Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)
Doctoral Consortium Track

Safety Analysis of Deep Neural Networks

Dario Guidotti
University of Genoa
dario.guidotti@edu.unige.it

Abstract

Deep Neural Networks (DNNs) are popular machine learning models which have found successful application in many different domains across computer science. Nevertheless, providing formal guarantees on the behavior of neural networks is hard and therefore their reliability in safety-critical domains is still a concern. Verification and repair emerged as promising solutions to address this issue. In the following I will present some of my recent efforts in this area.

1 Introduction

The area of verification of DNNs is concerned with providing methods to certify whether a DNN satisfies a given input-output specification. Numerous methodologies have been presented for different specifications and architectures, *e.g.*, [Katz *et al.*, 2019; Balunovic *et al.*, 2019; Tjeng *et al.*, 2019]. Despite remarkable progress in the field, verification of neural networks is computationally intensive and is still challenging for non-trivial architectures. The first main focus of my PhD research is this very problem as I will describe in the following sections. While verification aims to prove whether a neural network is compliant with a certain specification, how to efficiently repair a network that has been proven to be faulty remains an open question. A straightforward repair methodology is to retrain the network augmenting the dataset with counterexamples obtained by verification [Pulina and Tacchella, 2010; Dreossi *et al.*, 2018]. Other approaches that do not require re-training have been proposed [Goldberger *et al.*, 2020; Papusha *et al.*, 2020]. These approaches use Satisfiability Modulo Theory (SMT) or Mixed Integer Linear Programming (MILP) to perform minimum modification to the parameters of the networks so that the specification of interest is guaranteed to be satisfied. Repair is the second main focus of my PhD research. In the following, I will present some results obtained during my PhD and, after that, I will outline the activities I plan to carry out to complete my PhD studies.

2 Current Results

The aim of my PhD thesis is to provide a unified framework in which networks can be trained, analyzed and repaired with emphasis on the ease of use and compatibility among different learning frameworks and verification methodologies. My motivation is that providing capabilities traditionally pertaining the learning community (*e.g., pruning, quantization etc.*) together with ones concerning verification and repair enables the user to leverage synergies between such capabilities. I studied an example of such synergies in [Guidotti *et al.*, 2020], where I investigated the possible interplay between DNN pruning and their verification. Pruning consists in removing or blending components of a NN to reduce its complexity: its main application until now has been to reduce the dimension of NNs so that they can be deployed on hardware with little computational resources. I showed how different pruning techniques can be used to produce networks that are easier to verify but have accuracy and robustness comparable with the original unpruned ones. My results were consistent in showing that pruned networks were easier to verify then the unpruned ones for all the verification and pruning methodology I considered. With the purpose of gaining a better understanding on how learning techniques and verification methodology inter- act, I decided to develop my own verification methodology, which is inspired by the work done in [Tran *et al.*, 2019]. The original methodology is based on abstract interpretation and, in particular, on the star set abstraction. Given a DNN and a specification ϕ defining input and output constraints, the methodology first computes the output reachable set corresponding to the input set identified by input constraints in ϕ. Then, the intersection is computed between the output reachable set and the set obtained by negating output constraints in ϕ. If the intersection is empty then the property is verified and the net is safe. The methodology provides both an incomplete and a complete algorithm for verification. The incomplete algorithm leverages over-approximation to speed up computation, with the downside of a loss of precision which can cause a network to be erroneously identified as unsafe. One of my aims was to enhance the precision of the over-approximate algorithm without increasing too much its computational complexity. In particular I proposed a new verification methodology that abstracts only selected portion of a DNN based on novel decision heuristics. Experimental results showed that my methodology is able to verify properties which the over-approximate version alone is not able to verify and it does so in less time than the complete version. While [Tran *et al.*, 2019] only support DNNs with ReLU acti-

vations, my approach also supports sigmoids via a new over-approximation scheme which can operate at different levels of precision and is easily adaptable to other non-linear activation functions. The methodologies presented in this section has been implemented in the tool NEVER2 [Guidotti *et al.*, 2021] which provides capabilities for the training, pruning, verification, repair of neural networks.

3 Planned Contribution and Future Research

In the following I will outline some research directions I intend to pursue in the last months of my PhD.

3.1 Over-approximation Refinement and Repair

I plan to enhance my verification methodology using a counter-example guided refinement [Clarke *et al.*, 2000] of the over-approximation. To do so I am currently working on sampling and search methodologies needed for extracting input counter-examples from the reachable set computed by the verification algorithm. I plan to leverage spurious counter-examples (*i.e.*, counter-example which are reported as unsafe from the verification procedure but that are actually compliant with the property of interest) to guide the refinement of the over-approximation, whereas I intend to use the valid counter-examples in a repair methodology continuing the work I started in [Guidotti *et al.*, 2019a; Guidotti *et al.*, 2019b; Guidotti *et al.*, 2019c].

