Anomaly Detection With Conditional Variational Autoencoders

Adrian Pol, Victor Berger, Gianluca Cerminara, Cécile Germain, Maurizio Pierini

Anomaly Detection With Conditional Variational Autoencoders

Adrian Alan Pol[1,2], Victor Berger[2], Gianluca Cerminara[1], Cecile Germain[2], Maurizio Pierini[1]

[1] European Organization for Nuclear Research (CERN)
Meyrin, Switzerland

[2] Laboratoire de Recherche en Informatique (LRI)
Université Paris-Saclay, Orsay, France

***Abstract*— Exploiting the rapid advances in probabilistic inference, in particular variational Bayes and variational autoencoders (VAEs), for anomaly detection (AD) tasks remains an open research question. Previous works argued that training VAE models only with inliers is insufficient and the framework should be significantly modified in order to discriminate the anomalous instances. In this work, we exploit the deep conditional variational autoencoder (CVAE) and we define an original loss function together with a metric that targets hierarchically structured data AD. Our motivating application is a real world problem: monitoring the trigger system which is a basic component of many particle physics experiments at the CERN Large Hadron Collider (LHC). In the experiments we show the superior performance of this method for classical machine learning (ML) benchmarks and for our application.**

I. Introduction

AD is expected to evolve significantly due to two factors: the explosion of interest in representation learning and the rapid advances in inference and learning algorithms for deep generative models. Both go well beyond the traditional fully supervised setting, which is generally not applicable for most AD tasks. Particularly relevant is the variational learning framework of deep directed graphical model with Gaussian latent variables i.e. variational autoencoder (VAE), and deep latent Gaussian model, introduced by [1], [2].

Relatively little work has been devoted to exploit for AD the advances in modeling complex structured representations that perform probabilistic inference effectively. In most of them (discussed in section III-D), it has been argued that vanilla VAE architectures may not be adequate for AD, and that they must be specifically tweaked for specific sub-cases of AD with complex extensions.

This work is motivated by a real world problem: improving AD for the trigger system, which is the first stage of event selection process in many experiments at the LHC at CERN. To be acceptable in this high-end production context, any method must abide to stringent constraints: certainly performance, but also simplicity and robustness, for long-term maintainability. Because of the nature of our target application the algorithm has to be conditional. In layman terms, some of the structure of the model is known and associated observables are available. This points towards CVAE architectures [3]. CVAE is a conditional directed graphical model where input observations modulate the prior on latent variables that generate the outputs, in order to model the distribution of high-dimensional output space as a generative model conditioned on the input observation.

The goal of this paper is to explore the relevance of the alleged limitations for an ordinary CVAE, both in a generic setting, and for our specific application. We address two categories of limitations: general issues of (C)VAEs and specific to AD. Our main contributions are as follows.

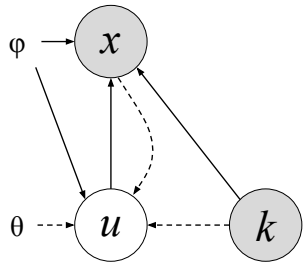

Fig. 1. Illustration of CVAE as a directed graph. Solid lines denote the generative model $p_\theta(x|u,k)p_\theta(u)$. Dashed lines denote variational approximation $q_\phi(u|x,k)$. Both variational parameters θ and generative model parameters ϕ are learned jointly.

- We define a new loss function that allows the model to learn the optimal reconstruction resolution.
- We design a new anomaly metric associated with the CVAE architecture that provides superior performance on both classical ML and particle physics specific datasets.
- We propose an alternative experimental setup for AD on MNIST dataset.

The remainder of this paper is organized as follows. Section II outlines the problem we want to solve. Section III summarizes the theoretical background, proposed method and related work. We consider a toy experiment, the MNIST and Fashion-MNIST dataset in Section IV. Finally we apply the proposed method to a real problem, related to the monitoring of the CMS experiment at the CERN LHC in Section V.

II. Problem Statement

We are operating in a semi-supervised setup, where the examples of anomalous instances are not available. However we know the design of the system and we directly observe some factors of variation in data. The observable x is a function of k (*known*) and u (*unknown*) latent vectors: $x = f(k,u)$. For a collection of samples $x = [x_1, x_2, ..., x_n]$ we are interested in highlighting instances where we observe:

- big change on single feature, we later call **Type A** anomaly, *and*
- small but systematic change on a group of features in the same configuration group (generated using the same k, as we further explain in Section IV-B), called **Type B** anomaly.

On the contrary, samples with a problem of small severity and on a group of uncorrelated features should be considered as an inlier, purely caused by expected statistical fluctuations.

