Unsupervised Hybrid Deep Generative Models for Photovoltaic Synthetic Data Generation

Dan A. Rosa de Jesús, *Student Member, IEEE*, Paras Mandal, *Senior Member, IEEE*, Tomonobu Senjyu, *Senior Member, IEEE*, and Sukumar Kamalasadan, *Senior Member, IEEE*

***Abstract*—This paper contributes to the field of deep generative learning applied to solar photovoltaic (PV) synthetic data generation problems by exploring Deep Generative Model (DGM) that combines Variational Autoencoders (VAE) and Generative Adversarial Networks (GAN), i.e., VAEGAN. We build upon knowledge in the area of deep learning to incorporate our Hybrid Deep Neural Network (HDNN), combining convolutional and Long Short-Term Memory (LSTM) layers at the encoding level for producing robust latent representations and subsequently high-quality synthetic PV data samples. The major advantage of these approaches is that it allows the DGMs to perform better feature extraction as well as to capture the historical trends in data effectively. The simulations on actual data acquired from a real PV system demonstrate the effectiveness of the DGMs to produce high-quality samples for multiple seasons of the year.**

***Index Terms*—Deep Learning, Generative Adversarial Networks, Solar Photovoltaic, Synthetic Data, Variational Autoencoders.**

I. Introduction

MODELING electric power generation from solar photovoltaic (PV) panels is essential due to its significant impact on realizing smart and intelligent power grids. However, as energy prosumers comprise heterogeneous renewable energy sources, modeling the PV system components becomes extremely complex. Additionally, the spreading of advanced metering infrastructures in the distribution network has led to enormous amounts of energy generation data. While transmission datasets are relatively easier to get a hand on, the accessibility and availability of distribution level datasets are particularly challenging due to security and privacy considerations [1]. Moreover, PV power output primarily depends on solar radiation, which is highly intermittent due to the presence and movement of clouds at a particular location. This calls for novel techniques for the generation of high-quality synthetic PV datasets. Synthetic data generation is considered to be the way of alleviating the aforementioned issues.

Existing literature includes several methods for generating PV synthetic data. For instance, a Markov model was applied in [2] for simulating solar radiance time-series, which were converted to PV power time-series. The considered metrics showed the model's capability to reproduce the ground truth data with low variability. Lave *et al.* [3] generated PV synthetic data is generated through a three-step method that builds relationships between high and low-frequency data for different locations obtained from hourly satellite irradiance and sub-minute ground measured solar irradiance, respectively. Simulation results demonstrated the method's ability to produce unique, high-resolution PV synthetic data. Power *et al.* [4] reported a Bayesian method for residential demand and PV generation. The mean absolute error for the weekday profile is computed to be less than 10%, which demonstrates the accuracy of the model to resemble the real data. However, these methods are model-based, meaning that they require modeling the weather conditions and PV systems based on mathematical formulas or characteristics, increasing the synthetic data generation complexity. An alternative to these methods to alleviate these issues is the development and application of novel Deep Generative Models (DGM) [5].

DGMs learn the underlying ground-truth data probability distributions through unsupervised learning. In recent years, DGMs has gained popularity due to their superiority to perform a wider number of tasks, including but not limited to, feature extraction for dimensionality and noise reduction. DGMs have a number of parameters significantly smaller than the amount of data it is trained on. This encourages the models to discover and efficiently internalize the time-series sequential dependencies from historical trends. Notably, two of the most commonly used and efficient DGMs are Variational Autoencoders (VAE) and Generative Adversarial Networks (GAN). VAEs aims at maximizing the lower bound of the data log-likelihood, whereas GANs achieve an equilibrium between two networks, the generator, and discriminator. In the literature, VAEs and GANs have been used for data compression and denoising [6], [7] as well as for anomaly detection [8] applications. Furthermore, the combination of the VAE and GAN into the VAEGAN has improved the generation of synthetic data in terms of its quality and diversity [9], [10]. However, to the best of our knowledge, the application of DGMs has not been thoroughly studied in the context of power system time-series data applications, in particular, solar PV power synthetic data generation.

