Fast simulation of the electromagnetic calorimeter response using Self-Attention Generative Adversarial Networks

Fedor Ratnikov[1,*] and *Alexander* Rogachev[1,**]

[1]HSE University, 11 Pokrovsky Boulevard, Moscow, Russia

Abstract. Simulation is one of the key components in high energy physics. Historically it relies on the Monte Carlo methods which require a tremendous amount of computation resources. These methods may have difficulties with the expected High Luminosity Large Hadron Collider need, so the experiment is in urgent need of new fast simulation techniques. The application of Generative Adversarial Networks is a promising solution to speed up the simulation while providing the necessary physics performance. In this paper we propose the Self-Attention Generative Adversarial Network as a possible improvement of the network architecture. The application is demonstrated on the performance of generating responses of the LHCb type of the electromagnetic calorimeter.

1 Introduction

The Large Hadron Collider (LHC) built by the European Organization for Nuclear Research (CERN) is the world's largest collider. The Large Hadron Collider beauty (LHCb) experiment is a specialized b-physics experiment, designed to measure the parameters of CP violation in the interactions of b-hadrons. The whole detector consists of several parts: the vertex detector VELO, Ring imaging Cherenkov detector RICH, the main tracking system, followed by another RICH, the electromagnetic/hadronic calorimeters and the muon system. Calorimeters specialize in measuring specific parameters of passing particles, providing measurements of the energy of electrons, photons, and hadrons. This data allows the reconstruction of the initial collision, so the detector's response simulation for a given input becomes the challenge. Using a baseline simulation software Geant4 [9] requires a tremendous number of CPU hours, being most computationally expensive part of the experimental program.

It was shown [1, 2] that it is possible to simulate the calorimeter response using deep learning techniques and Generative Adversarial Networks, but in order to apply GANs practically their performance should be properly evaluated using quality metrics and improved using additional techniques if it is possible. In this work, we propose adding Self-Attention module into previously published architecture to improve the quality of generating objects and compare previous results with the new ones generated by Self-Attention Generative Adversarial Network in terms of Area Under Precision-Recall Distribution Curve (PRD-AUC).

---

*e-mail: fedor.ratnikov@cern.ch
**e-mail: airogachev@hse.ru

2 Generative neural networks

Generative adversarial networks (GANs) are types of generative model that consist of two networks competing against each other in order to provide an ability to reproduce a given distribution. In this section a brief overview of Conditional GANs is given. Then, we review an approach based on Self-Attentions, because SA-module helps Convolutional Neural Networks (CNNs) with modeling long range and multi-level dependencies across image regions [3], and analyze the possibility of applying Self-Attention Generative Adversarial Network (SAGAN) [3] to reproduce the response of the calorimeters.

2.1 Conditional Generative Adversarial Networks

The key idea of generative adversarial networks is a contest of two neural networks, a generator (G) and a discriminator (D). The first one generates samples and another one evaluates them, so the generator training objective is to increase the error of the discriminative network. Basically, the GAN learns to map from a latent space to a data distribution, so its input can be represented by a noise (a simple distribution, e.g. Gaussian). Formally, it is possible to represent the minimax game of two networks:

$$\min_G \max_D \mathrm{E}_{x \sim p_{data}(x)}[log D(x)] + \mathrm{E}_{z \sim N(0,I)}[log(1 - D(G(z)))], \tag{1}$$

where p_{data} is a true data distribution.

Adding the conditional vector along with the noise one as an input of the GAN it is possible to perform a transition to the conditional generation. Thus, the generator utilizes additional information that comes from a class label, in case of multi-class dataset, or just like another continuous input for generation in order to generate more appropriate sample according to the condition. The discriminator uses the conditional input to classify real and generated objects as well.

2.2 Self-Attention Generative Adversarial Networks

Previous CaloGAN architectures [1, 2] used many convolutional layers, but common CNN GANs have difficulties in figuring out some geometric patterns that occur consistently in case of some fixed cluster conditions. We noticed some outliers to appear in generating objects in case of WGAN (Fig. 3). One of the possible explanations can be the weakness of local receptive field of the neurons: it becomes impossible to process some long-distance dependencies because the convolution has only a local receptive field, thus in order to take far regions into account several convolutional layers should be applied or the size of the convolution kernel should be increased. Such architectural changes lead to the loss of the computational and statistical efficiency obtained by using CNN structure. Increasing the receptive field with a small computational cost becomes the challenge. Attention-based approach may be the solution.

