

LogNNet Neural Network

Subjects: Computer Science, Artificial Intelligence

Contributor: Andrei Velichko

LogNNet - neural network which uses filters based on logistic mapping. LogNNet has a feedforward network structure, but possesses the properties of reservoir neural networks. The input weight matrix, set by a recurrent logistic mapping, forms the kernels that transform the input space to the higher-dimensional feature space. The most effective recognition of a handwritten digit from MNIST-10 occurs under chaotic behavior of the logistic map. The correlation of classification accuracy with the value of the Lyapunov exponent was obtained. An advantage of LogNNet implementation on IoT devices is the significant savings in memory used. At the same time, LogNNet has a simple algorithm and performance indicators comparable to those of the best resource-efficient algorithms available at the moment. The presented network architecture uses an array of weights with a total memory size from 1 to 29 kB and achieves a classification accuracy of 80.3–96.3%. Memory is saved due to the processor, which sequentially calculates the required weight coefficients during the network operation using the analytical equation of the logistic mapping. The proposed neural network can be used in implementations of artificial intelligence based on constrained devices with limited memory, which are integral blocks for creating ambient intelligence in modern IoT environments. From a research perspective, LogNNet can contribute to the understanding of the fundamental issues of the influence of chaos on the behavior of reservoir-type neural networks.

Keywords: logistic map ; constrained devices ; IoT ; neural network ; reservoir computing ; handwritten digits recognition ; ambient intelligence ; Lyapunov exponent ; chaos

1. Introduction or History

In the age of neural networks and Internet of Things (IoT), the search for new neural network architectures capable of operating on devices with small amounts of memory (10s of kB of RAM) is becoming an urgent agenda ^{[1][2][3]}. The constrained devices possess significantly less processing power and memory than a regular smartphone or modern laptop, and usually do not have a user interface ^[4]. The constrained devices form the basis for ambient intelligence (Aml) in IoT environments ^[5], and can be divided into three categories based on code and memory sizes: class 0 (less than 100 KB Flash and less than 1 KB RAM), class 1 (≈100 KB Flash and ≈ 10 KB RAM) and class 2 (≈ 250 KB Flash and ≈ 50 KB RAM) ^[6]. Smart objects, which are built on constrained devices, possess limited system resources, and these limitations prohibit the application of security mechanisms common in the Internet environment. Complex cryptographic mechanisms require significant time and high energy resources, and in addition, the storage of a large number of keys for secure data transition is not possible on the constrained devices. The storage issue presents significant challenges for the integration of blockchain ^[7] and artificial intelligence (AI) ^{[1][8]} with IoT and calls for new approaches and research efforts to resolve this limitation. Intelligent IoT devices should be able to process the incoming information without sending it to the cloud. Smart objects equipped with their own AI capabilities spend significantly less time on the analysis of incoming data and development of a final solution, creating new possibilities, for example, in the development of Aml in the medical industry ^[9], predicting the behavior of mechanisms ^[10], local semantic processing of video data ^[11] and “smart” services in e-tourism ^[12]. This approach facilitates the development of the concepts of smart spaces and fog computing, when devices detect each other, for example, using wireless technologies ^[13]; redistribute computational tasks; and optimize the distribution of responses ^{[14][15]}. Therefore, the integration of AI and IoT creates new social, economic and technological benefits.

Neural networks create the foundation for AI. The well-known types of neural networks are feedforward neural networks, convolutional neural networks and recurrent neural networks ^[16]. In addition to neural networks, the popular methods for classification and efficient prediction include tree-based algorithms ^[17] and the k-nearest neighbors algorithm (kNN) ^[18]. The problem of implementing compact and efficient algorithms for the operation of neural networks on constrained devices is the main constraining factor in the development of the integration of AI and IoT. Training of neural networks is the process of optimal selection of the weight coefficients of neuron couplings and filter parameters that occupy a significant amount of memory. It highlights the importance of the development of the methods for reducing the memory consumed by a neural network.