The work described in this paper contributes to developing DGMs for PV synthetic data generation and leveraging from previously developed performance measures for other

D. A. Rosa de Jesús and P. Mandal are with the Power and Renewable Energy Systems (PRES) Lab. within the Department of Electrical and Computer Engineering, The University of Texas at El Paso, El Paso, TX 79968, USA (e-mail: danrosa@miners.utep.edu; pmandal@utep.edu).

T. Senjyu is with the Department of Electrical and Electronics Engineering, University of the Ryukyus, Nishihara, Okinawa 9030213, Japan (email: b985542@tec.u-ryukyu.ac.jp).

S. Kamalasadan is with the Energy Production Infrastructure Center, and the Department of Electrical and Computer Engineering, University of North Carolina at Charlotte, Charlotte, NC 28223 USA (e-mail: skamalas@uncc.edu).

2021 IEEE Power & Energy Society General Meeting (PESGM) | 978-1-6654-0507-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/PESGM46819.2021.9637844

978-1-6654-0507-2/21/$31.00 ©2021 IEEE

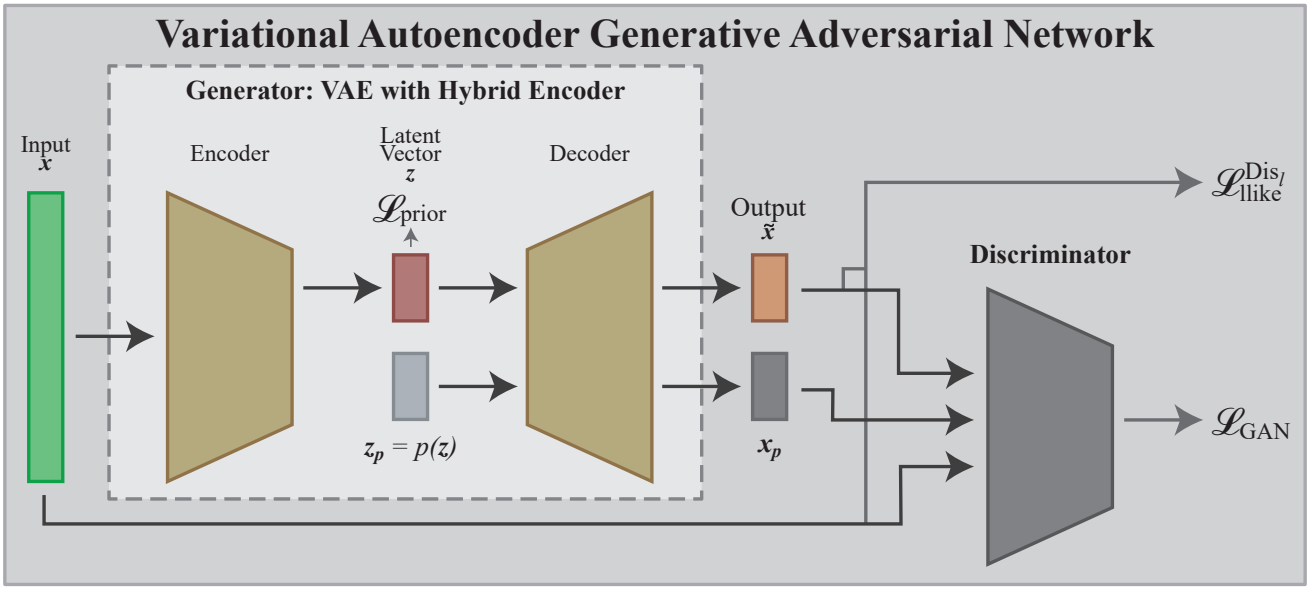

Fig. 1: Hybrid deep generative model illustrating VAEGAN's high-level architecture incorporating the HDNN in the encoder. Data samples forward pass through the encoder, which gives the distribution parameters to sample latent representations. The regularization term ($\mathcal{L}_{prior}$) is computed using the Kullback-Leibler divergence. The decoder maps latent representations back to data space, and the reconstruction error ($\mathcal{L}_{llike}^{Dis_l}$) is computed. A latent representation ($\mathbf{z_p}$) is sampled from prior ($\mathcal{N}(\mathbf{0}, \mathbf{I})$) and its associated reconstruction is passed to the discriminator along with the ones of the true and generated samples. Finally, the VAE and GAN combined models produce the robust $\mathcal{L} = \mathcal{L}_{prior} + \mathcal{L}_{llike}^{Dis_l} + \mathcal{L}_{GAN}$ loss.

applications to measure the performance of the generated PV data samples. This paper explores different DGMs, such as VAE and a hybrid model using the combination of VAE and GAN, i.e., VAEGAN, incorporating convolutional and Long Short-Term Memory Networks (LSTM) layers at the encoding level of the generator. The major advantages of these DGMs over the existing techniques are: (1) lower degree of risk of overfitting, (2) avoiding the need for extensive feature engineering or hyperparameter tuning, and (3) better generalization and reconstruction diversity through learning the joint distribution over the real data and their corresponding latent representations.

