Noise Learning-Based Denoising Autoencoder

Woong-Hee Lee, Mustafa Ozger, Ursula Challita, and Ki Won Sung, *Member, IEEE*

***Abstract*— This letter introduces a new denoiser that modifies the structure of denoising autoencoder (DAE), namely noise learning based DAE (nlDAE). The proposed nlDAE learns the noise of the input data. Then, the denoising is performed by subtracting the regenerated noise from the noisy input. Hence, nlDAE is more effective than DAE when the noise is simpler to regenerate than the original data. To validate the performance of nlDAE, we provide three case studies: signal restoration, symbol demodulation, and precise localization. Numerical results suggest that nlDAE requires smaller latent space dimension and smaller training dataset compared to DAE.**

***Index Terms*— Machine learning, noise learning based denoising autoencoder, signal restoration, symbol demodulation, precise localization.**

I. Introduction

MACHINE learning (ML) has recently received much attention as a key enabler for future wireless communications [1]–[3]. While the major research effort has been put to deep neural networks, there are enormous number of Internet of Things (IoT) devices that are severely constrained on the computational power and memory size. Therefore, the implementation of efficient ML algorithms is an important challenge for IoT devices, as they are energy and memory limited. Denoising autoencoder (DAE) is a promising technique to improve the performance of IoT applications by denoising the observed data that consists of the original data and the noise [4]. DAE is a neural network model for the construction of the learned representations robust to an addition of noise to the input samples [5], [6]. The representative feature of DAE is that the dimension of the latent space is smaller than the size of the input vector. It means that the neural network model is capable of encoding and decoding through a smaller dimension where the data can be represented.

The main contribution of this letter is to improve the efficiency and performance of DAE with a modification of its structure. Consider a noisy observation Y which consists of the original data X and the noise N, i.e., $Y = X + N$. From the information theoretical perspective, DAE attempts to minimize the expected reconstruction error by maximizing a lower bound on mutual information $I(X;Y)$. In other words, Y should capture the information of X as much as possible although Y is a function of the noisy input. Additionally, from the manifold learning perspective, DAE can be seen as a way to find a manifold where Y represents the data into a low dimensional latent space corresponding to X. However, we often face the problem that the stochastic feature of X to be restored is too complex to regenerate or represent. This is called the curse of dimensionality, i.e., the dimension of latent space for X is still too high in many cases.

What can we do if N is simpler to regenerate than X? It will be more effective to learn N and subtract it from Y instead of learning X directly. In this light, we propose a new denoising framework, named as noise learning based DAE (nlDAE). The main advantage of nlDAE is that it can maximize the efficiency of the ML approach (e.g., the required dimension of the latent space or size of training dataset) for capability-constrained devices, e.g., IoT, where N is typically easier to regenerate than X owing to their stochastic characteristics. To verify the advantage of nlDAE over the conventional DAE, we provide three practical applications as case studies: signal restoration, symbol demodulation, and precise localization.

The following notations will be used throughout this letter.

- $\mathrm{Ber}, \mathrm{Exp}, \mathcal{U}, \mathcal{N}, \mathcal{CN}$: the Bernoulli, exponential, uniform, normal, and complex normal distributions, respectively.
- $\mathbf{x}, \mathbf{n}, \mathbf{y} \in \mathbb{R}^P$: the realization vectors of random variables X, N, Y, respectively, whose dimensions are P.
- $P' (< P)$: the dimension of the latent space.
- $\mathbf{W} \in \mathbb{R}^{P' \times P}, \mathbf{W}' \in \mathbb{R}^{P \times P'}$: the weight matrices for encoding and decoding, respectively.
- $\mathbf{b} \in \mathbb{R}^{P'}, \mathbf{b}' \in \mathbb{R}^P$: the bias vectors for encoding and decoding, respectively.
- $\mathcal{S}$: the sigmoid function, acting as an activation function for neural networks, i.e., $\mathcal{S}(a) = \frac{1}{1+e^{-a}}$, and $\mathcal{S}(\mathbf{a}) = (\mathcal{S}(\mathbf{a}[1]), \cdots, \mathcal{S}(\mathbf{a}[P]))^T$ where $\mathbf{a} \in \mathbb{R}^P$ is an arbitrary input vector.
- f_θ: the encoding function where the parameter θ is $\{\mathbf{W}, \mathbf{b}\}$, i.e., $f_\theta(\mathbf{y}) = \mathcal{S}(\mathbf{W}\mathbf{y} + \mathbf{b})$.
- $g_{\theta'}$: the decoding function where the parameter θ' is $\{\mathbf{W}', \mathbf{b}'\}$, i.e., $g_{\theta'}(f_\theta(\mathbf{y})) = \mathcal{S}(\mathbf{W}' f_\theta(\mathbf{y}) + \mathbf{b}')$.
- M: the size of training dataset.
- L: the size of test dataset.

