Fibre Optic Nonlinear Technologies [FONTE] - A European Industrial Doctorate [GA766115]

Document Details

Title	Deliverable D4.2
Deliverable number	D4.2
Deliverable Type	Report (public)
Deliverable title	Multi-user communication and information theory
Work Package	WP4 - Network applications of the NFT technology
Description	Due to the computational complexity of NFT, performing both nonlinear modulation and nonlinear multiplexing is hard, in optical fiber communication. A variety of approaches has been proposed to enhance NFDM to address this problem. We, however, in this report, propose a machine learning approach to tackle the nonlinearity problem in optical fiber communication.
Deliverable due date	31/05/2019
Actual date of submission	26/03/2020
Lead beneficiary	Telecom Paris
Version number	V1.1
Status	FINAL

Dissemination level

PU	Public	X
CO	Confidential, only for members of the consortium (including Commission Services	

Project Details

Grant Agreement	766115
Project Acronym	FONTE
Project Title	Fibre Optic Nonlinear Technologies
Call Identifier	H2020-MSCA-ITN-2017
Project Website	fonte.astonphotonics.uk
Start of the Project	1 June 2018
Project Duration	48 months

Consortium

Aston University

NOKIA Bell Labs

TELECOM ParisTech

TU Delft

DTU Technical University of Denmark

EC Funding

European Commission

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 766115

Executive Summary

Nonlinear frequency-division multiplexing (NFDM) has been introduced to address nonlinearity in optical fiber communication. This approach is a signal multiplexing scheme based on the nonlinear Fourier transform (NFT). Problematically, the computational complexity of the inverse nonlinear Fourier transform (NFT), typically conducted by integral equations, is high. This makes it hard, in multi-user nonlinear optical fiber communication, to perform both nonlinear modulation and multiplexing. A variety of approaches has been proposed to enhance NFDM to address this problem. We, however, consider a machine learning approach to tackle the nonlinearity problem in optical fiber communication.

An end to end communication system was simulated. The equalizer at the receiver, which was based on Backpropagation (BP) algorithm, was targeted to be optimized, as BP has high complexity. A neural network (NN) was succeeded to be devised that approximates the high-complexity BP, with the accuracy of 99.9% in noiseless condition, while having **87.3% lower complexity**. This NN equalizer also works well in noisy environments. In this article, after providing a brief background on machine learning and neural networks, we elaborate on this NN equalizer and analyse the results. We also discuss how machine learning can enhance the achievable information rate of fiber-optic communication systems by reducing the receiver complexity.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ACRONYMS

AiPT	Aston Institute of Photonic Technologies
AIR	Achievable information rate
BP	Backpropagation
CNN	Convolutional Neural Network
DL	Deep Learning
ESR	Early Stage Researcher
EC	European Commission
EID	European Industrial Doctorates
EPF	Expected Prediction Error
FONTE	Fibre Optic Nonlinear Technologies
ML	Machine Learning
MLP	Multi Layer Perceptron
NFDM	Nonlinear frequency-division multiplexing
NFT	Nonlinear Fourier transform
NN	Neural Network
NBL	Nokia Bell Labs
RNN	Recurrent Neural Network
SSFM	Split-Step Fourier Method

1. A BRIEF BACKGROUND ON MACHINE LEARNING AND NEURAL NETWORKS

Machine learning (ML) is a field of artificial intelligence trying to find a sort of algorithms that enables systems to learn and advance from past experiences without the need to be explicitly programmed [1]. These algorithms start with observations or data in a vector space E to finally output a determination or prediction in the same or another vector space K. ML algorithms are commonly categorized as supervised or unsupervised.

