Echo-State Conditional Variational Autoencoder for Anomaly Detection

Suwon Suh*, Daniel H. Chae†, Hyon-Goo Kang‡, Seungjin Choi§
*§Department of Computer Science and Engineering
Pohang University of Science and Technology
77 Cheongam-ro, Nam-gu, Pohang 37673, Korea
Email: *caster@postech.ac.kr, §seungjin@postech.ac.kr
†‡Access Network Lab, SK Telecom, Korea
Email: †dani75.chae@sk.com, ‡hyongoo.kang@sk.com

***Abstract*—Anomaly detection involves identifying the events which do not conform to an expected pattern in data. A common approach to anomaly detection is to identify outliers in a latent space learned from data. For instance, PCA has been successfully used for anomaly detection. Variational autoencoder (VAE) is a recently-developed deep generative model which has established itself as a powerful method for learning representation from data in a nonlinear way. However, the VAE does not take the temporal dependence in data into account, so it limits its applicability to time series. In this paper we combine the echo-state network, which is a simple training method for recurrent networks, with the VAE, in order to learn representation from multivariate time series data. We present an *echo-state conditional variational autoencoder* (ES-CVAE) and demonstrate its useful behavior in the task of anomaly detection in multivariate time series data.**

I. Introduction

Anomaly detection has been researched in many fields from engineering [1] to astronomy [2]. It is closely related to predictive maintenance in engineering and novelty detection to find new type of galaxy in astronomy. Because it has been tackled in so many fields, it is hard to get what the anomaly detection is really about. The good starting point is following the basic definition and a point of view in the survey paper [3] , which categorized lots of literatures in a single point of view.

According to the survey, there are three kinds of anomalies: First, a point anomaly is a data point that can not be assigned to any other existing groups. If there are multiple data points gather and they can not be assigned to the existing groups, they are in a group anomaly. Second, a contextual anomaly is anomalous based on contextual information. For example, in timeseries, a contextual anomaly may not be a point anomaly but it is anomalous based on the previous or the next values to come. Third, a data point in collective anomalies is neither a point anomaly nor a contextual anomaly. However, the data points are collectively anomalous as a group.

In this paper, we adopt probabilistic framework to detect anomalies especially contextual anomalies in timeseries. As we can tackle point anomaly detection with density estimator, we tackle contextual anomaly detection with a conditional density estimator of the current observation given history.

This model is developed to address the anomaly detection problem with vast amounts[1] of real-time event logs from a nation-wide telecommunication company. Because there is serious restriction on computational resources to handle all the event logs, we need to learn this model online. For this reason, we propose simple model to embrace this constraint. *Stochastic Gradient Variational Bayes (SGVB)* is a type of doubly stochastic inference algorithm. It approximates the gradient of a utility function not only by sampling a set of minibatch from the entire training set but also by sampling posterior distribution to approximate variational lower bound of the utility function, which makes this algorithm suitable to so called *Big Data Analysis* (high volume, high velocity, high variety information).

We formulate the detection of contextual anomaly as a learning conditional density estimator and introducing a simple threshold based classifier. To model the conditional density estimator, we propose *Echo-State Conditional Variational Auto Encoder (ES-CVAE)*. In experiment, our algorithm shows competitive performance compared to baselines and it can be easily expanded to multivariate timeseries. Moreover, it can be easily implemented in distributed environment.

II. Related Work

Variational Recurrent Neural Network (VRNN) (dynamic prior) [4] and *Stochastic Recurrent Network (STORN)* (static prior) [5] are closely related models where a state for a history are updated by using *Recurrent Neural Network (RNN)* or *Long Short Term Memory (LSTM)*. Such an update needs not only observation at time t, $\boldsymbol{x}_t$, but also a latent state at time t, $\boldsymbol{z}_t$, which makes these state must be updated whenever the new parameters are learned, which makes it hard to use these algorithm in online learning. Recently, Sohn at al propose *Conditional Variational Auto Encoder (CVAE)* to predict structured output [6], in our model, we extends this idea to handle sequential data as well as input data per each time slot.

[1]A summary of logs are recorded in every 10 second per broadcast station over the country.

978-1-5090-0620-5/16/$31.00

In the following subsections, we briefly review *Variational Autoencoder (VAE)* [7] and *Echo State Network (ESN)* [8] to explain the proposed model.

A. Variational Auto Encoder

VAE is a recently developed Deep Generative Model (DGM), which conveys an idea that only the top layer of deep structure has randomness, of which is modeled with a factorized distribution as multivariate normal distribution with an isotropic covariance matrix. For this reason, we can easily get independent samples from VAE, which is not possible for other deep generative models such as Deep Belief Networks (DBNs) [9] and Deep Boltzmann Machines (DBMs) [10], where we must wait until Markov chains converges with Gibb sampling. VAE has the following joint probability distribution with observation variable $\boldsymbol{x} \in \mathbb{R}^D$ and latent variable $\boldsymbol{z} \in \mathbb{R}^K$:

$$\begin{aligned} p(\boldsymbol{x}, \boldsymbol{z}) &= p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}) \ , \\ p(\boldsymbol{z}) &= \mathcal{N}(0, \boldsymbol{I}) \ , \\ p(\boldsymbol{x}|\boldsymbol{z}) &= \mathcal{N}(\boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma}^2)) \ , \end{aligned}$$

where D is the dimension of an observation, K is the dimension of a latent variable, $\boldsymbol{\mu} \triangleq MLP_\mu(\boldsymbol{z})$ denotes the mean of the conditional distribution, $\boldsymbol{\sigma}^2 \triangleq MLP_\sigma(\boldsymbol{z})$ denotes D dimensional variance vector over the dimension D respectively, $MLP(\cdot)$ denotes *multi-layer perceptron (MLP)* and the MLPs used to generate parameters of the conditional distribution, $p(\boldsymbol{x}|\boldsymbol{z})$, are called as *generative networks* as a group.

