Latent Variable Mining with its Applications to Anomalous Behavior Detection

Shunsuke Hirose* and Kenji Yamanishi
Common Platform Software Research Laboratories, NEC , Kawasaki , Japan

Received 01 May 2008; revised 17 November 2008; accepted 18 February 2009
DOI:10.1002/sam.10032
Published online 26 May 2009 in Wiley InterScience (www.interscience.wiley.com).

Abstract: In this paper, we propose a new approach to anomaly detection by looking at the latent variable space to make the first step toward *latent anomaly detection*. Most conventional approaches to anomaly detection are concerned with tracking data which are largely deviated from the ordinary pattern. In this paper, we are instead concerned with the issue of how to track changes occurring in the latent variable space consisting of the meta information existing behind directly observed data. For example, in the case of masquerade detection, the conventional task was to detect anomalous command lines related to masqueraders' malicious behaviors. Meanwhile, we rather attempt to track changes of behavioral patterns such as *writing mails*, *making software*, etc. which are information of more abstract level than command lines. The key ideas of the proposed methods are: (i) constructing the model variation vector, which is introduced relative to the latent variable space, and (ii) the latent anomaly detection is reduced to the issue of change-point detection for the time series that the model variation vector forms. We demonstrate through the experimental results using an artificial data set and a UNIX command data set that our method has significantly enhanced the accuracy of existing anomaly detection methods. © 2009 Wiley Periodicals, Inc. Statistical Analysis and Data Mining 2: 70–86, 2009

Keywords: latent anomalies; anomalous behavior detection; masquerade detection; hidden Markov model

1. INTRODUCTION

1.1. Motivation of Mining Latent Anomalies

We are concerned with the issue of detecting anomalies from time series as accurately as possible. Typical examples include the issue of masquerade detection, the goal of which is to detect masqueraders' anomalous behaviors as early as possible with the lowest false alarm rates. A number of studies on anomaly detection have been investigated in the scenario of security, failure detection, fraud detection, and so on. Most of them adopt the approach of detecting *observed anomalies*, which are observed data significantly deviated from the ordinary regularity. Meanwhile, we propose a new approach to anomaly detection on the basis of tracking *latent anomalies*. Here, the latent anomalies are the incidents recognized as anomalies not necessarily in an observed data space but in a latent variable space specifying the data generation mechanism.

Let us illustrate the issue of latent anomaly detection through an example of masquerade detection from UNIX command sequences. The observed data space consists of UNIX commands themselves such as `ls`, `lpr`, `ftp`, and so on. Meanwhile the latent variable space consists of meta information behind command lines, such as *writing text*, *writing a code*, etc., which are not explicitly recognized from observed data. When we observe UNIX commands, there are two latent variables, namely two kinds of meta-information. The first one is which behavioral pattern (user's action e.g. *writing text* or *writing a code*) the commands are generated from. The second one is which hidden state of generating commands the user stays. Hidden states represent user's internal states which involve *way of typing commands*. In the same action, which commands are used and the order of them are dependent on a user. We represent these two, how to conduct an action, as way of typing commands. For example, when writing text, User 1 use `vi` generated by *hidden state 1* and User 2 use `emacs` generated by *hidden state 2*. Taking notices of these two latent variables, we define latent anomalies as those which are induced by structural changes of the variables, namely, sudden changes of (i) behavioral patterns, and (ii) hidden states. They are *latent* because patterns or states are not directly observed. Figure 1 shows an example of latent variables and latent anomalies. We conventionally look at the observed data space only to detect anomalous command lines themselves or irregular command patterns. In contrast, in the new scenario of latent anomaly detection, we rather track anomalous events that occur in the latent variable

Correspondence to: Shunsuke Hirose
(s-hirose@ap.jp.nec.com)

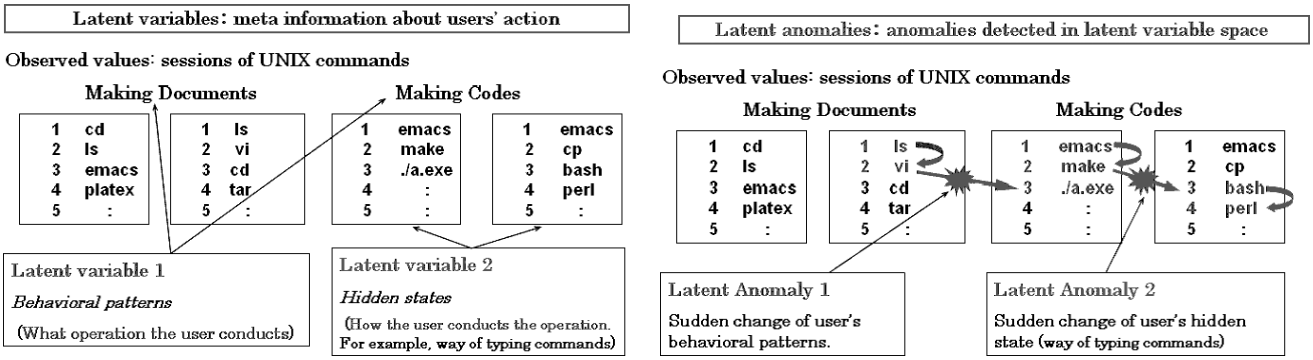

Fig. 1 An example of latent variables (left panel) and latent anomalies (right panel).

spaces, e.g. the periodic pattern consisting of *writing text* and *writing a code* terminates and new behavior emerges.

Why is latent anomaly detection to be explored? The most important reason is that we can possibly enhance the accuracy of detecting anomalous incidents by looking at not only the observed data space but also the latent variable space. In the scenario of masquerade detection from UNIX command sequence, for example, a masquerader may often conduct malicious behaviors, which do not necessarily induce any change of occurrence frequency of commands but induce drastic change of usage patterns that are recognized as latent information behind the commands. This implies that there exist new types of anomalies that may not be detected by looking at the observed data space only. Hence, one could detect anomalous events more accurately if one could track such latent anomalies in addition to observed ones.

The purpose of this paper is twofold. The first one is to develop a method for detecting latent anomalies from time series. The method consists of a probabilistic modeling with latent variables and a design of a most fundamental algorithm for detecting latent anomalies. As we discuss in Section 4.5 and Appendix C, our proposed algorithm is extendable to more general one. Thus, we call the algorithm *most fundamental algorithm*.

The second one is to demonstrate the effectiveness of the methods through the experiments using an artificial data set including changes of behavioral patterns and a UNIX command data set including masquerade sequence [1]. We empirically show that the latent anomaly detection significantly improves conventional methods in which the latent anomalies are not taken into account.

The key ideas of techniques for latent anomaly detection are summarized as follows: First, we define the *symmetrized model variation* by quantifying how the probabilistic model with latent variables changes. Then, we show that the symmetrized model variation is decomposed into a number of subscores, each of which corresponds to the one derived from a latent variable of behavioral patterns and that of hidden states, or an observed data. Then, we define a *model variation vector* as a tuple of these subscores. We reduce the issue of latent anomaly detection into that of *change-point detection over a stream of model variation vectors.*

Throughout the paper, we illustrate the methodology as above using the hidden Markov model (HMM) mixture model, but we can easily extend it to a more general model with hidden variables.

1.2. Latent Anomaly Detection and Conventional Methods

In this section, let us explain latent anomaly detection by comparing it with conventional anomaly detection method.

We assume that a set of latent states generates the observation sequences. Namely, there exist latent variables which control the dynamics of the sequences. This assumption does not always hold true. However, the assumption is natural in the case of anomalous behavior detection because at least there are two kinds of latent variables discussed above.

We illustrate latent anomalies in Fig. 2. In the figure, symbols (circles, squares, stars and triangles) denote sessions of observed commands, z-axis denotes probability density of sessions, horizontal plane denotes latent variable space, and two clusters denote two behavioral patterns. We aim to detect latent anomalies defined as sudden changes in latent variable space. Under the above-mentioned assumption, it is expected that, in normal states, sessions gradually move in latent variable space like circles in the figure. It is also expected that sessions move suddenly in the space when anomalous behavior starts. This is because latent variables, behavioral patterns and hidden states (how to conduct an operation), do not change drastically in normal states and they drastically change when anomalous behavior starts. Thus, we can detect anomalous behaviors by tracking latent anomalies.

Latent anomaly detection tracks sudden jumps in latent variable space such as the arrows a, a', b, and b' in

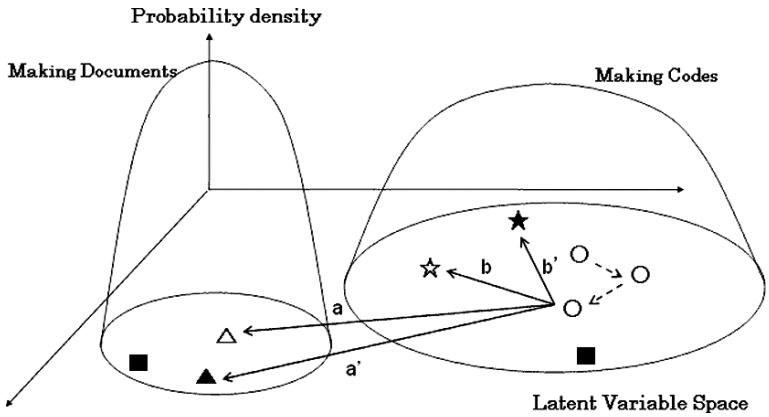

Fig. 2 Illustration of latent anomalies.

Fig. 2. There are two kinds of jumps in latent variable space. One is latent anomaly (1) such as a and a'. The other is latent anomaly (2) such as b and b'. On the other hand, conventional methods detect irregular sessions, whose occurrence probabilities are small, such as the black symbols in the figure. Thus, by taking into account latent variable space, we can detect anomalies undetectable by conventional methods, such as the arrows a and b (jumps to high probability session).

What is important is that anomalies detected by latent anomaly detection and conventional methods are sometimes of different types for the following reasons. First, there exist anomalies undetectable by conventional methods such as the arrows a and b. Second, there exist anomalies undetectable by latent anomaly detection such as low probability session reached by gradual motion. Of course, they are not always different because they have overlaps such as the arrows a' and b'.

In Section 5.2, we conduct anomaly detection from artificial data including jumps from a high probability session of a behavioral pattern to a high probability session of another pattern as anomalies (this anomaly corresponds to the arrow a in the figure). By the experiment, we show that latent anomaly detection can track anomalies, which are undetectable by conventional methods.

In the real-world dataset, of course, anomalies are not always undetectable by conventional methods. However, if the occurrence frequency of such anomalies is not negligible, detection accuracy is enhanced by employing latent anomaly detection. In Section 5.3, we conduct masquerade detection from UNIX command sequence and show that detection accuracy is enhanced by employing latent anomaly detection. This implies that some real-world anomalies are classified to such anomalies.