This paper is organized as follows. Section II describes the data set. Section III introduces the considered DGMs for PV synthetic data generation. In Section IV, the experimental results are presented and compared. Finally, the conclusion and future work are discussed in Section V.

II. Data Description

The PV dataset, which is acquired from Ashland, Oregon for the purpose of training and testing of the considered DGMs, consists of hourly data samples of PV power, Global Solar Radiation (GSR), and temperature [11], [12]. The first step in preparing the dataset involves segregating the samples into spring, fall, summer, and winter seasons. The next step involves the normalization of the features within the unit range [0, 1] using the min-max function defined in [13]. The last step requires framing the season datasets as multivariate unsupervised learning problems where the inputs and outputs for the considered DGMs correspond to hourly samples, including the features mentioned above.

III. Deep Generative Models

This section provides a background of VAE and GAN models and further describes how they are incorporated into the VAEGAN (see Fig. 1) with the purpose of improving the VAE with a robust loss function to measure the reconstruction performance during training.

A. Variational Autoencoder

The VAE [14] comprises two networks that *encode* data samples $\mathbf{x}$ into latent vectors $\mathbf{z} = Enc(\mathbf{x}) = q(\mathbf{z}|\mathbf{x})$ and *decode* latent representations back to data space $\tilde{\mathbf{x}} = Dec(\mathbf{z}) = p(\mathbf{x}|\mathbf{z})$. The encoder is regularized by making the latent distribution $p(\mathbf{z})$ as similar as possible to the prior $\mathbf{z}$, which is typically chosen as $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The loss is equal to the negative expected log likelihood plus the prior regularization (Kullback-Leibler divergence) term:

$$\mathcal{L}_{VAE} = -\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[logp(\mathbf{x}|\mathbf{z})] + D_{KL}(q(\mathbf{z}|\mathbf{x})||p(\mathbf{z})). \quad (1)$$

B. Generative Adversarial Network

The GAN [15] incorporates the *generator* and *discriminator* networks that maps latent representations z to data space and assigns probability $y = Dis(\mathbf{x})$ that $\mathbf{x}$ is an actual data sample and probability $1 - y$ that $\mathbf{x}$ is generated using the generator network, respectively. The loss function objective is to obtain the binary classifier that discriminates between the true and generated data samples while encouraging the generator to fit the underlying distribution of the true data samples $\mathcal{L}_{GAN} = log(Dis(\mathbf{x})) + log(1 - Dis(Gen(\mathbf{z})))$, where $\mathbf{x}$ and $\mathbf{z}$ are the data samples and latent vector, respectively.