In fact, the above mentioned probabilistic model with the generative networks was proposed in 1999 by Mackay [11], where he named it as density networks. The only difference between VAE and density networks is the way to infer the latent variable in the probabilistic model. In density network, Mackay used Gibbs sampler to infer the continuous latent variable, which made it hard to use in case that latent variable is in high dimensional space. In VAE, Kingma showed a efficient way to infer the continuous latent variable with reparameterization trick using recognition networks, which consists of MLPs generating parameters of variational posterior distribution, $q(\boldsymbol{z}|\boldsymbol{x})$.

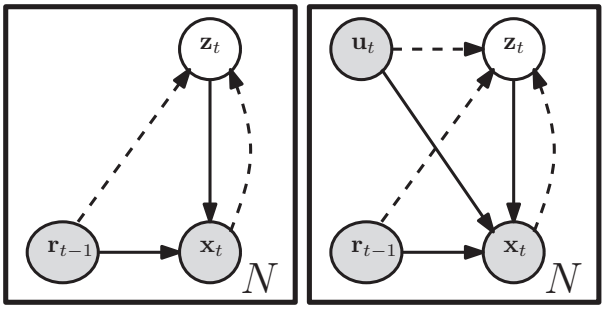
B. Echo State Networks

ESN is a type of RNN where only parameters for outputs are learned, which results in careful parameter initialization of the other parameters in the recurrent loop:

$$\boldsymbol{r}_t = \alpha \boldsymbol{r}_{t-1} + (1-\alpha)\tanh(\boldsymbol{A}\boldsymbol{r}_{t-1} + \boldsymbol{B}[1; (\boldsymbol{\Lambda}^{obs}\boldsymbol{x}_t)^\top]^\top) \ ,$$

where $\boldsymbol{r}_t$ denotes echo state at time t and $\boldsymbol{A}$ state transition matrix that is initialized in a way that the largest absolute eigen value of sparse matrix $\boldsymbol{A}$ is less than 1 for Lyapunov stability, which ensure that the influence of echo state at time t fades away after series of updates. Moreover, $\boldsymbol{B}$ is a dense matrix, α denotes leaking rate of echo state and $\boldsymbol{\Lambda}^{obs}$ denotes a diagonal matrix of which entries are inverse variance of observation variable $\boldsymbol{x}$. We follow the practical guideline for the parameter setting of ESN [12]. The parameters $\boldsymbol{A}$ and $\boldsymbol{B}$ are not updated after initialization. Only parameters that interpret the echo state are learned, which help us to get a concise learning algorithm for online learning.

III. Model



(a) ES-CVAE. (b) ES-CVAE with input.

Fig. 1: ES-CVAE and ES-CVAE with input

We want to model the sequential data $\mathcal{X} = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \cdots, \boldsymbol{x}^{(t)}\}$ and corresponding inputs $\mathcal{I} = \{\boldsymbol{u}^{(1)}, \boldsymbol{u}^{(2)}, ..., \boldsymbol{u}^{(t)}\}$, where $\boldsymbol{x} \in \mathbb{R}^D$ and $\boldsymbol{u} \in \mathbb{R}^G$. For ease of understanding, we omit the inputs hereafter. However, it is straightforward to include inputs in the model. Now we have the following joint probability density function (PDF) for the sequential data :

$$p(\boldsymbol{x}^{(1:t)}) = p(\boldsymbol{x}^{(1)}) \prod_{i=2}^{t} p(\boldsymbol{x}^{(i)}|\boldsymbol{x}^{(1:i-1)}) \ , \tag{1}$$

where $\boldsymbol{x}^{(1:t)}$ denotes a short-hand notation for $\{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, ..., \boldsymbol{x}^{(t)}\}$. Instead of using $\boldsymbol{x}^{(1:t)}$ as variable length history variables, we introduce echo state, $\boldsymbol{r}^{(t)}$, to maintain history as fixed length. As a result, we can rewrite Eq. 1 as follows:

$$p(\boldsymbol{x}^{(1:t)}) = p(\boldsymbol{x}^{(1)}) \prod_{i=2}^{t} p(\boldsymbol{x}^{(i)}|\boldsymbol{r}^{(i-1)}) \ , \tag{2}$$

where we set $p(\boldsymbol{x}^{(1)})$ as $\mathcal{N}(0, \sigma_{init}^2 \boldsymbol{I})$ and σ_{init}^2 has a large positive value. As a next step, we introduce the following parametric model for $p(\boldsymbol{x}^{(i)}|\boldsymbol{r}^{(i-1)})$:

$$p(\boldsymbol{x}^{(i)}|\boldsymbol{r}^{(i-1)}) = \int p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}, \boldsymbol{r}^{(i-1)}) p(\boldsymbol{z}^{(i)}|\boldsymbol{r}^{(i-1)}) d\boldsymbol{z}^{(i)} \ ,$$

where $\boldsymbol{z}^{(i)} \in \mathbb{R}^K$ is the latent variable at time i, $\boldsymbol{r}^{(i)} \in \mathbb{R}^R$ denotes the history variable at time i, $p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}, \boldsymbol{r}^{(i-1)}) \triangleq \mathcal{N}(\boldsymbol{\mu}^{(i)}, \mathrm{diag}(\boldsymbol{\sigma}^{2(i)}))$, $p(\boldsymbol{z}^{(i)}|\boldsymbol{r}^{(i-1)}) \triangleq \mathcal{N}(\boldsymbol{\psi}^{(i)}, \boldsymbol{I})$. $\boldsymbol{\mu}^{(i)} \triangleq MLP_\mu(\boldsymbol{z}^{(i)}, \boldsymbol{r}^{(i-1)})$, $\log \boldsymbol{\sigma}^{(i)} \triangleq MLP_\sigma(\boldsymbol{z}^{(i)}, \boldsymbol{r}^{(i-1)})$ and $\boldsymbol{\psi}^{(i)} \triangleq MLP_\psi(\boldsymbol{r}^{(i-1)})$. You can find graphical representation of the model in Fig. 1, where solid lines indicate the generative networks and dashed lines denote the recognition networks for the variational posterior distribution.

Now, let us introduce variational approach for the conditional log-likelihood of $\boldsymbol{x}^{(i)}$ given $\boldsymbol{r}^{(i-1)}$ as follows:

$$\begin{aligned}
&\log p(\boldsymbol{x}^{(i)}|\boldsymbol{r}^{(i-1)}) \\
&\quad \geq \mathbb{E}_{q(z^{(i)}|x^{(i)},r^{(i-1)})}\Big[\log p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)},\boldsymbol{r}^{(i-1)})\Big] \\
&\quad -KL\Big[q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)},\boldsymbol{r}^{(i-1)})||p(\boldsymbol{z}^{(i)}|\boldsymbol{r}^{(i-1)})\Big] \qquad (3)\\
&\quad = \mathcal{F}^{(i)} \ ,
\end{aligned}$$

where $q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)},\boldsymbol{r}^{(i-1)}) \triangleq \mathcal{N}(\boldsymbol{\mu}^{(i)}_{latent}, \mathrm{diag}(\boldsymbol{\sigma}^{2(i)}_{latent}))$ denotes the variational posterior distribution at time i and the recognition networks at time i consists of two MLPs: $(1)\boldsymbol{\mu}^{(i)}_{latent} \triangleq MLP_{l\mu}(\boldsymbol{x}_i, \boldsymbol{r}_{i-1})$, $(2)\log\boldsymbol{\sigma}^{(i)}_{latent} \triangleq MLP_{l\sigma}(\boldsymbol{x}_i, \boldsymbol{r}_{i-1})$. In addition, $\mathcal{F}^{(i)}$ denotes the variational lower bound of the conditional log-likelihood, which will be tight when the variational posterior distribution $q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)},\boldsymbol{r}^{(i-1)})$ is the true posterior distribution, $p(\boldsymbol{z}^{(i)}|\boldsymbol{r}^{(i-1)})$.

Following the approach in [7], the KL divergence term in the Eq. 3 has the closed form as follows:

$$\begin{aligned}
&KL\Big[q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)},\boldsymbol{r}^{(i-1)})||p(\boldsymbol{z}^{(i)}|\boldsymbol{r}^{(i-1)})\Big] \qquad (4)\\
&\quad = -\frac{1}{2}\sum_{k=1}^{K}(1+\log\sigma^{2(i)}_{latent,k} - (\psi^{(i)}_k - \mu^{(i)}_{latent,k})^2 - \sigma^{2(i)}_{latent,k}) \ .
\end{aligned}$$

To sum up, we can rewrite the variational lower bound in Eq. 3 using Eq. 4 and the reparameterization trick as follows:

$$\begin{aligned}
\mathcal{F}^{(i)} &= \mathbb{E}_{q(z^{(i)}|x^{(i)},h^{(i-1)})}\Big[\log p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)},\boldsymbol{r}^{(i-1)})\Big] \\
&+ \frac{1}{2}\sum_{k=1}^{K}(1+\log\sigma^{2(i)}_{latent,k} - (\psi^{(i)}_k - \mu^{(i)}_{latent,k})^2 - \sigma^{2(i)}_{latent,k}) \\
&= \mathbb{E}_{p(\boldsymbol{\epsilon})}\Big[\log p(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}^{(i)}_{latent} + \sigma^{(i)}_{latent}\odot\boldsymbol{\epsilon},\boldsymbol{r}^{(i-1)})\Big] \\
&+ \frac{1}{2}\sum_{k=1}^{K}(1+\log\sigma^{2(i)}_{latent,k} - (\psi^{(i)}_k - \mu^{(i)}_{latent,k})^2 - \sigma^{2(i)}_{latent,k}) \\
&\approx -\frac{1}{2}(D\log 2\pi + \sum_{d=1}^{D}\log\sigma^{2(i)}_d \\
&+ ((\boldsymbol{x}^{(i)} - \boldsymbol{\mu}^{(i)})\odot\boldsymbol{\sigma}^{-(i)})^T((\boldsymbol{x}^{(i)} - \boldsymbol{\mu}^{(i)})\odot\boldsymbol{\sigma}^{-(i)})) \\
&+ \frac{1}{2}\sum_{k=1}^{K}(1+\log\sigma^{2(i)}_{latent,k} - (\psi^{(i)}_k - \mu^{(i)}_{latent,k})^2 - \sigma^{2(i)}_{latent,k}) \ ,
\end{aligned}$$