In supervised learning, the vector space E of all possible inputs and and the vector space K of all possible outputs and all the known ground truth input-output pairs (x_i, y_i) are taken first. Then the algorithm seeks the best function, $f: E \to K$, predicting the output $\lambda \in K$ for an input $x \in E$. The quality metric of this function is based on a loss function, $L(Y, f(X))$, to penalize error in the prediction, where $X \in E$ and $Y \in K$ are random variables, and Y is ground truth. Several different loss functions can be considered as a metric. Some of these functions are listed in Table 1. However, so far, the most common loss function is squared error loss: $L\big(Y, f(X)\big) = \big(Y - f(X)\big)^2$. The loss function produces a criterion for choosing f, called expected prediction error (EPF) or cost function. In case of choosing squared error loss [2, Ch. 2.4],

$$\begin{aligned} EPE(f) &= E\left[\big(Y - f(X)\big)^2\right] \\ &= \int [y - f(x)]^2 \Pr(dx, dy). \end{aligned} \tag{1}$$

By conditioning on X, EPF can be written as

$$EPE(f) = E_X E_{Y|X}([Y - f(X)]^2 \mid X). \tag{2}$$

Hence, it suffices to minimize EPF pointwise:

$$f(x) = argmin_c E_{Y|X}([Y - c]^2 \mid X = x). \tag{3}$$

This yields the solution

$$f(x) = E(Y|X = x), \tag{4}$$

the conditional expectation, also known as the regression function. Therefore, when the optimal solution metric is the mean squared error, the conditional mean is the best prediction of Y at any point $X = x$.

Neural networks (NNs) is a branch of supervised learning which attempts to figure out (4), or any function leading to the minimum EPF, by performing a sequence of forward and backward propagation over training samples. The idea of neural networks was firstly proposed in 1943 [3], and have been going in and out of fashion for more than 70 years. However, thanks to recent remarkable advances in this field, that we will discuss later in this article, NNs has again dramatically come on the scene [4], and it has widely been adopted by different fields of study to obtain better results [5]-[7]. Deep learning (DL) is in fact, just a new name for neural networks.

Neural networks usually are arranged in interconnected layers. Each layer consists of a number of nodes called hidden units containing an activation function. Training samples are provided to the NN through the input layer, passing them to the hidden layers where the actual processing is done via a system of weighted connections [8, Ch. 1]. Finally, the answer is provided as the output of the output layer.

Table 1. Some of commonly used loss functions

	$L(Y, f(x))$
Mean Square Error *(MSE)*	$\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2$
Mean Absolute Error (MAE)	$\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))$
Root Mean Square Error *(RMSE)*	$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2}$
Cross Entropy Loss	$-\frac{1}{n}\sum_{i=1}^{n}[y_i log\big(f(x_i)\big) + (1-y_i)\,(1 - log\big(f(x_i)\big))]$
*Huber loss**	$\frac{1}{n}L_\delta\big(y_i, f(x_i)\big),\ L_\delta\big(y_i, f(x_i)\big) = \begin{cases}\frac{1}{2}\big(y_i - f(x_i)\big)^2, & if\ \lvert y_i - f(x_i)\rvert < \delta \\ \delta\lvert y_i - f(x_i)\rvert - \frac{1}{2}\delta^2, & o.w.\end{cases}$

The activation function of a node, $\sigma(.)$, applies a nonlinearity to it. This is to address the so-called expressive power of the neural network [9]. By this means, the neural network functional form matches the hierarchical or Markovian structure. Most data of the practical interest is generated by some form of these processes [4]. A number of routinely used activation functions are listed in Table 2.

Most of the neural networks obey some kind of learning rule, which continually corrects the weights of connections based on the computed final loss of the presented batch of training samples. A correction is done after a cycle of forward activation flow of outputs and backward propagation of error, called backpropagation. Backpropagation performs a gradient over the vector space K towards the global/local minimum producing the steepest descent along the error surface [10]. Simply speaking, a neural network initially sets its weights randomly, and based on the presented input, it sees how far its guess is from the ground truth. Then based on the result, it performs the correction through a backward propagation of the computed error.

There are different types of neural networks, among well-known of which, one can list: multi-layer perceptron (MLP) [9], convolutional neural network (CNN) [9], recurrent neural network (RNN) [11, Ch. 10], and generative adversarial neural network (GAN) [11, Ch. 16]. In the following subsections, we very briefly review the MLPs and CNNs (RNNs and GANs are out of the scope of this article).

