaluate these differences, we used the Jensen–Shannon (JS) divergence, which is widely used to

evaluate differences in probability distributions. The JS divergence is zero when the distributions are completely consistent and increases with increasing differences. To discuss the degree of differences between correct distributions and state appearance distributions from the proposed model, we also calculated JS divergence between the uniform distribution and each distribution. Here, because the correct distribution combined states 2 and 9 and states 3 and 7 as described above, the JS divergence was calculated after the state appearance probabilities for those state pairs were summed.

Table 5 shows the mean and standard deviation for the JS divergence calculated between each distribution. These results demonstrate that the difference between the state appearance distribution from the proposed model and each correct distribution is smaller than differences between the uniform distribution and each correct distribution. We performed paired multiple comparison using the Bonferroni method to evaluate whether significant differences are confirmed for those JS divergence means. Table 6 shows the results. In that table, *A* (or *B*) indicates evaluator *A* (or *B*), *X/Y* refers to the JS divergence between distributions of methods *X* and *Y*, and values in each cell are the *p*-value for the mean difference of the JS divergences. For example, the cell in the first row and first column shows the *p*-value of the mean difference between the JS divergences of *A/B* and those of *A/Proposed*. The results show that JS divergences between state appearance distributions from the proposed model and the correct distributions as calculated by both evaluators present no significant differences, while those between uniform and correct distributions reveal significant differences.

These results indicate that state appearance distributions from the proposed model have trends similar to the interpretations by the expert evaluators. This suggests that using the proposed model to analyze subtask appearance patterns for each learner is appropriate.

Analysis of the Relation Between Skills and the Writing Process

As discussed in “[Introduction](#)”, the writing process and writing skills are known to be related. Therefore, if analyses of this relation based on the proposed model are consistent with findings from existing studies, the validity of analyses using the proposed model can be confirmed.

For this evaluation, we classified learners with similar processes and analyzed relations between these clusters and writing skills. Specifically, we performed hierarchical clustering using the JS divergence of the state appearance distribution between learners as the distance function. We used the pseudo-F criterion to determine the

Table 5 JS divergence between methods

| | Evaluator A | Evaluator B | Uniform |
|-------------|---------------|---------------|---------------|
| Proposed | 0.695 (0.297) | 0.796 (0.335) | 1.361 (0.319) |
| Evaluator A | – | 0.379 (0.207) | 1.615 (0.409) |
| Evaluator B | – | – | 1.542 (0.435) |

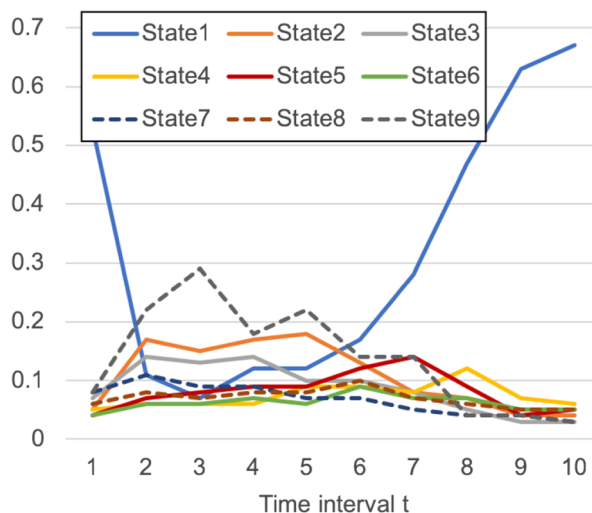
Table 6 Results of statistical tests

| | A/Proposed | B/Proposed | A/Uniform | B/Uniform | Proposed/Uniform |
|------------|------------|------------|-----------|-----------|------------------|
| A/B | 0.829 | 0.187 | < 0.001 | < 0.001 | < 0.001 |
| A/Proposed | – | 1.000 | < 0.001 | < 0.001 | < 0.010 |
| B/Proposed | – | – | < 0.001 | < 0.001 | < 0.050 |
| A/Uniform | – | – | – | 1.000 | 1.000 |
| B/Uniform | – | – | – | – | 1.000 |

optimum cluster number, and two clusters were supported. Therefore, in this experiment we classified learners into two clusters, with 26 learners in one group and 46 learners in another. Figures 10 and 11 show mean values of state appearance probabilities for learners belonging to each cluster. Those figures show the following characteristics for each cluster.