II. Method of nlDAE

In the traditional estimation problem of signal processing, N is treated as an obstacle to the reconstruction of X. Therefore, most of the studies have focused on restoring X as much as possible, which can be expressed as a function of X and N. Along with this philosophy, ML-based denoising

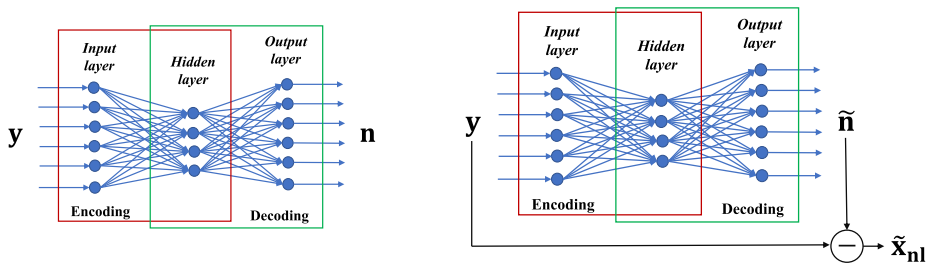
Manuscript received May 17, 2021; accepted June 14, 2021. Date of publication June 23, 2021; date of current version September 10, 2021. This work was supported by a Korea University Grant. This research was supported by the BK21 FOUR (Fostering Outstanding Universities for Research) funded by the Ministry of Education (MOE, Korea) and National Research Foundation of Korea (NRF). This work was partly funded by the European Union Horizon 2020 Research and Innovation Programme under the EU/KR PriMO-5G project with grant agreement No 815191. The associate editor coordinating the review of this letter and approving it for publication was C. Studer. *(Corresponding author: Ki Won Sung.)*

Woong-Hee Lee is with the Department of Control and Instrumentation Engineering, Korea University, Sejong-si 30019, Republic of Korea (e-mail: woongheelee@korea.ac.kr).

Mustafa Ozger and Ki Won Sung are with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, 164 40 Stockholm, Sweden (e-mail: ozger@kth.se; sungkw@kth.se).

Ursula Challita is with Ericsson Research, 164 83 Stockholm, Sweden (e-mail: ursula.challita@ericsson.com).

Digital Object Identifier 10.1109/LCOMM.2021.3091800



(a) Training phase (b) Test phase

Fig. 1. An illustration of the concept of nlDAE.

techniques, e.g., DAE, have also been developed in various signal processing fields with the aim of maximizing the ability to restore X from Y. Unlike the conventional approaches, we hypothesize that, if N has a simpler statistical characteristic than X, it will be better to subtract from Y after restoring N.