A recent work [3] proposed to use Self-Attention modules for convolutional layers of neural networks in order to improve the model's performance. Basically, SA-module calculates response at a position as a weighted sum of the features at all positions, where the weights are expected to be calculated with a small cost, allowing networks to model relationships between widely separated spatial regions.

According to the algorithm, the features that come from the hidden layer $x \in \mathbb{R}^{C \times N}$ (C is the number of channels, N is the number of feature locations of features from the previous hidden layer) are transformed into two feature spaces $f(x) = W_f x$, and $g(x) = W_g x$ to calculate the attention.

$$\beta_{j,i} = \frac{\exp(s_{ij})}{\sum_{i=1}^{N} \exp(s_{ij})}, \tag{2}$$

where $s_{ij} = f(x_i)^T g(x_j)$ and $\beta_{j,i}$ indicates the extent to which the model attends to the ith location when synthesizing the jth region. The component of the output of the SA layer can be represented as follows:

$$o_j = v\left(\sum_{i=1}^{N}\beta_{j,i}h(x_i)\right),\ h(x_i) = W_h x_i,\ v(x_i) = W_v x_i. \tag{3}$$

All the matrices W are learned through the training process as well as the layer output's scale parameter γ, that is initialized as zero, so SA-module doesn't affect the output in the beginning of the training procedure, allowing the convolutional layer to learn some close-range dependencies first and then increase the complexity of the task.

2.3 GANs for electromagnetic calorimeter response simulation

The application of Generative Adversarial Networks to simulation in High Energy Physics has been proposed by Paganini et al [4]. It was shown that the implementation of DCGAN could be used for calorimeter simulation, however improved accuracy was required in order to replace Geant4[9]. The idea was further developed in [2] with the use of Wasserstein CGAN [6]. In addition to the previous DCGAN implementation, particle coordinates and momentum projections were used as an input, meanwhile particle energy was an output. Lately, Fedor Sergeev et al. [1] compared the performance of Conditional Variational autoencoder, an improved Conditional GAN and combined CVAE+CGAN architectures, considering CGAN to achieve the best results, boosting previous accuracy even more.

In this paper, we improve the generated samples quality by adding Self-Attention layers to the previous best-performing architecture to help CNN-based generator and discriminator to process long-range relationships between image regions through the training process, as it was claimed in [3], expecting the generation of clusters with more appropriate shape according to the given conditions. During the training process Spectral Normalization [5] was implemented to improve training dynamics. We test our model the same way as it was done in [1], boosting the PRD metrics evaluated on the generated samples.

3 SAGAN for calorimeter response generation

In this section we review the given dataset and data preprocessing, describe the architecture of the model and discuss the metrics we use to evaluate generated samples.

3.1 Dataset

In this paper, the dataset contains electrons interactions inside the electromagnetic calorimeter data inspired by the LHCb detector at the CERN LHC [8]. The electromagnetic calorimeter uses "shashlik" technology of alternating layers of lead and scintillation plates. The prototype consists of 5×5 blocks of size $12\,\text{cm} \times 12\,\text{cm}$, the cell granularity corresponds to each block being 6×6 of size $2\,\text{cm} \times 2\,\text{cm}$. As an electron enters the calorimeter, it collides with a metal plate, producing a few secondary photons and electrons. The produced particles hit metal plates, creating more ones. This chain reaction produces a stream of electrons and photons, called an electromagnetic shower. All energies deposited in the scintillator layers of one cell are summed up.