Writers in cluster 1 wait for a certain amount of time, followed by a fast-writing formulation stage (states 9 and 2) and then, in the latter half, transition to the waiting state with a certain ratio of overall revision (state 5) and minor edits (state 4). This can be considered as a good writing process, because there is good balance between planning, formulation, and revision, and writing is completed with time to spare.

Writers in cluster 2 are relatively slow to start writing, and the start of writing is followed by slow-writing formulation (states 7 and 3) for a long time, with minor edits (state 4) and edits at specific locations (state 6) conducted just before the end of the writing period. This can be seen as a cluster of learners who formulate slowly and cannot secure sufficient time for revisions.

**Fig. 10** Average state appearance probabilities of cluster 1

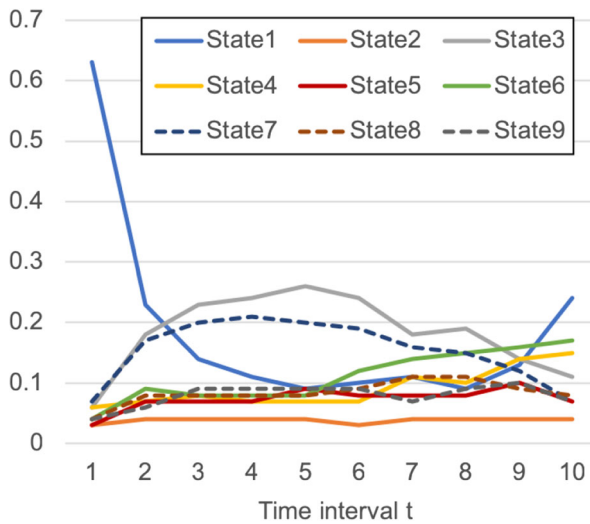


Fig. 11 Average state appearance probabilities of cluster 2

Learners who write fast and spend more time performing revisions are generally known to have high writing skills (Sasaki 2000, 2002; Larios et al. 2008). Therefore, if the product quality in cluster 1 is high, analyses based on the proposed model are validated.

To evaluate the quality of writing products, we had two experts score the writing of each learner by two perspectives: 1) *organization*, and 2) *readability*. Organization was evaluated using five scores: 1. *extremely poor*, 2. *poor*, 3. *neither poor nor skilled*, 4. *skilled*, and 5. *extremely skilled*. Readability was evaluated using four categories: 1. *very difficult to read and understand*, 2. *difficult to read and understand*, 3. *somewhat difficult to read and understand*, and 4. *no problem with readability*. Evaluators were not informed of which cluster learners belonged to. The average of the scores by the two experts was used as the final score. We conducted the Wilcoxon rank-sum test to examine differences in mean score between clusters for each evaluation point. We also performed the same calculation for the total score of the two evaluation points.

Table 7 shows the results. The experimental result shows that the score for cluster 1 is higher than that for cluster 2 for each evaluation point. In addition, the readability and total scores are significantly different. Here, we also confirmed the

Table 7 Scores for each cluster

| | Organization | Readability | Total score |
|-----------------|--------------|--------------|--------------|
| Cluster 1 | 2.673 (.114) | 3.615 (.198) | 6.288 (.484) |
| Cluster 2 | 2.467 (.146) | 3.065 (.431) | 5.533 (.917) |
| <i>p</i> -value | .027 | <.001 | <.001 |

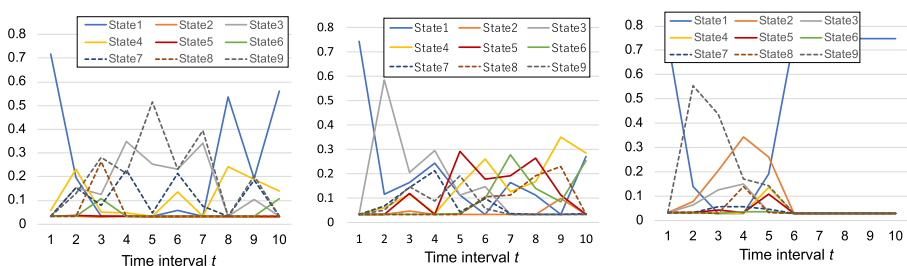
Table 8 Scores for each attribute of subjects

| Gender | | Background | |
|---------|-------------|-----------------|-------------|
| Men | 5.71 (1.04) | Arts/Humanities | 6.02 (0.86) |
| Women | 5.92 (0.84) | Scientific | 5.85 (1.01) |
| p-value | 0.35 | | 0.56 |

relations between the attributes of the subjects and their scores. Concretely, Table 8 shows the averaged total scores (standard deviations) and p -values of the Wilcoxon rank-sum test for different genders and different backgrounds. These tests found no significant difference for gender or background. In addition, the correlation between age and total score was 0.17, with no significance ($p = 0.15$). These results show that the effects of subjects' attributes did not significantly affect the outcome of this experiment.