C. Variational Autoencoders Generative Adversarial Network

The VAEGAN takes advantage of the observation that while the discriminator discerns between true and generated data, it also learns the joint distribution over the real data and its corresponding latent representations. The characteristics of sequential dependencies from historical trends in the data learned by the discriminator are integrated into the VAE in terms of a robust reconstruction loss. This results in the combination of the GAN's high-quality generative capabilities and the encoding effectiveness of the VAE –as a feature extractor– to produce latent representations.

In order to achieve a robust VAE error, the expected log likelihood is replaced with a reconstruction error in terms of the GAN discriminator, which introduces a Gaussian observation model for the hidden representation of the l-th layer of the discriminator $p(Dis_l(\mathbf{x})|\mathbf{z}) = \mathcal{N}(Dis_l(\mathbf{x})|Dis_l(\tilde{\mathbf{x}}), \mathbf{I})$, where $\tilde{\mathbf{x}} \sim Dec(\mathbf{z})$ is sample from the decoder of $\mathbf{x}$ and the VAE error becomes $\mathcal{L}_{llike}^{Dis_l} = -\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[logp(Dis_l(\mathbf{x})|\mathbf{z})]$. The combined model considers the following loss function:

$$\mathcal{L} = \mathcal{L}_{prior} + \mathcal{L}_{llike}^{Dis_l} + \mathcal{L}_{GAN}. \tag{2}$$

A detailed description of the VAEGAN's encoder, decoder, and discriminator network architectures is presented next, and for which, we refer to Fig. 1 for the steps followed through the training procedure.

1) Input Layer: This layer distributes the data feature vectors to the VAEGAN model. The data samples contain vectors with 24 time-steps (one per hour) and three features, including the historical PV power output, GSR, and temperature.

2) Encoder Network: The encoder includes a modified version of the HDNN [11], combining convolutional and LSTM units at the layer level (see Fig. 2), which allows it to obtain meaningful latent representations through feature extraction and sequential dependency capture from historical trends using the convolutional and LSTM layers, respectively.

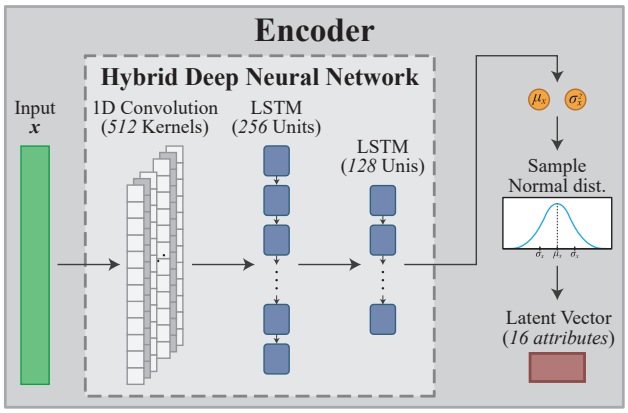

Fig. 2: VAEGAN encoder network high-level architecture incorporating the HDNN that combines a convolutional layer with 512 feature extractors and two LSTM layers of 256 and 128 units. The network computes the parameters –mean and variance– to sample the values of the latent vector using the normal distribution.

The convolutional deep learning algorithm that resembles the organization and functionality of the animal visual cortex [16], where kernels extract features through convolutional operations:

$$\mathbf{S}(\mathbf{n_1}, ..., \mathbf{n_M}) = (\mathbf{I} * \mathbf{K})(n_1, ..., n_M), \tag{3}$$

where $*$, $\mathbf{I}$, and $\mathbf{K}$ are the convolutional operation, input, and kernel. This layer contains 512 kernels with size 3 and stride 2. The LSTM is a type of recurrent neural network with units containing gates that manage the storage and flow of information to avoid the vanishing gradient problem [17]:

$$\mathbf{h_t^j} = \mathbf{o_t^j} tanh(\mathbf{c_t^j}), \tag{4}$$

where $\mathbf{h_t^j}$, $\mathbf{o_t^j}$, and $\mathbf{c_t^j}$ are the recurrent state output, output of the gate that controls the amount of memory outputted, and memory cell, respectively. The subsequent LSTM layers contain 256 and 128 units, respectively. The dense layer at the end of the encoder network outputs two vectors describing the mean ($\boldsymbol{\mu_x}$) and variance ($\boldsymbol{\sigma_x^2}$) of the latent distributions.