where $\boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ denotes a sample from standard Gaussian distribution for ith observation when we approximate variational lower bound with only one sample per observation.

For parameter estimation of the model, let $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ be a set of parameters of MLPs for generative networks and for recognition networks respectively. Additionally, we assume that we are at time $(N+1)$. Then we have the following optimization problem with N observation (We omit the first observation in estimating model parameters):

$$[\boldsymbol{\theta},\boldsymbol{\phi}] \quad = \quad \arg\max_{\boldsymbol{\theta},\boldsymbol{\phi}} \sum_{i=2}^{N+1}\mathcal{F}^{(i)}.$$

Instead of considering all the observations of size N, we adopt stochastic gradient ascent by sampling a minibatch of size M, $\mathcal{X}_M = \{\boldsymbol{x}^{(m)}\}_{m=1}^{M}$, in each Adagradient update [13].

$$[\boldsymbol{\theta},\boldsymbol{\phi}] \quad = \quad \arg\max_{\boldsymbol{\theta},\boldsymbol{\phi}} \sum_{i=2}^{N+1}\mathcal{F}^{(i)} \approx \arg\max_{\boldsymbol{\theta},\boldsymbol{\phi}} \frac{N}{M}\sum_{m=1}^{M}\mathcal{F}^{(m)} \ ,$$

where the variational lowerbound for the mth obervation in a minibatch, $\mathcal{F}^{(m)}$, is defined as follows:

$$\begin{aligned}
\mathcal{F}^{(m)} = \Big(&-\frac{1}{2}(D\log 2\pi + \sum_{d=1}^{D}\log\sigma^{2(m)}_d \\
&+ ((\boldsymbol{x}^{(m)} - \boldsymbol{\mu}^{(m)})\odot\boldsymbol{\sigma}^{-(m)})^T((\boldsymbol{x}^{(m)} - \boldsymbol{\mu}^{(m)})\odot\boldsymbol{\sigma}^{-(m)})) \\
&+ \frac{1}{2}\sum_{k=1}^{K}(1+\log\sigma^{2(i)}_{latent,k} - (\psi^{(i)}_k - \mu^{(i)}_{latent,k})^2 - \sigma^{2(i)}_{latent,k})\Big) \ .
\end{aligned}$$

Until this point, we adopt approximation twice to approximate evidence lower bound (ELBO) of observations of size N; First, we approximate the expectation in ELBO with regard to variational posterior distribution using sampling with reparameterization trick. Second, we approximate the ELBO of N observations with ELBO of a randomly sampled minibatch of size M. For this reason, this approach is named as Stochastic Gradient Variational Bayes (SGVB) [7].

A. Online Parameter Estimation

In stochastic variational inference (SVI) [14], batches are collected along with stream and local sufficient statistics are exactly calculated and global sufficient statistics are approximated by local sufficient statistics and parameters are updated as moving average. Because sufficient statistics are not dealt in SGVB, we uses more direct manner to maintain slide window along with data stream and learn parameters by sampling on the window periodically.

For online learning of ES-CVAE, we first initialize parameters for ESN and updates echo states along with the sequential data $\mathcal{X}$. These parameters are fixed after the first update, which is valuable property of ESN for online learning. Second, we apply stochastic gradient updates ξ times whenever new observation $\boldsymbol{x}^{(i)}$ comes where each minibatch of size M is chosen from the most recently received N observations.

IV. Anomaly Detection

What is Anomaly? Anomaly is an observation that doesn't conform with prediction, which is highly subjective for operators or analysts who use this model as a tool to detect anomalies. For this reason, we first generate an anomaly score by applying the current observation to the learned predictive model and decide the current observation as an anomaly only if its anomaly score exceeds a certain threshold, which is the parameter that reflects various level of sensitivity of operators.

As mentioned, we first predict the current observation with conditional probability density function [2] in Eq. 3. Based on the model, we define anomaly score as follows:

$$\text{Anomaly Score}(\boldsymbol{x}^{(i)}) = -\log p(\boldsymbol{x}^{(i)}|\boldsymbol{r}^{(i-1)}) \ . \tag{5}$$

Second, let $s^{(i)} \in \{0, 1\}$ be the anomaly indication variable for time i, we can derive the anomaly classifier as follows:

$$s^{(i)} = I(\text{Anomaly Score}(\boldsymbol{y}^{(i)}) > \tau) \ , \tag{6}$$

where τ denotes real valued threshold and $I(\cdot)$ denotes indication function.