From the experimental results, we can confirm that the product quality in cluster 1 was higher than that in cluster 2, as expected. This suggests that writing process analyses based on the proposed model derive findings consistent with those of previous studies (e.g., Sasaki 2000, 2002; Larios et al. 2008), suggesting that such analyses are appropriate.

Finally, to show some examples of the writing process of subjects with advanced skills and those with lesser skills, Figs. 12 and 13 depict the state appearance probabilities of the subjects with the top 3 and bottom 3 scores, respectively. Figure 12 shows that high-performing subjects have high ratios of formulation stages (State 2, 3, 7, 9) in the first half of the total writing time. In the second half, the ratios of waiting (State 1) and revision actions (States 4, 5, 6, 8) increase. Although the time allocated to each stage differs among the subjects, they tend to divide the formulation and revision phases consciously. In contrast, Fig. 13 shows that low-performing subjects have a low ratio of waiting (State1) across all time intervals, meaning that they continued to write until just before the end of the writing period. Concretely, the subject shown in the center of Fig. 13 has high ratios of formulation stages (State 3, 7, 9) overall, and the subjects shown in the left and right of Fig. 13 have high ratios of editing/revision actions (State 4, 5, 6, 8) across all time intervals. These results suggest that the common characteristics of good writers are that (1) the formulation

**Fig. 12** State appearance probabilities of subjects with top 3 scores

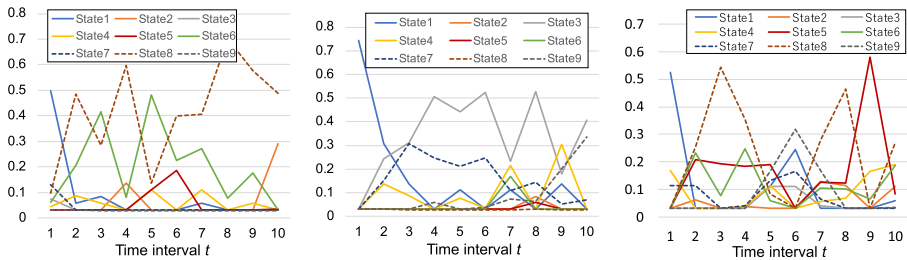


Fig. 13 State appearance probabilities of subjects with bottom 3 scores

and revision phases are divided, and (2) time management is practiced. In addition, the figures show that the temporal changes in subtasks differ between the subjects, although good (and bad) writers share roughly similar trends. The interpretation of the detailed temporal changes would help learners and instructors grasp the writing characteristics of each learner quantitatively, as we discussed in “[Introduction](#)”. Furthermore, such data also provide useful information for instructors to use in providing feedback and instruction based on the writing patterns of each learner.

Conclusion

In this study, we proposed a method for machine learning from keystroke log data to estimate temporal change in learners’ subtask appearance patterns. Specifically, we defined keystroke log data as series data of writing features, and developed an expanded GHMM model that estimates the subtask appearance distribution from these data. The proposed model considered latent states in GHMM as subtasks, and incorporated parameters that express state appearance probabilities for each time interval for each learner. Furthermore, we proposed a Bayesian estimation method via collapsed Gibbs sampling as a method for estimating parameters for the proposed model.

We used actual data to show that the proposed model produces higher fit than does GHMM. Furthermore, we demonstrated that the writing process of each learner could be quantitatively interpreted based on the state appearance distribution from the proposed model, and that the distribution was valid and similar to interpretations by expert evaluators. We also showed that differences in learner writing processes can be measured based on JS divergence in state appearance distributions, and that clustering of writing processes can be conducted using that measure. We also demonstrated the validity of writing process analysis based on the clustering.

In the future, we would like to examine generalizability of the proposed method through applications to various actual data. We will also examine an extension of the proposed method to deal with reports that writing activity depends on emotion (Epp et al. 2011; Salmeron-Majadas et al. 2018). In the proposed method, the state appearance probabilities for each learner ϕ_i will reflect the effects of emotion. The effects of emotion and writing characteristics cannot be differentiated, but the effects might be explicitly captured by incorporating emotion parameters. This extension will be

explored in future work. We also aim to develop a writing learning support system that visualizes estimated results as feedback to learners.