3.2 NeVer2

I also aim to improve the usability of NEVER2 by supporting an increasing variety of neural networks architectures. One of the first planned enhancement is the support for convolutional layers and other popular activation functions besides ReLU and Sigmoid. At present I am also working on a graphical user interface for the tool and on extending the range of supported formats for loading and saving of network models. My aim is to provide support for the main learning frameworks, namely ONNX, PyTorch and Tensorflow/Keras.

3.3 Definition of a Standard for NNs Verification

Another relevant issue for the verification of neural networks is the current absence of a standard for verification benchmarks, *i.e.*, networks and properties thereof. I am currently working on this topic with other members of the VNN-LIB [1] initiative, whose aim is to encourage collaboration and facilitate research and development in verification of neural networks.

References

[Balunovic *et al.*, 2019] Mislav Balunovic, Maximilian Baader, Gagandeep Singh, Timon Gehr, and Martin T. Vechev. Certifying geometric robustness of neural networks. In *Proc. of NeurIPS'19*, pages 15287–15297, 2019.

[Clarke *et al.*, 2000] Edmund M. Clarke, Orna Grumberg, Somesh Jha, Yuan Lu, and Helmut Veith. Counterexample-guided abstraction refinement. In *Proc. of FM'00*, pages 154–169, 2000.

[Dreossi *et al.*, 2018] Tommaso Dreossi, Shromona Ghosh, Xiangyu Yue, Kurt Keutzer, Alberto L. Sangiovanni-Vincentelli, and Sanjit A. Seshia. Counterexample-guided data augmentation. In *Proc. of IJCAI'18*, pages 2071–2078, 2018.

[Goldberger *et al.*, 2020] Ben Goldberger, Guy Katz, Yossi Adi, and Joseph Keshet. Minimal modifications of deep neural networks using verification. In *Proc. of LPAR'20*, pages 260–278, 2020.

[Guidotti *et al.*, 2019a] Dario Guidotti, Francesco Leofante, Claudio Castellini, and Armando Tacchella. Repairing learned controllers with convex optimization: A case study. In *Proc. of CPAIOR'19*, pages 364–373, 2019.

[Guidotti *et al.*, 2019b] Dario Guidotti, Francesco Leofante, Luca Pulina, and Armando Tacchella. Verification and repair of neural networks: A progress report on convolutional models. In *Proc. of AI*IA'19*, pages 405–417, 2019.

[Guidotti *et al.*, 2019c] Dario Guidotti, Francesco Leofante, Armando Tacchella, and Claudio Castellini. Improving reliability of myocontrol using formal verification. *IEEE TNSRE*, 27(4):564–571, 2019.

[Guidotti *et al.*, 2020] Dario Guidotti, Francesco Leofante, Luca Pulina, and Armando Tacchella. Verification of neural networks: Enhancing scalability through pruning. In *Proc. of ECAI'20*, volume 325, pages 2505–2512, 2020.

[Guidotti *et al.*, 2021] Dario Guidotti, Luca Pulina, and Armando Tacchella. NeVer 2.0: A Framework for Learning and Verification of Neural Networks. In *(under review)*, 2021.

[Katz *et al.*, 2019] Guy Katz, Derek A. Huang, Duligur Ibeling, Kyle Julian, Christopher Lazarus, Rachel Lim, Parth Shah, Shantanu Thakoor, Haoze Wu, Aleksandar Zeljic, David L. Dill, Mykel J. Kochenderfer, and Clark W. Barrett. The marabou framework for verification and analysis of deep neural networks. In *Proc. of CAV'19*, pages 443–452, 2019.

[Papusha *et al.*, 2020] Ivan Papusha, Rosa Wu, Joshua Brulé, Yanni Kouskoulas, Daniel Genin, and Aurora Schmidt. Incorrect by construction: Fine tuning neural networks for guaranteed performance on finite sets of examples. *CoRR*, abs/2008.01204, 2020.

[Pulina and Tacchella, 2010] Luca Pulina and Armando Tacchella. An abstraction-refinement approach to verification of artificial neural networks. In *Proc. of CAV'10*, pages 243–257, 2010.

[Tjeng *et al.*, 2019] Vincent Tjeng, Kai Y. Xiao, and Russ Tedrake. Evaluating robustness of neural networks with mixed integer programming. In *Proc. of ICLR'19*, 2019.

[Tran *et al.*, 2019] Hoang-Dung Tran, Diego Manzanas Lopez, Patrick Musau, Xiaodong Yang, Luan Viet Nguyen, Weiming Xiang, and Taylor T. Johnson. Star-based reachability analysis of deep neural networks. In *Proc. of FM'18*, pages 670–686, 2019.

[1] http://www.vnnlib.org/