How photonic reservoirs improve the performance of deep neural networks

I. Bauwens,[1,2] G. Van der Sande,[1] P. Bienstman,[2] and G. Verschaffelt[1]

[1] Vrije Universiteit Brussel, Applied Physics Research Group, Pleinlaan 2, 1050 Brussels, Belgium
[2] Ghent University-IMEC, Photonics Research Group, Department of Information Technology, Technologiepark Zwijnaarde 126, 9052 Ghent, Belgium

Large deep neural networks (DNNs) have been successfully used for solving complex problems. However, they are often very energy-intensive and time-consuming to train. In this work, we propose to use a photonic reservoir to preprocess the input instead of directly injecting it into the DNN. The advantage of using such a photonic reservoir is that it consists of a network of many randomly connected nodes which do not need to be trained. We numerically show that using such a photonic reservoir as preprocessor results in an improved performance for deep neural networks on a time-series prediction task. We observe that using a stand-alone DNN results in a high test error. If we combine this DNN with a photonic reservoir as preprocessor, we achieve a better performance, shown by a lower test error. Furthermore, it is able to outperform the long short-term memory (LSTM) network with a lower test error and lower total training time. Photonic reservoirs are ideal candidates as physical preprocessors to deep neural networks due to their fast computation times and low-energy consumption [1].

Introduction

Deep neural networks (DNNs) have shown to be very useful in the field of machine learning and artificial neural networks due to their applicability to various problems, e.g. natural language processing, drug discovery, detection of medical images, analyzing big data and many other applications. Because of their ability to generalize underlying patterns and features, they are able to achieve very high accuracy on these tasks which explains their popularity. Artificial neural networks function by training the network, which is achieved by fitting a large dataset of input samples to a model by optimizing internal network weights. The number of these parameters weights can become very large for complex and deep networks with many nodal connections, which is often the case in order to achieve high accuracies. However, by increasing the number of layers and parameters in such DNNs, they become computationally difficult to train, i.e. the training requires much memory, time and energy.

In order to circumvent this problem, we can introduce a photonics-based preprocessor in front of the DNN architecture. By doing so, we are able to transform the input data into a state in a higher dimensional state-space, while harnessing the advantages of photonics, such as a low-energy consumption, the possibility of high inherent parallelism and fast computation times [2]. The photonic preprocessor we investigate in this work is based on reservoir computing (RC). RC systems are recurrent neural networks (RNNs) which consist of three layers: an input layer, a reservoir and an output layer. The input layer is used to inject input data into the system, while the output layer is used to make predictions, typically using a linear read-out with linear weights. The reservoir contains a large amount of randomly interconnected nonlinear nodes. Fig. 1 shows an illustration of a general RC system, with its three different layers. Opposed to DNNs, the internal weights of the reservoir are fixed and remain constant. The only training occurs in the output layer, so that the training time of such networks is drastically reduced compared to DNNs. The RC systems have already been successfully used for different prediction tasks, such as speech recognition, time-series predictions and nonlinear channel equalization tasks.

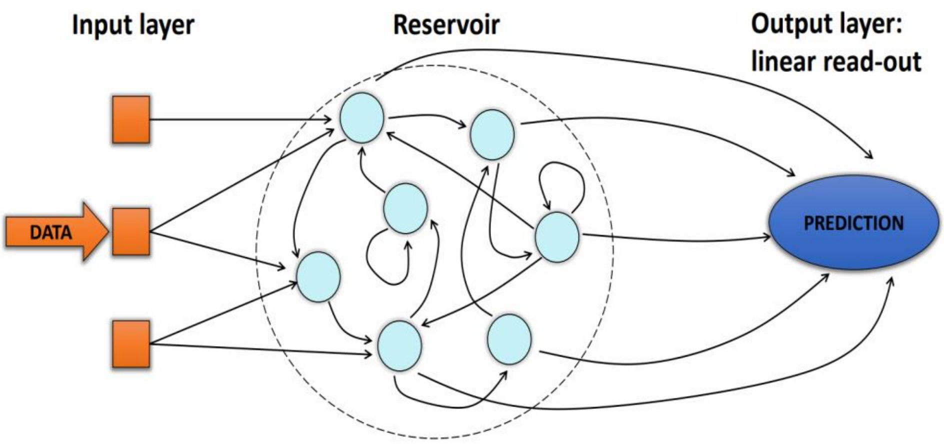

Figure 1. Illustration of a general RC system consisting of an input layer, reservoir and output layer. The randomly connected nodes of the reservoir are shown as blue circles.