Calorimeter cells and the energy stored in them are represented as a 30×30 energy distribution matrix being the target output of the generation process. We use the momentum of

particles $\boldsymbol{p} = (p_x, p_y, p_z)$ and the coordinates where they enter the calorimeter $\boldsymbol{r} = (x, y)$ as a conditional input of the model. All the energy distributions for the given $\boldsymbol{r}, \boldsymbol{p}$ were precomputed using the Geant4 framework. According to [1] we use logarithmic transformation to achieve better performance:

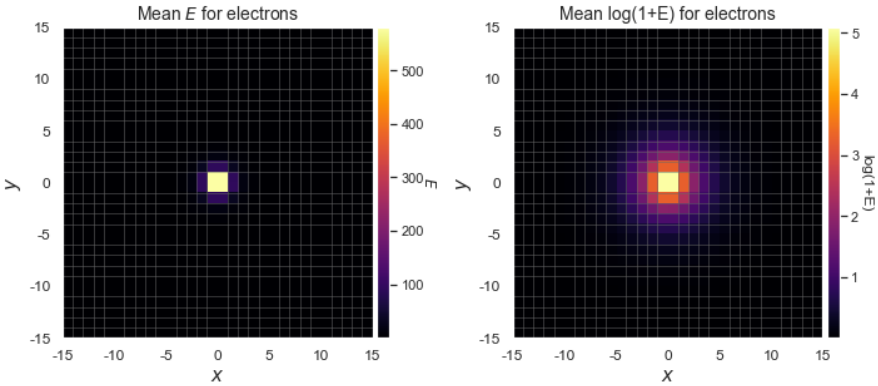$$f(E) = \log\left(1 + \frac{E}{MeV}\right), \tag{4}$$

Figure 1: Visualisation of the dataset.

3.2 Model

We added two self-attention layers in the architecture that was introduced in [1]. During the experiments in order to stabilize and speed up the training process we applied spectral normalization to discriminator only and to both networks, achieving the best performance in the second case.

Comparing Wasserstein loss in case of baseline, the hinge version of the adversarial loss through SAGAN training was minimized:

$$
\begin{aligned}
L_D = & -\mathbb{E}_{(x,y)\sim p_{data}}[\min(0, -1 + D(x,y))] \\
& - \mathbb{E}_{z\sim p_z, y\sim p_{data}}[\min(0, -1 - D(G(z), y))], \\
L_G = & -\mathbb{E}_{z\sim p_z, y\sim p_{data}} D(G(z), y),
\end{aligned}
\tag{5}
$$

3.3 Metric

To evaluate the quality of generated samples and the performance of the models PRD-AUC [7] is used :

$$\text{PRD}(Q, P) = \{(\theta\alpha(\lambda), \theta\beta(\lambda))|\lambda \in (0, \infty), \theta \in [0, 1]\}, \tag{6}$$

where P and Q are distributions, that are defined on a finite state space,

$$\alpha(\lambda) = \sum_{\omega\in\Omega} \min(\lambda P(\omega), Q(\omega)),$$

Discriminator

| input **E** (30x30) |
| conv2D (32, stride=1) |
| conv2D (64, stride=2) |
| Self-Attention (64) |
| conv2D (128, stride=2) |
| Self-Attention (128) |
| conv2D (256, stride=2) |
| flatten & concat with conditions |
| FC (256+5 → 64 → 32) |
| FC (32 → 1) |

Generator

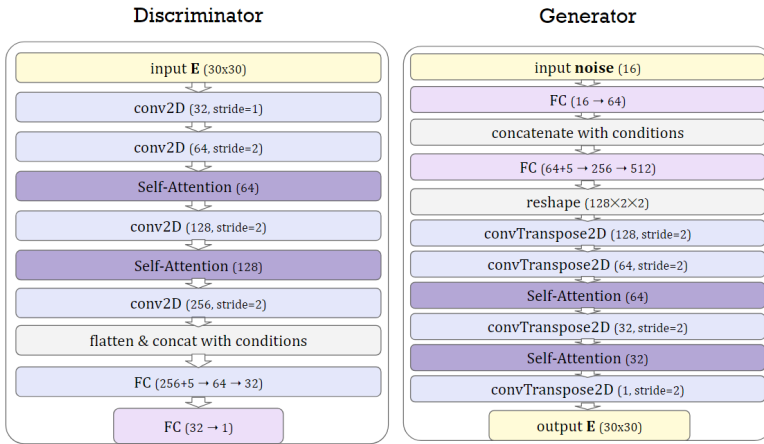| input **noise** (16) |
| FC (16 → 64) |
| concatenate with conditions |
| FC (64+5 → 256 → 512) |
| reshape (128×2×2) |
| convTranspose2D (128, stride=2) |
| convTranspose2D (64, stride=2) |
| Self-Attention (64) |
| convTranspose2D (32, stride=2) |
| Self-Attention (32) |
| convTranspose2D (1, stride=2) |
| output **E** (30x30) |