3) Latent Vector: Latent representations ($\mathbf{z}$) are sampled from $\mathcal{N} \sim (\boldsymbol{\mu_x}, \boldsymbol{\sigma_x^2})$. This process requires a *reparameterization* to randomly sample an epsilon $\boldsymbol{\epsilon} \sim (\mathbf{0}, \mathbf{I})$ to compute the network's relationship of the distribution parameters with respect to the loss using backpropagation.

4) Decoder Network: The decoder network maps latent representations ($\mathbf{z}$) back to data space ($\tilde{\mathbf{x}}$). The architecture comprises two fully-connected layers of 288 and 144, a reshape layer, and one convolutional transpose layer. The reshape layer simply transforms the output of the last fully-connected layer from 144 to 72. Subsequently, the 1D convolutional layer transforms the data from 72 to 24×3, corresponding to the number of hours and features, respectively, with the number of filters, kernels, and strides set to 3.

5) Discriminator Network: The discriminator network tells if a sample is real or generated with probability $y \in [0, 1]$. Its architecture includes three convolutional layers with 512, 256, and 128 kernels of size 3 and stride 2 and the output with one neuron.

IV. Simulation Results and Discussion

We compare the performance of the VAEGAN with that of a VAE through the *precision*, *recall*, *density*, *coverage*, Discrete Fréchet Distance (DFD), and Maximum Mean Discrepancy (MMD) [18]–[20] in terms of the quality of the generated solar PV synthetic data. Given a real distribution $P(X)$ and a generative model $Q(Y)$, let us assume that samples X_i and Y_j can be sampled, respectively. The *precision* is defined as the portion of $Q(Y)$ that can be generated by $P(X)$, where N and M are the number of real and generated data samples, respectively, and $1_{(\cdot)}$ is the indicator function (see (5)). Similarly, the *recall* is defined as the portion of $P(X)$ that can be generated by $Q(Y)$. The *density* improves upon the *precision* dealing with the overestimation of the manifold around real outliers. The *coverage* improves upon the *recall* by building nearest neighbor manifolds around real samples. The DFD measures the similarity between curves considering the location and order of their points, where L, d, and $d(u_a, v_{b_i})$ are the length of the longest link in L (the coupling between

TABLE I: Comparison of PV synthetic data generation performance between VAE and VAEGAN.

			Season											
			Spring			Summer			Fall			Winter		
		Metric	PV	GSR	T	PV	GSR	T	PV	GSR	T	PV	GSR	T
Model	VAE	Precision	0.4946	0.5679	0.9565	0.4084	0.4826	0.9121	0.4075	0.4730	0.9565	0.3418	0.4003	0.8226
		Recall	0.5100	0.5888	0.9352	0.4286	0.5046	0.9332	0.4277	0.5037	0.8320	0.3600	0.4373	0.9761
		Coverage	0.4792	0.5530	0.9049	0.3910	0.4579	0.8919	0.3915	0.4707	0.7811	0.3254	0.4162	0.8610
		Density	0.4785	0.5597	0.9673	0.3737	0.4551	0.9109	0.3771	0.4517	0.9601	0.2908	0.3713	0.7460
		DFD	0.1025	0.0898	0.0785	0.1139	0.0931	0.0708	0.1373	0.1078	0.0856	0.1384	0.1085	0.1074
		MMD	0.0013	0.0015	0.0055	0.0008	0.0012	0.0070	0.0011	0.0015	0.0097	0.0008	0.0011	0.0106
	VAEGAN	Precision	0.5000	0.5811	0.9606	0.3933	0.4702	0.9693	0.4048	0.4812	0.9103	0.3464	0.4265	0.9106
		Recall	0.5059	0.5883	0.9778	0.4281	0.5046	0.9029	0.4249	0.4977	0.8951	0.3596	0.4368	0.9059
		Coverage	0.4742	0.5507	0.9438	0.4006	0.4776	0.8814	0.3933	0.4725	0.8503	0.3277	0.4176	0.8310
		Density	0.4983	0.5772	0.9698	0.3665	0.4475	0.9665	0.3995	0.4840	0.8810	0.3099	0.3858	0.8812
		DFD	0.1092	0.0911	0.0634	0.0850	0.0746	0.0646	0.1154	0.0916	0.0670	0.1129	0.0958	0.1012
		MMD	0.0015	0.0017	0.0031	0.0007	0.0009	0.0044	0.0021	0.0023	0.0130	0.0008	0.0009	0.0095