We first look into the mechanism of DAE to build neural networks. Recall that DAE attempts to regenerate the original data $\mathbf{x}$ from the noisy observation $\mathbf{y}$ via training the neural network. Thus, the parameters of a DAE model can be optimized by minimizing the average reconstruction error in the training phase as follows:

$$\theta^*, \theta'^* = \arg\min_{\theta,\theta'} \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}\big(\mathbf{x}^{(i)}, g_{\theta'}(f_\theta(\mathbf{y}^{(i)}))\big), \tag{1}$$

where $\mathcal{L}$ is a loss function such as squared error between two inputs. Then, the j-th regenerated data $\tilde{\mathbf{x}}^{(j)}$ from $\mathbf{y}^{(j)}$ in the test phase can be obtained as follows for all $j \in \{1, \cdots, L\}$:

$$\tilde{\mathbf{x}}^{(j)} = g_{\theta'^*}(f_{\theta^*}(\mathbf{y}^{(j)})). \tag{2}$$

It is noteworthy that, if there are two different neural networks which attempt to regenerate the original data and the noise from the noisy input, the linear summation of these two regenerated data would be different from the input. This means that either $\mathbf{x}$ or $\mathbf{n}$ is more effectively regenerated from $\mathbf{y}$. Therefore, we can hypothesize that learning N, instead of X, from Y can be beneficial in some cases even if the objective is still to reconstruct X. This constitutes the fundamental idea of nlDAE.

The training and test phases of nlDAE are depicted in Fig. 1. The parameters of nlDAE model can be optimized as follows for all $i \in \{1, \cdots, M\}$:

$$\theta^*_{nl}, \theta'^*_{nl} = \arg\min_{\theta,\theta'} \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}\big(\mathbf{n}^{(i)}, g_{\theta'}(f_\theta(\mathbf{y}^{(i)}))\big). \tag{3}$$

Notice that the only difference from (1) is that $\mathbf{x}^{(i)}$ is replaced by $\mathbf{n}^{(i)}$. Let $\tilde{\mathbf{x}}^{(j)}_{nl}$ denote the j-th regenerated data based on nlDAE, which can be represented as follows for all $j \in \{1, \cdots, L\}$:

$$\tilde{\mathbf{x}}^{(j)}_{nl} = \mathbf{y}^{(j)} - g_{\theta'^*_{nl}}(f_{\theta^*_{nl}}(\mathbf{y}^{(j)})). \tag{4}$$

To provide the readers with insights into nlDAE, we examine two simple examples where the standard deviation of X is fixed as 1, i.e., $\sigma_X = 1$, and that of N varies. $Y = X + N$ is comprised as follows:

- Example 1: $X \sim \mathcal{U}(0, 2\sqrt{3})$ and $N \sim \mathcal{N}(0, \sigma_N)$.
- Example 2: $X \sim \text{Exp}(1)$ and $N \sim \mathcal{N}(0, \sigma_N)$.

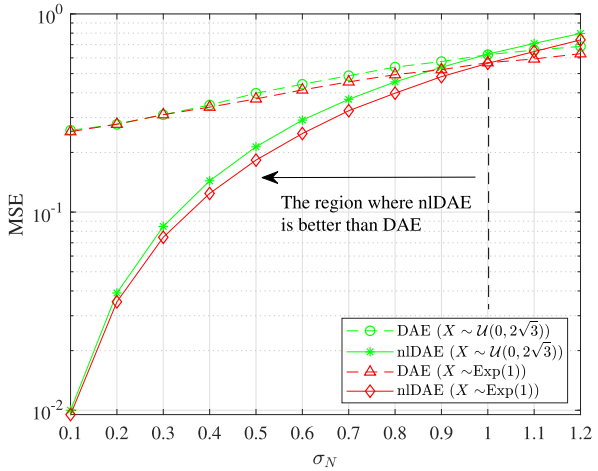


Fig. 2. A simple example of comparison between DAE and nlDAE: reconstruction error according to σ_N.

Fig. 2 describes the performance comparison between DAE and nlDAE in terms of mean squared error (MSE) for the two examples.[1] Here, we set $P = 12$, $P' = 9$, $M = 10000$, and $L = 5000$. It is observed that nlDAE is superior to DAE when σ_N is smaller than σ_X in Fig. 2. The gap between nlDAE and DAE widens with lower σ_X. This implies that the standard deviation is an important factor when we select the denoiser between DAE and nlDAE.

