On The Expressivity of Recurrent Neural Cascades (Extended Abstract)*

Nadezda A. Knorozova[1,†], Alessandro Ronca[2,*,†]

[1] Relational AI
[2] University of Oxford

Abstract

Recurrent Neural Cascades (RNCs) are the recurrent neural networks with no cyclic dependencies among recurrent neurons. This class of recurrent neural networks is successfully used in practice. Besides training methods for a fixed architecture such as backpropagation, the cascade architecture naturally allows for constructive learning methods, where recurrent nodes are added incrementally one at a time, often yielding smaller networks. Furthermore, acyclicity amounts to a structural prior that even for the same number of neurons yields a more favourable sample complexity compared to a fully-connected architecture. A central question is whether the advantages of the cascade architecture come at the cost of a reduced expressivity. We provide new insights into this question. We show that the regular languages captured by RNCs with sign and tanh activation with positive recurrent weights are the *star-free* regular languages. In order to establish our results we develop a novel framework where the capabilities of RNCs are assessed by analysing which semigroups and groups a single neuron is able to implement. A notable implication of our framework is that RNCs can achieve the expressivity of all regular languages by introducing neurons that can implement groups.

Keywords

Recurrent Neural Networks, Recurrent Neural Cascades, Theory of Neural Networks, Expressivity, Star-free Regular Languages, Automata Theory, Algebraic Automata Theory, Krohn-Rhodes Theory, Semigroups, Groups, Formal languages, Temporal Logic

1. Introduction

Recurrent Neural Cascades (RNCs) are the subclass of recurrent neural networks where recurrent neurons are cascaded. Namely, they can be layed out into a sequence so that every neuron has access to the state of the preceding neurons as well as to the external input; and, at the same time, it has no dependency on the subsequent neurons. RNCs have been successfully applied in many different areas, including information diffusion in social networks [1], geological hazard predictions [2], automated image annotation [3], intention recognition [4], and optics [5].

RNCs offer several advantages over fully-connected recurrent networks. First, RNCs have a more favourable *sample complexity*, or dually better *generalisation capabilities*. This comes for the reduced number of weights, half the one of a fully-connected recurrent network, which implies a smaller VC dimension [6]. Second, the acyclic structure of the cascade architecture naturally allows for so-called *constructive learning* techniques [7, 8]. These techniques construct the network architecture dynamically during the training, often yielding smaller networks, faster training and improved generalisation. One such method is *recurrent cascade correlation*, which builds the architecture incrementally adding one recurrent neuron at a time [7]. RNCs emerge naturally here from the fact that every node does not depend on a node added later. RNCs also admit learning methods for fixed architectures, such as *backpropagation through time* [9], where only the weights are learned. For these methods the advantage of the cascade architecture comes from the reduced number of weights.

LNSAI 2024: First International Workshop on Logical Foundations of Neuro-Symbolic AI, August 05, 2024, Jeju, South Korea

* The full paper is available on arXiv: https://arxiv.org/abs/2312.09048.

*Corresponding author.
† These authors contributed equally.
nadezda.knorozova@relational.ai (N. A. Knorozova); alessandro.ronca@cs.ox.ac.uk (A. Ronca)
https://www.cs.ox.ac.uk/people/alessandro.ronca/ (A. Ronca)
0000-0002-2239-5141 (N. A. Knorozova); 0000-0002-0131-2087 (A. Ronca)

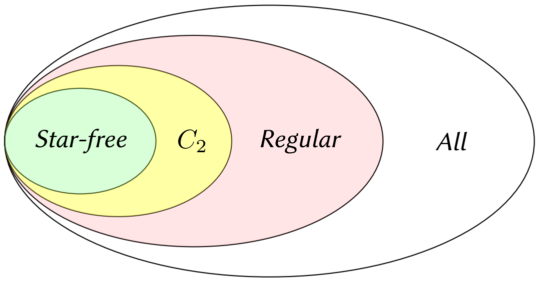

Figure 1: Relevant classes of languages. The label *'Star-free'* denotes the star-free regular languages, the label *'C_2'* denotes an extension of the star-free regular languages characterised by the group C_2, the label *'Regular'* denotes the regular languages, and the label *'All'* denotes all formal languages.