1.3. Related Work

Most typical approaches to anomaly detection include outlier detection [2-5], change-point detection [6-8], and anomalous behavior detection [9,10]. The task of outlier detection is to extract data largely deviated relative to the ordinary data pattern. Relevant to the problem being addressed in this paper, system call intrusion detection methods based on outlier detection have been proposed [2,3]. The task of change-point detection is to track the time point when a sudden change occurs in a time series. The task of anomalous behavior detection is to track from a time series an abnormal subsequence which is largely deviated relative to the ordinary behavior pattern. Many other methods such as episode rule-based ones [11], clustering-based ones [12], sequential-pattern mining [13], Markov monitoring [14] have also been studied. In all of them, however, anomalies to be detected were considered on the basis of the observed data space only, without looking at the latent variable space.

The technique of dynamic model selection (for short, DMS [9,10]) has been proposed in the scenario of estimating probabilistic model sequences under the assumption that the model may change over time. In the case of finite mixture models, for example, the model denotes the mixture size, i.e. the number of components in the mixture. DMS has been applied to anomaly detection, in which anomalies are tracked by detecting the change of the mixture size. In this sense, DMS is also closely related to latent anomaly detection, but cannot be applied to the case where one has to detect any change in the latent variable spaceeven when models are kept the same over time.

Probabilistic models with latent variables including finite mixture models, HMMs, independent component analysis model, etc. have extensively been studied. However, the anomalies in the latent variable space have been

paid scat attentions. For example, though several HMM based anomaly detection techniques for system call intrusion detection have been proposed [15,16] and they have performed very well, they were not designed for detecting latent anomalies but for outlying observations. Very recently techniques of change detection in the latent variable space have been applied to the context of bursty topic mining from a text stream [17].

The rest of this paper is organized as follows: In Section 2, we describe our problem settings. Section 3 yields a description of probabilistic modeling we employ. Section 4 yields detailed descriptions of our proposed methodology of latent anomaly detection. Section 5 shows experimental results obtainedusing an artificial data and the UNIX command sequence [1] for masquerade detection [1,18,19]. Section 6 gives concluding remarks.

2. PROBLEM SETTINGS

Let us first describe our problem settings.

Let $\mathcal{Y}$ be a finite set of discrete symbols. We consider time t to be discrete. At each time step t, we observe a session which consists of discrete symbols belonging to $\mathcal{Y}$, which we denote as $\mathbf{y}_t = (y_{t,1}, \cdots, y_{t,T_t}) \in \mathcal{Y}^{T_t}$, where T_t is the length of the session. For example, when the input data is a set of UNIX commands such as ($command1 =$ `ls`, $command2 =$ `cat`, $command3 =$ `netscape`, $command4 =$ `netscape`, $\cdots$), we divide them into *sessions* e.g. $session1 = \mathbf{y}_1 =$ (`ls`, `cat`), $session2 = \mathbf{y}_2 =$ (`netscape`, `netscape`), $\cdots$. For example, we can construct a session as a command sequence of fixed length.

Our problem is to detect latent anomalies from the abovementioned observations. As mentioned in Section 1.1, latent anomalies are defined as those which are induced by structural changes of two latent variables, namely, sudden changes of (i) behavioral patterns and (ii) hidden states.

As a solution for the problem, we develop an anomaly detection method consisting of a probabilistic modeling with latent variables and a most fundamental algorithm for detecting latent anomalies. In Section 3, we employ a probabilistic modeling with latent variables in order to take into account information of latent variables. In Section 4, we propose a fundamental algorithm for detecting latent anomalies.

3. PROBABILISTIC MODELING

3.1. Probabilistic Models

For the modeling of data generation of sessions mentioned in Section 2, we consider a probabilistic model in which a finite number of sequential behavioral patterns are mixed and state transitions exist in each pattern. As a concrete probabilistic model of such kinds, we employ the mixture of HMMs, without loss of generality. In it, the behavioral pattern and state transitions may both be specified by latent variables.

Let K be a positive integer. We employ HMM mixtures having K HMMs for representing a probability distribution $p_t(\mathbf{y})$ of a session $\mathbf{y}$ at time t:

$$p_t(\mathbf{y}) = \sum_{i=1}^{K} \pi_{i,t}\, p_{i,t}(\mathbf{y}), \tag{1}$$

where for each $i \in \{1, .., K\}$, $\pi_{i,t}$ denotes the mixture coefficient such that $\sum_i \pi_{i,t} = 1$ and $\pi_{i,t} > 0$, and $p_{i,t}(\mathbf{y})$ denotes the ith component of the mixture represented as an HMM. We call each component of the mixture a *cluster*. Then, a behavioral pattern is represented by the probability distribution over the clusters.

Let $\mathcal{X}$ be a finite set of discrete symbols. We call an element in $\mathcal{X}$ a *state*. For each i and t, let $a_{i,t}(x|x')$ $(x, x' \in \mathcal{X})$ be a state transition matrix and let $b_{i,t}(y|x)$ $(x \in \mathcal{X},\ y \in \mathcal{Y})$ be a probability that a state x generates an observed value y. Let $\gamma_{i,t}(x)$ be an initial probability distribution over $\mathcal{X}$. Then the probability distribution of the ith cluster is represented as follows:

$$p_{i,t}(\mathbf{y}) = \sum_{x_1, \cdots, x_{T_t}} \gamma_{i,t}(x_1) \prod_{j=1}^{T_t - 1} a_{i,t}(x_{j+1}|x_j) \prod_{j=1}^{T_t} b_{i,t}(y_j|x_j). \tag{2}$$

The notations used in this paper are summarized in Table 1.

Note that two kinds of latent variables are included in this model. One is the cluster index i indicating which behavioral pattern is generated from. The other is the state x in each cluster indicating which state appears in the fixed behavioral pattern. Hence the *latent variable space* in this model consists of $\{i = 1, ..., K\} \times \mathcal{X}$.

We call the anomalies induced in the latent variable space *latent anomalies*.

There are two reasons for employing HMM mixtures. First, HMM mixtures are appropriate for our purpose. It is possible to represent the behavioral patterns by mixtures for detecting latent anomalies of type (1), and the hidden state by a and b matrix of each HMM for detecting latent anomalies of type (2) (the latent anomalies are defined in Section 1.1). Second, it has been demonstrated in [10,19] that HMM mixtures are effective for representing behavioral patterns and can be successfully applied to anomalous behavior detection from UNIX commands, syslogs, and so on.

Table 1. Notation used in this paper.

Symbol	Definition
y	Observed symbol, discrete finite, $y \in Y$.
Y	Finite set of discrete symbols.
t	Time stamp, discrete, $t = 1, 2, \ldots$
$\mathbf{y}_t$	tth session, $\mathbf{y}_t = (y_{t,1}, \cdots, y_{t,T})$.
T_t	Session length of the tth session, number of symbols included in the session.
$y_{t,j}$	jth symbol in the tth session
$p_t(\mathbf{y})$	Probability distribution of session $\mathbf{y}$ at time t, HMM mixtures.
K	Number of HMMs in the HMM mixture.
$\pi_{i,t}$	Mixture coefficient of the i-th HMM, $\sum_i \pi_{i,t} = 1, \pi_{i,t} > 0 (i = 1, \ldots, K)$.
$p_{i,t}(\mathbf{y})$	Probability distribution of the ith HMM at time t, $p_{i,t}(\mathbf{y}) = \sum_{x_1,\ldots,x_{T_t}} \gamma_{i,t}(x_1) \prod_{j=1}^{T_t-1} a_{i,t}(x_{j+1}\|x_j) \prod_{j=1}^{T_t} b(y_j\|x_j)$
x	Hidden state of HMM, finite discrete, $x \in X$.
X	Finite set of discrete symbols.
$\gamma_{i,t}(x)$	Initial probability distribution over X.
$b_{i,t}(y\|x)$	Probability that a state x generates an observed value y $(x \in X, y \in Y)$.
$a_{i,t}(x\|x')$	Matrix of state transition (from x' to x) probability, $(x, x' \in X)$.
$r_{i,t}(x)$	Steady state probability distribution of the ith HMM, $\sum_{x'} a_{i,t}(x\|x') r_{i,t}(x') = r_{i,t}(x)$.
α_t, β_t, s_t	Model variation vectors, K-dimensional vectors.
$S_\alpha(\mathbf{y}_t), S_\beta(\mathbf{y}_t), S_s(\mathbf{y}_t)$	Change-point score of α_t, β_t, s_t.
$\mathbf{z}_T$	Combination of a session and its corresponding hidden states, $\mathbf{z} = (\mathbf{x}, \mathbf{y})$. T denotes the dimensionality of $\mathbf{y}$ (or $\mathbf{x}$).

3.2. Learning the Model

In order to detect latent anomalies for a given observed sequence, first, the model has to be learned and then, each data (session) has to be given a score on the basis of the estimated model. In this subsection, we describe how to learn a mixture of HMMs.

We can straightforwardly apply the *on-line discounting learning algorithm* proposed in [10] to the learning of our model. The parameters of the model, $(\pi_{i,t}, \gamma_{i,t}(\cdot), a_{i,t}(\cdot|\cdot), b_{i,t}(\cdot|\cdot))$ for $i = 1, \cdots, K$, are dynamically learned at each time step using this algorithm, which conducts parameter estimation by gradually forgetting out-of-date statistics as time goes on. This makes the model adaptive to the nonstationary environment.

4. LATENT ANOMALY DETECTION

4.1. Model Variation and Model Decomposition

In this section, we introduce a method of scoring the anomalousness of each data on the basis of the learned model as in the previous section.

For detecting latent anomalies, we introduce a method of scoring the anomalousness of each data on the basis of the learned model as mentioned in the previous section.

We make the following two steps.

