

Investigation of using neural networks for temperature and relative humidity measurement with the Rayleigh scattering-based distributed optical fiber sensor

Mateusz Mądry,* Bogusław Szczupak, Mateusz Śmigieński, and Bartosz Matysiak

Faculty of Information and Communication Technology, Wrocław University of Science and Technology,
ul. Wybrzeże Wyspiańskiego 27, 50-370 Wrocław

Received February 11, 2025; accepted March 28, 2025; published March 31, 2025

Abstract—The paper presents an investigation of neural networks for temperature and relative humidity (RH) measurement by Rayleigh-based distributed optical fiber sensor (DOFS). The sensor consists of bare and polyimide-coated fibers placed side by side, ensuring different sensitivities to temperature and RH. Two neural networks have been thoroughly examined in sensor data processing: Multilayer Perceptron (MLP) and Convolutional Neural Networks (CNN). These models were assessed in terms of mean square errors (MSE) and training time. The MLP model achieves better results with lower training time compared to CNN. The proposed solution enables fast and automatic sensor data analysis after model training.

Optical frequency domain reflectometry (OFDR) based on Rayleigh scattering enables relatively fast, high-resolution distributed measurements along an optical fiber [1]. This technique exploits naturally occurring Rayleigh scattering, originating from the fiber's inherent microscopic inhomogeneities. Rayleigh-based OFDR utilizes swept-wavelength interferometry to generate interference patterns from Rayleigh backscatter at various points along the optical fiber. Variations in temperature and/or strain affect both the refractive index and the length of the fiber, causing a shift in the spectrum of the backscattered light. This shift is then analyzed to determine temperature and/or strain along the fiber [2]. Rayleigh-based OFDR has been demonstrated for measuring various parameters, including temperature [3], axial strain [4], and relative humidity (RH) [5]. Its high spatial resolution and distributed sensing capability make it attractive for applications such as structural health monitoring [6]. In the standard approach, multiparameter measurements using Rayleigh-based OFDR require manual analysis of sensitivities based on the type of optical fiber. This typically involves constructing a sensitivity matrix equation [7, 8]. In this case, the matrix relates the sensitivity of various fiber sections to physical parameters, as demonstrated in a study using standard and reduced-cladding fibers to measure strain and temperature in OFDR [8] simultaneously. However, the process of multiparameter prediction is both error-prone and time-consuming due to its reliance on manual analysis. Machine learning (ML) has been explored in the context

of fiber optic sensors, such as fiber Bragg gratings [9, 10], interferometers [11, 12], and Rayleigh- or Brillouin-based distributed optical fiber sensors (DOFS) [13-15]. A linear regression model has been used to predict temperature and RH values in Rayleigh-based OFDR [15].

To further explore ML techniques in distributed fiber sensors, we have studied, for the first time to the best of our knowledge, the use of neural networks (NNs) for predicting temperatures and RH in Rayleigh-based OFDR. In our approach, the selected NN models were trained using spectral shift values obtained from measurements performed on partially polyimide-coated and uncoated (bare) fibers arranged side-by-side. This paper focuses on evaluating the efficiency and comparing the performance, in terms of mean square error (MSE) and processing time, of two NN architectures: the Multilayer Perceptron (MLP) and the Convolutional Neural Network (CNN). Our results clearly demonstrate the feasibility of the proposed NN-based method for automated, accurate, and efficient multiparameter sensing without the need for manual sensitivity calibration. Initially, measurements were performed to create the dataset for the NN models using the setup presented in Fig. 1.

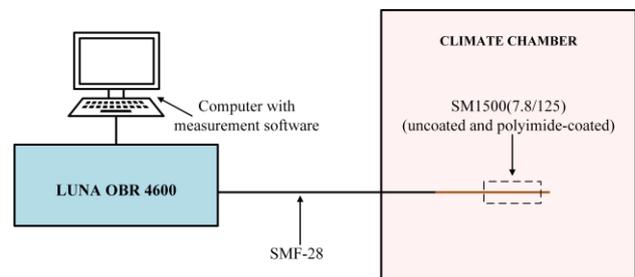


Fig. 1. The scheme of measurement setup.

