DynamicBlocks: A Generalized ResNet Architecture

Derek Onken[1], **Simion Novikov**[2], **Eran Treister**[2], **Eldad Haber**[3], **and Lars Ruthotto**[4,1]

1 Department of Computer Science, Emory University **2** Ben-Gurion University of the Negev **3** University of British-Columbia **4** Department of Mathematics, Emory University

DOI: 00.00000/joss.00000

Software

- Review
- Repository
- Archive

Submitted: 00 January 0000
Published: 00 January 0000

License
Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License (CC-BY).

Summary

Deep Residual Neural Networks (ResNets) demonstrate impressive performance on several image classification tasks (He et al. 2016). ResNets feature a skip connection, which observably increases a model's robustness to vanishing and exploding gradients. ResNets also possess interesting theoretical properties; for example, the forward propagation through a ResNet can be interpreted as an explicit Euler method applied to a nonlinear ordinary differential equation (ODE) (Weinan 2017; Haber and Ruthotto 2017). Similarly, Ruthotto and Haber (2018) introduces a similar interpretation of convolutional ResNets as partial differential equations (PDE). These insights provide more theoretical narrative and insight while motivating new network architectures from different types of differential equations, e.g., reversible hyperbolic network (Chang et al. 2018) or parabolic networks (Ruthotto and Haber 2018). Recent attention focuses on improving the time integrators used in forward propagation, e.g., higher-order single and multistep methods (Lu et al. 2018), black-box time integrators (Chen et al. 2018), and semi-implicit discretizations (Haber et al. 2019).

This toolbox exists primarily to facilitate further research and development in convolutional residual neural networks motivated by PDEs. To this end, we generalize the notation of ResNets, referring to each of its parts that amend a PDE interpretation (i.e., each set of several consecutive convolutional layers of fixed number of channels) as a `dynamic block`. A dynamic block can then be compactly described by a layer function and parameters of the time integrator (e.g., time discretization points). We provide the flexibility to model several state-of-the-art networks by combining several dynamic blocks (acting on different image resolutions and number of channels) through connective units (i.e., a convolutional layer to change the number of channels followed by a pooling layer).

Our `DynamicBlocks` toolbox provides a general framework to experiment with different time discretizations and PDE solvers. In its first version, we include capabilities to obtain a more general version of ResNet based on a forward Euler (Runge-Kutta 1) module that can handle arbitrary, non-uniform time steps. By allowing different time discretizations for the features and parameters, we provide the ability to decouple the states (features) and controls (parameters). Through this decoupling, we can determine the separate relationships of weights and layers with model performance. Furthermore, we include a fourth-order accurate Runge-Kuttta 4 block to demonstrate the generalizability of the architecture to accept other PDE solvers.

In the training, we primarily focus on discretize-optimize learning methods, which are popular in optimal control and whose favorable properties for ResNets are shown in Gholami, Keutzer, and Biros (2019). However, dynamic blocks can also employ optimize-then-discretize approaches as in Neural ODEs (Chen et al. 2018).

Acknowledgements

This material is in part based upon work supported by the National Science Foundation under Grant Number DMS-1751636. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Chang, Bo, Lili Meng, Eldad Haber, Lars Ruthotto, David Begert, and Elliot Holtham. 2018. "Reversible Architectures for Arbitrarily Deep Residual Neural Networks." In *AAAI Conference on AI*. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/viewPaper/16517.

Chen, Tian Qi, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. 2018. "Neural Ordinary Differential Equations." In *Advances in Neural Information Processing Systems 31*, 6571–83. http://papers.nips.cc/paper/7892-neural-ordinary-differential-equations.pdf.

Gholami, Amir, Kurt Keutzer, and George Biros. 2019. "ANODE: Unconditionally Accurate Memory-Efficient Gradients for Neural Odes." *arXiv:1902.10298*. http://arxiv.org/abs/1902.10298.

Haber, Eldad, Keegan Lensink, Eran Treister, and Lars Ruthotto. 2019. "IMEXnet a Forward Stable Deep Neural Network." In *Proceedings of the 36th International Conference on Machine Learning*, 97:2525–34. http://proceedings.mlr.press/v97/haber19a.html.

Haber, Eldad, and Lars Ruthotto. 2017. "Stable Architectures for Deep Neural Networks." *Inverse Problems* 34 (1): 014004. https://doi.org/10.1088/1361-6420/aa9a90.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. "Deep Residual Learning for Image Recognition." In *Proceedings of the Ieee Conference on Computer Vision and Pattern Recognition*, 770–78. https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper.pdf.

Lu, Yiping, Aoxiao Zhong, Quanzheng Li, and Bin Dong. 2018. "Beyond Finite Layer Neural Networks: Bridging Deep Architectures and Numerical Differential Equations." In *Proceedings of the 35th International Conference on Machine Learning*, 80:3276–85. http://proceedings.mlr.press/v80/lu18d.html.

Ruthotto, Lars, and Eldad Haber. 2018. "Deep Neural Networks Motivated by Partial Differential Equations." *arXiv:1804.04272*. https://arxiv.org/abs/1804.04272.

Weinan, E. 2017. "A Proposal on Machine Learning via Dynamical Systems." *Comm. Math. Statist.* 5 (1): 1–11. https://doi.org/10.1007/s40304-017-0103-z.