Step 1. Model Variation Vector Construction We define *model variation vectors* $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$ by Eqs. (3) and (4), respectively.

$$(\boldsymbol{\alpha}_t)_i \stackrel{\text{def}}{=} \sum_x \left\{ \pi_{i,t-1} r_{i,t-1}(x) D(a_{i,t-1}(\cdot|x) \| a_{i,t}(\cdot|x)) + \pi_{i,t} r_{i,t}(x) D(a_{i,t}(\cdot|x) \| a_{i,t-1}(\cdot|x)) \right\}, \quad (3)$$

$$(\boldsymbol{\beta}_t)_i \stackrel{\text{def}}{=} \sum_x \left\{ \pi_{i,t-1} r_{i,t-1}(x) D(b_{i,t-1}(\cdot|x) \| b_{i,t}(\cdot|x)) + \pi_{i,t} r_{i,t}(x) D(b_{i,t}(\cdot|x) \| b_{i,t-1}(\cdot|x)) \right\}. \quad (4)$$

Here, $(\boldsymbol{\alpha}_t)_i$ and $(\boldsymbol{\beta}_t)_i$ are the ith components of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$, respectively. $D(p(\cdot)\|q(\cdot))$ denotes the Kullback-Leibler (KL) divergence between two probability distributions. p and q are probability distributions and KL divergence between them are defined as Eq. (5), $r_{i,t}(x)$ is the steady-state probability distribution of the HMM and is derived as the eigenvector of the matrix $a_{i,t}$ corresponding to the eigenvalue 1 (see Eq. (6)).

$$D(p\|q) \stackrel{\text{def}}{=} \sum_x p(x) \log\left(\frac{p(x)}{q(x)}\right), \quad (5)$$

$$\sum_{x'} a_{i,t}(x|x') r_{i,t}(x') = r_{i,t}(x). \quad (6)$$

Here, the log is taken with the natural base. The parameters of the model, $(\pi_{i,t}, \gamma_{i,t}(\cdot), a_{i,t}(\cdot|\cdot), b_{i,t}(\cdot|\cdot))$ for $i = 1, \cdots, K$, are dynamically estimated at each time step using the on-line discounting learning algorithm. Each component of $\boldsymbol{\alpha}_t$ can be thought of as the variation of the matrix $a_{i,t}(x'|x)$. Thus, the value of $\boldsymbol{\alpha}_t$ measures how significantly

the overall hidden state transition has changed. Meanwhile, each component of $\boldsymbol{\beta}_t$ can be thought of as the variation of the matrix $b_{i,t}(y|x)$. Thus, the value of $\boldsymbol{\beta}_t$ measures how significantly the relation between observed symbols and hidden states has changed.

Note that we can calculate model variation vectors, regardless of whether the hidden states are identical for each HMM. This is because there appears no inter-cluster term in Eqs. (3) and (4). Thus, in this paper, we do not deal with the correspondence between hidden states of different HMMs.

Both of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$ have information of the degree to what extent the anomalies appearing in the latent variable space of hidden states have affected the model variation. Thus, we can detect the latent anomalies of type (2) (those in the space of hidden states) by tracking sudden changes of these two vectors.

Finally, we define a vector $\mathbf{s}_t$ as the sum of the two model variation vectors $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$:

$$\mathbf{s}_t \stackrel{\text{def}}{=} \boldsymbol{\alpha}_t + \boldsymbol{\beta}_t. \tag{7}$$

Each component of $\mathbf{s}_t$ can be thought of as the variation of the probability distribution of each cluster. Hence, the 1-norm of $\mathbf{s}_t$ can be recognized as the variation of the total distribution. Thus, we can detect the latent anomalies of type (1) (those in the space of behavioral patterns) by tracking sudden changes of $\mathbf{s}_t$ because the ith component of $\mathbf{s}_t$ represents how largely the model change has been caused by that of ith behavioral pattern.

Step 2. Change-Point Score Computation Once we have vectors $\mathbf{s}_t$, $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$ for each time step t, according to the method of Step 1, we may conduct the procedure of *change-point detection* for the time series $\{\mathbf{s}_t\}_t$, $\{\boldsymbol{\alpha}_t\}_t$ and $\{\boldsymbol{\beta}_t\}_t$ in order to track the significant changes in them. Here, the change-point detection is the process of giving to each time point a score of measuring how significantly the nature of time series has changed at the time point. We describe the details of the methodology of change-point scoring in Section 4.3. In this section, we assume that some change-point scoring method is given, and discuss how to evaluate latent anomalies and observed anomalies.

We define the following scores of five kinds:

$$S_\alpha(\mathbf{y}_t) = (\text{change-point score of vector } \boldsymbol{\alpha}_t), \tag{8}$$

$$S_\beta(\mathbf{y}_t) = (\text{change-point score of vector } \boldsymbol{\beta}_t), \tag{9}$$

$$S_s(\mathbf{y}_t) = (\text{change-point score of vector } \boldsymbol{s}_t), \tag{10}$$

$$S_\alpha(\mathbf{y}_t) + S_\beta(\mathbf{y}_t) = (\text{sum of change-point scoreof } \boldsymbol{\alpha}_t \text{ and change-point score of } \boldsymbol{\beta}_t), \tag{11}$$

$$\mathbf{e}^\dagger \mathbf{s}_t \times S_s(\mathbf{y}_t) = (\text{1-norm of vector } \boldsymbol{s}_t) \times (\text{change point score of vector } \boldsymbol{s}_t). \tag{12}$$

The score S_α measures how significantly the rates of hidden state transitions have changed. The score S_β measures how significantly the occurrence frequency of each symbol has changed. The score S_s measures how significantly the hidden state transition rates and the frequencies have changed. Thus, high S_α and/or S_β indicate the appearance of latent anomalies in the space of hidden states. On the other hand, high S_s indicates the appearance of latent anomalies in the space of behavioral patterns.

The score $S_\alpha + S_\beta$ represents the logical sum of S_α and S_β for anomaly detection. Note that this score is different from S_s because the sum of change-point scores is different from the change-point score of the sum of the vectors. The score in Eq. (12) is defined as a product in order to take into account the model variation itself (namely 1-norm of $\mathbf{s}_t$). By multiplying a 1-norm of $\mathbf{s}_t$, we aim to suppress the effect of noisy fluctuations of $\mathbf{s}_t$ around the origin.

Each scoring corresponds to the detection of anomalies of different types. We consider here that data of higher scores have caused latent anomalies. We emphasize here that our methods enable us to score the total anomalousness by decomposing it into the anomalies of different types in an explicit way.

4.2. Interpretation of Model Variation Vector

We give an interpretation to the model variation vectors, $\mathbf{s}_t$, $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$ from the view of the model decomposition and give their rationale.

In order to detect anomalies, either observed or latent anomalies, it is necessary to estimate anomalousness of the observed value $\mathbf{y}$. For this purpose, we employ the KL-divergence $D(p_{t-1}||p_t)$ as an anomalousness measure. This quantity represents how large p_t has moved from p_{t-1} after learning with $\mathbf{y}_t$. Therefore, $\mathbf{y}_t$ with large KL-divergence is anomalous in the sense that it greatly contributes to changing a probabilistic model.

In order to detect latent anomalies, we decompose the anomalousness measure $D(p_{t-1}||p_t)$ into a number of parts so that each part represents how large the model change has been caused by the change in one latent variable space. For the sake of notational simplicity, we denote a combination of a hidden state $\mathbf{x}$ and an observed value $\mathbf{y}$ as $\mathbf{z}$ ($(\mathbf{x}, \mathbf{y}) = \mathbf{z}$) and denote $\mathbf{z}$ as $\mathbf{z}_T$ when we like to express that $\mathbf{x}$ and $\mathbf{y}$ consist of T components. Hereafter, we compute the KL-divergence under the following assumptions:

Assumption 1: The length T_t of a session at each time step t is large enough ($T_t = T \to \infty$) independently of t. In

addition, the stochastic process of Markovian transition of a hidden state x has a stationary state r ($\lim_{T\to\infty} a^T\gamma = r$).

Assumption 2: For most of $\mathbf{z}$, each $\mathbf{z}$ belongs to a single cluster[1] (in other words, the overlaps among clusters are negligible) so that the following equalities hold:

$$\lim_{T\to\infty}\frac{1}{T}\sum_{i=1}^{K}\sum_{\mathbf{z}_T}\pi_{i,t-1}p_{i,t-1}(\mathbf{z}_T)\log\frac{p_{t-1}(i|\mathbf{z}_T)}{p_t(i|\mathbf{z}_T)}=0, \quad (13)$$

where

$$p_t(i|\mathbf{z}_T)\stackrel{\text{def}}{=}\frac{\pi_{i,t}p_{i,t}(\mathbf{z}_T)}{\sum_k\pi_{k,t}p_{k,t}(\mathbf{z}_T)}. \quad (14)$$

Assumption 2 states that the overlaps among clusters are negligibly small and thus we can consider one session is generated from one behavioral pattern. Let us qualitatively explain the connection between this assumption and Eq. (13). The left-hand side of Eq. (13) represents a kind of $\mathbf{z}_T$ average of variation of posterior distribution $p_{t-1}(\cdot|\mathbf{z}_T)$. Eq. (13) holds in the following cases. First, if the distribution of the clusters given the session does not change over time, variation of the posterior distributions does not change and the equation holds. Second, if overlaps among clusters are small, $p_\cdot(\cdot|\mathbf{z}_T)$ usually becomes 0 or 1 and does not change over time very much and thus, variation of the posterior distributions is small. In this case, the equation holds, regardless of whether the distribution of the clusters given the session changes over time. In order to represent the second case, we employed the equation.

We have made the assumptions from the following theoretical reasons. First, we need to fix the session length T when we compute the KL-divergence. However, we do not know either typical or optimal T. It is natural to choose $T=\infty$ because we can compare two models under the same condition, in the limit of $T=\infty$. Note that Assumption 1 is not required for learning models but for computing the KL-divergence. Namely, we do *not* assume that the observed sessions used for learning models have infinitely long T. Assumption 2 implies that most of sessions belong to a single behavioral pattern. Under our problem settings, it is natural to employ this assumption because it becomes difficult to find clearly separated behavioral patterns if the assumption is violated.

Below we show the decomposition of the KL-divergence. We define the model variation quantity for the ith cluster at time t by $D(p_{i,t-1}(\mathbf{z})||p_{i,t}(\mathbf{z}))$, and the symmetrized model variation at time t by $D(p_{t-1}||p_t)+D(p_t||p_{t-1})$. Under the assumptions we have made, the symmetrized model variation can be decomposed into a number of parts so that each part represents how large the model change has been caused by the change of one latent variable space.

Under the assumptions, we have made, we can decompose the symmetrized model variation according to the following theorem:

THEOREM 1: Under Assumptions 1 and 2, the symmetrized model variation is decomposed as follows:

$$\lim_{T\to\infty}\tfrac{1}{T}\big\{D(p_{t-1}||p_t)+D(p_t||p_{t-1})\big\}=\sum_{i=1}^{K}\Big((\boldsymbol{\alpha}_t)_i+(\boldsymbol{\beta}_t)_i\Big). \quad (15)$$

This theorem describes that the symmetrized model variation is decomposed into a number of parts so that each part represents how large the model change has been caused by the change of one latent variable space. From the theorem, it is shown that the model variation is decomposed by two steps. In Fig. 3, the flow of the two-step decomposition is summarized taking $K=2$ case for instance. The first step is decomposition into sum of each mixture's variation in the space of the latent variable representing behavioral patterns. The second step is decomposition into sum of two parts, α and β, in the space of the latent variable representing hidden states.

This is formally proven using the following two lemmas. The proofs of the lemmas are shown in Appendices A and B.