In summary, we need an algorithm that exploits the known causal structure in data, spots both types of problems listed above, generalizes to unseen cases and uses data instead of relying on feature engineering. Inference time is negligible in the context of the target application (see Section V).

III. Background and Proposed Method

A. Variational Autoencoders

VAEs ([1], [2]) are a class of likelihood-based directed graphical generative models, maximizing the likelihood of the training data according to the generative model $p_\theta(Data) = \prod_{x \in Data} p_\theta(x)$. To achieve this in a computable way, the generative distribution is augmented by the introduction of a latent variable z: $p_\theta(x) = \int p_\theta(x|z)p(z)dz$. This allows to choose $p_\theta(x|z)$ as a simple distribution (like a normal law) and still have the marginal $p_\theta(x)$ to be very expressive, as an infinite mixture controlled by z.

The parameter estimation of the graph is problematic due to intractable posterior inference. The VAEs parameters are efficiently trained using an inference distribution $q_\phi(z|x)$ in a fashion very similar to autoencoders, using stochastic gradient variational Bayes framework. The recognition model $q_\phi(z|x)$ is included to approximate the true posterior $p_\theta(z|x))$. Ref. [1] shows that for any such distribution:

$$\begin{aligned}\log p_\theta(x) - \mathbb{D}_{\mathrm{KL}}(q_\phi(z|x)\|p_\theta(z|x)) = \\ \mathbb{E}_{z\sim q}[\log p_\theta(x|z)] - \mathbb{D}_{\mathrm{KL}}(q_\phi(z|x)\|p(z))\end{aligned} \tag{1}$$

Given the Kullback-Leibler (KL) divergence is always positive, the right-hand term of this equality is thus a low-bound of $\log p_\theta(x)$ for all x (called the Evidence Lower Bound, or ELBO). Optimizing it is a proxy for optimizing the log-likelihood of the data, defining the training loss as:

$$\mathcal{L}_{\mathrm{ELBO}}(x) = \mathbb{E}_{z\sim q}[\log p_\theta(x|z)] - \mathbb{D}_{\mathrm{KL}}(q_\phi(z|x)\|p(z)) \tag{2}$$

The model choice for $q_\phi(z|x)$, $p(z)$ is generally considered a factorized normal distributions, allowing easy computation of the $\mathbb{D}_{\mathrm{KL}}$ term, and sampling of z through the reparameterization trick [1].

It is typical when using VAEs to model the reconstruction as a mean squared error (MSE) between the data x and the output of the decoder. However, this is equivalent to setting the observation model $p_\theta(x|z)$ as a normal distribution of fixed variance $\sigma = 1$. Indeed, the log-likelihood of a normal distribution with fixed variance of 1 is given as:

$$-\log \mathcal{N}(x; \mu, 1) = \|x - \mu\|^2 + \log(\sqrt{2\pi}) \tag{3}$$

We argue that fixing the variance this way can be detrimental to learning as it puts a limit on the accessible resolution for the decoder: this defines the generative model as having a fixed noise of variance 1 on its output, making it impossible for it to accurately model patterns with a characteristic amplitude smaller than that. However, unless *a priori* knowledge suggests it, there is no guarantee that all patterns of interest would have such a large characteristic amplitude. This is actually trivially false for some very common cases: when the dataset has been normalized to a global variance of 1, or when it is composed of data constrained to a small interval of values, such as images whose pixels are constrained to $[0; 1]$. Rather than adding a weighting term between the two parts of the loss like has often been done ([4] for example) we rather let the model learn the variance of the output of the decoder feature-wise (i running as the dimensionality of the data vectors x):

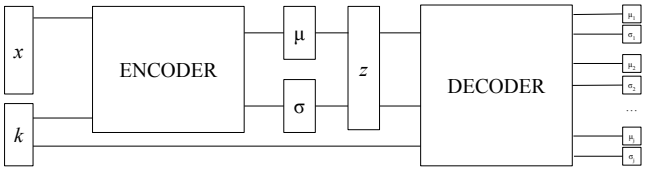

Fig. 2. Architecture of CVAE based model for AD.

$$-\log p_\theta(x|z) = \sum_i \frac{(x_i - \mu_i)^2}{2\sigma_i^2} + \log\left(\sqrt{2\pi}\sigma_i\right) \tag{4}$$

Learning the variance of the MSE reconstruction allows the model to find the optimal resolution for the reconstruction of each feature of the data, separating the intrinsic noise from the correlations. This empirically gives similar results to associating a fine-tuned weighing parameter, while removing the need to tune said hyper-parameter.