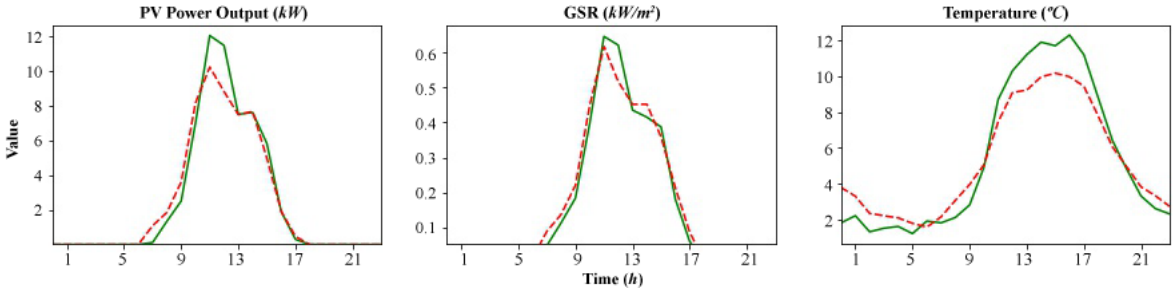

Fig. 3: Comparison between the real (green) and synthetic data generated (red) over a spring day.

P and Q), distance function, and distinct pairs of points from the coupling between P and Q, respectively. The MMD is a distance on the space of probability measures based on the idea of embedding probabilities in a reproducing kernel Hilbert space.

$$
\begin{aligned}
Precision &= \frac{1}{M}\sum_{j=1}^{M} 1_{Y_j \in manifold(X_1,\ldots,X_N)} \\
Recall &= \frac{1}{N}\sum_{i=1}^{N} 1_{X_i \in manifold(Y_1,\ldots,Y_M)} \\
Density &= \frac{1}{kM}\sum_{j=1}^{M}\sum_{i=1}^{N} 1_{Y_j \in B(X_i, NND_k(X_i))} \\
Coverage &= \frac{1}{N}\sum_{i=1}^{N} 1_{\exists j \;\; s.t. \;\; Y_j \in B(X_i, NND_k(X_i))} \\
DFD &= min\{\|L\| = \max_{i=1,\ldots,m} d(u_a, v_{b_i})\} \\
MMD &= \tfrac{1}{N^2}\textstyle\sum_{i=1}^{N}\sum_{j=1}^{N} k(X_i, X_j) \\
&\quad + \tfrac{1}{M^2}\textstyle\sum_{i=1}^{M}\sum_{j=1}^{M} k(Y_i, Y_j) \\
&\quad - \tfrac{2}{NM}\textstyle\sum_{i=1}^{N}\sum_{j=1}^{M} k(X_i, Y_j)
\end{aligned}
\tag{5}
$$