LEMMA 1: Under Assumptions 1 and 2, the symmetrized model variation is decomposed into the sum of $(\mathbf{s}_t)_i$s;

$$\lim_{T\to\infty}\tfrac{1}{T}\big\{D(p_{t-1}||p_t)+D(p_t||p_{t-1})\big\}=\lim_{T\to\infty}\tfrac{1}{T}\sum_{i=1}^{K}D(p_{i,t-1}||p_{i,t})+D(p_{i,t}||p_{i,t-1}) \quad (16)$$

$$=\sum_{i=1}^{K}(\mathbf{s}_t)_i. \quad (17)$$

First, the symmetrized model variation is decomposed into $(\mathbf{s}_t)_i$s, components of the vector $\mathbf{s}_t$. Lemma 1 shows that the symmetrized model variation coincides with the $1-$norm of the vector $\mathbf{s}_t$, and each $(\mathbf{s}_t)_i$ represents the degree of a change of each model of a behavioral pattern (namely the degree of a change of each component of the mixtures).

[1] Assumption 2 does not state that all sessions belong to only one behavioral pattern, but states that a session is generated from one behavioral pattern and different session can belong to different pattern.

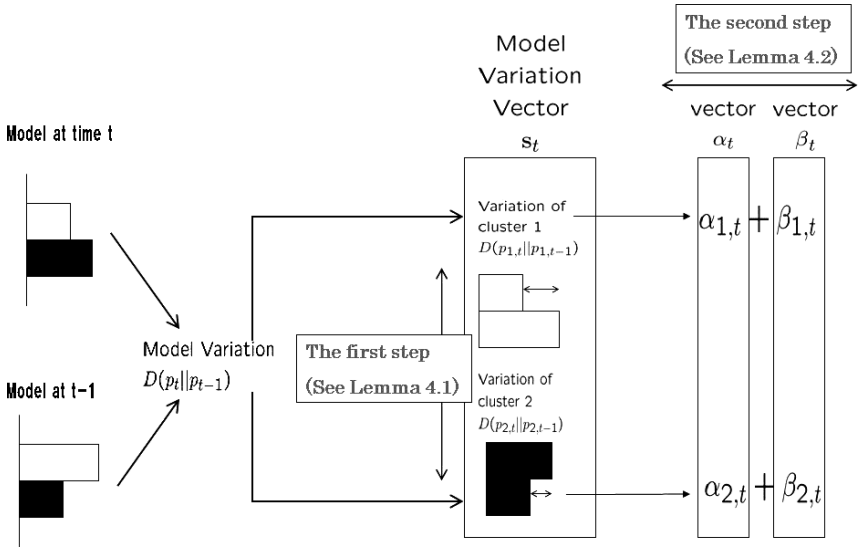

Fig. 3 Flow of the decomposition of the symmetrized model variation in the case of $K = 2$. The decomposition consists of two steps.

This gives a rationale of $\mathbf{s}_t$ since the symmetrized model variation can be a measure of how significantly the model has changed. Hence, the latent anomalies induced by sudden change of behavioral patterns can be detected by tracking sudden change of the vector $\mathbf{s}_t = ((\mathbf{s_t})_1, \cdots, (\mathbf{s}_t)_K)$.

LEMMA 2: Under Assumption 1, the vector $\mathbf{s}_t$ is decomposed into the sum of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$;

$$(\mathbf{s}_t)_i = (\boldsymbol{\alpha}_t)_i + (\boldsymbol{\beta}_t)_i \ \big(i = 1, \cdots, K\big). \tag{18}$$

Second, each component of $\mathbf{s}_t$, $(\mathbf{s}_t)_i$, is decomposed into the sum of $(\boldsymbol{\alpha}_t)_i$ and $(\boldsymbol{\beta}_t)_i$. Lemmas 1 and 2 show that the symmetrized model variation $\frac{1}{T}\big\{D(p_{t-1}||p_t) + D(p_t||p_{t-1})\big\}$ is also decomposed into two parts. One is a model variation consisting of $r_{i,t}$, $\pi_{i,t}$ and $D(a_{i,t-1}(\cdot|x)||a_{i,t}(\cdot|x))$, which represents how significantly the overall hidden state transition pattern has changed. The other is a model variation consisting of $r_{i,t}$, $\pi_{i,t}$ and $D(b_{i,t-1}(\cdot|x)||b_{i,t}(\cdot|x))$, which represents how significantly a symbol generation pattern from a fixed state has changed. The former coincides with $\sum_i(\boldsymbol{\alpha}_t)_i$ while the latter coincides with $\sum_i(\boldsymbol{\beta}_t)_i$. Thus, by tracking sudden change of $\boldsymbol{\alpha}$ and/or $\boldsymbol{\beta}$, we can detect the latent anomalies induced by sudden change of hidden states, from different aspects.

4.3. Change-point Scoring

As for the change-point scoring in Step 2 in Section 4.1, we employ the method proposed in 7, which we briefly summarize below.

Let $\mathbf{u}_t$ be either $\boldsymbol{\alpha}_t$, $\boldsymbol{\beta}_t$ or $\mathbf{s}_t$. A time series $\{\mathbf{u}_t : t = 1, 2, ...\}$ is learned with an autoregression (AR) model. Here, the model parameters of the AR model are estimated using an on-line discounting learning algorithm for which the out-of-date statistics are gradually forgotten as time goes on. We denote the probability density function of the learned model as $p_{AR}(\mathbf{u}_t|\mathbf{u}^{t-1})$ $(t = 1, 2, \cdots, \mathbf{u}^{t-1} = \{\mathbf{u}_i\}_{i=1}^{t-1})$.

For each time t the *logarithmic score* for $\mathbf{u}_t$ relative to p_{AR} is written as $-\log p_{AR}(\mathbf{u}_t|\mathbf{u}^{t-1})$. We then define a time series $\{v_t : t = 1, 2, ...\}$ by a moving average of the logarithmic score for $\{\mathbf{u}_t : t = 1, 2, ...\}$:

$$v_t = -\frac{1}{w}\sum_{i=t-w+1}^{t} \log P_{AR}\,(\mathbf{u}_t|\mathbf{u}^{t-1}), \tag{19}$$

where $w(\geq 1)$ is a given window size.

Then v_t is further learned with another AR model $\{v_t : t = 1, 2, ...\}$ using the on-line discounting learning algorithm as above. We denote the probability density function of the learned model as $\tilde{p}_{AR}(v_t|v^{t-1})$.

Finally, we compute a *change-point score* S_t at each time step t by

$$S_t = -\log \tilde{p}_{AR}(v_t|v^{t-1}), \tag{20}$$

which measures how significantly the nature of the time series has changed. A higher score indicates that a bursty change has occurred in the time series. The validity of this change-point scoring method has been demonstrated in [7].

4.4. Number of Hidden States and Clusters and Construction of Sessions

One of the limitations of the proposed methods is that a user has to set mixture size K and the number of hidden states. In the experiments in Section 5, we fixed K to 2 or 5 and the number of hidden states to 3, which are empirically good values for anomaly detection.

Of course, it is much better to determine them optimally than to employ fixed values. They can be determined by DMS [9,10] in an on-line manner. However, it may happen that optimal values change in time. Then, it becomes difficult to conduct change-point detection from model variation vectors, because dimensionality of a vector changes in time and it is not trivial how to treat such vectors. Thus, we employed fixed values. Note that we cannot employ batch optimization algorithms because our aim is to detect anomalies early. Therefore, we cannot use time-invariant values determined by batch model selection.

In the case of session construction, the situation is the same. There exist degrees of freedom, the session length and how to cut the windows (nonoverlapping or sliding window). In the experiments, we used the fixed value, session length 10 and nonoverlapping windows, while optimally determining them is a challenging research issue. However, the proposed methods work in any case, namely, in the case of any session length and in the case of nonoverlapping or sliding window.

4.5. Generalization of the Methodology

In this paper, we illustrate our method using an HMM mixture. However, it is extendable to a more general class of probabilistic models. We discuss on the extension in Appendix C.

5. EXPERIMENTS AND DISCUSSIONS

We performed two experiments in order to validate the effectiveness of the proposed methods in comparison with existing methods which do not take into account the latent variable space. We evaluated them in terms of how early anomalies could be detected for the same level of false alarm rates.

5.1. Data Set

We employed an artificial data set and UNIX command data set [1] for masquerade detection. Both data sets consisted of 15 000 symbols. A symbol represented a unit of action, for example, a UNIX command in Section 5.3. We divided each data set into 1500 blocks consisting of 10 symbols each and called a block *session*. We represented the order of the sessions using the index t ($t = 1, \cdots, 1500$). Each data set had labels which indicate whether each session is anomalous or not. Using these labels, we evaluated the accuracy of anomaly detection for each method.

As a probabilistic model of session generation, we employed an HMM mixture as in Section 3. Here the mixture size (the number of components in the mixture model) was fixed to 2 in Section 5.2 and was fixed to 5 in Section 5.3. The number of hidden states in each HMM was fixed to 3. At each time step, one session was input and the model was learned using the on-line discounting learning algorithm (see Section 3.2).

After the learning process was finished, we gave scores to each data using our proposed methods (see Eqs. (8)$\sim$(12) and existing methods. When we conducted the change-point scoring mentioned in Section 4.3, we set $w = 4$, where w was the window size in Eq. (19). As for the existing methods, we employed the HMM mixture based scoring (without taking into account the latent variable space for scoring) and the naive Bayes (NB) based scoring (without taking into account the latent variable even in a probabilistic model). Their corresponding scoring functions S_{HMM} and S_{NB} were, respectively, calculated as follows:

$$S_{HMM}(\mathbf{y}_t) = -\frac{1}{T_t}\log P_{HMM}(\mathbf{y}_t|\theta^{(t-1)}), \tag{21}$$

$$S_{NB}(\mathbf{y}_t) = -\frac{1}{T_t}\log P_{NB}(\mathbf{y}_t|\theta^{(t-1)}). \tag{22}$$

Here, P_{HMM} and P_{NB} represent probability distributions when we employ an HMM mixture and NB as probabilistic models respectively. Eqs. (21) and (22) are the logarithmic losses for a session when a probabilistic model of session generation is an HMM mixture (θ denotes the parameters of the model) and NB model, respectively. Note that NB is reduced to the special case of an HMM mixture where $K = 1$ and hidden state set $\mathcal{X}$ consists of a single state, which we write as σ. With the same notation as Eqs. (1) and (2), a probability distribution of NB is represented as

$$p_t(\mathbf{y}) = \prod_{i=1}^{T_t} b_t(y_i|\sigma). \tag{23}$$

We employed the scoring methods as above to conduct anomaly detection. Data to which our proposed methods give higher scores include latent anomalies, while those to which the existing ones give higher scores are considered to include observed anomalies only. We evaluated the performance of all of the methods in terms of how early anomalies could be detected for constant false alarm rates.