Table 2. Routinely used activation functions in neural networks

	$g(z)$	*Derivate:* $g'(z)$
Sigmoid	$\frac{1}{1+e^{-z}}$	$g(z)(1-g(z))$
Tanh (Hyperbolic Tangent)	$\frac{e^z - e^{-z}}{e^z + e^{-z}}$	$1 - g(z)^2$
ReLU (Rectified Linear Units)	$\max(0, z)$	$\begin{cases}1, & z > 0 \\ 0, & o.w.\end{cases}$
*ELU (Exponential Linear Unit)**	$\begin{cases}z, & z > 0 \\ \delta(e^z - 1), & o.w\end{cases}$	$\begin{cases}1, & z > 0 \\ \delta e^z, & o.w.\end{cases}$

* $\delta > 0$ is an arbitrary constant.

1.1. MULTI-LAYER PERCEPTRON

What was just discussed is exactly true for MLPs. An MLP is a stack of interconnect layers on top of each other over a system of weighted connections. MLPs are in the category of feed-forward neural networks wherein connections between the nodes do not form a cycle [11, Ch. 6]. Each layer j with N_j hidden units performs the mapping $R^{(j)}: \mathbb{R}^{N_{j-1}} \times \mathbb{R}^{N_{j-1} \times N_j} \rightarrow \mathbb{R}^{N_j}$ on the output of the previous layer, using the weights, as parameters, and the activations of the nodes. If we denote I as the input of MLP and O as the output, then

$$O = R(I, W) = R^{L-1}(R^{L-2}(\dots R^1(I^1, W^1) \dots, W^{L-2}), W^{L-1}), \quad (5)$$

where W^j is the weights matrix of the input connections to the layer j.

To find the best set of weights, at the end of each iteration t, the final loss/error is computed, and through a process of backpropagation, the error is propagated to the previous layers so that they update their weights by gradient descent according to

$$W_{t+1} = W_t - \alpha \nabla \hat{L}(W_t), \quad (6)$$

where $\alpha > 0$ is the learning rate, and $\hat{L}$ is the estimated loss based on the current weights.

A schematic diagram of an MLP with two hidden layers is illustrated in Fig. 1.

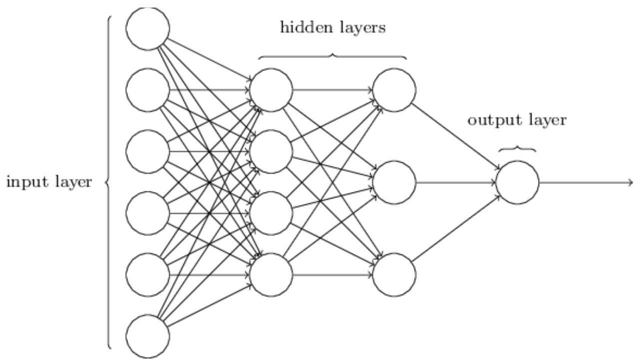

Fig. 1. The schematic diagram of an MLP, receiving an array of 6 cells, with 2 hidden layers having 4 and 3 hidden units, respectively. The NN outputs a single value at the end.

1.2. CONVOLUTIONAL NEURAL NETWORKS

CNNs are like MLPs with a number of intermediate convolutional layers. The idea of CNN was originally proposed for efficient learning in 2D images [11, Ch. 9]. Convolution layers apply a convolution via a kernel over the input:

$$I(t) * h(t) = \int I(k)h(t-k)dk \quad (7)$$

where I is the input and h is the kernel. As in the digital world, values are discrete, (7) turns to:

$$I[n] * h[n] = \sum I[k]h[n-k] \quad (8)$$

By this means, a CNN is able to profitably capture the spatial and temporal dependencies in the data by adopting the relevant, suitable kernels/filters. Using convolution, CNNs also highly reduce the dimension of data to streamline the underlying computation. This is usually done by pooling layers or by increasing the stride.

This feature of CNNs is their primary point of attraction over MLPs in optical fiber communication research. In the next section we discuss the application of neural networks in optical fiber communication.