B. Setup Description

In our setup we have three types of variables, see Figure 1. For random observable variable x, u (*unknown*, unobserved) and k (*known*, observed) are independent random latent variables. The conditional likelihood function $p_\theta(x|u, k)$ is formed by a non-linear transformation, with parameters θ. ϕ is another non-linear function that approximates inference posterior $q_\phi(u|k, x) = N(\mu, \sigma I)$. The latent variables u allow for modeling multiple modes in conditional distribution of x given k making the model sufficient for modeling one-to-many mapping. To approximate ϕ and θ we optimize the following modified ELBO term:

$$\begin{aligned}\log p_\theta(x) \geq & \mathbb{E}_{q_\phi(z|k,x)}[\log p_\theta(x|k)] \\ & - \mathbb{D}_{\mathrm{KL}}(q_\phi(z|x, k)||p(z))\end{aligned} \tag{5}$$

where z (a Gaussian latent variable) intends to capture non-observable factors of variation u. The loss is computed as:

$$\begin{aligned}\mathcal{L}_{\mathrm{CVAE}}(x, k, \theta, \phi) = & \sum_i \frac{(x_i - \mu_i)^2}{2\sigma_i^2} + \log\left(\sqrt{2\pi}\sigma_i\right) \\ & + \mathbb{D}_{\mathrm{KL}}(q_\phi(z|x, k)||p(z)).\end{aligned} \tag{6}$$

Our model is built upon CVAE framework but we focus on conditional distribution of output variables for AD tasks.

We address the difference to VAE setup in Section III-C. The schema of the network architecture, corresponding to a graph from Figure 1 is shown in Figure 2. Depending on the experiment, the number and type of hidden layers will vary. We train the model using Keras [5] with TensorFlow [6] as a backend using Adam [7] optimizer and with early stopping [8] criterion. Once the model parameters are learned, we can detect anomalies using different metrics:

- for Type A with average infinity norm of the reconstruction loss $||\frac{1}{\sigma}(x-\hat{x})^2||_\infty$ ($\hat{x}$ as the reconstructed mean and σ as the reconstructed variance of decoder output), performing multiple sampling of z (we arbitrarily choose 30);
- for Type B with mean KL-divergence term $\mu(\mathbb{D}_{\text{KL}})$.

C. A metric for anomaly detection with CVAE

For a given datapoint (x, k), the evaluation of the loss of the VAE at this datapoint $\mathcal{L}(x, k)$ is an upper-bound approximation of $-\log p_\theta(x|k)$, measuring how unlikely the measure x is to the model given k. Thresholding the value of this loss is thus a natural approach to AD as explored with good results in [9]. The CVAE thus provides here a model that naturally estimates how anomalous x is given k, rather than how anomalous the couple (x, k) is. This means that a rare value for k associated with a proper value for x should be treated as non-anomalous, which is our goal. The CVAE was successfully used for intrusion detection tasks before [10]. However authors approach did not use $\mathbb{D}_{\text{KL}}$ as anomaly indicator.

The loss function from Equation 6 can be broken up to target two independent problems. Because of two separate failure scenarios we do not combine the metrics in one overall score but rather use logical `OR` to determine anomalous instances. In the first case we are interested in identifying an anomaly on a single feature. Typically used mean of reconstruction error would likely be an incorrect choice when most of the features do not manifest abnormalities and lower the anomaly score. In the second case we expect μ_z to land on a the tail of the distribution for anomalous cases. As argued in [11] the $\mathbb{D}_{\text{KL}}$ measures the amount of additional information needed to represent the posterior distribution given the prior over the latent variable being used to explain the current observation. The lower the absolute value of $\mathbb{D}_{\text{KL}}$ the more predictable state is observed.

Finally, the use of the VAE framework guarantees that the method generalizes to unseen observations as argued in [12].

D. VAE for anomaly detection

Deep architectures have become increasingly popular in semi-supervised AD [13]. They cope with the issues of the classical methods, μ-SVM [14], and Isolation Forest [15] (IF). As argued by [16], the μ-SVM kernel-based classification does not scale to high data dimensionality, as it requires that the function to learn be smooth enough to achieve generalization by local interpolation between neighboring examples. Isolation assumes that anomalies can be isolated in the native feature space.

The need for agnostically learning a representation from the data can be addressed indirectly by deep networks in a classification or regression context [17], and be exploited for semi-supervised AD [18]. Autoencoders are particularly adapted to semi-supervised AD. When trained on the nominal, testing on unseen faulty sample tend to yield sub-optimal representations, indicating that a sample is likely generated by a different process. Until relatively recently, the autoencoding approach was restricted to learning a deterministic map of the inputs to the representation, because the inference step with these representations would suffer from high computational cost [19]. A considerable body of work has been devoted to evolve these architectures towards learning density models implicitly [20].