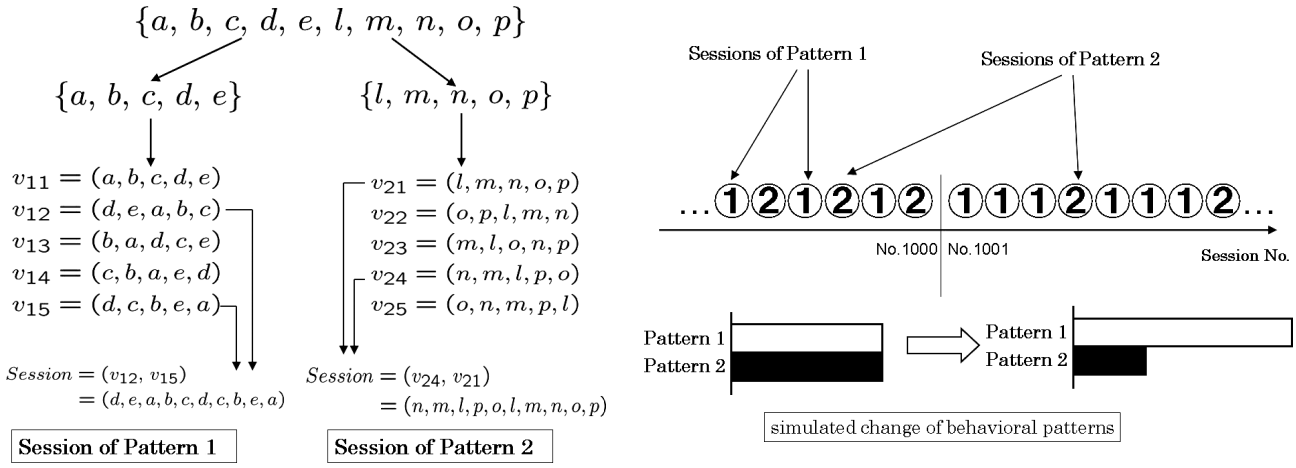

Fig. 4 Construction of the artificial data used in Section 5.2. (Left) Randomly combining two tuples of $v_{1\cdot}$ or $v_{2\cdot}$, we made a session consisting of ten symbols. (Right) We changed the occurrence frequency of each pattern after the 1000th session. The change represents a simulated change of behavioral patterns.

As a performance measure, we used what percentage of the first Δ sessions of anomalous sequence were detected for constant false alarm rates. Namely, we did not consider all anomalous sessions but rather those which appeared just after the starting point of the anomaly to be detected. We denote the number of sessions to be detected as Δ. In the experiments, we set $\Delta = 5$, 10 or 50.

Then, we drew a curve in which the horizontal axis showed the false-positive rate and the vertical axis showed what percentage of the first Δ sessions of anomalous sessions were detected. The quantity which the vertical axis showed can be thought of being similar to what we call the *average score* defined in [6]. Hence, hereafter, we call this quantity, detection rate of the first Δ sessions after an anomaly started, *average score*. It was a measure of how early anomalies were detected. For each of the methods, we employed the area under its corresponding curve (area under the curve (AUC)) $\mathcal{R}$, which was normalized to 1 ($0 \leq \mathcal{R} \leq 1$), as the criterion for its detection accuracy. A method with higher $\mathcal{R}$ indicates its higher detection accuracy. This is because a curve is lifted up and $\mathcal{R}$ becomes larger when the detection accuracy becomes higher for the same level of false-positive rates. A method with $\mathcal{R} = 1$ can detect all starting points of anomalies with no time delay. On the other hand, that with $\mathcal{R} = 0$ cannot detect any starting points of anomalies in defined time range Δ.

5.2. Behavioral Pattern Change Detection for Artificial Data

First, we conducted experiments on anomaly detection using an artificial data set.

5.2.1. *Data set*

The artificial data set was sequence of two patterns and included an anomaly where the occurrence frequency of each pattern changed. In this experiment, we tested whether our proposed methods were able to detect the behavioral pattern change of this kind earlier than the existing methods which do not take into account anomalies recognized in latent variable space.

We made an artificial data as shown in Fig. 4. It consisted of 15 000 symbols and each symbol was included in a finite set, $\{a, b, c, d, e, l, m, n, o, p\}$. We defined ten kinds of tuples of five symbols: $v_{11} = (a, b, c, d, e)$, $v_{12} = (d, e, a, b, c)$, $v_{13} = (b, a, d, c, e)$, $v_{14} = (c, b, a, e, d)$, $v_{15} = (d, c, b, e, a)$, $v_{21} = (l, m, n, o, p)$, $v_{22} = (o, p, l, m, n)$, $v_{23} = (m, l, o, n, p)$, $v_{24} = (n, m, l, p, o)$ and $v_{25} = (o, n, m, p, l)$. We combined two tuples as a *session* (see the left panel of Figure 4). Namely, the data of 15 000 symbols consisted of 1500 sessions. From the 1st to the 1000th sessions, each of the $(4n + 1)$th and $(4n + 3)$th ($n = 0, \cdots, 249$) sessions consisted of two tuples of v_{1i} ($i = 1, \cdots, 5$) which were randomly chosen, and each of the $(4n + 2)$th and $(4n + 4)$th ($n = 0, \cdots, 249$) sessions consisted of two tuples of v_{2i} ($i = 1, \cdots, 5$) which were randomly chosen. Next, from 1001st to 1052nd sessions, each of the $(4n + 1)$th,$(4n + 2)$th and $(4n + 3)$th ($n = 250, \cdots, 262$) sessions consisted of two tuples of $v_{1i}(i = 1, \cdots, 5)$ which were randomly chosen, and each of the $(4n + 4)$th($n = 250, \cdots, 262$) sessions consisted of two tuples of $v_{2i}(i = 1, \cdots, 5)$ which were randomly chosen. Remaining sessions were constructed in the same manner as the first 1000 sessions.

This data included a simulated change of behavioral patterns (see the right panel of Fig. 4). The change is recognized as a latent anomaly for two reasons. First,

Table 2. Results of pattern change detection. The average area under the curve, $R(0 \leqslant R \leqslant 1)$, are summarized. As for the definition of each score, see Eqs. (8)–(12), (21) and (22). The *average score* explained in Section 5.1 represents what percentage of the first Δ sessions of anomalous sessions is detected.

	Score	Δ 5	10	50
Existing method	S_{HMM}	0.102	0.085	0.079
	S_{NB}	0.407	0.385	0.401
	S_α	0.758	0.727	0.586
	S_β	0.943	0.871	0.668
Proposed method	S_β	0.944	0.881	0.673
	$S_\alpha + S_\beta$	0.931	0.854	0.620
	$\mathbf{e}\times S\mathbf{s}\times S_S$	0.932	0.867	0.671

it is difficult to detect this change by tracking observed data significantly deviated from the ordinary regularity because, in the data, rare events never appeared (Patterns 1 and 2 are both popular) even after the anomaly started. Second, on the other hand, it can be detected as sudden change of behavioral patterns. In this data, index j of v_{ji} $(i = 1, \cdots, 5)$ represents a pattern because there is no transition between the two symbol sets $\{a, b, c, d, e\}$ and $\{l, m, n, o, p\}$. Thus, the data is considered to be a sequence of Pattern 1 consisting of v_1.and Pattern 2 consisting of v_2.. From the 1001st to the 1052nd sessions, the occurrence frequency of each pattern changes. This change corresponds to that of behavioral patterns.

Considering the 1001st∼1052nd sessions to be anomalous sessions, we conducted an experiment on anomaly detection. By learning an HMM mixture with the on-line discounting learning algorithm, we gave scores to each session according to the methods in Section 4.1. The definition of the scores follows Eqs. (8)∼(12), (21) and (22). We repeated this procedure 50 times and estimated $\mathcal{R}$ as in Section 5.1. Employing the average of $\mathcal{R}$ as an evaluation criterion, we compared performance of our proposed methods with those of the existing methods.

5.2.2. Results and discussions

As results of anomaly detection using seven kinds of scores, the average area under the curve, $\mathcal{R}$ $(0 \leq \mathcal{R} \leq 1)$, are summarized in Table 2.

From Table 2 we observed that our proposed methods outperformed NB and HMM. This indicates that our proposed methods detected the change of behavioral patterns earlier than the existing methods.

HMM and NB did not perform well in this experiment. At the starting point of the anomalous sessions, sessions of Pattern 1 appear. After the anomaly started, the scores of HMM and NB of Pattern 1 decreased because the occurrence frequency of Pattern 1 increased. Therefore, in the case of pattern change detection of this kind, performance of a better learner becomes worse when only occurrence frequency is taken into account. Thus, HMM performed worse than NB.

This anomaly can not be detected even by DMS [9,10], which detects anomalies in the space of behavioral patterns. This is because the number of patterns did not change after the anomaly started, while DMS detects a change of the number.

From these results, it is shown that our proposed methods detected the anomalies which could not be detected by the existing methods since the former detected latent anomalies as well as observed ones while the latter detected observed ones only. In this experiment, from which pattern a session is generated is latent information and the sudden change of the occurrence frequency of each pattern is a latent anomaly.

When we employed larger Δ, the performance of our proposed methods became worse. The reason for this phenomenon is as follows: The proposed methods detected a change point of model variation vectors. The change point corresponded to a starting point of an anomalous sequence. Therefore, the proposed method was inadequate for detecting outliers which were not induced from changes of data structures.

5.3. Masquerade Detection from UNIX Command Sequence

Second, we applied our methods to masquerade detection.

5.3.1. Experimental settings

We used the data which prepared by Schonlau *et al* [1] for masquerade detection. As described in [1], the data set included 70 users' UNIX command sequence. Data for each user consisted of 15 000 commands. The 70 users were divided into the class of 50 target users and that of 20 masqueraders. The first 5000 commands of all the target 50 users included no masquerade sequence. The remaining 10 000 commands of each target user were divided into 100 blocks of 100 commands each. These blocks were seeded with masquerading users, i.e. with data of a user in 20 masqueraders. At any given block after the 5000 commands, a masquerade started with a probability of 1%. If a previous block was a masquerade, then the next one will also be a masquerade with a probability of 80%. This data set is available for download from `http://www.schonlau.net/`.

Each user's data had labels that indicated whether each block was masquerader or not. Using these labels, we evaluated the masquerade detection accuracy of each method. We divided each user into 1500 blocks of 10 commands and called a block *session*.

Table 3. Results of masquerade detection. The average area under the curve, $R(0 \leqslant R \leqslant 1)$, of 18 users is summarized. As for the definition of each score, see Eqs. (8)–(12), (21) and (22). The *average score* explained in Section 5.1 represents what percentage of the first Δ sessions of masquerade sessions is detected.

	Score	Δ 5	10	50
Existing method	S_{HMM}	0.747	0.745	0.735
	S_{NB}	0.638	0.622	0.621
	S_α	0.735	0.751	0.670
	S_β	0.804	0.775	0.651
"Proposed method	S_β	0.805	0.776	0.651
	$S_\alpha + S_\beta$	0.819	0.810	0.680
	$\mathbf{e}\times S\mathbf{s}\times S_S$	0.842	0.818	0.720

From the 50 users' data sets, we selected 18 data sets for the evaluation. These data sets were all of the ones such that the length of a masquerade sequence was not less than 500 commands. The reason why we selected these data sets is that we concentrate our research on the detection of bursty change of behavioral patterns or data structures, rather than outlier detection.