2. Applications of Neural Networks in optical fiber communication

Thanks to the recent impressive advances in the field of neural networks, NNs have recently been integrated into numerous approaches to various problems in different areas of study. These advances have resulted in a highly faster convergence rate of the NN, using efficient optimization algorithms, e.g., Adam [12]; and extremely higher robustness against overfitting via well-planned regularization schemes, e.g., Dropout [13].

The field of optical fiber communication has not been exempt from this fact. A variety of papers on the topic of applications of neural networks in optical fiber communication have been published [14]-[16]. Generally speaking, these papers, based on their purpose, can be divided into three groups:

a- NNs to reduce complexity [17]-[19]. The approach of this group of papers is based on the fact that most of the real-world data is generated by some form of hierarchical or Markovian process, and deep NNs perform well because their functional form matches these structures [4]. The goal of this group of approaches is to find an NN modeling the current in use function well.
b- NNs to reduce error [20]-[21]. This sort of papers looks for the mechanisms that boost the accuracy of the NN approaches by reducing the bit error rate.
c- NNs to achieve both mentioned goals by developing a fundamentally new approach in designing communication systems [14], [23], [24]. This kind of approaches look for an auto-encoder which tries to learn representations x of the messages 'm' that are robust concerning the channel impairments distorting x to y (i.e., noise, fading, distortion, etc.), so as to the transmitted message can be recovered with low error probability. Put another way, instead of removing redundancy from the input data to do compression, this auto-encoder usually adds redundancy to it by learning an intermediate representation robust to channel perturbations [9].

Among the mentioned group above, motivations in groups a and b are more pragmatic. To elaborate on this set of motivations, let us bring up our recent research and experimentation, as an instance. It is known that the signal evolution, $q(t, z)$, in a lossless optical fiber obeys nonlinear Schrödinger (NLS) equation [25]:

$$j\frac{\partial q}{\partial z} = \frac{\partial^2 q}{\partial t^2} + 2|q|^2 q + n(t, z), \tag{9}$$

where $n(t, z)$ is (zero-mean) white circular symmetric complex Gaussian noise; and t and z denote time and distance, respectively . It has been shown that NLS can be computed by the split-step Fourier method [26]. Therefore, the inverse of SSFM can be used as an equalizer at the receiver. This algorithm is called Backpropagation (BP) algorithm [27]. Yet SSFM has very high complexity due to having to iterate recursively over fiber segments (see D. 4.1 [28], page. 12):

$$\underline{q}(z_{k+1}) = \mathcal{F}^{-1} D_L \mathcal{F} D_N\, \underline{q}(z_k) + n(z_{k+1}). \tag{10}$$

where $\mathcal{F}$ is the Fourier transform, $\underline{q}(z_k)$ is the $L \times 1$ length-vector of sample values $q(z_k, t_l), l = 0,1, \dots, L - 1$, at position z_k, and n is the Gaussian noise $L \times 1$ length-column-vector. D_N is a diagonal matrix with entries:

$$e^{j\gamma |q(z_k,t_l)|^2 \Delta z}, l = 0,1, \dots, L - 1, \tag{11}$$

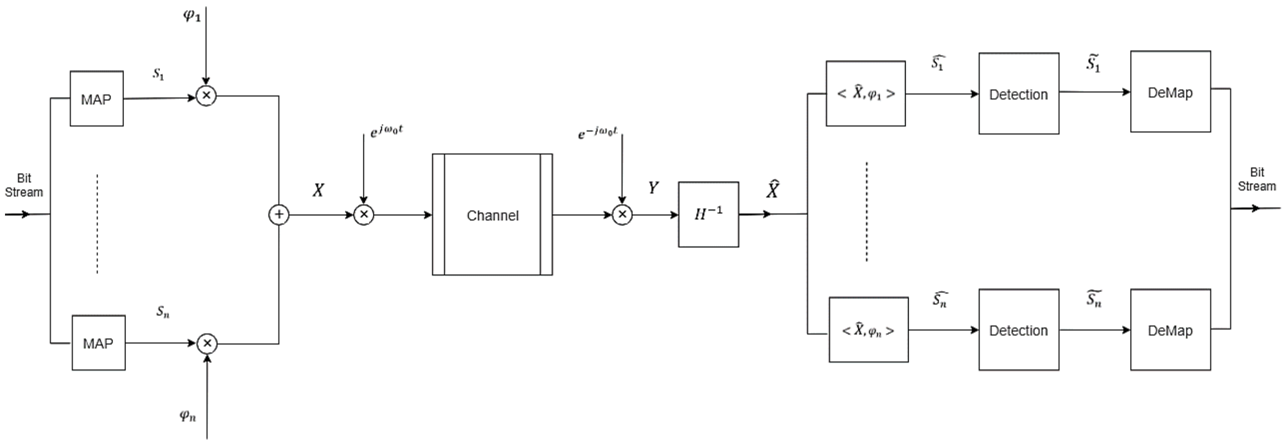

Fig. 2. The schematic diagram of the simulated end-to-end communication system.

where γ is the nonlinearity parameter. D_L is a diagonal matrix with entries:

$$e^{-j\left(\frac{\beta_2}{2}\right)l^2/(L\Delta t)^2\Delta z}, l = 0,1, \dots, \frac{L}{2} - 1, \tag{12}$$

$$e^{-j\left(\frac{\beta_2}{2}\right)(L-l)^2/(L\Delta t)^2\Delta z}, l = \frac{L}{2}, \dots, L - 1, \tag{13}$$

where β_2 is the dispersion parameter.

This leads BP to have high complexity. Now, if an NN approximating BP via a lower complexity is devised, it leads communication systems to have a higher achievable rate, in practice. Because the receiver can process the received data faster, and therefore, there is potential to input more data to the channel.

This has been the subject of our research so far. In the next section, we will confer the details of this approach and discuss the results.

3. Experimentation and Results

An end to end communication system was simulated, according to Fig. 2. The performance of the equalizer at the receiver, which was based on BP, was analysed. The equalizer was targeted to be optimized, as BP has high complexity due to the reason discussed in the previous section.

4-QAM constellation, illustrated in Fig. 3, was chosen for the transmitter and the receiver.

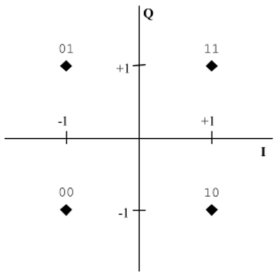

Fig. 3. The used constellation in the experimentations.

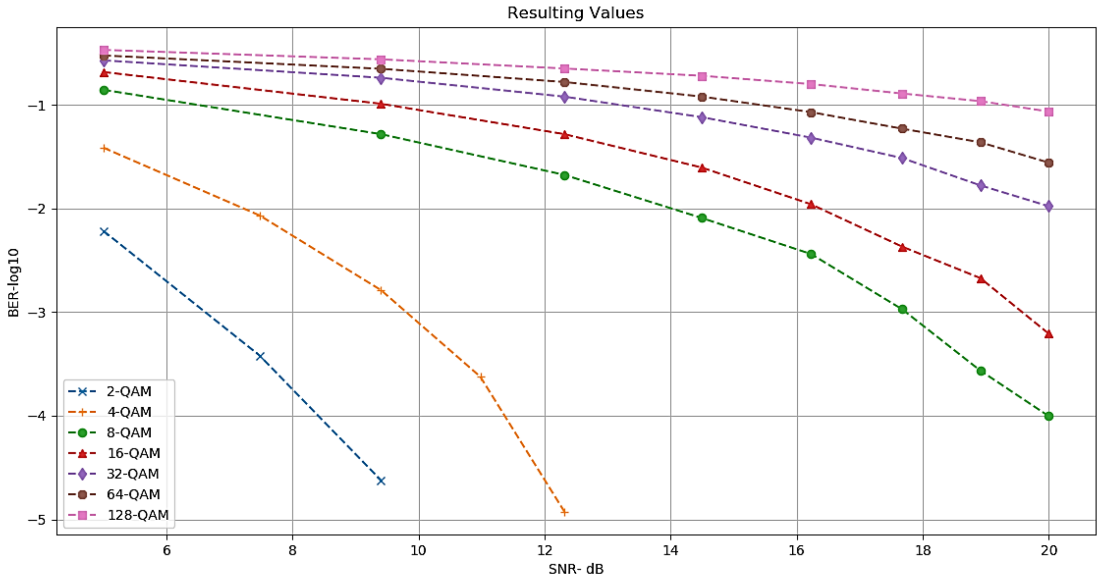

Fig. 4. Bit error rate (BER) plot of the communication system, based on BP, over different constellations.