Acknowledgements This work was supported by JSPS KAKENHI Grant Numbers 17H04726 and 17K20024. Data collection was performed with the support of the Assessment Research and Development Office, Benesse Educational Research and Development Institute.

References

- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19, 716–723.
- Ali, M.L., Thakur, K., Tappert, C.C., Qiu, M. (2016). Keystroke biometric user verification using hidden Markov model. In *IEEE 3rd international conference on cyber security and cloud computing* (pp. 204–209).
- Allen, J.F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11), 832–843.
- Barkaoui, K. (2016). What and when second-language learners revise when responding to timed writing tasks on the computer: The roles of task type, second language proficiency, and keyboarding skills. *The Modern Language Journal*, 100(1), 320–340.
- Bayat, N. (2014). The effect of the process writing approach on writing success and anxiety. *Educational Sciences: Theory & Practice*, 14(3), 1133–1141.
- Bishop, C.M. (2006). Pattern recognition and machine learning (information science and statistics). Springer.
- Brooks, S., Gelman, A., Jones, G., Meng, X. (2011). Handbook of Markov chain Monte Carlo. CRC Press.
- Chan, S. (2017). Using keystroke logging to understand writers processes on a reading-into-writing test. *Language Testing in Asia*, 7(1), 1–27.
- Chen, W., & Chang, W. (2004). Applying hidden Markov models to keystroke pattern analysis for password verification. In: *Proceedings of IEEE International Conference on Information Reuse and Integration* (pp. 467–474).
- Conijn, R., van der Loo, J., van Zaanen, M. (2018). What's (not) in a keystroke? automatic discovery of students writing processes using keystroke logging. In: *Proceedings of the 8th International Conference on Learning Analytics & Knowledge* (pp. 1–6).
- Deane, P., & Zhang, M. (2015). Exploring the feasibility of using writing process features to assess text production skills (Rapport technique). ETS Research Report.
- Epp, C., Lippold, M., Mandryk, R.L. (2011). Identifying emotional states using keystroke dynamics. In *Proceedings of the Sigchi Conference on Human Factors in Computing Systems* (pp. 7150–724).
- Flower, S., & Hayes, R. (1981). A cognitive process theory of writing. *College Composition and Communication*, 32, 365–387.
- Fox, J.-P. (2010). Bayesian item response modeling: Theory and applications. Springer.
- Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis*, 1(3), 515–534.
- Griffiths, T.L., & Steyvers, M. (2004). Finding scientific topics. In *Proc. National Academy of Sciences of the United States of America* (pp. 5228–5235).
- Griffiths, T.L., Steyvers, M., Blei, D.M., Tenenbaum, J.B. (2004). Integrating topics and syntax. In: *Proceedings of the 17th International Conference on Neural Information Processing Systems* (pp. 537–544).
- Hayes, J., & Flower, L. (1980). Identifying the organization of writing processes. In *Cognitive Processes in Writing* (pp. 1–28). Erlbaum.
- Karnan, M., Akila, M., Krishnaraj, N. (2011). Biometric personal authentication using keystroke dynamics: a review. *Applied Soft Computing*, 11(2), 1565–1573.
- de Larios, J.R., Manchón, R., Murphy, L., Marín, J. (2008). The foreign language writer's strategic behaviour in the allocation of time to writing processes. *Journal of Second Language Writing*, 17(1), 30–47.
- Leijten, M., & Waes, L.V. (2013). Keystroke logging in writing research. *Written Communication*, 30(3), 358–392.