These examples show the consideration of whether X or N is easier to be regenerated, which is highly related to differential entropy of each random variable, $H(X)$ and $H(N)$ [7]. The differential entropy is normally an increasing function over the standard deviation of the corresponding random variable, e.g., $H(N) = \log(\sigma_N \sqrt{2\pi e})$. Naturally, it is efficient to reconstruct a random variable with a small amount of information, and the standard deviation can be a good indicator.

[1]Throughout this letter, the squared error and the scaled conjugate gradient are applied as the loss function and the optimization method, respectively.

III. Case Studies

To validate the advantage of nlDAE over the conventional DAE in practical problems, we provide three applications for IoT devices in the following subsections. We assume that the noise follows Bernoulli and normal distributions, respectively, in the first two cases, which are the most common noise modeling. The third case deals with noise that follows a distribution expressed as a mixture of various random variables. For all the studied use cases, we select the DAE as the conventional denoiser as a baseline for performance comparison. We present the case studies in the first three subsections. Then, we discuss the experimental results in Sec. III-D.

A. Case Study I: Signal Restoration

In this use case, the objective is to recover the original signal from the noisy signal which is modeled by the corruptions over samples.

1) Model: The sampled signal of randomly superposed sinusoids, e.g., the recorded acoustic wave, is the summation of samples of k damped sinusoidal waves which can be represented as follows:

$$\mathbf{x} = \Big\{ \sum_{l=1}^{k} V_l e^{-\gamma_l n \Delta t} \cos(2\pi f_l n \Delta t) \Big\}_{n=0}^{P-1}, \tag{5}$$

where V_l, γ_l, and f_l are the peak amplitude, the damping factor, and the frequency of the l-th signal, respectively. Here, the time interval for sampling, Δt, is set to satisfy the Nyquist theorem, i.e., $\frac{1}{2\Delta t} > \max\{f_1, \cdots, f_k\}$. To consider the corruption of $\mathbf{x}$, let us assume that the probability of corruption for each sample follows the Bernoulli distribution $\text{Ber}(p_{cor})$, which indicates the corruption with the probability p_{cor}. In addition, let $\mathbf{b} \in \{0,1\}^P$ denote the realization of $\text{Ber}(p_{cor})$ over P samples. Naturally, the corrupted signal, $\mathbf{y} \in \mathbb{R}^P$, can be represented as follows:

$$\mathbf{y} = \mathbf{x} + C\mathbf{b}, \tag{6}$$

where C is a constant representing the sample corruption.

2) Application of nlDAE: Based on (6), the denoised signal $\tilde{\mathbf{x}}_{nl}^{(j)}$ can be represented by

$$\tilde{\mathbf{x}}_{nl}^{(j)} = \mathbf{x}^{(j)} + C\mathbf{b}^{(j)} - g_{\theta'^{*}_{nl}}(f_{\theta^{*}_{nl}}(\mathbf{x}^{(j)} + C\mathbf{b}^{(j)}), \tag{7}$$

where

$$\theta^{*}_{nl}, \theta'^{*}_{nl} = \arg\min_{\theta,\theta'} \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}\big(C\mathbf{b}^{(i)}, g_{\theta'}(f_{\theta}(\mathbf{x}^{(i)} + C\mathbf{b}^{(i)}))\big).$$

3) Experimental Parameters: We evaluate the performance of the proposed nlDAE in terms of the MSE of restoration. For the experiment, the magnitude of noise C is set to 1 for simplicity. In addition, V_l, γ_l, and f_l follow $\mathcal{N}(0,1)$, $\mathcal{U}(0, 10^3)$, and $\mathcal{U}(0, 10\text{ kHz})$, respectively, for all l. The sampling time interval Δt is set to 0.5×10^{-4} second, and the number of samples P is 12. We set $P' = 9$, $p_{cor} = 0.9$, and $M = 10000$ unless otherwise specified.

B. Case Study II: Symbol Demodulation

Here, the objective is to improve the symbol demodulation quality through denoising the received signal that consists of channel, symbols, and additive noise.