The Rayleigh-based OFDR (OBR 4600, *Luna Innovations*) was connected to SMF-28 pigtails and then to the investigated fiber. The fiber used was a polyimide-coated fiber, SM 1500 (7.8/125), produced by *Fibercore*, with a cladding diameter of 125 μm , and an overall

* E-mail: mateusz.madry@pwr.edu.pl



diameter (including the polyimide coating) of 155 μm . The investigated fiber consisted of bare and polyimide-coated sections. In this approach, the bare fiber was sensitive only to temperature, whereas the polyimide-coated fiber responded to both temperature and RH, as thoroughly investigated in one of our studies [16]. All measurements were performed in a controlled climate chamber (KMF 115, *Binder*). The measurements were conducted at temperatures ranging from 30 to 80 $^{\circ}\text{C}$ in steps of 10 $^{\circ}\text{C}$ and at RH values ranging from 20% to 70% in steps of 10%. A spatial resolution of 0.1 cm was set for the measurements. The spectral shifts along the fiber were recorded for all states within the measured temperature and RH ranges. An example of spectral shift values for the investigated sensor fiber at three different environmental conditions (50 $^{\circ}\text{C}$, 40% RH; 60 $^{\circ}\text{C}$, 50% RH; and 70 $^{\circ}\text{C}$, 60% RH) is shown in Fig. 2.

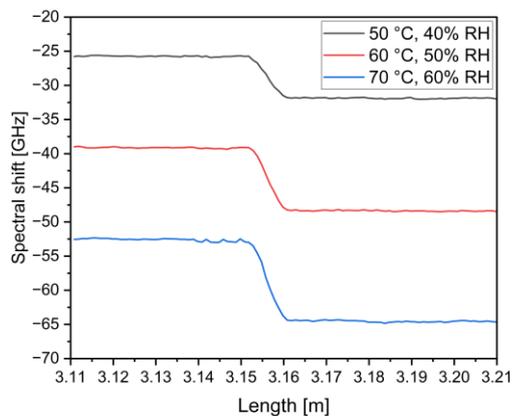


Fig. 2. The example of measurement results of spectral shift changes for three different environmental conditions (temperature/RH).

All measured spectral shift values, corresponding to different temperature and RH conditions, were used as input data for the neural networks. The dataset consists of measurement files for each temperature and RH states, each containing spectral shift values and the corresponding positions along the fiber length.

The first examined model was a fully-connected MLP, which is shown in Fig. 3.

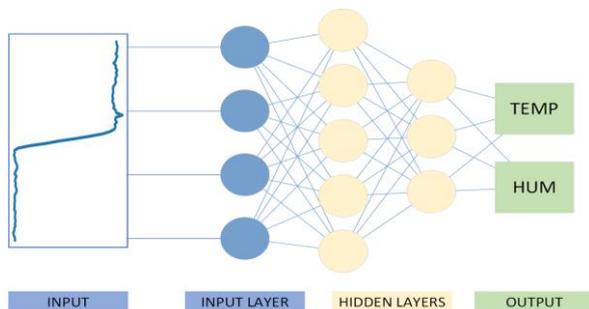


Fig. 3. The architecture of MLP.

In this architecture, first described by Rosenblatt in 1958, layers of processing units called neurons are connected in a feed-forward approach [17]. Each neuron first computes a linear combination of its input values with a vector of parameters called weights and then passes it to a non-linear activation function to capture more complex dependencies between inputs and outputs to the model. The outputs are temperature and RH values. The second architecture was CNN, presented in Fig. 4.

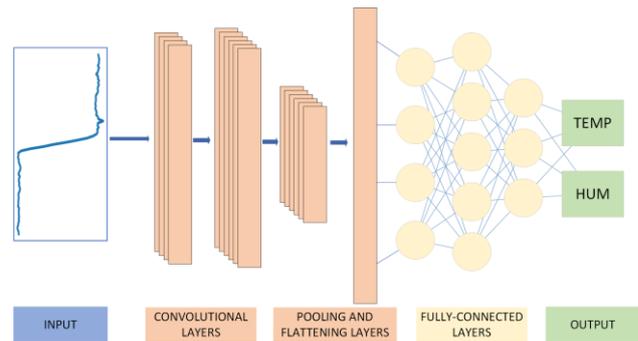


Fig. 4. The architecture of CNN.

CNN can be adapted for processing 1D signals, such as time-series data, audio signals, and sensor readings [18]. Unlike in image processing, where CNNs work on 2D grids of pixels, in the 1D context, the input is a sensor measurement along the fiber length, and the convolution operation is applied along a single dimension. The architecture of a 1D CNN typically consists of convolutional layers, a pooling layer, and fully connected layers. The convolutional layers apply 1D filters to the input signal to detect features such as peaks, trends, and periodic patterns. The pooling layer reduces the dimensionality of the feature maps by down-sampling, which helps retain the most important features while reducing computational complexity; common pooling methods include max pooling and average pooling. After the convolution and pooling operations, the extracted features are flattened and fed into fully connected layers identical to those of a typical MLP.