V. Experiment

A. Toy Problem

To verify our algorithm, we make toy dataset shown in Fig. 3a, which is a sequential data consisting of three components: The first component is a seasonal component with the period 7. The second component is a random walk component. And, the last component is an anomaly component that follows Bernoulli distribution with very low activation rate. When it activates, it generates an anomalous value that is sampled from a Gaussian distribution with a large variance.

We initiate ESN and encode the current history along the timeline as shown in Fig. 5c. We implement our algorithm in Matlab and execute it on a machine with a CPU with 4 cores (3.10GHz) and 16 GB memory to learn the parameters of the model. It takes about 13 seconds in 800 iterations.

As a result, we get anomaly scores along with the time line in offline setting in Fig. 2a and ROC curve as shown in Fig. 2b. It shows reasonable performance.

We adopt online learning scheme with Toy problem as shown in Fig. 3. Looking closely into Fig . 5b, variance of the score has decreased along with time axis, which means ES-CVAE models the dynamics of the given timeseries well as time passes. For this reason, the classification accuracy at the last stage is higher than other parts. One of the notable property of ES-CVAE is the temporal manifold that is conditional representation of the current observation x_t given history $\boldsymbol{r}_{t-1}$. With this embedding, we can intuitively recognize how the relationship between the current observation and the history has changed over the timeseries. As shown in Fig. 5d, for i th observation, the dynamic prior $p(\boldsymbol{z}^{(i)}|\boldsymbol{r}^{(i-1)})$ is determined, which denotes a probable encoding region over the latent space at time i. Assuming that the reconstruction of ES-CVAE is high enough and there is an anomaly at time i, variational posterior or probable encoding region $q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)}, \boldsymbol{r}^{(i-1)})$ differs significantly[3].

B. Anomaly Detection in Yahoo S5 dataset

To verify the algorithm described in the previous section on a real dataset, we decide to use Yahoo S5 dataset[4] that contains 66 univariate timeseries, each of which contains about 1,423 long univariate real values labeled with anomalies [15]. We use the first 44 timeseries as training set and the next 11 timeseries as validation set to select hyper-parameters(model parameters $(K, H^5, R, \boldsymbol{A}, \boldsymbol{B})$ and threshold(τ)). Using the parameter setting that gives the best result in validation set, we get the final result from the test set that consists of 11 timeseries. We use random hyper parameter searching for 300 times. These timeseries are collected as performance measures of anonymous services in Yahoo and each timeseries have different scales. Thus, we first normalize them with mean and variance to get zero mean and unit variance timeseries.

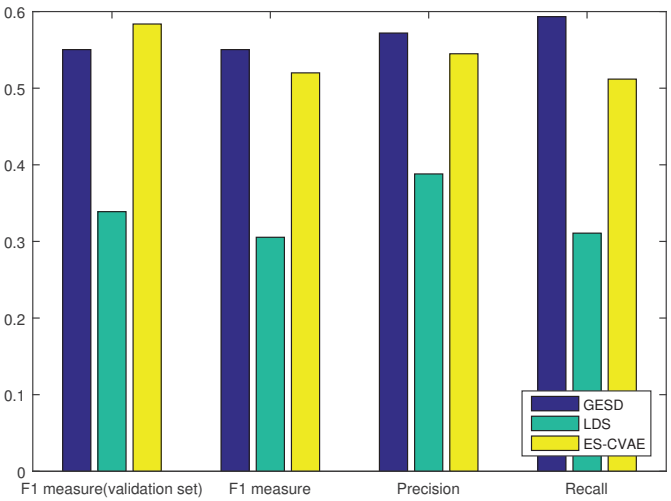


Fig. 4: Results of Anomaly Detection.

We test two algorithms as baselines; *Generalized Extreme Studentized Deviation (GESD) test* [16] [17] and *Linear Dynamic System (LDS)* [18].

In GESD test, we first apply robust local regression (or STL decomposition) to smooth a given timeseries, then we apply GESD test to draw anomalies from the residuals with optimally chosen maximum number of anomaly with validation set. In LDS, we use EM algorithm to learn the model and choose threshold τ and the dimension of latent variable by random hyperparameter search in the validation set. We choose these algorithms as baseline because GESD test represents the classical time series algorithm and LDS represents linear dynamic model.