1) Model: Consider an orthogonal frequency-division multiplexing (OFDM) system with P subcarriers where the subcarrier spacing is expressed by Δf. Let $\mathbf{d} \in \mathbb{C}^P$ be a sequence in frequency domain. $\mathbf{d}[n]$ is the n-th element of $\mathbf{d}$ and denotes the symbol transmitted over the n-th subcarrier. In addition, let K denote the pilot spacing for channel estimation. Furthermore, the channel impulse response (CIR) can be modeled by the sum of Dirac-delta functions as follows:

$$h(t,\tau) = \sum_{l=0}^{L_p-1} \alpha_l \delta(t - \tau_l), \tag{8}$$

where α_l, τ_l, and L_p are the complex channel gain, the excess delay of l-th path, and the number of multipaths, respectively. Let $\mathbf{x} \in \mathbb{C}^P$ denote the discrete signal obtained by P-point fast Fourier transform (FFT) after the sampling of the signal experiencing the channel at the receiver, which can be represented as follows:

$$\mathbf{x} = \mathbf{d} \odot \mathbf{h} = \{\mathbf{d}[n] \sum_{l=0}^{L_p-1} \alpha_l e^{-j2\pi n \Delta f \tau_l}\}_{n=0}^{P-1}, \tag{9}$$

where $\odot$ denotes the operator of the Hadamard product. Here, $\mathbf{h} \in \mathbb{C}^P$ is the channel frequency response (CFR), which is the P-point FFT of $h(t,\tau)$. In addition, let $\mathbf{n} \in \mathbb{C}^P$ denote the realization of the random variable $N \sim \mathcal{CN}(0, \sigma_N)$. Finally, $\mathbf{y}(= \mathbf{d} \odot \mathbf{h} + \mathbf{n})$ is the noisy observed signal.

Our goal is to minimize the symbol error rate (SER) over $\mathbf{d}$ by maximizing the quality of denoising $\mathbf{y}$. We assume the method of channel estimation is fixed as the cubic interpolation [8] to focus on the performance of denoising the received signal.

2) Application of nlDAE: To consider the complex-valued data, we separate it into real and imaginary parts. $\Re$ and $\Im$ denote the operators capturing real and imaginary parts of an input, respectively. Thus, $\tilde{\mathbf{x}}_{nl}^{(j)}$ is the regenerated $\mathbf{d}^{(j)} \odot \mathbf{h}^{(j)}$ by denoising $\mathbf{y}^{(j)}$, which can be represented by

$$\begin{aligned}\tilde{\mathbf{x}}_{nl}^{(j)} = {} & \Re(\mathbf{y}^{(j)}) - g_{\theta'^{*}_{nl,R}}(f_{\theta^{*}_{nl,R}}(\Re(\mathbf{y}^{(j)}))) \\ & + i(\Im(\mathbf{y}^{(j)}) - g_{\theta'^{*}_{nl,I}}(f_{\theta^{*}_{nl,I}}(\Im(\mathbf{y}^{(j)})))),\end{aligned} \tag{10}$$

where

$$\theta^{*}_{nl,R}, \theta'^{*}_{nl,R} = \arg\min_{\theta,\theta'} \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}\big(\Re(\mathbf{n}^{(i)}), g_{\theta'}(f_{\theta}(\Re(\mathbf{y}^{(i)})))\big),$$

$$\theta^{*}_{nl,I}, \theta'^{*}_{nl,I} = \arg\min_{\theta,\theta'} \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}\big(\Im(\mathbf{n}^{(i)}), g_{\theta'}(f_{\theta}(\Im(\mathbf{y}^{(i)})))\big).$$

Finally, the receiver estimates $\mathbf{h}$ with the predetermined pilot symbols, i.e., $\mathbf{d}[nK+1]$ where $n = 0, 1, \cdots$, and demodulates $\mathbf{d}$ based on the estimate of $\mathbf{h}$ and the regenerated $\tilde{\mathbf{x}}_{nl}$.