The prominent tree-based algorithms include the GBDT algorithms [19] and Bonsai from Microsoft [17], which allows efficient use of memory and runs on constrained devices with 2–16 kB RAM. An algorithm called ProtoNN [20], based on the kNN method, operates on devices with 16 kB RAM. Currently, effective memory redistribution algorithms in convolutional neural networks (CNN) are available and do not exceed 2 kB of consumed RAM [21], and demonstrated impressive results of classification accuracy $\approx 99.15\%$ on MNIST-10 database. However, the complexity of the algorithms leads to the large size of the program itself. Algorithms based on the recurrent network architecture, such as Spectral-RNN [22] and FastGRNN [18], achieve $\approx 98\%$ classification accuracy on MNIST-10 with a model size of ≈ 6 kB.

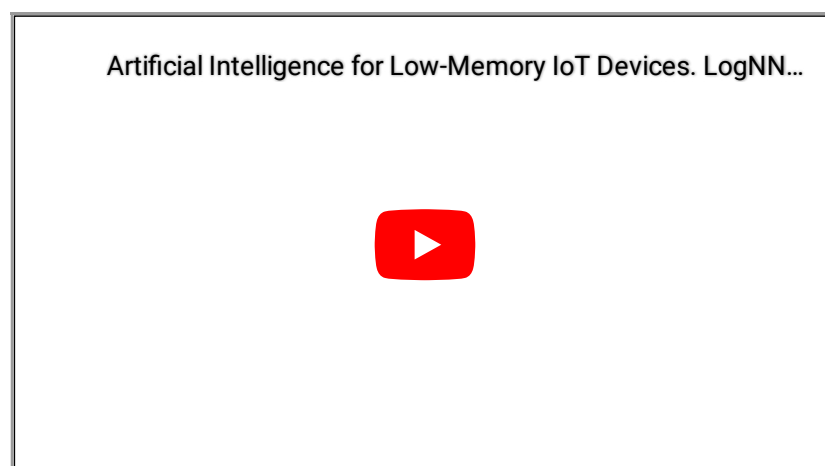
An alternative approach to reduce the number of trained weights is based on physical reservoir computing (RC) [23]. RC uses complex physical dynamic systems (coupled oscillators [24][25][26], memristor crossbar arrays [27], opto-electronic feedback loop [28]), or recurrent neural networks (echo state networks (ESNs) [29] and liquid state machines (LSMs) [30]), as reservoirs with rich dynamics and powerful computing capabilities. The couplings in the reservoir are not trained, but are specified in a special way. The reservoir translates the input data into a higher dimensionality of space (kernel trick [31]), and the output neural network, after training, can classify the result more accurately. The search for simple algorithms for simulating complex reservoir dynamics is an important research task.

2. Model

Figure 1. LogNNet architecture

The network architecture proposed by Andrei Velichko, called LogNNet, is shown in the figure 1. The signal propagates in the same way as in a feedforward network. The weights W_1 are not trained, but are set in a special way. The first line of the array of weights is set by the sine equation, depending on two parameters A and B . The rest of the lines are calculated using the recurrent logistic mapping, with a simple equation with one parameter r . In this way, a reservoir with fixed weights W_1 is created, and training is carried out only for the output weights W_2 .

The entry is from <https://doi.org/10.3390/electronics9091432>



[1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20][21][22][23][24][25][26][27][28][29][30]

References

1. Merenda, M.; Porcaro, C.; Iero, D. Edge Machine Learning for AI-Enabled IoT Devices: A Review. *Sensors* 2020, 20, 2533, doi:10.3390/s20092533.
2. Abdel Magid, S.; Petrini, F.; Dezfouli, B. Image classification on IoT edge devices: Profiling and modeling. *Clust. Comput.* 2020, 23, 1025–1043, doi:10.1007/s10586-019-02971-9.