Table I presents the average performance measure values obtained from the considered DGMs. The *coverage* values obtained from the VAE model range between 0.3254 and 0.4792, 0.4162 and 0.5530, and 0.7811 and 0.9049 for the PV power output, GSR, and temperature, respectively, over the seasons. The VAEGAN model obtained *coverage* values in the range of 0.3277 and 0.4742, 0.4176 and 0.5507, and 0.8310 and 0.9438, respectively. The VAE obtained *density* values between 0.2908 and 0.4785, 0.3713 and 0.5597, and 0.7460 and 0.9673, whereas the VAEGAN generated between 0.3099 and 0.4983, 0.3858 and 0.5772, and 0.8810 and 0.9698 across the seasons. Both models showed DFD values as low as 0.0708 for the VAE and 0.0634 for the VAEGAN. In the context of the MMD, the VAE obtained values as low as 0.0008 for the PV power output in summer, whereas the VAEGAN improved it to 0.0007. Both VAE and VAEGAN models show similar performance with high values for the *precision*, *recall*, *coverage*, and *density*; and low values for the DFD and MMD. Both of these VAE and VAEGAN models excel better at generating synthetic data over the spring with *density* as high as 0.9673 and 0.9698 and in the summer with MMD as low as 0.0008 and 0.0007, respectively, based on the average values obtained across the seasons.

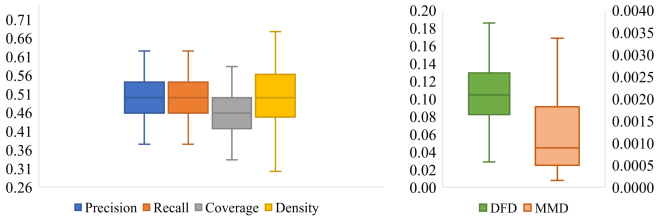

Fig. 4: Performance measure deviation for the PV power output over the spring season obtained from the VAEGAN model.

Fig. 3 shows the synthetic data generated (red) for the true data sample (green) corresponding to March 11, 2011. From the reconstructions, it can be noticed that the VAEGAN can perform feature extraction through the generator, which is a VAE incorporating a modified version of the HDNN at the encoding level. This and the robust loss function present in

the VAEGAN make the best use of the features extracted to produce high-quality synthetic samples. A box-whisker plot in Fig. 4 shows the distribution of the values for each measure obtained for the PV power output by the VAEGAN over the spring, the season in which it obtained the best measure values. The model can produce reconstructions for the PV power output with *precision*, *recall*, *coverage*, and *density* values spanning between 0.33 and 0.625, 0.33 and 0.625, 0.333 and 0.583, and 0.250 and 0.677, respectively. In terms of DFD and MMD, the measures spread from 0.029 to 0.289 and from 0.0002 to 0.0186, respectively. However, the test results and the overall performance demonstrate that both individual (VAE) and hybrid (VAEGAN) deep generative models are capable of generating high-quality reconstructions case-wise over the seasons.

Research on generative models for synthetic data generation is a relevant challenge in the context of smart power grid. This paper contributed to solving the problem of PV synthetic data generation over the seasons through the exploration of DGMs and presented the appropriate performance metrics in terms of their generalization and diversity considering the underlying distributions of the real and generated PV samples.

V. Conclusion

This paper presented the VAE and the GAN as the two most relevant and popular DGMs due to their good performance in other applications (e.g., data compression and denoising, anomaly detection, etc.) and described the procedure of combining these two efficient models to form the VAEGAN for a robust loss function with an application to generate PV synthetic data. Furthermore, the associated model's encoder network implements a modified version of the HDNN, combining convolutional and LSTM layers, which allows both VAE and VAEGAN models to perform efficient feature extraction obtaining high-quality synthetic PV data samples. Additionally, we described a set of previously developed metrics –*precision*, *recall*, *density*, *coverage*, DFD, and MMD– that measure the generalization and diversity capabilities of both VAE and VAEGAN models in generating synthetic PV data. The performance of the DGMs using the accuracy measures presented in this paper indicated high-quality generated samples by both models, which can also be applied and extended to other down-the-stream tasks including microgrid simulation for cybersecurity without compromising the security and the privacy associated with the power grid and prosumers, respectively. The DGMs are successful to this end because they include the HDNN in the encoder, which performs salient feature extraction and capture sequence dependency trends in data. Future work includes applying the considered DGMs to (i) cybersecurity and (ii) interoperability in the context of smart grid applications.