The dissemination of the generative models, and specifically the VAE, offer a more general and principled avenue to autoencoding-based AD. [21] describes a straightforward approach for VAE-based AD. It considers a simple VAE, and the Monte-Carlo estimate of the expected reconstruction error (termed reconstruction probability), which is similar to our metric for Type A problem. Experiments on MNIST and KDD demonstrate a majority of superior performance of VAE over AE and spectral methods.

However, [22] argues that the probabilistic generative approach of VAE could suffer from an intrinsic limitations when the goal is AD, with two arguments. Firstly, because the model is trained only on inliers, the representation will not be discriminative, and will essentially overfit the normal distribution. Secondly, the representation might even be useless, falling back to the prior; technically because the generator is too powerful, the regularization by the $\mathbb{D}_{\text{KL}}$ vanishes [23].

[24] addresses this issue with specific hypotheses on the distributions of inliers and anomalies. A more general approach [25], [22] is to learn a more conservative representation by exposing the model to out-of-distribution (abnormal) examples, still without knowledge of the actual anomaly distribution, with adversarial architectures and specific regularizations. While [25] simply defines an ad-hoc regularization and hyperparameter optimization, [22] proposes an adversarial architecture for generating the anomalies and exploiting them to create a less overfitted representation. Neither of these approaches would meet the robustness and simplicity specifications of our motivating application.

IV. Experiments on benchmarks

A. Anomaly Detection on MNIST and Fashion-MNIST

We asses the performance of the proposed method using the MNIST [26], and the more recent and more challenging Fashion-MNIST [27] datasets as examples of possible real-world applications. Both datasets contain gray-level images of handwritten digits and pieces of clothing respectively. Handwritten digits in MNIST dataset belong to a manifold of dimension much smaller than the dimension of x (28x28 pixels), because the majority of random arrangements of pixel intensities do not look like handwritten digits. Intuitively, we would expect this dimension to be at least the size of 10 as

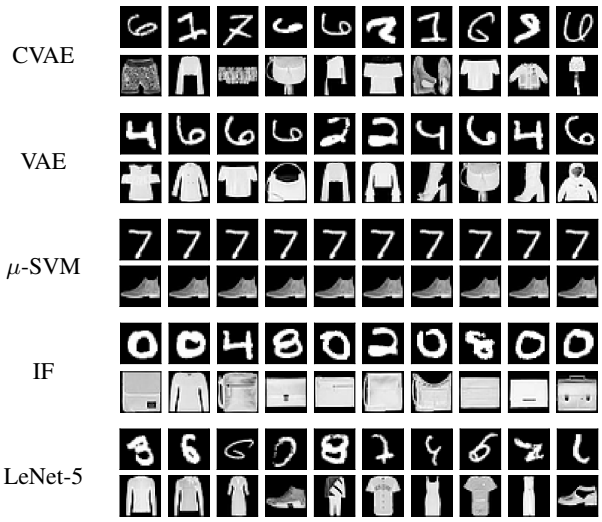

Fig. 3. Most anomalous samples in the test set for MNIST (top) and Fashion-MNIST (bottom) datasets and for each AD method and LeNet-5 classifier.

the number of classes suggests. But we need to accommodate larger latent space as each digit can be written in different style. Similar intuition applies to Fashion-MNIST as this dataset also has 10 target classes of clothing types but there is variability inside a class e.g. type of shoe.

Past works on AD with MNIST dataset arbitrarily assigned one of the classes as anomalous. For instance digit 0 was considered abnormal while other digits were considered as inliers. We propose a different, more intuitive setup. Firstly we can subjectively asses performance of AD algorithms using test dataset simply by reporting instances regarded as most anomalous (see Figure 3). Human observer regards the digits as outliers because of the latent features not captured by class label describing the original, unconventional handwriting styles. For instance digit 4 with style resembling digit 9 should be considered as anomalous. To proxy this behavior we train a classifier M and label each sample having classification error higher than threshold t as anomalous. We apply the exact same procedure for Fashion-MNIST dataset. In our study we use LeNet-5 [26] model. In summary, each pre-trained AD algorithm A is evaluated as in Algorithm 1.

Algorithm 1 AD on MNIST and Fashion-MNIST datasets

procedure LABEL(Model M, Data X_{train}, Data X_{test})
  $M \leftarrow X_{train}$ ▷ Training Classifer
  $s = M(X_{test})$ ▷ Evaluate Log Loss
  return s
procedure DETECTION(Algo A, Data X_{test})
  $t = 0.01$
  while $t < 1$ **do**
    $labels \leftarrow s > t$ ▷ Get Binary Labels
    $scores \leftarrow A(X_{test})$ ▷ Get Anomaly Score
    $p \leftarrow AUC(labels, scores)$ ▷ Get ROC AUC
    $t = t + 0.01$
  return p

In our experimental setup we assign a class label to vector k while u should accommodate information about other factors of variation e.g. hand used to write a digit. The problem of detecting anomalies is analogous to Type B problem. In this case we would expect $\mu(\mathbb{D}_{\text{KL}})$ to be higher in cases of mislabelling or uncommon style.