3. Li, S.; Dou, Y.; Xu, J.; Wang, Q.; Niu, X. mmCNN: A Novel Method for Large Convolutional Neural Network on Memory-Limited Devices. In Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, Japan, 23–27 July 2018; Volume 1, pp. 881–886.
4. Gerdes, S.; Bormann, C.; Bergmann, O. Keeping users empowered in a cloudy Internet of Things. In *The Cloud Security Ecosystem: Technical, Legal, Business and Management Issues*; Elsevier Inc.: Amsterdam, The Netherlands, 2015; pp. 231–247. ISBN 978-0128-017-807.
5. Korzun, D.; Balandina, E.; Kashevnik, A.; Balandin, S.; Viola, F. *Ambient Intelligence Services in IoT Environments: Emerging Research and Opportunities*; IGI Global: Hershey, Pennsylvania, USA, 2019, 1–199.
6. El-Haii, M.; Chamoun, M.; Fadlallah, A.; Serhrouchni, A. Analysis of Cryptographic Algorithms on IoT Hardware platforms. In Proceedings of the 2018 2nd Cyber Security in Networking Conference, CSNet 2018, Paris, France, 24–26 October 2019.
7. Fernández-Caramés, T.; Fraga-Lamas, P. A Review on the Use of Blockchain for the Internet of Things. *IEEE Access* 2018, 6, 32979–33001.
8. Ghosh, A.; Chakraborty, D.; Law, A. Artificial intelligence in Internet of things. *CAAI Trans. Intell. Technol.* 2018, 3, 208–218.
9. Meigal, A.; Korzun, D.; Gerasimova-Meigal, L.; Borodin, A.; Zavyalova, Y. Ambient Intelligence At-Home Laboratory for Human Everyday Life. *Int. J. Embed. Real-Time Commun. Syst.* 2019, 10, 117–134, doi:10.4018/IJERTCS.2019040108.
10. Qian, G.; Lu, S.; Pan, D.; Tang, H.; Liu, Y.; Wang, Q. Edge Computing: A Promising Framework for Real-Time Fault Diagnosis and Dynamic Control of Rotating Machines Using Multi-Sensor Data. *IEEE Sens. J.* 2019, 19, 4211–4220, doi:10.1109/JSEN.2019.2899396.
11. Bazhenov, N.; Korzun, D. Event-Driven Video Services for Monitoring in Edge-Centric Internet of Things Environments. In Proceedings of the Conference of Open Innovation Association (FRUCT), Helsinki, Finland, 5–8 November 2019; pp. 47–56.
12. Kulakov, K. An Approach to Efficiency Evaluation of Services with Smart Attributes. *Int. J. Embed. Real-Time Commun. Syst.* 2017, 8, 64–83, doi:10.4018/IJERTCS.2017010105.
13. Marchenkov, S.; Korzun, D.; Shabaev, A.; Voronin, A. On applicability of wireless routers to deployment of smart spaces in Internet of Things environments. In Proceedings of the 2017 IEEE 9th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2017), Bucharest, Romania, 21–23 September 2017; Volume 2, pp. 1000–1005.
14. Korzun, D.; Varfolomeyev, A.; Shabaev, A.; Kuznetsov, V. On dependability of smart applications within edge-centric and fog computing paradigms. In Proceedings of the 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT 2018), Kiev, Ukraine, 24–27 May 2018; pp. 502–507.
15. Korzun, D.; Kashevnik, A.; Balandin, S.; Smirnov, A. The smart-M3 platform: Experience of smart space application development for internet of things. In *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*; Springer Verlag: Berlin/Heidelberg, Germany, 2015; Volume 9247, pp. 56–67.
16. Types of Artificial Neural Networks—Wikipedia. Available online: https://en.wikipedia.org/wiki/Types_of_artificial_neural_networks (accessed on 22 July 2020).
17. Kumar, A.; Goyal, S.; Varma, M. Resource-Efficient Machine Learning in 2 KB RAM for the Internet of Things. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 1935–1944.
18. Kusupati, A.; Singh, M.; Bhatia, K.; Kumar, A.; Jain, P.; Varma, M. FastGRNN: A Fast, Accurate, Stable and Tiny Kilobyte Sized Gated Recurrent Neural Network. In Proceedings of the Advances in Neural Information Processing Systems 2018, Montreal, QC, Canada, 3–8 December 2018.
19. Friedman, J. Stochastic gradient boosting. *Comput. Stat. Data Anal.* 2002, 38, 367–378, doi:10.1016/S0167-9473(01)00065-2.
20. Gupta, C.; Suggala, A.; Goyal, A.; Simhadri, H.; Paranjape, B.; Kumar, A.; Goyal, S.; Udupa, R.; Varma, M.; Jain, P. ProtoNN: Compressed and Accurate kNN for Resource-scarce Devices. In Proceedings of the 34th International Conference on Machine Learning; Precup, D., Teh, Y.W., Eds.; International Convention Centre: Sydney, Australia, 2017; Volume 70, pp. 1331–1340.
21. Gural, A.; Murmann, B. Memory-Optimal Direct Convolutions for Maximizing Classification Accuracy in Embedded Applications. In Proceedings of the 36th International Conference on Machine Learning; Chaudhuri, K., Salakhutdinov, R., Eds.; PMLR: Long Beach, CA, USA, 2019; Volume 97, pp. 2515–2524.