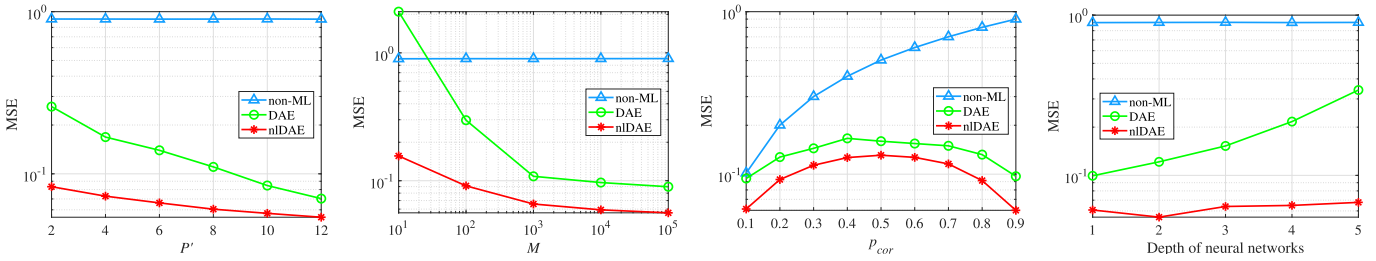


Fig. 3. Case study I (signal restoration): MSE according to (a) the dimension of latent space; (b) the size of training dataset; (c) p_{cor}; and (d) the depth of neural networks.

3) Experimental Parameters: The performance of the proposed nlDAE is evaluated when $L = 5000$. For the simulation parameters, we set 4 QAM, $P = 12$, $\Delta f = 15$ kHz, $L_p = 4$, and $K = 3$. We further assume that $\alpha \sim \mathcal{CN}(0, 1)$ and $\tau \sim \mathcal{U}(0, 10^{-6})$. Furthermore, $P' = 9$, SNR$= 5$ dB, and $M = 10000$ unless otherwise specified. We also provide the result of non-ML (i.e., only cubic interpolation).

C. Case Study III: Precise Localization

The objective of this case study is to improve the localization quality through denoising the measured distance which is represented by the quantized value of the mixture of the true distance and error factors.

1) Model: Consider a 2-D localization where P reference nodes and a single target node are randomly distributed. We estimate the position of the target node with the knowledge of the locations of P reference nodes. Let $\mathbf{x} \in \mathbb{R}^P$ denote the vector of true distances from P reference nodes to the target node when X denotes the distance between two random points in a 2-D space. We consider three types of random variables for the noise added to the true distance as follows:

- N_N: ranging error dependent on signal quality.
- N_U: ranging error due to clock asynchronization.
- N_B: non line-of-sight (NLoS) event.

We assume that N_N, N_U, N_B follow the normal, uniform, and Bernoulli distributions, respectively. Hence, we can define the random variable for the noise N as follows:

$$N = N_N + N_U + R_{NLoS} N_B, \tag{11}$$

where R_{NLoS} is the distance bias in the event of NLoS. Note that N does not follow any known probability distribution because it is a convolution of three different distributions. Besides, we assume that the distance is measured by time of arrival (ToA). Thus, we define the quantization function $\mathcal{Q}_B$ to represent the measured distance with the resolution of B, e.g., $\mathcal{Q}_{10}(23) = 20$. In addition, the localization method based on multi-dimensional scaling (MDS) is utilized to estimate the position of the target node [9].

2) Application of nlDAE: In this case study, we consider the discrete values quantized by the function $\mathcal{Q}_B$. Here, $\tilde{\mathbf{x}}_{nl}^{(j)}$ can be represented as follows:

$$\tilde{\mathbf{x}}_{nl}^{(j)} = \mathcal{Q}_B(\mathbf{y}^{(j)}) - g_{\theta'^{*}_{nl,R}}(f_{\theta^{*}_{nl,R}}(\mathcal{Q}_B(\mathbf{y}^{(j)}))), \tag{12}$$

where

$$\theta^{*}_{nl,R}, \theta'^{*}_{nl,R} = \arg\min_{\theta,\theta'} \frac{1}{M} \times \sum_{i=1}^{M} \mathcal{L}\big(\mathcal{Q}_B(\mathbf{n}^{(i)}), g_{\theta'}(f_{\theta}(\mathcal{Q}_B(\mathbf{y}^{(i)})))\big).$$

Thus, $\tilde{\mathbf{x}}_{nl}$ is utilized for the estimation of the target node position in nlDAE-assisted MDS-based localization.