Throughout the experiments, we use the original train-test splits with 10000 test samples. For changing classification error threshold values t we report Receiver Operating Characteristic (ROC) Area Under the Curve (AUC) p for popular AD algorithms and a vanilla VAE, see Figure 4. We use μ-SVM and IF as baselines, for which we concatenate class label to input pixel values for fair comparison. We notice that for vanilla VAE the $\mathbb{D}_{\text{KL}}$ is not a useful anomaly indicator, as we expect the latent information to be mostly dominated by the class-label value. Changing architecture to CVAE turns $\mathbb{D}_{\text{KL}}$ to anomaly indicator, which outperforms other baseline techniques. The Fashion-MNIST dataset was designed to be more challenging replacement for MNIST. We notice observable drop in ROC AUC as the dataset has more ambiguity between classes. However, compared to baseline methods the CVAE-based model exceeds their detection performance.

For generative purposes our setup is insufficient. As shown in [28] we would need additional adversarial system for such objective. However, the AD task is in fact simpler as it is not necessary to generate realistic outputs of the generator. Such regularization will not help with training set contamination with outliers. This can give the encoder possibility to store too much information about the anomalies and harm the detection performance of the algorithm.

B. Synthetic Problem

The synthetic dataset uses normally distributed ($\mu = 0$, $\sigma = 1$), continues and independent latent variables u and k. Observable x is simply a product of u, k and additional noise ϵ given configuration constraints: $x_j = f_j(\vec{u}) \cdot \sum_{i=0}^{m} \mathbf{S}_{ji} k_i + \epsilon$, where j is a feature index for $\vec{x}$ in $\mathbb{R}^n$. A binary matrix $\mathbf{S}$ describes which k is used to compute feature j:

$$\mathbf{S} = \begin{array}{c} \\ x_0 \\ x_1 \\ \vdots \\ x_n \end{array} \begin{array}{c} \begin{array}{cccc} k_0 & k_1 & \cdots & k_m \end{array} \\ \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \end{array},$$

and function $f(\vec{u})$ describes which u enters the product that defines each feature j: $f_j(\vec{u}) = \prod_o u_o$. S and $f(\vec{u})$ stay unchanged across each sample in the dataset but the values of k and u do change. For simplicity, we ensure that each j depends only on one k and the dependence is equally distributed. Finally we can manipulate values of o and m. For instance, the first column x_0 can use k_0, u_1 and u_4: $x_0 = k_0 u_1 u_4$; x_{99} may be generated using k_4 and u_0 etc.

We generate samples with x being 100-dimensional ($n = 100$) and $m = o = 5$. An example of correlation matrix

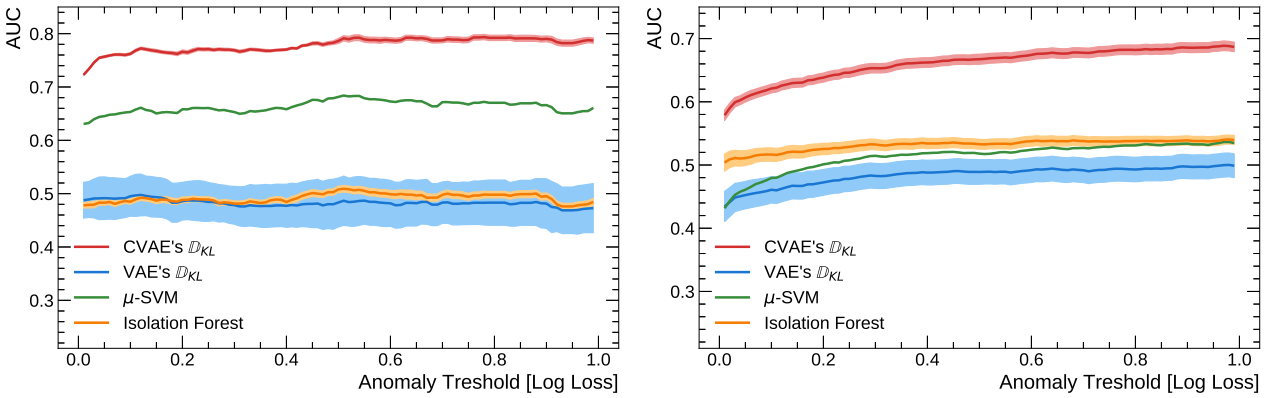

Fig. 4. Reported ROC AUC for **MNIST** (left) **Fashion-MNIST** (right) datasets and different AD algorithms as a function of varying anomaly threshold t based on LeNet-5 model classification log loss s. Overall classifier accuracy is 98.95% and 89.62% for MNIST and Fashion-MNIST respectively. The curves stay relatively flat due to high performance of the classifier: most of the test samples have log loss smaller than 0.01.