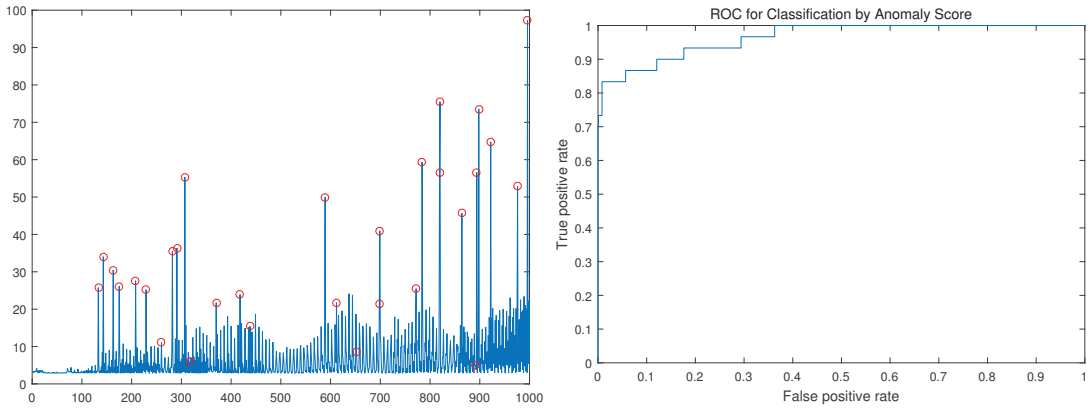
Although ES-CVAE shows the best F1-measure in the validation set as shown in Fig. 4, GESD turns out to be generalized better than ES-CVAE in the test set. Looking into the anomalies that the models miss, they are collective anomalies, which are anomaly as a collective set. This is ineveitable because the models are designed to detect contextual anomaly in the first place. To cope with this problem, we need an additional structure.

[2]Instead of calculating more accurate value with sampling method, we use variational lowerbound as approximation.

[3]You can prove this straight forwardly with Eq. 3.

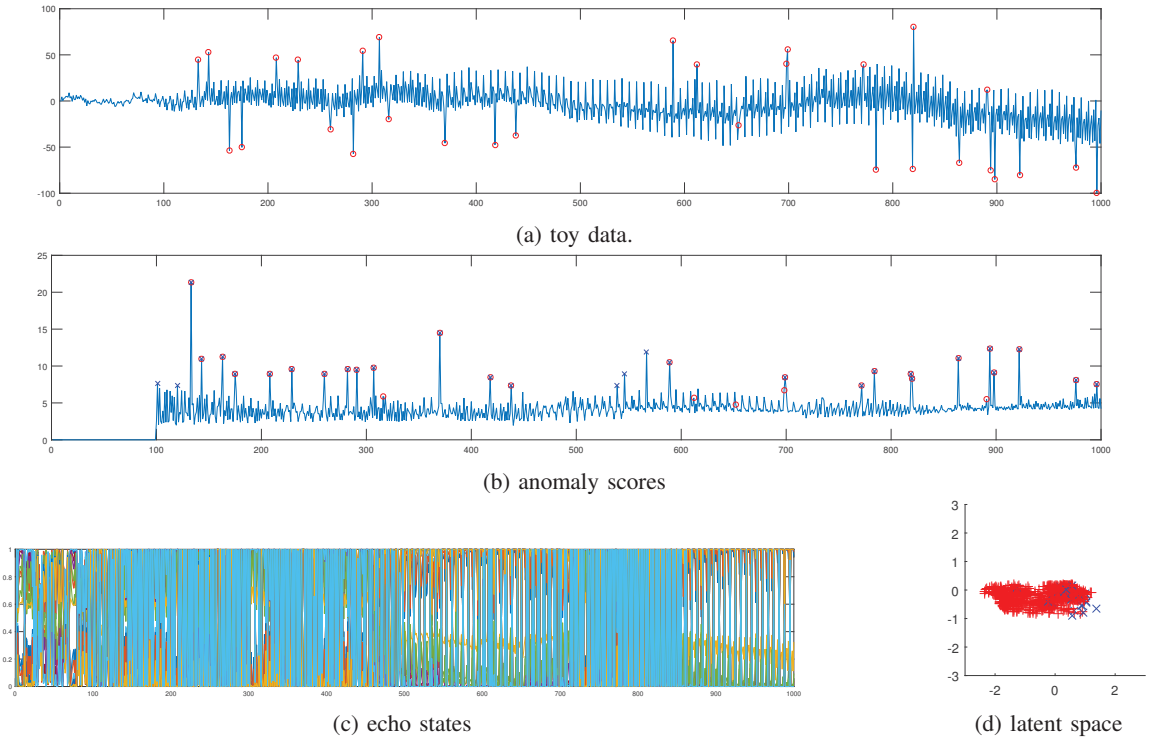
[4]we omit 45th timeseries out of the total 67 timeseries.

[5]The number of hidden units



(a) Anomaly Score with toy data. (b) ROC curve

Fig. 2: ES-CVAE(K=4, H=10, R=10, 800 iteration)



(a) toy data.

(b) anomaly scores

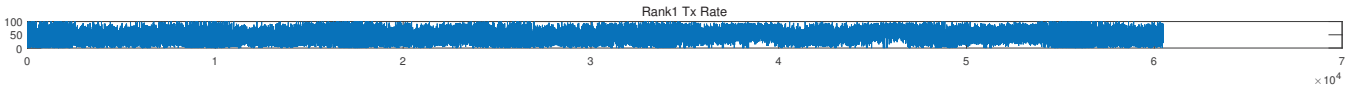
(c) echo states (d) latent space

Fig. 3: Online Learning with ES-CVAE(K=2, H=5, R=20, 50 iteration per new sample)