A central question is whether the advantages of the cascade architecture come at the cost of a reduced *expressivity* compared to the fully-connected architecture. The studies so far have shown that there exist regular languages that are not captured by RNCs with monotone activation such as tanh [10]. However, an exact characterisation of their expressitivity is still missing. Furthermore, it is unclear whether the inability to capture all regular languages is a limitation of the cascade architecture, or rather of the considered activation functions. We continue this investigation and provide new insights into the capabilities of RNCs to capture regular languages.

2. Main Contributions

Expressivity Results. We develop an analysis of the capabilities of RNCs establishing the expressivity results described next, involving the classes of languages depicted in Figure 1.

1. RNCs with sign or tanh activations capture the star-free regular languages. The expressivity result already holds when recurrent weights are restricted to be positive. In terms of Figure 1, their expressivity includes the green area.
2. RNCs with sign or tanh activations and positive recurrent weights do not capture any regular language that is not star-free. In terms of Figure 1, their expressivity does not include the red and yellow areas.
3. Allowing for negative recurrent weights properly extends the expressivity of RNCs with sign and tanh activations beyond the star-free regular languages. One witnessing language is the language of all strings of even length, which is not star-free. In terms of Figure 1, the former language belongs to the yellow area, characterised by the group C_2, the cyclic group of order two.
4. We show that in principle the expressivity of RNCs can be extended gradually to all regular languages in a controlled way. It suffices to identify appropriate recurrent neurons. In particular, neurons that can implement finite simple groups. As a first step, we show that second-order sign and tanh neurons can implement the computations described by the group C_2, the cyclic group of order two. In terms of Figure 1, second-order neurons allow for capturing the languages corresponding to the yellow area. Thus, RNCs of first-order and second-order neurons capture the green and yellow areas.

Points 1 and 2 establish an important connection between recurrent neural cascades and *star-free regular languages*. Specifically, they establish the importance of the sign of recurrent weights, and hence they isolate the subclass RNC_+ of recurrent neural cascades with positive recurrent weights as a particularly important class. In fact, as a corollary of Points 1 and 2, the regular languages recognised by RNC_+ are exactly the star-free regular languages.

A Novel Framework to study the expressivity of RNNs. As a result of our investigation we develop a novel framework where recurrent neural networks are analysed through the lens of *Semigroup and Group Theory*. The framework is of independent interest, as its potential goes beyond our current results. The framework allows for establishing the expressivity of RNCs by analysing the capabilities of a single neuron from the point of view of which semigroups and groups it can implement. If a neuron can implement the so-called *flip-flop monoid*, then cascades of such neurons capture the star-free regular languages. To go beyond that, it is sufficient to introduce neurons that implement *groups*. Our framework can be readily used to analyse the expressivity of RNCs with neurons that have not been considered in this work. In particular, we introduce abstract flip-flop and group neurons, which are the neural counterpart of the flip-flop monoid and of any given group. To show expressivity results, it is sufficient to instantiate our abstract neurons. Specifically in this work we show how to instantiate flip-flop neurons with (first-order) sign and tanh, as well as a family of grouplike neurons with second-order sign and tanh. In a similar way, other results can be obtained by instantiating the abstract neurons with different activation functions.

3. Significance of the Results

Our expressivity results provide a more comprehensive understanding of the expressivity of recurrent neural cascades. Our analysis is fine-grained, and it highlights the role of different aspects of the architecture of a neural network such as the role of cyclicity and the sign of recurrent weights. Notably, our results establish an important connection between the subclass RNC_+ and the star-free regular languages. This makes RNC_+ a strong candidate for learning temporal patterns, since the star-free regular languages are a central class that corresponds to the expressivity of many well-known formalisms. Such formalisms include *star-free regular expressions* from where they take their name [11], *monadic first-order logic* on finite linearly-ordered domains [12], *past temporal logic* [13], and *linear temporal logic* on finite traces [14]. They are also the languages recognised by *counter-free automata* as well as *group-free automata* [11]. On one hand, our result introduces an opportunity of employing RNC_+ for learning targets that one would describe in any of the above formalisms. For such targets, RNCs are sufficiently expressive and, compared to fully-connected recurrent neural networks, offer a more favorable sample complexity along with a wider range of learning algorithms. On the other hand, it places RNCs alongside well-understood formalisms with the possibility of establishing further connections and leveraging many existing fundamental results.

Our results establish a formal correspondence between continuous systems such as recurrent neural networks and discrete abstract objects such as automata, groups, and semigroups. Effectively they bridge recurrent neural networks with *algebraic automata theory*, cf. [11], two fields that developed independently and so far have not been considered to have any interaction.