The initial BP, whose result is illustrated in Fig. 4, was based on a blueprint of SSFM with 1000 segmentation, and therefore it must iterate over 1000 steps. We reduced the number of steps and analyzed the performance. The results are brought in Fig. 5.

It is clear from Fig. 5 that the performance has dramatically been diminished. At this point, the question was whether or not we can make a NN approximating the 1000-step BP via a lower complexity.

After many efforts, we succeeded in devising a neural network that approximates the 1000-step BP (highest complexity), with the accuracy of 99.9% in noiseless condition, while having **87.3% lower complexity**. The mean absolute error (MAE) of this NN is 8.0827×10^{-5}. This NN also works well in noisy environments. The bit error rate (BER) of the system, based on this NN equalizer, in a noisy environment, is illustrated in Fig. 5. This plot also compares the NN with all the versions of BP (the reduced ones and the pure).

Many mathematical solutions were applied to devise this NN. Also, the regularization was performed such that the NN produced almost the same history graph of train and test error, as it is illustrated in Fig. 6. This guarantees that the NN is well applicable to untested data.

4. Conclusion

We provided a neural network approximating Backpropagation with high accuracy while having considerable lower complexity. We conferred the potential of this approach for contributing to achieving a higher AIR in optical fiber communication systems. We also debated the role of machine learning in reaching higher AIRs in communication systems, and we reviewed a number of papers taking advantage of NNs brought about this result. We furthermore showed that a more accurate NN can be achieved in case of assigning more processing power for training the NN.

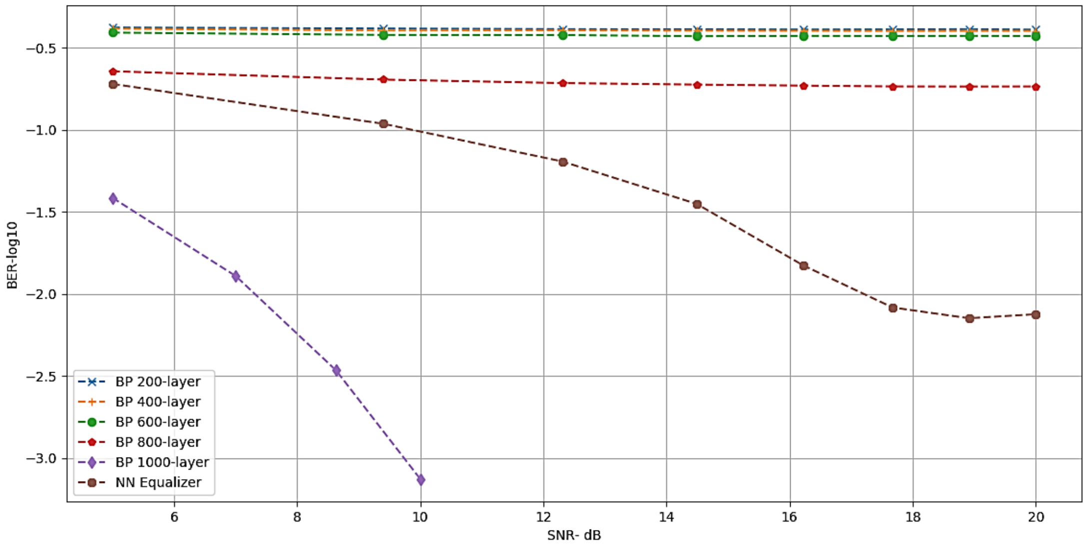

Fig. 5. Bit error rate (BER) plot of the system based on the specified equalizers (BP in the different number of steps and the NN).