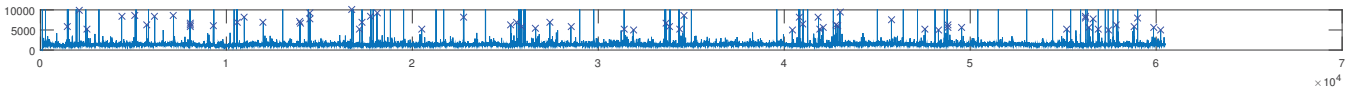
C. Anomaly Detection on Appolo dataset

In the previous subsection, we verify our model works in univariate timeseries dataset and, in this subsection, we apply our model on real Big dataset (Appolo dataset). Appolo datatset consists of multivariate (305 dimensions) time series gathered from 1391 cells scattered over a Busan city. The data had been accumulated over one month. For the proof of concept (POC), we decide to use a subset of it. We choose to use timeseries from only five cells over 7 days. As shown in Fig. 5, we can draw anomaly scores for the multivariate timeseries using ES-CVAE with minimal effort.

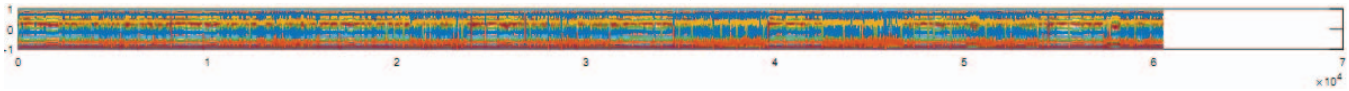
By introducing one hot encoding vectors of day of weeks and hour of day to ES-CVAE as inputs $\boldsymbol{u}$, we can directly gauge the influence of long-term seasonal dependencies. Moreover, we also include latitude and longitude of the cell to gauge geospatial influence. To consider large scale implementation, following the idea of [19], we adopt distributed architecture as shown in Fig.7 and the results of distributed learning are as shown in Fig.6. In this distributed setting, we apply asynchronous parameter updates. Each base station maintains its logs with sliding window and periodically learns parameters by getting the current model parameters from the central server. After few updates of parameters with Stochastic Gra-



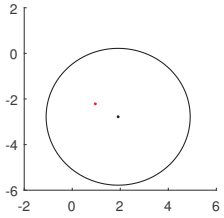
(a) One dimension of SKT dataset(Rank1 Tx Rate).



(b) anomaly scores

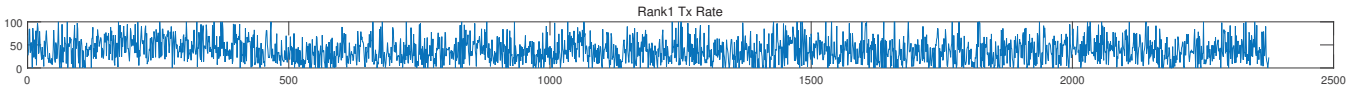


(c) echo states

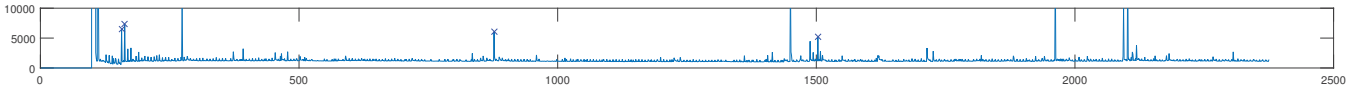


(d) Latent states with dynamic prior contour(black) and mean of variational posterior(red)

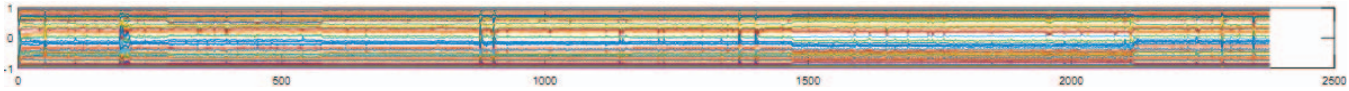
Fig. 5: Online Learning with ES-CVAE(K=10, H=100, R=100, 50 iteration per new sample) 5a) one dimension out of 305 dimensions. 5b) anomaly scores for one week in one cell.



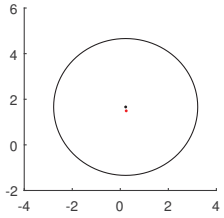
(a) One dimension of SKT dataset(Rank1 Tx Rate).



(b) anomaly scores



(c) echo states



(d) Latent states with dynamic prior contour(black) and mean of variational posterior(red)

Fig. 6: Distributed Online Learning with ES-CVAE(K=10, H=100, R=100, 50 iteration per new sample) 5a) one dimension out of 305 dimensions. 5b) anomaly scores for one week in five cells.