4. Relevance to Neuro-Symbolic AI

The paper is relevant for Neuro-Symbolic AI since it studies the expressivity of recurrent neural networks in terms of formal languages, establishing connections between neural formalisms such as Recurrent Neural Cascades and symbolic formalisms such as automata and linear temporal logic. From a methodological point of view, the paper analyses neural networks using methods typical of symbolic AI such as automata theory.

5. Related Work

In our work, the connection between RNNs and automata plays an important role. Interestingly, the connection appears to exist from the beginning of automata theory [15]: "*In 1956 the series Automata Studies (Shannonon and McCarthy [1956]) was published, and automata theory emerged as a relatively autononmous discipline. [...] much interest centered on finite-state sequential machines, which first arose not in the abstract form [...], but in connection with the input-output behaviour of a McCulloch-Pitts net [...]*". The relationship between automata and the networks by [16] is discussed both in [17] and [18]. Specifically, an arbitrary automaton can be captured by a McCulloch-Pitts network. Our results reinforce this result, extending it to sign and tanh activation. The extension to tanh is important because of its differentiability, and it requires a different set of techniques since it is not binary, but rather real-valued. Furthermore, our results extend theirs by showing a correspondence between RNCs and group-free automata.

The Turing-completeness of RNNs as an *offline model* of computation are studied in [19, 20, 21, 22]. In this setting, an RNN is allowed to first read the entire input sequence, and then return the output after an arbitrary number of iterations, triggered by blank inputs. This differs from our study, which focuses on the capabilities of RNNs as *online machines*, which process the input sequence one element at a time, outputting a value at every step. This is the way they are used in many practical applications such as Reinforcement Learning, cf. [23, 24, 25, 26].

The expressivity of RNNs in terms of whether they capture all *rational series* or not has been analysed in [27]. This is a class of functions that includes all regular functions. Thus, it is a coarse-grained analysis compared to ours, which focuses on subclasses of the regular languages.

The problem of *latching* one bit of information has been studied in [28] and [29]. This problem is related to star-free regular languages, as it amounts to asking whether there is an automaton recognising a language of the form sr^* where s is a set command and r is a read command. This is a subset of the functionalities implemented by a flip-flop semiautomaton. Their work established conditions under which a tanh neuron can latch a bit. Here we establish conditions guaranteeing that a tanh neuron homomorphically represents a flip-flop semiautomaton, implying that it can latch a bit. An architecture that amounts to a restricted class of RNCs has been considered in [30].

Transformers are another class of neural networks for sequential data [31]. They are a *non-uniform model of computation*, in the sense that inputs of different lengths are processed by different networks. This differs from RNNs which are a *uniform model of computation*. The expressivity of transformers has been studied in [32, 33, 34, 35, 36, 37].

Acknowledgments

Alessandro Ronca is supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 852769, ARiAT). For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript (AAM) version arising from this submission.

References

[1] J. Wang, V. W. Zheng, Z. Liu, K. C.-C. Chang, Topological recurrent neural network for diffusion prediction, in: IEEE ICDM, 2017.

[2] L. Zhu, L. Huang, L. Fan, J. Huang, F. Huang, J. Chen, Z. Zhang, Y. Wang, Landslide susceptibility prediction modeling based on remote sensing and a novel deep learning algorithm of a cascade-parallel recurrent neural network, Sensors 20 (2020).

[3] H.-C. Shin, K. Roberts, L. Lu, D. Demner-Fushman, J. Yao, R. M. Summers, Learning to read chest x-rays: Recurrent neural cascade model for automated image annotation, in: IEEE/CVF CVPR, 2016.

[4] D. Zhang, L. Yao, X. Zhang, S. Wang, W. Chen, R. Boots, B. Benatallah, Cascade and parallel convolutional recurrent neural networks on eeg-based intention recognition for brain computer interface, in: AAAI, 2018.

[5] Z. Xu, C. Sun, T. Ji, J. H. Manton, W. Shieh, Cascade recurrent neural network-assisted nonlinear equalization for a 100 gb/s pam4 short-reach direct detection system, Optics Letters 45 (2020).

[6] P. Koiran, E. D. Sontag, Vapnik-chervonenkis dimension of recurrent neural networks, Discret. Appl. Math. 86 (1998).