In this work, we focus on a photonic delay-based reservoir that uses a semiconductor laser as nonlinear node. We compare the photonic reservoir, combined with either a linear read-out layer such as in conventional reservoir computing, or when it is combined with a DNN. In case of the combined reservoir and DNN system, we feed the output of the photonic reservoir as input to the DNN, so that the reservoir actually functions as a preprocessor to the deep neural network. Such a photonic preprocessor network combined with a DNN is shown in Fig. 2.

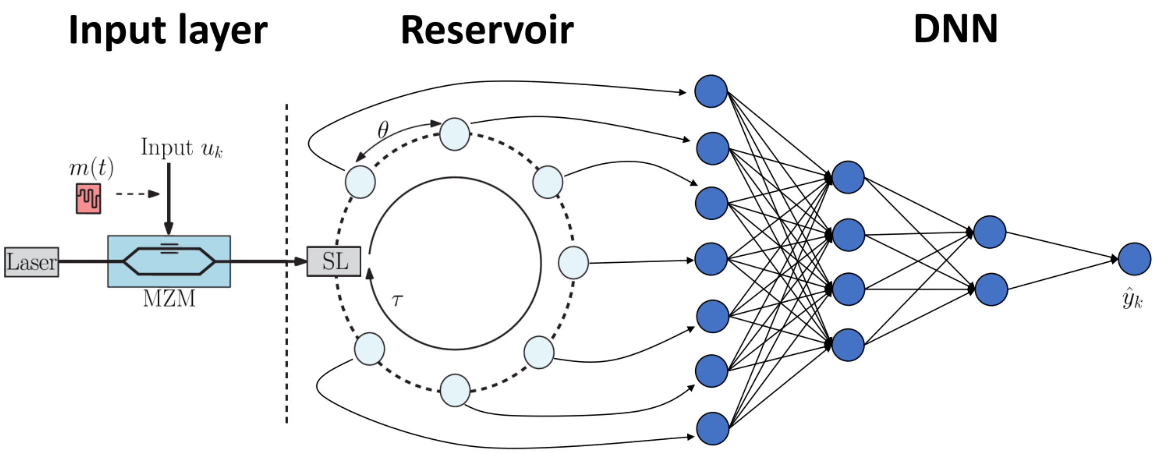

Figure 2. Illustration of a photonic delay-based reservoir, using a semiconductor laser (SL), combined with a deep neural network. Also shown are the input data u_k, preprocessing mask $m(t)$, nodal separation θ and delay time τ . The light blue circles represent the virtual nodes and the output layer is defined by the reservoir output and network weights.

Numerical implementation of the photonic delay-based RC

We implement the photonic delay-based RC system by using a semiconductor laser with delayed feedback, with a time-multiplexing approach to inject data [2]. We measure the intensity at virtual nodes along the delay line as the reservoir output and record the state of the nodes in the state matrix A. We used in this study 200 nodes in the reservoir. Training the linear weights w in this RC approach [2] is done by calculating $w = A^{\dagger}y_{train}$, with † representing the Moore-Penrose pseudoinverse. The optical injection of data u_k is done by using a Mach-Zehnder modulator. Note that due to the time-multiplexed approach, we have to implement a mask $m(t)$ to ensure coupling between nodes. The dynamics of the single-mode laser is modelled using the rate-equations from Ref. [3]

Numerical implementation of the deep neural network

The DNN we use in this work takes as input data a vector with 200 features (i.e. equal to the number of reservoir nodes) and predicts a single value. The DNN consists of three fully-connected layers, where the amount of nodes is decreased throughout the network. The first (hidden) layer in the DNN consists of a fully-connected layer with 100 nodes, with a ReLU activation function in each of the nodes. The following hidden layer is again a fully-connected layer, with 50 nodes, and again with the ReLU activation function. This signal is then injected into the final layer, which only has one node and is used to make a prediction $\hat{y}_k$. We train the DNN using the Adam optimizer, with a learning rate of 10^{-3} for 10^4 epochs. We use early stopping on a validation set, which is different from the training and test set, to avoid overfitting.