22. Zhang, J.; Lei, Q.; Dhillon, I. Stabilizing Gradients for Deep Neural Networks via Efficient {SVD} Parameterization. In Proceedings of the 35th International Conference on Machine Learning; Dy, J., Krause, A., Eds.; PMLR: Stockholm Sweden, 2018; Volume 80, pp. 5806–5814.
23. Tanaka, G.; Yamane, T.; Héroux, J.B.; Nakane, R.; Kanazawa, N.; Takeda, S.; Numata, H.; Nakano, D.; Hirose, A. Recent advances in physical reservoir computing: A review. *Neural Netw.* 2019, 115, 100–123, doi:10.1016/j.neunet.2019.03.005.
24. Velichko, A.; Ryabokon, D.; Khanin, S.; Sidorenko, A.; Rikkiev, A. Reservoir computing using high order synchronization of coupled oscillators. *IOP Conf. Ser. Mater. Sci. Eng.* 2020, 862, 52062, doi:10.1088/1757-899x/862/5/052062.
25. Yamane, T.; Katayama, Y.; Nakane, R.; Tanaka, G.; Nakano, D. Wave-Based Reservoir Computing by Synchronization of Coupled Oscillators BT—Neural Information Processing; Arik, S., Huang, T., Lai, W.K., Liu, Q., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2015; pp. 198–205.
26. Velichko A Method for Evaluating Chimeric Synchronization of Coupled Oscillators and Its Application for Creating a Neural Network Information Converter. *Electronics* 2019, 8, 756, doi:10.3390/electronics8070756.
27. Donahue, C.; Merkel, C.; Saleh, Q.; Dolgovs, L.; Ooi, Y.; Kudithipudi, D.; Wysocki, B. Design and analysis of neuromemristive echo state networks with limited-precision synapses. In Proceedings of the 2015 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), Verona, NY, USA, 26–28 May 2015; pp. 1–6.
28. Larger, L.; Baylón-Fuentes, A.; Martinenghi, R.; Udaltsov, V.; Chembo, Y.; Jacquot, M. High-Speed Photonic Reservoir Computing Using a Time-Delay-Based Architecture: Million Words per Second Classification. *Phys. Rev. X* 2017, 7, 11015, doi:10.1103/PhysRevX.7.011015.
29. Ozturk, M.; Xu, D.; Príncipe, J. Analysis and Design of Echo State Networks. *Neural Comput.* 2006, 19, 111–138, doi:10.1162/neco.2007.19.1.111.
30. Wijesinghe, P.; Srinivasan, G.; Panda, P.; Roy, K. Analysis of Liquid Ensembles for Enhancing the Performance and Accuracy of Liquid State Machines. *Front. Neurosci.* 2019, 13, 504, doi:10.3389/fnins.2019.00504.
31. Mantas Lukoševičius; Herbert Jaeger; Reservoir computing approaches to recurrent neural network training. *Computer Science Review* 2009, 3, 127-149, [10.1016/j.cosrev.2009.03.005](https://doi.org/10.1016/j.cosrev.2009.03.005).

Retrieved from <https://encyclopedia.pub/entry/history/show/8899>