Acknowledgment

The authors would like to express sincere gratitude to the National Science Foundation (NSF) (Awards #2021470 and #1840424) for supporting this project partially. The results presented in this work were obtained using the Chameleon testbed supported by the NSF.

References

[1] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125–1142, 2017.

[2] C. O. Inácio and C. L. T. Borges, "Stochastic model for generation of high-resolution irradiance data and estimation of power output of photovoltaic plants," *IEEE Transactions on Sustainable Energy*, vol. 9, no. 2, pp. 952–960, 2018.

[3] M. Lave, M. J. Reno, and R. J. Broderick, "Creation and value of synthetic high-frequency solar inputs for distribution system qsts simulations," in *2017 IEEE 44th Photovoltaic Specialist Conference (PVSC)*, 2017, pp. 3031–3033.

[4] T. Power, G. Verbič, and A. C. Chapman, "A nonparametric bayesian methodology for synthesizing residential solar generation and demand data," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2511–2519, 2020.

[5] C. G. Turhan and H. S. Bilge, "Recent trends in deep generative models: A review," in *2018 3rd International Conference on Computer Science and Engineering (UBMK)*, 2018, pp. 574–579.

[6] G. Yang, S. Yu, H. Dong, G. Slabaugh, P. L. Dragotti, X. Ye, F. Liu, S. Arridge, J. Keegan, Y. Guo, and D. Firmin, "Dagan: Deep de-aliasing generative adversarial networks for fast compressed sensing mri reconstruction," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1310–1321, 2018.

[7] J. M. Wolterink, T. Leiner, M. A. Viergever, and I. Išgum, "Generative adversarial networks for noise reduction in low-dose ct," *IEEE Transactions on Medical Imaging*, vol. 36, no. 12, pp. 2536–2545, 2017.

[8] T. Wang, M. Qiao, Z. Lin, C. Li, H. Snoussi, Z. Liu, and C. Choi, "Generative neural networks for anomaly detection in crowded scenes," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1390–1399, 2019.

[9] A. B. L. Larsen, S. K. Sønderby, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," *CoRR*, vol. abs/1512.09300, 2015.

[10] Y. Xian, S. Sharma, B. Schiele, and Z. Akata, "F-vaegan-d2: A feature generating framework for any-shot learning," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 10 267–10 276.

[11] D. R. de Jesús, P. Mandal, S. Chakraborty, and T. Senjyu, "Solar pv power prediction using a new approach based on hybrid deep neural network," in *2019 IEEE Power Energy Society General Meeting*, 2019.

[12] D. A. R. De Jesús, P. Mandal, M. Velez-Reyes, S. Chakraborty, and T. Senjyu, "Data fusion based hybrid deep neural network method for solar pv power forecasting," in *2019 North American Power Symposium (NAPS)*, 2019, pp. 1–6.

[13] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.

[14] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, *Autoencoding beyond pixels using a learned similarity metric*, 2016.

[15] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, *Generative adversarial networks*, 2014.

[16] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998, ISSN: 0018-9219.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[18] M. F. Naeem, S. J. Oh, Y. Uh, Y. Choi, and J. Yoo, *Reliable fidelity and diversity metrics for generative models*, 2020.

[19] T. Eiter and H. Mannila, "Computing discrete frechet distance," May 1994.

[20] I. O. Tolstikhin, B. K. Sriperumbudur, and B. Schölkopf, "Minimax estimation of maximum mean discrepancy with radial kernels," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29, Curran Associates, Inc., 2016, pp. 1930–1938.