3) Experimental Parameters: The performance of the proposed nlDAE is evaluated via $L = 5000$. In this simulation, 12 reference nodes and one target node are uniformly distributed in a 100×100 square. We assume that $N_N \sim \mathcal{N}(0, 10)$, $N_U \sim \mathcal{U}(0, 20)$, $N_B \sim$ Ber(0.2), and $R_{NLoS} = 50$. The distance resolution B is set to 10 for the quantization function $\mathcal{Q}_B$. Note that $P' = 9$, $p_{NLoS} = 0.2$, and $M = 10000$ unless otherwise specified. We also provide the result of non-ML (i.e., only MDS based localization).

D. Analysis of Experimental Results

Fig. 3(a), Fig. 4(a), and Fig. 5(a) show the performance of the three case studies with respect to P', respectively. nlDAE outperforms non-ML and DAE for all ranges of P'. Particularly with small values of P', nlDAE continues to perform well, whereas DAE loses its merit. This means that nlDAE provides a good denoising performance even with an extremely small dimension of latent space if the training dataset is sufficient.

The impact of the size of training dataset is depicted in Fig. 3(b), Fig. 4(b), and Fig. 5(b). nlDAE starts to outperform non-ML with M less than 100. Conversely, DAE requires about an order higher M to perform better than non-ML. Furthermore, nlDAE converges faster than DAE, thus requiring less training data than DAE.

In Fig. 3(c), Fig. 4(c), and Fig. 5(c), the impact of a noise-related parameter for each case study is illustrated. When the noise occurs according to a Bernoulli distribution in Fig. 3(c), the performance of ML algorithms (both nlDAE and DAE) exhibits a concave behavior. This is because the variance of Ber(p) is given by $p(1-p)$. Similar phenomenon is observed in Fig. 5(c) because the Bernoulli event of NLoS constitutes a part of localization noise. As for non-ML, the performance worsens as the probability of noise occurrence increases in both cases. Fig. 4(c) shows that the SER performance

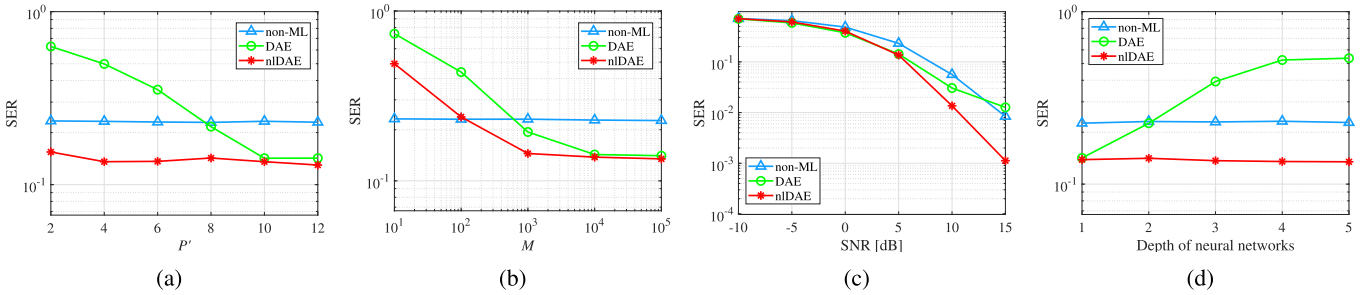


Fig. 4. Case study II (symbol demodulation): SER according to (a) the dimension of latent space; (b) the size of training dataset; (c) SNR; and (d) the depth of neural networks.