By learning an HMM mixture with the on-line discounting learning algorithm and scoring each session, we performed masquerade detection. The definition of the scores follows Eqs. (8)∼(12), (21) and (22). Employing area under the curve as in Section 5.1, $\mathcal{R}$, as goodness measure, we compared the masquerade detection performance of our proposed methods with those of an HMM mixture and NB.

5.3.2. *Results and discussions*

As results of masquerade detection using seven kinds of scores, the average area under the curve, $\mathcal{R}$ ($0 \leq \mathcal{R} \leq 1$), of 18 users are summarized in Table 3. The curves for the score of Eq. (12), HMM scoring and NB scoring in the case of $\Delta = 5, 10, 50$ are in Figure 5.

From Table 3, we observed that our proposed methods outperformed NB and HMM for a small Δ. When $\Delta = 5$, the average $\mathcal{R}$ of our proposed methods using five kinds of scores was about 8% larger than HMM and about 25% larger than NB. This result indicates that the proposed methods detected masquerades 8∼25% earlier than HMM and NB. It is also shown that the change detection of behavioral patterns was effective for early detection of anomalous behaviors.

In contrast to the case of $\Delta = 5$, HMM outperformed the proposed methods for $\Delta = 50$(500 commands). The proposed methods detected a change point of model variation vectors, and a change point corresponded to a starting point of a masquerade sequence. Therefore, the proposed methods were inadequate for detecting *outlier sessions* which were located away from the point. Here outlier sessions represent a kind of observed anomalies which are not caused by bursty behavior changes but are recognized as statistical outliers. On the other hand, the existing methods detected outlier sessions which were deviated relative to the learned model of data generation. Thus, they were able to detect masquerade sessions which were located away from the starting point. We may say that the proposed methods are adequate for early detection of anomalies where model structure changes, while HMM is adequate for detecting isolated anomalous events.

Even though the performance is not so well for higher values of Δ, the proposed methods are better for masquerade detection. This is because for masquerade detection analyzing with lower Δ values are sufficient. In masquerade detection, it is important to detect apparition of a masquerader and thus it is unnecessary to isolate all masquerade sessions.

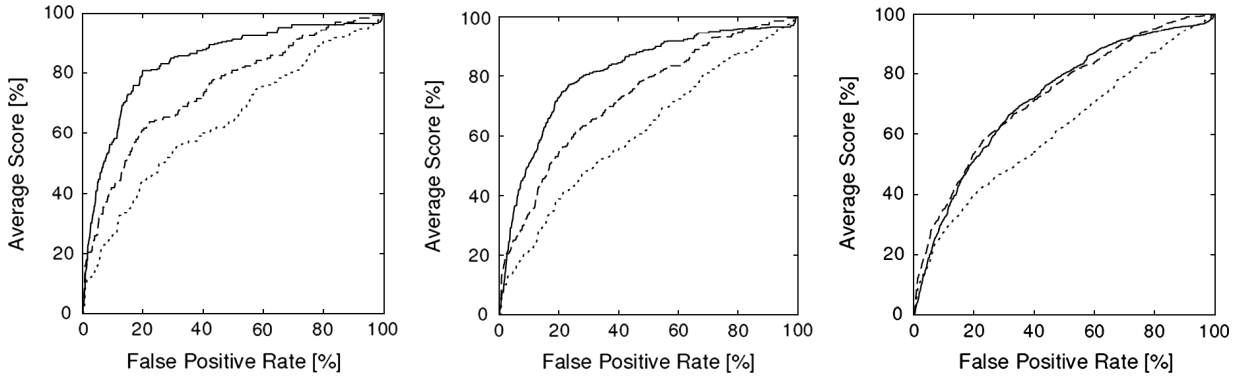

Fig. 5 The curve for evaluating detection accuracy in the case of $\Delta = 5$ (left panel), $\Delta = 10$ (middle panel) and $\Delta = 50$ (right panel). The horizontal axis represents false-positive rate and the vertical axis represents average score. The explanation of this curve is in Section 5.1 Dotted, dashed and solid lines represent the result of NB scoring, HMM scoring and the proposed method with the score of Eq. (12), respectively.

Regardless of which methods were employed, the performance became worse for larger Δ. There are two reasons for this degradation. One is that the proposed methods were inadequate for detecting outlier sessions. The other is that, with the on-line discounting learning algorithm, models were learned adaptively to nonstationary environments. If many sessions consisting of the same command sequence are included in the masquerade sequence, a session of this kind coming after many masquerade sessions is given a lower score value than that included in the first several sessions of the sequence because models have been adapted to generation of repeated masquerade sessions. Thus, the both methods that we compared each other are insensitive to masquerade sessions which have been repeated many times.

We observed from Table 3 that S_s and S_β lead to approximately the same results, and $S_\alpha + S_\beta$ lead to better results than S_s. This observation indicates the following two things: First, $\boldsymbol{\beta}_t$ contributed to $\mathbf{s}_t$ much more than $\boldsymbol{\alpha}_t$. Second, $\boldsymbol{\alpha}_t$ played an important role in enhancing a detection accuracy. In the case of $S_\alpha + S_\beta$, the change point scores of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$ were treated with even weight, though the absolute value of $\boldsymbol{\alpha}_t$ was much smaller than that of $\boldsymbol{\beta}_t$. Namely, it is possible to enhance the detection accuracy in this experiment by emphasizing the hidden state variable space. This implies that it is effective for early detection of anomalous behaviors to utilize not only information of a latent variable indicating a behavioral pattern but also that of a hidden state variable x in each behavioral pattern. From Table 3, we observed that the masquerade detection performance became worse when we neglected $\boldsymbol{\beta}_t$. Therefore, we may employ a score such as $S(\mathbf{y}_t) = cS_\alpha(\mathbf{y}_t) + (1-c)S_\beta(\mathbf{y}_t)$ with optimal coefficient c, though we do not deal with its optimization issue in this paper.

We also observed that $S_s \times \mathbf{e}^\dagger \mathbf{s}$ outperformed S_s. By taking into account the 1-norm of $\mathbf{s}_t$, $S_s \times \mathbf{e}^\dagger \mathbf{s}$ became insensitive to noisy fluctuation of $\mathbf{s}_t$ around the origin. Thus, it is effective to remove the effect of such noisy fluctuations from anomalousness scores.

Summarizing the results of this experiment, it was shown that the proposed methods work better than an HMM mixture and NB as the former can detect latent anomalies as well as observed ones, while the latter detect observed anomalies only. Differently from Unlike the case of the artificial data, it is difficult to identify latent anomalies. However, there are two possible candidates for latent anomalies. One is, same as the case of the artificial data, a sudden change of the occurrence frequency of each pattern. This is because we may expect that sudden changes of behavioral patterns occur when masquerade sequence starts. The other is a sudden change of structures of a hidden state transition. This change corresponds to that of the appearance order of UNIX commands in some behavioral patterns. The change is latent because behavioral patterns are not observed directly. From the experimental results it is shown that detection accuracy is enhanced, it was possible to enhance detection accuracy, by giving a more weight to $\boldsymbol{\alpha}_t$ than $\boldsymbol{\beta}_t$. Components of $\boldsymbol{\alpha}_t$ represent how significantly the hidden state transition has changed. Thus, a change of the appearance order of commands in a behavioral pattern can be thought of as one of latent anomalies.

6. CONCLUDING REMARKS

In this paper, we have proposed methods of detecting latent anomalies. The key ideas of the methods are; (i) constructing the model variation vector, which is introduced relative to the latent variable space, and (ii) the latent anomaly detection is reduced to the issue ofchange-point detection for the time series that the model variation vector forms.We have demonstrated through the experimental results usingartificial data set and UNIX command data set that our methods have significantly enhanced the accuracy of existinganomaly detection methods, namely, the HMM mixture basedscoring without using the latent variable space and the NB based scoring. Future works include further extension of our proposed methods by combining it with the existing DMS method [10] to establish a more general framework of anomaly detection using the latent variable space.

APPENDIX A

A.1. Proof of Lemma 1

In this appendix, we give the proofof Lemma 1 in Section 4.2.

First, we derive the asymptotic form of $D(p_{i,t-1}(\mathbf{z})||p_{i,t}(\mathbf{z}))$ as follows:(In the following expansion, the cluster index i and timeindex $t-1$ are omitted, and time index t is replaced with$'$ for the sake of notational simplicity.)

$$\lim_{T\to\infty}\frac{1}{T}D(p_{i,t-1}||p_{i,t}) = \lim_{T\to\infty}\frac{1}{T}\sum_{\mathbf{x},\mathbf{y}}\gamma(x_1)b(y_1|x_1)$$
$$\times\left(\prod_{j=2}^{T}a(x_j|x_{j-1})b(y_j|x_j)\right)$$
$$\times\log\left[\frac{\gamma(x_1)b(y_1|x_1)\prod_{j=2}^{T}a(x_j|x_{j-1})b(y_j|x_j)}{\gamma'(x_1)b'(y_1|x_1)\prod_{j=2}^{T}a'(x_j|x_{j-1})b'(y_j|x_j)}\right]$$

$$\lim_{T\to\infty}\frac{1}{T}\sum_{j=2}^{T}\left\{\sum_{x_j,y_j,x_{j-1}} r(x_{j-1})a(x_j|x_{j-1})b(y_j|x_j)\right.$$
$$\left.\times\log\frac{b(y_j|x_j)}{b'(y_j|x_j)}+\sum_{x_j,x_{j-1}} r(x_{j-1})a(x_j|x_{j-1})\log\frac{a(x_j|x_{j-1})}{a'(x_j|x_{j-1})}\right\}$$
$$=\sum_x r(x)\left\{\sum_y b(y|x)\log\frac{b(y|x)}{b'(y|x)}+\sum_{x'}a(x'|x)\log\frac{a(x'|x)}{a'(x'|x)}\right\}$$
$$=\sum_x r(x)\Big\{D(b(\cdot|x)||b'(\cdot|x))+D(a(\cdot|x)||a'(\cdot|x))\Big\}. \quad (A.1)$$

Here $r_{i,t}(x)$ is the eigenvector of matrix $a_{i,t}$ corresponding to eigenvalue 1 (see Eq. (6)). For the derivation of Eq. (24), we have used $\log\left[\prod_j a_j/a'_j\right]=\sum_j\log\left[a_j/a'_j\right]$ and $\lim_{n\to\infty}a^n\gamma=r$ (existence of stationary state, from Assumption 1 in Section 4.2). Let us define $\Delta(p_{i,t-1}||p_{i,t})$ by

$$\Delta(p_{i,t-1}||p_{i,t}) \stackrel{\text{def}}{=} \sum_x r_{i,t-1}(x)\left\{D(a_{i,t-1}(\cdot|x)||a_{i,t}(\cdot|x))\right.$$
$$\left.+D(b_{i,t-1}(\cdot|x)||b_{i,t}(\cdot|x))\right\}. \quad (A.2)$$