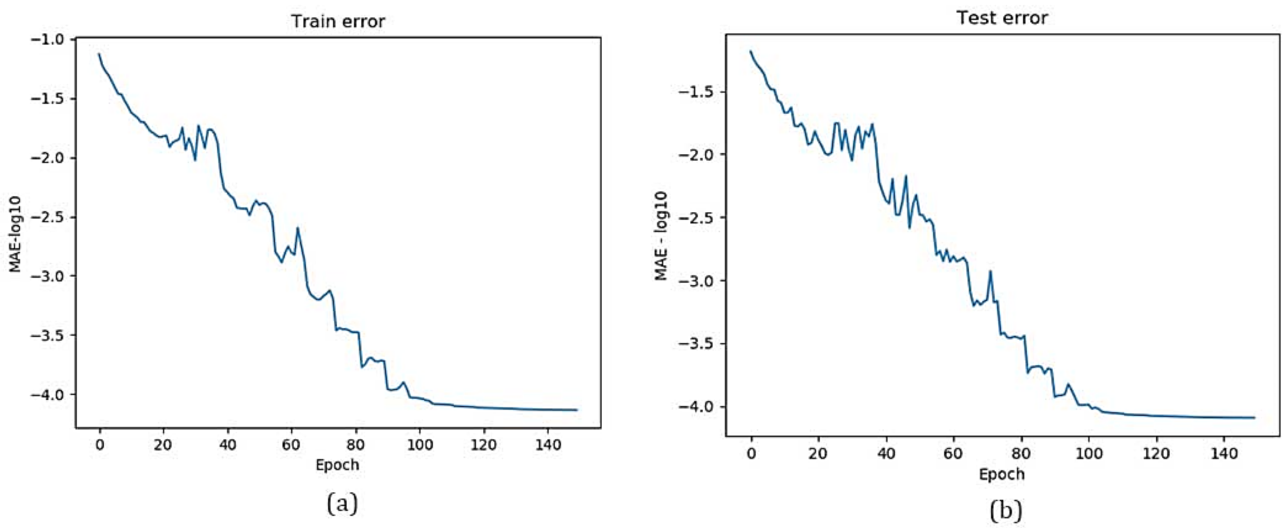

Fig. 6. a) The history of the calculated loss on the train set during training. b) The history of the calculated loss on the test set during training.

REFERENCES

[1] D. Michie, "'Memo' Functions and Machine Learning," *Nature*, vol. 218, no. 5138, pp. 306–306, 1968.

[2] T. Hastie, R. Tisbshirani, and J. Friedman, *The Elements of statistical learning: data mining, inference, and prediction*. New York: Springer, 2017.

[3] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.

[4] H. W. Lin, M. Tegmark, and D. Rolnick, “Why Does Deep and Cheap Learning Work So Well?,” *Journal of Statistical Physics*, vol. 168, no. 6, pp. 1223–1247, 2017.

[5] C. Catanese, A. Triki, E. Pincemin and Y. Jaouën, "A Survey of Neural Network Applications in Fiber Nonlinearity Mitigation," *2019 21st International Conference on Transparent Optical Networks (ICTON)*, Angers, FR, pp. 1–4, 2019.

[6] J. Townsend, T. Chaton and J. M. Monteiro, "Extracting Relational Explanations From Deep Neural Networks: A Survey From a Neural-Symbolic Perspective," *IEEE Transactions on Neural Networks and Learning Systems*.

[7] M. Markus, “Issues in designing automated minimal resource allocation neural networks,” *IEEE International Joint Conference on Neural Networks*, Montreal, CA, vol. 5, pp. 2671–2673, 2005.

[8] C. C. Aggarwal, *Neural networks and deep learning: a textbook*. Cham, Switzerland: Springer, 2018.

[9] T. O’Shea and J. Hoydis, "An Introduction to Deep Learning for the Physical Layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563-575, 2017.

[10] D. E. Rumelhart and J. L. McClelland, *Parallel distributed processing: explorations in the microstructure of cognition*. Cambridge, MA: MIT Press, 1999.