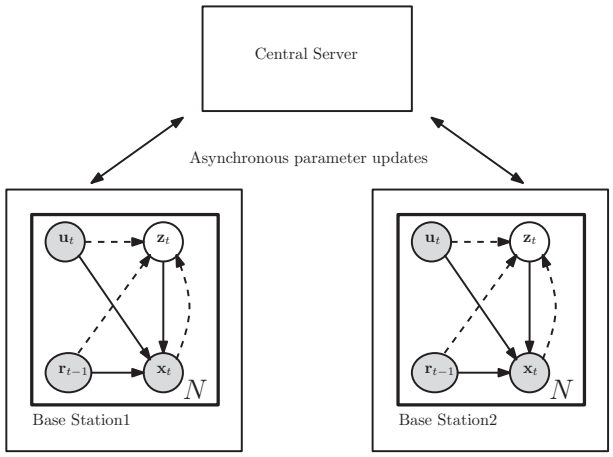


Fig. 7: POC of distributed learning algorithm of ES-CVAE.

dient Ascent (Adagradient) and asynchronously updates the current model parameters on the central server by averaging them. Unfortunately, we do not have any label to measure the performance of anomaly detector quantitatively due to the humongous size of event logs (over 360 GB). However, we tentatively say that the the distributed version works better than single cell version by comparing Fig. 5b and Fig. 6b. According the telecom company, there are very few anomalous events in this period and the anomaly score in distributed version shows a few such anomalous surges along the time line.

VI. Discussion

Although the GESD test shows the best result in the S5 dataset, ES-CVAE has many advantages over GESD test. First, it is based on the probabilistic framework, which enables us to extend the model to consider other metadata or to introduce an additional structure to embrace other problems at the same time. For examples, we can add an additional structure to detect collective anomalies. Second, we can model multivariate time series with ES-CVAE without modification of the existing algorithm. Third, multi-variate ES-CVAE gives dimension reduction as well, which enable us to visualize a high-dimensional observation with lower dimensional space. Fourth, as a generative model ES-CVAE can be used as a simulation tool and we can explain why it is perceived as a anomaly by comparing dynamic prior (what the model expect) and variational posterior distribution (what the model actually get). Finally, we can learn ES-CVAE with very large scale thanks to the doubly stochastic inference. In each parameter update, minibatch is constructed using uniform sampling. Instead of uniform sampling , we may use stratified sampling by randomly choosing a subset of cells and from which we gather randomly chosen observation to build a minibatch.[6]

[6]The average of this stochastic gradient would be the gradient.

VII. Conclusion

ES-CVAE models the sequential data. Unlike other related models, the model uses ESN instead of RNN or LSTM, which makes inference of the model simple. Because we only need to learn how to interpret the echo state, which itself need not to be updated whenever we update model parameters. By adopting ES-CVAE based anomaly detection algorithm, we can not only get anomaly scores of a given timeseries but also visualize in lower dimensional space. We adopt *Stochastic Gradient Variational Bayes(SGVB)* to learn the model, which is an efficient learning algorithm for Big data. In experiment, our algorithm shows competitive performance compared to baselines and it is applied to the Apollo dataset, where it is shown that it can be easily implemented in distributed environment.

Acknowledgments

We are grateful to Yun-Guen Lee and Young-Min Choi for private discussion. This work was supported by SK Telecom, Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) [B0101-16-0307; Basic Software Research in Human-level Lifelong Machine Learning (Machine Learning Center)], and National Research Foundation (NRF) of Korea [NRF-2013R1A2A2A01067464].

References

[1] L. H. Chiang, R. D. Braatz, and E. L. Russell, *Fault detection and diagnosis in industrial systems*. Springer Science & Business Media, 2001.

[2] L. Xiong, B. Póczos, J. Schneider, A. Connolly, and J. VanderPlas, "Hierarchical probabilistic models for group anomaly detection," 2011.

[3] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[4] J. Chung, K. Kastner, L. Dinh, K. Goel, A. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," *Advances in Neural Information Processing Systems*, 2015.

[5] J. Bayer and C. Osendorfer, "Learning stochastic recurrent networks," *stat*, vol. 1050, p. 5, 2015.

[6] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," pp. 3465–3473, 2015.

[7] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proceedings of the International Conference on Learning Representations (ICLR)*, Banff, Canada, 2014.

[8] H. Jaeger, "Adaptive nonlinear system identification with echo state networks," in *Advances in neural information processing systems*, 2002, pp. 593–600.

[9] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[10] R. Salakhutdinov and G. E. Hinton, "Deep boltzmann machines," in *International Conference on Artificial Intelligence and Statistics*, 2009, pp. 448–455.

[11] D. J. MacKay and M. N. Gibbs, "Density networks," 1999.

[12] M. Lukoševičius, "A practical guide to applying echo state networks," in *Neural Networks: Tricks of the Trade*. Springer, 2012, pp. 659–686.

[13] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *The Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, 2011.

[14] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic variational inference," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 1303–1347, 2013.

[15] N. Laptev and S. Amizadeh, *Yahoo anomaly detection dataset s5*. [Online]. Available: http://webscope.sandbox.yahoo.com/catalog.php?datatype=s&did=70

[16] O. Vallis, J. Hochenbaum, and A. Kejariwal, "A novel technique for long-term anomaly detection in the cloud," in *Proceedings of the 6th USENIX conference on Hot Topics in Cloud Computing*. USENIX Association, 2014, pp. 15–15.
[17] F. A. A. Garcia, "Tests to identify outliers in data series."
[18] Z. Ghahramani, "Parameter estimation for linear dynamical systems," Tech. Rep., 1996.
[19] T. Broderick, N. Boyd, A. Wibisono, A. C. Wilson, and M. I. Jordan, "Streaming variational bayes," in *Advances in Neural Information Processing Systems*, 2013, pp. 1727–1735.