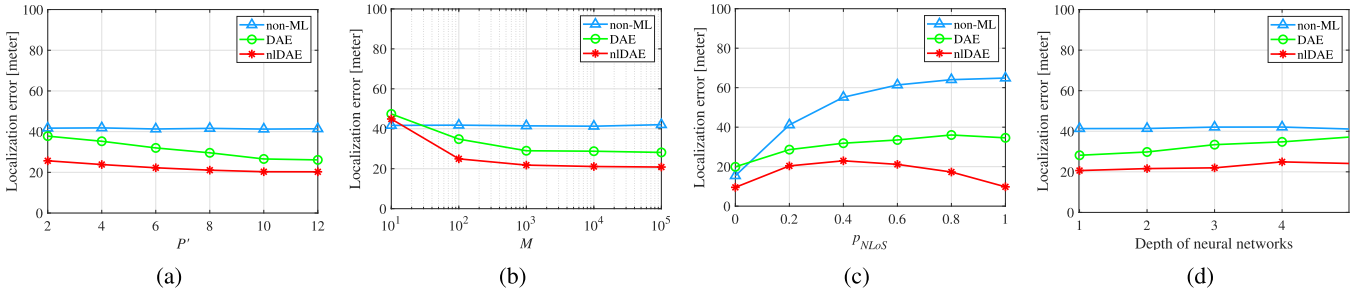


Fig. 5. Case study III (precise localization): Localization error according to (a) the dimension of latent space; (b) the size of training dataset; (c) p_{NLoS}; and (d) the depth of neural networks.

of nlDAE improves rapidly as the SNR increases. In all experiments, nlDAE gives superior performance than other schemes.

Thus far, the experiments have been conducted with a single hidden layer. Fig. 3(d), Fig. 4(d), and Fig. 5(d) show the effect of the depth of the neural network. The performance of nlDAE is almost invariant, which suggests that nlDAE is not sensitive to the number of hidden layers. On the other hand, the performance of DAE worsens quickly as the depth increases owing to overfitting in two cases.

In summary, nlDAE outperforms DAE over the whole experiments. nlDAE is observed to be more efficient for the underlying use cases than DAE because it requires smaller latent space and less training data. Furthermore, nlDAE is more robust to the change of the parameters related to the design of the neural network, e.g., the network depth.

IV. Conclusion and Future Work

We introduced a new denoiser framework based on the neural network, namely nlDAE. This is a modification of DAE in that it learns the noise instead of the original data. The fundamental idea of nlDAE is that learning noise can provide a better performance depending on the stochastic characteristics (e.g., standard deviation) of the original data and noise. We applied the proposed mechanism to the practical problems for IoT devices such as signal restoration, symbol demodulation, and precise localization. The numerical results support that nlDAE is more efficient than DAE in terms of the required dimension of the latent space and the size of training dataset, thus rendering it more suitable for capability-constrained conditions. Applicability of nlDAE to other domains, e.g., image inpainting, remains as a future work. Furthermore, information theoretical criteria of decision making for the selection between or a combination of DAE and nlDAE is an interesting further research.

References

[1] U. Challita, H. Ryden, and H. Tullberg, "When machine learning meets wireless cellular networks: Deployment, challenges, and applications," *IEEE Commun. Mag.*, vol. 58, no. 6, pp. 12–18, Jun. 2020.

[2] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3039–3071, 4th Quart., 2019.

[3] A. Azari, M. Ozger, and C. Cavdar, "Risk-aware resource allocation for URLLC: Challenges and strategies with machine learning," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 42–48, Mar. 2019.

[4] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3072–3108, 4th Quart., 2019.

[5] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 1096–1103.

[6] Y. Bengio, L. Yao, G. Alain, and P. Vincent, "Generalized denoising auto-encoders as generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 899–907.

[7] C. Marsh, "Introduction to continuous entropy," Dept. Comput. Sci., Princeton Univ., Princeton, NJ, USA, 2013.

[8] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in OFDM systems," *IEEE Trans. Broadcast.*, vol. 48, no. 3, pp. 223–229, Sep. 2002.

[9] I. Dokmanic, R. Parhizkar, J. Ranieri, and M. Vetterli, "Euclidean distance matrices: Essential theory, algorithms, and applications," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 12–30, Nov. 2015.