For MLP, the following parameters were taken into consideration: number of hidden layers, number of neurons in hidden layers, activation functions, dropout probabilities, and for CNN, different kernel sizes, number of convolution layers channels, pooling layers, and number of neurons in fully connected layers. Data preprocessing included scaling data to a range $(-1, 1)$, data sampling, averaging, and applying margin (cutting sensor measurement points on both sensor edges). MSE, one of the most widely used metrics in ML, was selected to measure prediction errors. The experiment was based on nested cross-validation. In each of the five outer folds, approximately 20% of (temperature, RH) pairs were

included in the testing set, and leave-one-out cross-validation (LOOCV) was performed on the remaining pairs to choose the model with the lowest MSE value in the current outer fold. The values of hyperparameters for the training process, which have been selected empirically, are presented in Tab. 1.

Table 1. Hyperparameters values used in NN models.

Hyperparameter name	Value
Batch size	32
Number of Epochs	100
Learning rate	0.01 (MLP); 0.001 (CNN)
Optimizer	Adam

The performed comprehensive experiments for MLP suggest that the preferable network configuration consists of 2 hidden layers, ReLU (rectified linear unit) activation function, and no dropout applied. Best models typically rely on a number of features from ranges 5-8 and 14-20 for temperature and RH, respectively. Scaling input data to a range (-1, 1) significantly improved the results.

In the case of experimental results for CNNs, deeper networks were preferred. The networks with 4-5 convolutional layers and 3-4 fully connected layers performed the best. However, data scaling before passing through the network was shown to be ineffective.

Based on the results for MLP, the MSE values for temperature fluctuated between approximately 0.02 °C and 0.17 °C, while the MSE for RH changed from 3.04% to 11.55%. Based on the results for CNN, the MSE values for temperature fluctuated between approximately 0.22 °C and 2.08 °C, while the MSE for RH changed from 3.53% to 9.19%. MLP performed better with temperature prediction than CNN, while at the same time, RH predictions were comparable.

MLP model training time was approximately 0.5 s while CNN training time was approximately 8 s – 54 s. From this, it can be concluded that a simpler neural network architecture results in shorter training times and that expanding the architecture (by adding more hidden layers or convolutional layers) does not automatically translate into better prediction results.

This paper presents a study on neural networks for measuring temperature and RH in Rayleigh-based OFDR. CNN and MLP models were thoroughly investigated for their potential application in fiber sensors. The results reveal that MLP models achieve better or comparable results to CNN, with very short training times below 1 s, which is significantly lower than those of CNN. This new

approach undoubtedly shows superior advantages over conventional methods by automating the prediction of parameters based on raw data and enabling fast processing.

This work was carried out as part of the research project no. 2024/08/X/ST7/01123 funded by the National Science Centre and the statutory funds of Telecommunications and Teleinformatics Department of Wrocław University of Science and Technology, no. 8253050500.

References

- [1] Z. Ding *et al.*, IEEE Sensors Journal **23**, 26925 (2023).
- [2] L. Palmieri, L. Schenato, M. Santagiustina, A. Galtarossa, Sensors **22**, 6811 (2022).
- [3] P. Bulot, R. Bernard, M. Cieslikiewicz-Bouet, G. Laffont, M. Douay, Sensors **21**, 3788 (2021).
- [4] J. Li *et al.*, Opt. Expr. **25**, 27913 (2017).
- [5] P.J. Thomas, J.O. Hellevang, Sensors and Actuators B: Chemical **247**, 284 (2017).
- [6] M. F. Bado, J. R. Casas, Sensors **21**, 1818 (2021).
- [7] Z. Han *et al.*, Opt. Comm. **530**, 129137 (2023).
- [8] Z. Ding *et al.*, IEEE Photon. J. **8**, 1 (2016).
- [9] B.K. Choi *et al.*, IEEE Sensors J. **24**, 27516 (2024).
- [10] S. Sarkar, D. Inupakutika, M. Banerjee, M. Tarhani, M. Shadaram, IEEE Photon. Technol. Lett. **33**, 876 (2021).
- [11] C. Zhu, O. Alsalman, Opt. Expr. **31**, 22250 (2023).
- [12] Y. Mei, T. Xia, H. Cai, Z. Liu, J. Lightw. Technol. **42**, 430 (2024).
- [13] C. Karapanagiotis, K. Hicke, A. Wosniok, K. Krebber, Opt. Expr. **30**, 12484 (2022).
- [14] C. Karapanagiotis, K. Hicke, K. Krebber, Opt. Expr. **31**, 5027 (2023).
- [15] M. Mądry, B. Szczupak, M. Śmigielski, B. Matysiak, Sensors **24**, 7913 (2024).
- [16] B. Szczupak *et al.*, IEEE Sensors Journal **21**, 20036 (2021).
- [17] F. Rosenblatt, Psycholog. Rev. **65**, 386 (1958).
- [18] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Proc. IEEE **86**, 2278 (1998).