Then we further obtain the following asymptotic form of $D(p_{t-1}||p_t)$.

$$\lim_{T\to\infty}\frac{1}{T}D(p_{t-1}||p_t)$$
$$=\lim_{T\to\infty}\frac{1}{T}\sum_{i=1}^{K}\sum_{\mathbf{z}}\pi_{i,t-1}p_{i,t-1}(\mathbf{z})\log\frac{\sum_j\pi_{j,t-1}p_{j,t-1}(\mathbf{z})}{\sum_k\pi_{k,t}p_{k,t}(\mathbf{z})}$$
$$=\lim_{T\to\infty}\frac{1}{T}\sum_{i=1}^{K}\sum_{\mathbf{z}}\pi_{i,t-1}p_{i,t-1}(\mathbf{z})$$
$$\times\left\{\log\frac{\pi_{i,t-1}}{\pi_{i,t}}+\log\frac{p_{i,t-1}(\mathbf{z})}{p_{i,t}(\mathbf{z})}-\log\frac{p_{t-1}(i|\mathbf{z})}{p_{i,t}(i|\mathbf{z})}\right\}$$
$$=\lim_{T\to\infty}\frac{1}{T}\sum_{i=1}^{K}\pi_{i,t-1}\left\{\log\frac{\pi_{i,t-1}}{\pi_{i,t}}\right.$$
$$\left.+\sum_{\mathbf{z}}p_{i,t-1}(\mathbf{z})\log\frac{p_{i,t-1}(\mathbf{z})}{p_{i,t}(\mathbf{z})}\right\} \quad (A.3)$$
$$=\lim_{T\to\infty}\frac{1}{T}\sum_{i=1}^{K}\pi_{i,t-1}D(p_{i,t-1}||p_{i,t}), \quad (A.4)$$
$$=\sum_{i=1}^{K}\pi_{i,t-1}\Delta(p_{i,t-1}||p_{i,t}). \quad (A.5)$$

Here we have used Assumption 2 in Section 4.2 for the derivation of Eq. (A.3) and Eqs. (A.1) and (A.2) for the derivation of Eq. (A.5), respectively.

On the other hand, note that by the definition of $(\mathbf{s}_t)_i$, it is rewritten as follows:

$$(\mathbf{s}_t)_i=\pi_{i,t-1}\Delta(p_{i,t-1}||p_{i,t})+\pi_{i,t}\Delta(p_{i,t}||p_{i,t-1}) \quad (A.6)$$

From Eqs. (A.1)∼(A.6), we can derive the asymptotic form of the symmetrized model variation $D(p_{t-1}||p_t)+D(p_t||p_{t-1})$ as follows;

$$\lim_{T\to\infty}\frac{1}{T}\left\{D(p_{t-1}||p_t)+D(p_t||p_{t-1})\right\}$$
$$=\lim_{T\to\infty}\sum_i\frac{1}{T}\left\{D(p_{i,t-1}||p_{i,t})+D(p_{i,t}||p_{i,t-1})\right\} \quad (A.7)$$
$$=\sum_i\pi_{i,t-1}\Delta(p_{i,t-1}||p_{i,t})+\pi_{i,t}\Delta(p_{i,t}||p_{i,t-1}) \quad (A.8)$$
$$=\sum_i(\mathbf{s}_t)_i. \quad (A.9)$$

Eqs. (A.7), (A.8) and (A.9) are derived from Eqs. (A.4), (A.5) and (A.6), respectively.

From Eqs. (A.9) and (A.8), it can be seen that Lemma 1 holds. ■

APPENDIX B

B.1. Proof of Lemma 2

In this appendix, we give the proof of Lemma 2 in Section 4.2.

Substituting Eq. (A.2) for Eq. (A.8), we can decompose $\mathbf{s}$ into the sum of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$;

$$\sum_i(\mathbf{s}_t)_i=\sum_i\pi_{i,t-1}\Delta(p_{i,t-1}||p_{i,t})$$
$$+\pi_{i,t}\Delta(p_{i,t}||p_{i,t-1}) \quad (B.1)$$
$$=\sum_{i,x}\left\{\pi_{i,t-1}r_{i,t-1}(x)D(a_{i,t-1}(\cdot|x)||a_{i,t}(\cdot|x))\right.$$
$$\left.+\pi_{i,t}r_{i,t}(x)D(a_{i,t}(\cdot|x)||a_{i,t-1}(\cdot|x))\right\}$$
$$+\sum_{i,x}\left\{\pi_{i,t-1}r_{i,t-1}(x)D(b_{i,t-1}(\cdot|x)||b_{i,t}(\cdot|x))\right.$$
$$\left.+\pi_{i,t}r_{i,t}(x)D(b_{i,t}(\cdot|x)||b_{i,t-1}(\cdot|x))\right\} \quad (B.2)$$
$$=\sum_i(\boldsymbol{\alpha}_t)_i+\sum_i(\boldsymbol{\beta}_t)_i. \quad (B.3)$$

Eq. (A.10) is derived from Eqs. (A.8) and (A.9). Eqs. (A.11) and (A.12) follow the definition of the vectors $\boldsymbol{\alpha}_t$ and $\boldsymbol{\beta}_t$, which are shown in Eqs. (3) and (4).

From Eqs. (A.9) and (B.3), it can be seen that Lemma 2 holds. ■

APPENDIX C

C.1. Two kinds of Generalization of the Methodology

In this appendix, we discuss on two kinds of generalization ofour proposed methods: (i) Generalization to probabilistic models with hierarchicallatent variables, and (ii) generalization to the case where latent variables take continuous values.

C.2. General Models of Hierarchical Hidden Variables

In the model which we employ, two types of latent variables are included in theprobabilistic model: One is that indicating which behavioral pattern appears while the other is indicatingwhich state appears.We may call the former a *global latent variable* and the latter a *local latent variable*.

A probabilistic model including such two latent variables may be written in the following general form:

$$p(Y) = \sum_{Z_1, Z_2} p(Z_1)p(Z_2|Z_1)p(Y|Z_1, Z_2). \quad \text{(C.1)}$$

Here, Y, Z_1 and Z_2 represent observed value, the global latent variable and the local latent variable, respectively. Y, Z_1 and Z_2 correspond to $\mathbf{y}$ (a *session* consisting of T_t components), a cluster index i, a hidden state x in the case that we employ an HMM mixture, respectively.

Let $\mathcal{Y}$, $\mathcal{Z}_1$ and $\mathcal{Z}_2$ be the rangesof Y, Z_1, andZ_2, respectively.In general, $\mathcal{Z}_1$ must be finite while $\mathcal{Z}_2$ can be either continuous or infinitely countableso that our framework works.There is no condition on $\mathcal{Y}$. Same as the case of HMM (our proposed methods), we consider that the probability distribution is updated at each time step. We represent the distributions at time t using an index t, such as $p_t(Y)$, $p_t(Z_1)$, $p_t(Z_2|Z_1)$, and $p_t(Y|Z_1, Z_2)$.

The model variation of p_t is derived as follows:

$$\begin{aligned} D(p_{t-1}||p_t) &= \sum_{Z_1\in\mathcal{Z}_1}\sum_{Z_2\in\mathcal{Z}_2}\sum_{Y\in\mathcal{Y}} p_{t-1}(Z_1)p_{t-1} \\ &\times(Z_2|Z_1)p_{t-1}(Y|Z_1, Z_2)\log\frac{p_{t-1}(Z_1)p_{t-1}(Z_2|Z_1)p_{t-1}(Y|Z_1,Z_2)}{p_t(Z_1)p_t(Z_2|Z_1)p_t(Y|Z_1,Z_2)} \\ &= \sum_{Z_1\in\mathcal{Z}_1} p_{t-1}(Z_1)\left\{\log\frac{p_{t-1}(Z_1)}{p_t(Z_1)} + \sum_{Z_2\in\mathcal{Z}_2} p_{t-1}(Z_2|Z_1)\right. \\ &\log\frac{p_{t-1}(Z_2|Z_1)}{p_t(Z_2|Z_1)} + \sum_{Z_2\in\mathcal{Z}_2} p_{t-1}(Z_2|Z_1) \\ &\left.\sum_{Y\in\mathcal{Y}_2} p_{t-1}(Y|Z_1, Z_2)\log\frac{p_{t-1}(Y|Z_1,Z_2)}{p_t(Y|Z_1,Z_2)}\right\}. \end{aligned} \quad \text{(C.2)}$$

Let us define model variarition vectors, $\boldsymbol{\alpha}_t$, $\boldsymbol{\beta}_t$, and $\mathbf{s}_t$ as vectors having the following components:

$$\begin{aligned} (\boldsymbol{\alpha}_t)_i &\stackrel{\text{def}}{=} p_{t-1}(Z_1)\left\{\log\frac{p_{t-1}(Z_1)}{p_t(Z_1)}\right. \\ &\left.+\sum_{Z_2\in\mathcal{Z}_2} p_{t-1}(Z_2|Z_1)\log\frac{p_{t-1}(Z_2|Z_1)}{p_t(Z_2|Z_1)}\right\}\Big|_{Z_1=i} \\ &+p_t(Z_1)\left\{\log\frac{p_t(Z_1)}{p_{t-1}(Z_1)} + \sum_{Z_2\in\mathcal{Z}_2} p_t(Z_2|Z_1)\right. \\ &\left.\times\log\frac{p_t(Z_2|Z_1)}{p_{t-1}(Z_2|Z_1)}\right\}|_{Z_1=i}, \end{aligned} \quad \text{(C.3)}$$

$$\begin{aligned} (\boldsymbol{\beta}_t)_i &\stackrel{\text{def}}{=} p_{t-1}(Z_1)\sum_{Z_2\in\mathcal{Z}_2} p_{t-1}(Z_2|Z_1) \\ &\times\sum_{Y\in\mathcal{Y}} p_{t-1}(Y|Z_1, Z_2)\log\frac{p_{t-1}(Y|Z_1,Z_2)}{p_t(Y|Z_1,Z_2)}\Big|_{Z_1=i} \\ &+p_t(Z_1)\sum_{Z_2\in\mathcal{Z}_2} p_t(Z_2|Z_1)\sum_{Y\in\mathcal{Y}} p_t(Y|Z_1, Z_2) \\ &\log\frac{p_t(Y|Z_1,Z_2)}{p_{t-1}(Y|Z_1,Z_2)}\Big|_{Z_1=i}, \end{aligned} \quad \text{(C.4)}$$

$$(\mathbf{s}_t)_i \stackrel{\text{def}}{=} (\boldsymbol{\alpha}_t)_i + (\boldsymbol{\beta}_t)_i. \quad \text{(C.5)}$$

Then, same as Theorem 1 in the case of HMM, the symmetrized model variation is decomposed as follows:

$$D(p_{t-1}||p_t) + D(p_t||p_{t-1}) = \sum_{i\in\mathcal{Z}_1}(\boldsymbol{\alpha}_t)_i + (\boldsymbol{\beta}_t)_i. \quad \text{(C.6)}$$

In addition, from Eqs. (C.3) and (C.4), we see that $\sum_i(\boldsymbol{\alpha})_i$ represents how significantly the overall latent state transition pattern has changedand $\sum_i(\boldsymbol{\beta})_i$ represents how significantly observed values' generation patterns from fixed latent states have changed. This is also same as the case of HMM.