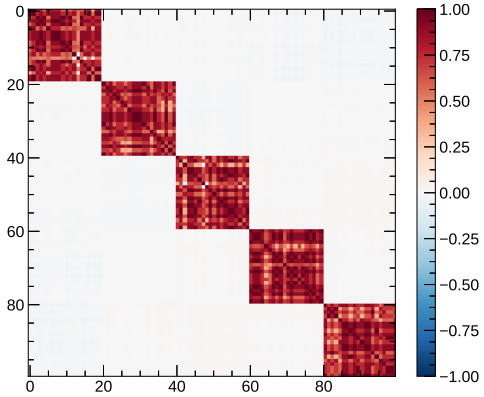

Fig. 5. Correlations between features for $m = o = 5$ and $n = 100$.

TABLE I
TYPES OF TEST DATA

Test set	**Description**
Type A Inlier	Generated in the same process as training data
Type A Anomaly	5σ change on ϵ for a random feature
Type B Inlier	3σ change on ϵ for a random set of $\frac{n}{m}$ features
Type B Anomaly	3σ change on ϵ for a random feature cluster

between features can be seen in Figure 5. For testing we generate samples according to Table I. The choice of 5σ and 3σ comes from legacy requirements of our target application. The AD is performed by: comparing output of the decoder with encoder input for problems observed only on one of the features - Type A problem; or comparing $\mathbb{D}_{\text{KL}}$ yield for a samples with problems present on all features belonging to the same causal group (using the same k column as input) - Type B problem.

The ROC curves corresponding to both of the problems are shown in Figure 6. Given the high order of the deviation on Type A anomalies, the algorithm easily spots those types of problems. In context of hierarchical structures, an algorithm needs to model a mapping from single input to multiple possible outputs. As argued in [3] we need a model that can make diverse predictions. The Type B detection provides good results outperforming vanilla VAE baseline confirming that CVAEs are suitable for such task.

V. Experiments on CMS Trigger Rate Monitoring

A. Motivation

This work emerges directly from the explicit urgency of extending monitoring of the CMS [29] experiment. The CMS experiment at CERN LHC [30] operates at the remarkable rate of 40 million particle collisions (*events*) per second. Each event corresponds to around 1 MB of data in unprocessed form. Due to understandable storage constrains and technological limitations (e.g. fast enough read-out electronics), the experiment is required to reduce the number of recorded data from 40 million to 1000 events per second in real time. To this purpose, a hierarchical set of algorithms collectively referred to as the *trigger system* are used to process and filter the incoming data stream which is the start of the physics event selection process.

Trigger algorithms [31] are designed to reduce the event rate while preserving the physics reach of the experiment. The CMS trigger system is structured in two stages using increasingly complex information and more refined algorithms:

1) **The Level 1** (L1) **Trigger**, implemented on custom designed electronics; reduces the 40 MHz input to a 100 kHz rate in < 10 μs.
2) **High Level Trigger** (HLT), a collision reconstruction software running on a computer farm; scales the 100 kHz rate output of L1 Trigger down to 1 kHz in < 300 ms.

Both the L1 and the HLT systems implement a set of rules to perform the selection (called *paths*). The HLT ones are seeded by the events selected by a configurable set of L1 Trigger paths.

Under typical running conditions, the trigger system regulates the huge data deluge coming from the observed collisions. The quality of the recorded data is guaranteed, by monitoring each detector subsystems independently (e.g.