- Lester, F., & Witte, S. (1981). Analyzing revision. *College Composition and Communication*, 32(4), 400–414.
- Liu, L., Cheng, L., Liu, Y., Jia, Y., Rosenblum, D. (2016). Recognizing complex activities by a probabilistic interval-based model.
- Liu, L., Wang, S., Hu, B., Qiong, Q., Wen, J., Rosenblum, D.S. (2018). Learning structures of interval-based Bayesian networks in probabilistic generative model for human complex activity recognition. *Pattern Recognition*, 81, 545–561.
- Newton, M., & Raftery, A. (1994). Approximate Bayesian inference by the weighted likelihood bootstrap. *Journal of the Royal Statistical Society. Series B: Methodological*, 56(1), 3–48.
- Paisley, J., & Carin, L. (2009). Hidden Markov models with stick-breaking priors. *IEEE Transactions on Signal Processing*, 57(10), 3905–3917.
- Quraishi, S.J., & Bedi, S. (2018). Keystroke dynamics biometrics, a tool for user authentication-review keystroke dynamics biometrics, a tool for user authentication-review. In *Proceedings of International Conference on System Modeling & Advancement in Research Trends* (pp. 248–254).
- Rodrigues, R.N., Yared, G.F.G., Costa, N., do, C.R., Yabu-Uti, J.B.T., Violaro, F., Ling, L.L. (2005). Biometric access control through numerical keyboards based on keystroke dynamics. In *Advances in Biometrics* (pp. 640–646). Springer: Berlin.
- Salmeron-Majadas, S., Baker, R.S., Santos, O.C., Boticario, J.G. (2018). A machine learning approach to leverage individual keyboard and mouse interaction behavior from multiple users in real-world learning scenarios. *IEEE Access*, 6, 39154–39179.
- Sasaki, M. (2000). Toward an empirical model of efl writing processes: an exploratory study. *Journal of Second Language Writing*, 9(3), 259–291.
- Sasaki, M. (2002). Building an empirically-based model of efl learners' writing processes. In *New Directions for Research in L2 Writing* (pp. 49–80). Dordrecht: Springer Netherlands.
- Schwarz, G. (1978). Estimating the dimensions of a model. *Annals of Statistics*, 6, 461–464.
- Seow, A. (2002). The writing process and process writing. In *Methodology in Language Teaching: An Anthology of Current Practice* (pp. 315–320). Cambridge University Press.
- Southavilay, V., Yacef, K., Calvo, R.A. (2010). Analysis of collaborative writing processes using hidden Markov models and semantic heuristics. In *IEEE International Conference on Data Mining Workshops* (pp. 543–548).
- Southavilay, V., Yacef, K., Calvo, R.A. (2010). Process mining to support students collaborative writing. In *Proceedings of International Conference on Educational Data Mining* (pp. 257–266).
- Stevenson, M., Schoonen, R., de Glopper, K. (2006). Revising in two languages: a multi-dimensional comparison of online writing revisions in L1 and FL. *Journal of Second Language Writing*, 15(3), 201–233.
- Taddy, M. (2012). On estimation and selection for topic models. In *Proc. international conference on artificial intelligence and statistics* (pp. 1184–1193).
- Tanizaki, H. (2008). A simple Gamma random number generator for arbitrary shape parameters. *Economics Bulletin*, 3(7), 1–10.
- Teh, P.S., Teoh, A.B.J., Yue, S. (2013). A survey of keystroke dynamics biometrics. *The Scientific World Journal*, 2013, 1–24.
- Uto, M., Louvigné, S., Kato, Y., Ishii, T., Miyazawa, Y. (2017). Diverse reports recommendation system based on latent dirichlet allocation. *Behaviormetrika*, 44(2), 425–444.
- Uto, M., & Ueno, M. (2015). Academic writing support system using bayesian networks. In *Proc. IEEE international conference on advanced learning technologies* (pp. 385–387).
- Uto, M., & Ueno, M. (2016). Item response theory for peer assessment. *IEEE Transactions on Learning Technologies*, 9(2), 157–170.
- Uto, M., & Ueno, M. (2018). Empirical comparison of item response theory models with rater's parameters. *Heliyon, Elsevier*, 4(5), 1–32.
- Wallach, H.M., Murray, I., Salakhutdinov, R., Mimno, D. (2009). Evaluation methods for topic models. In *Proc. international conference on machine learning* (pp. 1105–1112).
- Wang, Z., Wang, S., Ji, Q. (2013). Capturing complex spatio-temporal relations among facial muscles for facial expression recognition. In *IEEE conference on computer vision and pattern recognition* (pp. 3422–3429).
- Watanabe, S. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, pp. 3571–3594.

- Watanabe, S. (2013). A widely applicable Bayesian information criterion. *Journal of Machine Learning Research*, 14(1), 867–897.
- Zhang, M., Hao, J., Li, C., Deane, P. (2016). Classification of writing patterns using keystroke logs. In *Quantitative psychology research: The 80th annual meeting of the psychometric society* (pp. 299–314).
- Zhang, Y., Zhang, Y., Swears, E., Larios, N., Wang, Z., Ji, Q. (2013). Modeling temporal interactions with interval temporal Bayesian networks for complex activity recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(10), 2468–2483.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.