Therefore, our proposed methods can be conducted withthe probabilistic model defined as Eq. (C.1). After deriving the model variation vectors, we can calculate anomaly scores by conducting change-point detection from the vectors' time series.

C.3. Generalized State Space Model

As an example of the generalized model as in the previous section,we may consider the case where the probabilistic model takes the form of a mixture of generalized state space models (GSSMs) [20].Then the model $p_t(Y)$ of

Table 4. Difference between a hidden Markov model (HMM) mixture and a generalized state space model (GSSM) mixture for conducting latent anomaly detection.

	HMM	GSSM
Hidden state	x (1-dimensional and finite discrete)	$\mathbf{x}$ (m-dimensional and continuous)
Observed value	y (1-dimensional and finite discrete)	$\mathbf{y}$ (m-dimensional and continuous)
Input (a session)	$\mathbf{y}_t = (y_{t,1}, \cdots, y_{t,T})$ (T-dimensional vector)	$Y_t = (\mathbf{y}_{t,1}, \cdots, \mathbf{y}_{t,T_t})(m \times T$ matrix)
Transition matrix	$a(x\|x^{'})$ (finite dimensional matrix)	$a(\mathbf{x}\|\mathbf{x}^{'})$ (two-variable distribution)
Generation of observed values	$b(y\|x)$(finite dimensional matrix)	$b(\mathbf{y}\|\mathbf{x})$ (two-variable distribution)
Stationary states	$r(x)$ (eigenvector)	$r(\mathbf{x})$ (eigenfunction)
Summation	$\prod_{j=1}^{T} \sum_{x_j, y_j}$	$\prod_{j=1}^{T} \int d\mathbf{x}_j d\mathbf{y}_j$
Assumption 1	Existence of a stationary state	Existence of a stationary state
Assumption 2	Smallness of clusters' overlaps	Smallness of clusters' overlaps

a session Y at time step t is represented as follows:

$$p_t(Y) = \sum_{i=1}^{K} \pi_{i,t} p_{i,t}(Y), \tag{C.7}$$

where for each $i \in \{1, \cdots, K\}$, $\pi_{i,t}$ denotes the mixture coefficient such that $\sum_i \pi_{i,t} = 1$ and $\pi_{i,t} > 0$, and $p_{i,t}(Y)$ denotes the i-th component of the form of a GSSM; i.e. let $\mathbf{x}$ be an $m-$dimensionalcontinuous hidden state, let $a_{i,t}(\mathbf{x}|\mathbf{x}')$ be a state transition function at each index i and time$t$, let $b_{i,t}(\mathbf{y}|\mathbf{x})$ be a probability that a state $\mathbf{x}$ generates an observed value $\mathbf{y}$, let $\gamma_{i,t}(\mathbf{x})$ be an initial probability distribution of $\mathbf{x}$. Then $p_{i,t}(Y)$ is represented as follows:

$$p_{i,t}(Y) = \int \mathrm{d}\mathbf{x}_1 \cdots \mathrm{d}\mathbf{x}_{T_t}\ \gamma_{i,t}(\mathbf{x}_1) \prod_{j=1}^{T_t - 1} a_{i,t}(\mathbf{x}_{j+1}|\mathbf{x}_j) \times \prod_{j=1}^{T_t} b_{i,t}(\mathbf{y}_j|\mathbf{x}_j). \tag{C.8}$$

A GSSM mixture is a kind of generalization of an HMM mixture to the case where the state variable is continuous. Table 4 summarizes the differences between an HMM mixture and a GSSM mixture for conducting latent anomaly detection. In Table 4, we denote the eigenfunction of the transition function a of a GSSM corresponding to the eigenvalue 1 as r:

$$\int \mathrm{d}\mathbf{x}^{'}\ a_{i,t}(\mathbf{x}|\mathbf{x}^{'}) r_{i,t}(\mathbf{x}^{'}) = r_{i,t}(\mathbf{x}). \tag{C.9}$$

It is easy to check that Lemmas 1 and 2 hold for this-generalized model. Thus, under the same assumptions as Assumptions 1 and 2, Theorem 1 holds (we can decompose the model variation into model variation vectors) as in the case where the state variable is discrete.

In GSSM case, model variation vectors, $\boldsymbol{\alpha}_t$, $\boldsymbol{\beta}_t$ and $\mathbf{s}_t$ are defined as follows:

$$(\boldsymbol{\alpha}_t)_i \stackrel{\text{def}}{=} \int d\mathbf{x} \left\{ \pi_{i,t-1} r_{i,t-1}(\mathbf{x}) D(a_{i,t-1}(\cdot|\mathbf{x}) || a_{i,t}(\cdot|\mathbf{x})) + \pi_{i,t} r_{i,t}(\mathbf{x}) D(a_{i,t}(\cdot|\mathbf{x}) || a_{i,t-1}(\cdot|\mathbf{x})) \right\}, \tag{C.10}$$

$$(\boldsymbol{\beta}_t)_i \stackrel{\text{def}}{=} \int d\mathbf{x} \left\{ \pi_{i,t-1} r_{i,t-1}(\mathbf{x}) D(b_{i,t-1}(\cdot|\mathbf{x}) || b_{i,t}(\cdot|\mathbf{x})) + \pi_{i,t} r_{i,t}(\mathbf{x}) D(b_{i,t}(\cdot|\mathbf{x}) || b_{i,t-1}(\cdot|\mathbf{x})) \right\}, \tag{C.11}$$

$$(\mathbf{s}_t)_i \stackrel{\text{def}}{=} (\boldsymbol{\alpha}_t)_i + (\boldsymbol{\beta}_t)_i, \tag{C.12}$$

$$D(a_{i,t-1}(\cdot|\mathbf{x}) || a_{i,t}(\cdot|\mathbf{x})) = \int d\mathbf{x}' a_{i,t-1}(\mathbf{x}'|\mathbf{x}) \times \log \frac{a_{i,t-1}(\mathbf{x}'|\mathbf{x})}{a_{i,t}(\mathbf{x}'|\mathbf{x})}, \tag{C.13}$$

$$D(b_{i,t-1}(\cdot|\mathbf{x}) || b_{i,t}(\cdot|\mathbf{x})) = \int d\mathbf{y} b_{i,t-1}(\mathbf{y}|\mathbf{x}) \times \log \frac{b_{i,t-1}(\mathbf{y}|\mathbf{x})}{b_{i,t}(\mathbf{y}|\mathbf{x})}. \tag{C.14}$$

After deriving them, we can calculate anomaly scores by conducting change-point detection from the vectors' time series.

REFERENCES

[1] M. Schonlau, W. DuMouchel, W. H. Ju, A. F. Karr, M. Theus, and Y. Vardi, Computer intrusion: Detecting masquerades, Stat Sci 16(1) (2001), 58–74.

[2] S. Forrest, S. A. Hofmeyr, A. Somayaji, and T. A. Longstaff, A sense of self for Unix processes, in Proceedings of the 1996 IEEE ISRSP, Oakland, 1996.

[3] S. Hofmeyr, S. Forrest, and A. Somayaji, Intrusion detection using sequences of system calls, J Comput Secur 6(3) (1998), 151–180.

[4] S. Bay and M. Schwabache, Mining distance-based outlier in near linear time with randomization and a simple pruning rule, in Proceedings of Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD2003), Washington, DC, 2003.

[5] K. Yamanishi, J. Takeuchi, G. Williams, and P. Milne, On-line unsupervised oultlier detection using finite mixtures with discounting learning algorithms, in Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD2000), Boston, 2000.

[6] T. Fawcett and F. Provost, Noticing intereating changes in behavior, in Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD1999), San Diego, 1999.

[7] K. Yamanishi and J. Takeuchi, Unifying framework for detecting outliers and change points from non-stationary time series data, Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD2002), Edmonton, 2002.

[8] X. Song, M. Wu, C. Jermaine, and S. Ranka, Statistical change detection for multi-dimensional data, in Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD2007), San Jose, 2007.

[9] K. Yamanishi and Y. Maruyama, Dynamic Model Selection with its applications to novelty detection, in IEEE Transactions on Information Theory, Vol. 56(6) (2006), 2180–2189.

[10] K. Yamanishi and Y. Maruyama, Dynamic syslog mining for network failure monitoring, in Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD2005), Chicago, 2005, 499–508.

[11] R. K. Sahoo, A. J. Oliner, I. Rish, M. Gupta, J. E. Moreira, S. Ma, R. Vilalta, and A. Sivasubramaniam, Critical event prediction for proactive management in large-scale computer clusters, in Proceedings of Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD2003), Washington, DC, 2003.

[12] Risto Vaarandi, A data clustering algorithm for mining patterns from event logs, in Proceedings of IEEE IPOM2002, Kansas City, 2002.

[13] R. Agrawal and R. Srikant, Mining sequential patterns, in Proceedings of the Eleventh International Conference on Data Engineering(ICDE95), Taipei, 1995.

[14] P. Smyth, Markov monitoring with unknown states, in IEEE Journal on Selected Areas in Communications (JSAC), Vol. 12, Special Issue on Intelligent Signal Processing for Communications, 1994, 1600–1612.

[15] S. Forrest, C. Warrender, and B. Pearlmutter, Detecting intrusions using system calls: Alternate data models, in Proceedings of the 1999 IEEE ISRSP, Oakland, 1999.

[16] B Gao, H. Y. Ma, and Y. H. Yang, HMMs (Hidden Markov models) based on anomaly intrusion detection method, in Proceedings of International Conference on Machine Learning and Cybernetics 2002, Beijing, 2002.

[17] X. Wang, C. Zhai, X. Hu, and R. Sproat, Mining correlated bursty topic patterns from coordinated text streams, in Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD2007), San Jose, 2007.

[18] R. A. Maxion and T. N. Townsend. Masquerade detection using truncated command lines, in Proceedings of International Conference on Dependable Systems and Networks, Washington, DC, 2002, 219–228.

[19] Y. Matsunaga and K. Yamanishi, An Information-theoretic approach to detecting anomalous behaviors, in Proceedings of the Second Forum on Information Technologies (FIT2003), Ebetsu, Japan, 2003.

[20] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, in IEE Proceedings F, Radar and signal processing, Vol. 140(2), 1993, 107–113.