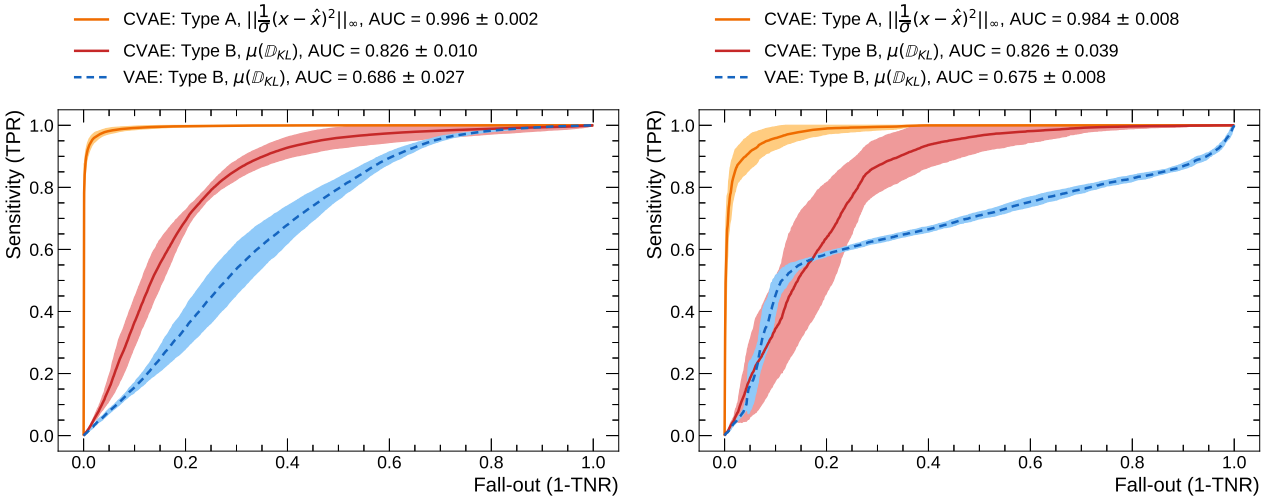

Fig. 6. The ROC curves for two AD problems using synthetic test dataset (left) and CMS trigger rates test dataset (right). The bands correspond to variance computed after running the experiment five times using random weight initialization. Anomaly score for Type B is computed using mean $\mathbb{D}_{\mathrm{KL}}$ of z. Anomaly score for Type A problem is computed using decoder outputs: μ and σ of each feature. For CMS trigger case with low fall-out, VAE slightly outperforms CVAE which could be caused by our specific choice of HLT paths.

measuring voltage), and by monitoring the trigger rates. The event acceptance rate is affected in presence of number of issues e.g. detector malfunctions, software problems etc. Depending on the nature of the problem, the rate associated to specific paths could change to unacceptable levels. Critical cases include dropping to zero or increasing to extreme values. In those cases, the system should alert the shift crew, calling for a problem diagnosis and intervention.

HLT paths are often very strongly correlated. This is due to the fact that groups of paths select similar physics objects (thus reconstructing the same event) and/or are seeded by the same selection of L1 Trigger paths. While critical levels of rate deviations for singular paths should be treated as anomaly, smaller deviations on number of random trigger paths are likely a result of statistical fluctuations. On the other hand an observable coherent drift (even small) on a group of trigger paths related by similar physics or making use of the same hardware infrastructure, is an indication of a likely fault present in the trigger system or hardware components.

We explore this hierarchical structure in our algorithm. Each HLT path has a direct, pre-configured link to set of L1 trigger paths through specified configuration as schematically shown in Figure 7. The configuration changes infrequently i.e. nodes are added, disabled or corrected. Consequently, the HLT system performance is directly linked with the status of L1 Trigger.

We do not focus on minimizing the inference time as the anomaly can be flagged within minutes which is long enough for all the algorithms considered.

B. Experiment

We apply CVAE architecture, where we treat HLT rates as x and L1 Trigger rates as k. Our prototype uses four L1 Trigger paths that seed six unique HLT paths each. We extract rates only from samples where all chosen paths are present in the configuration. We end up with 102895 samples which are then split into training, validation and test set. Our test set has 2800 samples. Operators set quality labels for each CMS sub-detector and for each sample. Since the global quality flag is composed by contribution from all subsystems, a sample could be regarded as bad due to under-performance of a detector component not related to the set of trigger paths we chose or not related to problem we try to solve. Hence we cannot use those labels in the test set. Instead, we consider hypothetical situations that are likely to happen in the production environment, similar to those used for synthetic problem. We generate four synthetic test datasets manipulating our test set in similar manner to the synthetic dataset. We detect isolated problems on one of the HLT paths - Type A; and problems present across HLT paths seeding the same L1 trigger path - Type B.

We report the results in Figure 6. The performance of the algorithm on CMS dataset is matching the performance we reported for the synthetic one. The CMS experiment currently does not provide any tools to track problems falling into Type B category. Given a good performance of the proposed method, we believe that the solution could be considered for deployment, provided further tests and refinements in the production environment.