[7] S. E. Fahlman, The recurrent cascade-correlation architecture, in: NIPS, 1990.

[8] R. Reed, R. J. Marks II, Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks, MIT Press, 1999.
[9] P. Werbos, Backpropagation through time: What it does and how to do it, Proc. of the IEEE 78 (1990).
[10] C. Giles, D. Chen, G.-Z. Sun, H.-H. Chen, Y.-C. Lee, M. Goudreau, Constructive learning of recurrent neural networks: Limitations of recurrent cascade correlation and a simple solution, IEEE Transactions on Neural Networks 6 (1995).
[11] A. Ginzburg, Algebraic Theory of Automata, Academic Press, 1968.
[12] R. McNaughton, S. A. Papert, Counter-Free Automata, The MIT Press, 1971.
[13] Z. Manna, A. Pnueli, Completing the temporal picture, Theor. Comput. Sci. 83 (1991).
[14] G. De Giacomo, M. Y. Vardi, Linear temporal logic and linear dynamic logic on finite traces, in: IJCAI, 2013.
[15] M. Arbib, Theories of Abstract Automata, Automatic Computation, Prentice-Hall, 1969.
[16] W. S. McCulloch, W. Pitts, A logical calculus of the ideas immanent in nervous activity, The bulletin of mathematical biophysics 5 (1943).
[17] S. C. Kleene, Representation of events in nerve nets and finite automata, Automata studies 34 (1956).
[18] M. L. Minsky, Computation: Finite and Infinite Machines, Prentice-Hall, 1967.
[19] H. T. Siegelmann, E. D. Sontag, On the computational power of neural nets, J. Comput. Syst. Sci. 50 (1995).
[20] J. Kilian, H. T. Siegelmann, The dynamic universality of sigmoidal neural networks, Inf. Comput. 128 (1996).
[21] J. N. Hobbs, H. T. Siegelmann, Implementation of universal computation via small recurrent finite precision neural networks, in: IJCNN, 2015.
[22] S. Chung, H. T. Siegelmann, Turing completeness of bounded-precision recurrent neural networks, in: NeurIPS, 2021.
[23] B. Bakker, Reinforcement learning with long short-term memory, in: NeurIPS, 2001.
[24] D. Ha, J. Schmidhuber, Recurrent world models facilitate policy evolution, in: NeurIPS, 2018.
[25] M. J. Hausknecht, P. Stone, Deep recurrent Q-learning for partially observable MDPs, in: AAAI Fall Symposia, 2015.
[26] S. Kapturowski, G. Ostrovski, J. Quan, R. Munos, W. Dabney, Recurrent experience replay in distributed reinforcement learning, in: ICLR, 2019.
[27] W. Merrill, G. Weiss, Y. Goldberg, R. Schwartz, N. A. Smith, E. Yahav, A formal hierarchy of RNN architectures, in: ACL, 2020.
[28] Y. Bengio, P. Y. Simard, P. Frasconi, Learning long-term dependencies with gradient descent is difficult, IEEE Trans. Neural Networks 5 (1994).
[29] P. Frasconi, M. Gori, M. Maggini, G. Soda, Unified integration of explicit knowledge and learning by example in recurrent networks, IEEE Trans. Knowl. Data Eng. 7 (1995).
[30] P. Frasconi, M. Gori, G. Soda, Local feedback multilayered networks, Neural Comput. 4 (1992).
[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: NeurIPS, 2017.
[32] M. Hahn, Theoretical limitations of self-attention in neural sequence models, Trans. Assoc. Comput. Linguistics 8 (2020).
[33] Y. Hao, D. Angluin, R. Frank, Formal language recognition by hard attention transformers: Perspectives from circuit complexity, Trans. Assoc. Comput. Linguistics 10 (2022).
[34] W. Merrill, A. Sabharwal, N. A. Smith, Saturated transformers are constant-depth threshold circuits, Trans. Assoc. Comput. Linguistics 10 (2022).
[35] B. Liu, J. T. Ash, S. Goel, A. Krishnamurthy, C. Zhang, Transformers learn shortcuts to automata, in: ICLR, 2023.
[36] D. Chiang, P. Cholak, A. Pillay, Tighter bounds on the expressivity of transformer encoders, in: ICML, 2023.
[37] W. Merrill, A. Sabharwal, A logic for expressing log-precision transformers, in: NeurIPS, 2023.