Figure 2: SAGAN architecture.

$$\beta(\lambda) = \sum_{\omega \in \Omega} \min(P(\omega), \frac{Q(\omega)}{\lambda}).$$

It is worthwhile noting that objects that are generated by the models can be considered as images to apply the same mechanisms to generate them, but we should take into account physics metrics through the evaluation as well. Thus, we use the minimum of two PRD-AUC scores, evaluated over raw images and over a set of physical metrics. These physical statistics are:

- shower asymmetry along and across the direction of inclination;

- shower width along and across direction;

- the number of cells with energies above a certain threshold (sparsity level).

    This approach allows us to verify that the generated samples are not only similar in terms of images but also have similar physical metrics distribution that seems more appropriate in terms of HEP application.

4 Results and discussion

All the models were trained for reproducing logarithmic energy distributions. We compared the PRD-AUC scores on the test dataset, computing it on with the generated samples and ground truth inverse transformation. Best performance training parameters are:

- batch size of 512;

- Adam optimizers with leaning rate $4 \cdot 10^{-4}$ and $1 \cdot 10^{-4}$ for discriminator generator respectively;

- 3 discriminator iterations per 1 generator iteration.

    SAGAN architecture without normalization performed too badly so we don't take in into account. Using Spectral Normalization (SN) only for discriminator changes the behaviour of

the model, however, normalization of both networks led to the most stable training process as well as the best results.

The most reasonable difference between the samples from WGAN and SAGAN models are the cells on the edge of the matrix and the shape of generated clusters. Sometimes there are some outliers that contain such relatively high-value energy cells in samples generated by WGAN (fig. 3), meanwhile such samples are more smoothed-out in SAGAN case (fig. 4) being closer to the original energy distribution according to PRD-AUC that was computed based on shower width along and across direction.
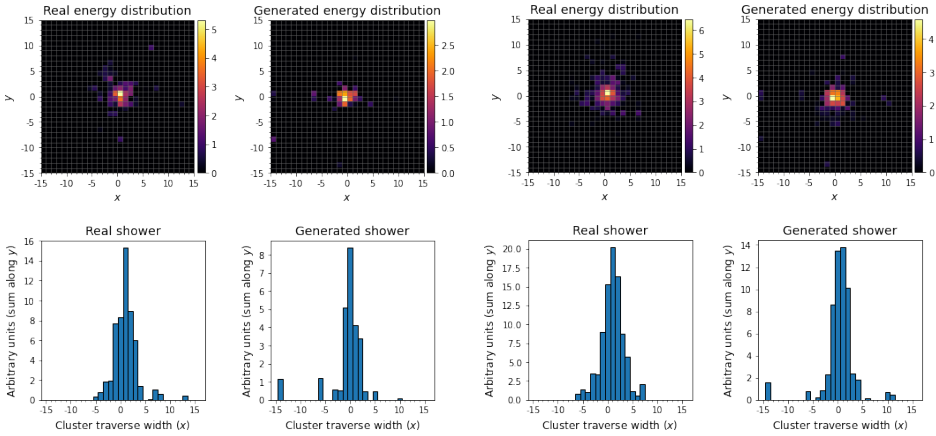

Figure 3: WGAN trained on log energies generated samples example.
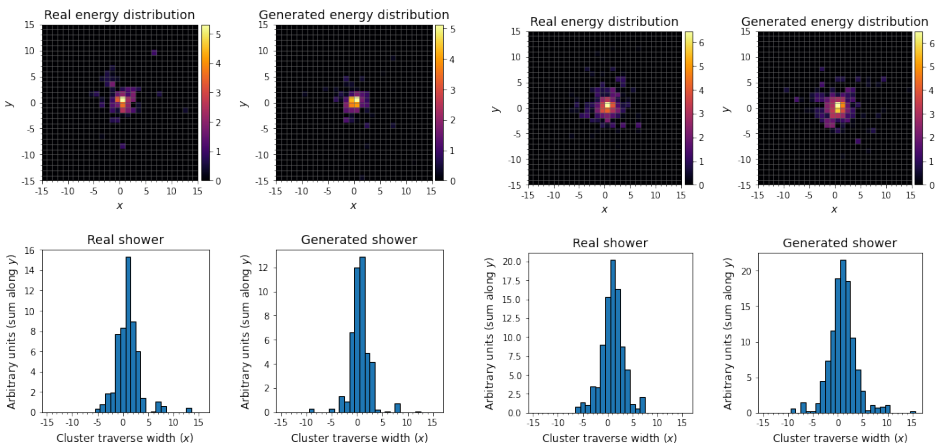

Figure 4: SAGAN trained on log energies generated samples example.