VI. Conclusions and Future Work

This paper shows how anomalous samples can be identified using CVAE. We considered the specific case of CMS trigger rate monitoring to extend the current monitoring functionality and showed good detection performance. The proposed algorithm does not rely on synthetic anomalies at training time or additional feature engineering. We demonstrated the method is not bound to CMS experiment specifics and has potential to work across different domains. However more tests on more difficult datasets are desirable, e.g. on

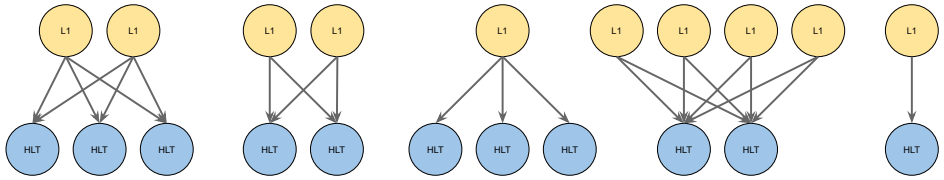

Fig. 7. Simplified, schematic graph inspired by the trigger system configuration. Blue nodes represent HLT paths while yellow L1 trigger paths. Each link is unidirectional starting from yellow nodes. The graph has few hundred nodes spread approximately equally between HLT and L1 triggers paths. The connection between L1 trigger and HLT paths can be seen as a hierarchical directional graph from L1 to HLT system.

CIFAR, which provides more classes and a higher variance. We did not perform hyper-parameter scan for any of the experiments thus we expect the results to get better if further optimized. Subsequent studies foresee using full configuration of the CMS trigger system. An interesting extension of the method would be learning correct encoding of unknown factors of variations in the latent space, which at this moment is unconstrained (e.g. a tilt or boldness of the digit in the MNIST dataset).

Acknowledgments: We thank the CMS collaboration for providing the dataset used in this study. We are thankful to the members of the CMS Physics Performance and Dataset project for useful discussions and suggestions. We acknowledge the support of the CMS CERN group for providing the computing resources to train our models. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement n^o 772369).

REFERENCES

[1] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[2] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, 2014.

[3] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*, pages 3483–3491, 2015.

[4] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR*, 2(5):6, 2017.

[5] François Chollet et al. Keras, 2015.

[6] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[8] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(Dec):3371–3408, 2010.

[9] Maximilian Soelch, Justin Bayer, Marvin Ludersdorfer, and Patrick van der Smagt. Variational inference for on-line anomaly detection in high-dimensional time series, 2016.

[10] Manuel Lopez-Martin, Belen Carro, Antonio Sanchez-Esguevillas, and Jaime Lloret. Conditional variational autoencoder for prediction and feature recovery applied to intrusion detection in iot. *Sensors*, 17(9):1967, 2017.

[11] Mevlana Gemici, Chia-Chun Hung, Adam Santoro, Greg Wayne, Shakir Mohamed, Danilo J Rezende, David Amos, and Timothy Lillicrap. Generative temporal models with memory. *arXiv preprint arXiv:1702.04649*, 2017.

[12] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in NeurIPS*, pages 3581–3589, 2014.

[13] Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *CoRR*, abs/1901.03407, 2019.

[14] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[15] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 6(1):3, 2012.

[16] Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 34(5):1–41, 2007.

[17] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *CoRR*, abs/1703.00810, 2017.

[18] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017.

[19] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[20] Guillaume Alain and Yoshua Bengio. What regularized auto-encoders learn from the data-generating distribution. *J. Mach. Learn. Res.*, 15(1):3563–3593, 2014.

[21] Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. Technical report, SNU Data Mining Center, 2015. Special Lecture on IE.

[22] Xuhong Wang, Ying Du, Shijie Lin, Ping Cui, and Yupu Yang. Self-adversarial variational autoencoder with gaussian anomaly prior distribution for anomaly detection. *CoRR*, abs/1903.00904, 2019.

[23] Shengjia Zhao, Jiaming Song, and Stefano Ermon. Infovae: Information maximizing variational autoencoders. *CoRR*, abs/1706.02262, 2017. AAAI'19.

[24] Yuta Kawachi, Yuma Koizumi, and Noboru Harada. Complementary set variational autoencoder for supervised anomaly detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2366—-2370, 2018.

[25] Dan Hendrycks, Mantas Mazeika, and Thomas G. Dietterich. Deep anomaly detection with outlier exposure. *CoRR*, abs/1812.04606, 2018. ICLR19.

[26] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[27] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[28] Michael F Mathieu, Junbo Jake Zhao, Junbo Zhao, Aditya Ramesh, Pablo Sprechmann, and Yann LeCun. Disentangling factors of variation in deep representation using adversarial training. In *Advances in Neural Information Processing Systems*, pages 5040–5048, 2016.

[29] Serguei Chatrchyan et al. The CMS experiment at the CERN LHC. *JINST*, 3:S08004, 2008.

[30] The LHC Study Group. The Large Hadron Collider, conceptual design. Technical report, CERN/AC/95-05 (LHC) Geneva, 1995.

[31] Vardan Khachatryan et al. The CMS trigger system. *JINST*, 12(01):P01020, 2017.