[11] A. Courville, I. Goodfellow, and Y. Bengio, *Deep learning*. Cambridge, MA: MIT Press, 2017.

[12] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” *arXiv.org*, 02-Mar-2015. [Online]. Available: https://arxiv.org/abs/1502.03167. [Accessed: 02-Nov-2019].

[13] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting.” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[14] B. Karanov, M. Chagnon, F. Thouin, T. A. Eriksson, H. Bulow, D. Lavery, P. Bayvel, and L. Schmalen, “End-to-End Deep Learning of Optical Fiber Communications,” *Journal of Lightwave Technology*, vol. 36, no. 20, pp. 4843–4855, 2018.

[15] J. Thrane, J. Wass, M. Piels, J. C. M. Diniz, R. Jones, and D. Zibar, “Machine learning techniques for optical performance monitoring from directly detected PDM-QAM signals,” *The Journal of Lightwave Technology*, vol. 35, no. 4, pp. 868–875, 2017.

[16] D. Zibar, M. Piels, R. Jones and C. G. Schäeffer, "Machine Learning Techniques in Optical Communication," *Journal of Lightwave Technology*, vol. 34, no. 6, pp. 1442-1452, 2016.

[17] C. Häger and H. D. Pfister, "Deep Learning of the Nonlinear Schrödinger Equation in Fiber-Optic Communications," *2018 IEEE International Symposium on Information Theory (ISIT)*, Vail, US, pp. 1590-1594, 2018.

[18] C. D. Häger and H. D. Pfister, “Nonlinear Interference Mitigation via Deep Neural Networks,” *Optical Fiber Communication Conference*, San Diego, US, paper W3A, 2018.

[19] T. J. O'Shea, J. Corgan and T. C. Clancy, "Unsupervised representation learning of structured radio communication signals," *2016 First International Workshop on Sensing, Processing and Learning for Intelligent Machines (SPLINE)*, Aalborg, DK, pp. 1-5, 2016.

[20] T. Gruber, S. Cammerer, J. Hoydis and S. t. Brink, "On deep learning-based channel decoding," *51st Annual Conference on Information Sciences and Systems (CISS)*, Baltimore, US, pp. 1-6, 2017.

[21] E. Nachmani, Y. Be'ery and D. Burshtein, "Learning to decode linear codes using deep learning," *54th Annual Allerton Conference on Communication, Control, and Computing (Allerton),* Monticello, US, pp. 341-346, 2016.

[22] S. Cammerer, T. Gruber, J. Hoydis and S. ten Brink, "Scaling Deep Learning-Based Decoding of Polar Codes via Partitioning," *IEEE Global Communications Conference*, Singapore, pp. 1-6, 2017.

[23] S. Li, C. Häger, N. Garcia and H. Wymeersch, "Achievable Information Rates for Nonlinear Fiber Communication via End-to-end Autoencoder Learning," *2018 European Conference on Optical Communication (ECOC)*, Rome, IT, pp. 1-3, 2018.

[24] F. A. Aoudia and J. Hoydis, "End-to-End Learning of Communications Systems Without a Channel Model," *52nd Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, US, pp. 298-303, 2018.

[25] M. I. Yousefi and F. R. Kschischang, "Information transmission using the nonlinear Fourier transform, Part I: Mathematical tools," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4312–4328, 2014.

[26] G. Kramer, M. I. Yousefi and F. R. Kschischang, "Upper bound on the capacity of a cascade of nonlinear and noisy channels," *2015 IEEE Information Theory Workshop (ITW)*, Jerusalem, IL, pp. 1-4, 2015.

[27] R. -. Essiambre and P. J. Winzer, "Fibre nonlinearities in electronically pre-distorted transmission," *European Conference on Optical Communication (ECOC)*, Glasgow, UK, vol. 2, pp. 191-192, 2005.

[28] "Principles of linear and nonlinear frequency-division multiplexing," *EID-FONTE Deliverable D4.1*. [Online]. Available: https://fonte.astonphotonics.uk/wp-content/uploads/sites/7/2019/09/20190925_D4.1-Final_as-submitted.pdf. [Accessed: 01-Oct-2019].