Self-attention can be the reason of such results, because model is expected to learn some geometric patterns of generated objects so model avoids such problematic cases and do not exaggerate the tails of the distribution, resulting in generation of clusters of more appropriate shape. This fact is

Comparing PRD-AUC scores, SAGAN model with spectral normalization apllied to generator and discriminator achieved the best result, boosting the previous published best value. PRD-AUC scores computed on the set of raw images for SAGAN and WGAN were almost same, however physics ones differ.

Table 1: PRD-AUC on test dataset

Model	Physics PRD-AUC	Raw Images PRD-AUC
WGAN	0.936	0.971
SAGAN + SN D	0.895	0.901
SAGAN + SN G and D	0.948	0.975

5 Conclusion

In this paper, we proposed to use Self-Attention Generative Adversarial Network as a possible efficient architecture for reproducing energy distribution in calorimeters using the case of the electromagnetic calorimeter of the LHCb experiment. Generative Adversarial Networks show promising results for the fast simulation and this research shows the importance of adding and studying new techniques and architectural approaches to improve the performance. The quality of generated result also relies on an appropriate training procedure with proper parameters. Adding Self-Attention layer made it possible to improve the shape of generated clusters, boosting the PRD-AUC score of the model, evaluated using physics-specific metrics.

Having proper evaluation measures is also important. PRD-AUC can be used as a basic solution, but it does not take any physics metrics into account and requires proxy evaluations of parameters that are chosen in advance. It is still an open question whether currently existing evaluation metrics generalize across different domains. So different ways of evaluating performance of GANs in case of simulation should be studied as well.

Future work will be focused on searching of the proper evaluation metrics and improving the quality of generated objects. Transformer GAN [10] seems like a next possible architecture to be studied because of its attention-based architecture, meanwhile density and coverage metrics [11] may solve some problems of precision-recall approach.

References

[1] Sergeev F., Jain N., Kostenkov G., Trofimova E. Fast simulation of the LHCb electromagnetic calorimeter response using VAEs and GANs // Journal of Physics: Conference Series. No. 1740, 2021

[2] Chekalina V., Orlova E., Ratnikov F., Ulyanov D., Ustyuzhanin A. and Zakharov E. Generative models for fast calorimeter simulation: the LHCb case EPJ Web Conf., 2019

[3] Zhang, H., Goodfellow, I., Metaxas, D., and Odena, A. Self-attention generative adversarial networks. In International conference on machine learning (pp. 7354-7363). PMLR., 2019

[4] Paganini M., Oliveira L., and Nachman B., CaloGAN: simulating 3D high energy particle showers in multi-layer electromagnetic calorimeters with generative adversarial networks *Phys. Rev. D* 97, 2018

[5] Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks. ICLR, 2018

[6] Gulrajani I., Ahmed F., Arjovsky M., Dumoulin V. and Courville A. Improved training of Wasserstein GANs *Advances in Neural Information Processing Systems* vol 30, pp 5769—5779, 2017

[7] Sajjadi M., Bachem O., Lucic M., Bousquet O. and Gelly S. Assessing generative models via precision and recall *Conf. on Neural Information Processing Systems*, 2018

[8] Alves A.A. Jr. et al. LHCb collaboration, JINST 3, 2008

[9] Agostinelli S et al 2003 GEANT4 — a simulation toolkit *Nucl. Instr. Meth. A* vol 506 no 3 pp 250–303

[10] Jiang Y., Chang S., Wang Z. Transgan: Two transformers can make one strong gan //arXiv preprint arXiv:2102.07074. – 2021.

[11] Naeem M. F. et al. Reliable fidelity and diversity metrics for generative models. //International Conference on Machine Learning.– PMLR, 2020, pp 7176-7185