Numerical results: Santa Fe time-series prediction task

We compare the performance of the different networks using a one-step ahead time-series prediction task. The input dataset is the Santa Fe dataset, which consists of data points sampled from a far-IR laser in a chaotic regime. We have taken 3000 data samples from the discrete Santa Fe dataset, u_k^{train} , as the training set in the RC system. As test set, u_k^{test}, we have taken 1000 different data samples. We use the normalized mean squared error (NMSE) as performance indicator. In Fig. 3 we show the performance of the RC system and the photonic reservoir coupled with the DNN as preprocessor, together with the relative improvement in performance. We show this for a 2D parameter scan of the feedback rate η to the semiconductor laser and the normalized excess pump current of the semiconductor laser which constitutes the reservoir.

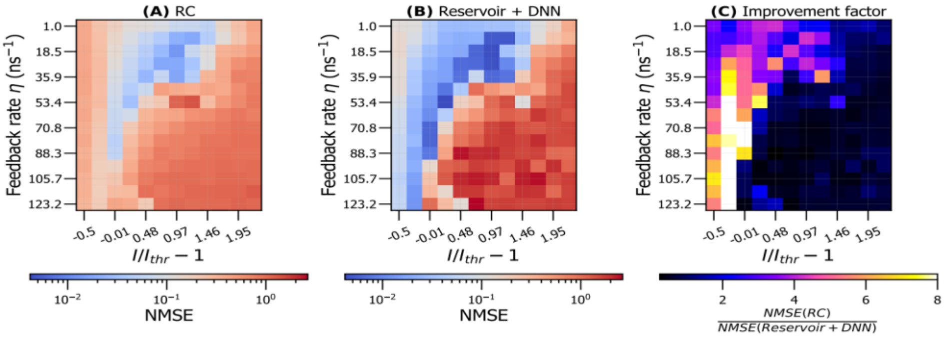

Figure 3. Performance resulting from a grid search of feedback rate and pump current, at constant injection rate, for conventional RC (A), for the photonic reservoir coupled with the DNN (B) and the improvement factor of the reservoir combined with the DNN compared to the RC system (C).

We observe that the reservoir coupled with the DNN is able to achieve an improved performance compared to the conventional RC system with a linear read-out. Furthermore, the reservoir coupled with the DNN has a wide parameter region for which it results in a good performance, indicating it does not need to be carefully fine-tuned for a good performance. In Table 1, we compare the test error and training time of different systems: the stand-alone DNN, the reservoir with DNN and an LSTM (Long short-term memory) network. Compared to the stand-alone DNN, we observe a much improved test error when using the combination of a reservoir and a DNN. Compared to the LSTM network (which is a specialized network with memory nodes), the photonic reservoir combined with the DNN has a much lower training time, and a better performance, indicated by the lower test error.

TABLE I. TEST ERROR AND TIME USED FOR TRAINING THE DIFFERENT SIMULATED NETWORK MODELS, ON THE SANTA FE ONE-STEP AHEAD PREDICTION TASK.

Network	**Test error ($\times 10^{-2}$)**	**Total training time (s)**
Stand-alone DNN	3.4	104
LSTM (recurrent neural network with memory nodes)	2.2	3072
Photonic reservoir with DNN	0.4	95

Conclusion

We have numerically investigated the effect of using a photonic reservoir combined with a DNN. Using such a photonic reservoir results in an improved performance and has the advantage of being a fast and energy-efficient preprocessor to DNNs. Furthermore, the parameters of this photonic reservoir do not need to be carefully fine-tuned for an increased performance.

References

[1] Bauwens, Ian, Guy Van der Sande, Peter Bienstman, and Guy Verschaffelt. "Using photonic reservoirs as preprocessors for deep neural networks." *Frontiers in Physics* 10 (2022): 1051941.

[2] Van der Sande, Guy, Daniel Brunner, and Miguel C. Soriano. "Advances in photonic reservoir computing." *Nanophotonics* 6, no. 3 (2017): 561-576.

[3] Lenstra, Daan, and Mirvais Yousefi. "Rate-equation model for multi-mode semiconductor lasers with spatial hole burning." *Optics express* 22, no. 7 (2014): 8143-8149.