Research Proposal

Remi Desmartin

May 24, 2022

1 Background

The increased complexity of critical software systems has made traditional testing tools insufficient for detecting edge failure cases. One solution, which has gained momentum in the recent years, is the use of formal methods, that use rigorous mathematical tools to model the system, formulate specifications and verify that they are met [9].

The introduction of artificial intelligence (AI) components, often relying on machine learning (ML) algorithms is a contributing factor in the complexification of safety-critical systems. Neural networks (NN), a group of ML algorithm, have shown exceptional performance at dealing with noisy data and have been deployed in safety-critical applications like autonomous cars [] or automated trading agents [2]. Because ML components' failure can lead to the of the whole system's failure [6], formally verifying the system necessarily involves verifying the ML component.

Imandra [10] is a verification tool that has been successfully applied to verifying complex systems in the FinTech domain [11], like trading algorithms and blockchain smart contract infrastructure. It combines features of both functional languages and interactive and automated theorem provers. Imandra's logic is based on a pure, higher-order subset of OCaml, and functions written in Imandra are at the same time valid OCaml code that can be executed.

Imandra's mode of interactive proof development is based on a typed, higher-order lifting of the *Boyer-Moore waterfall* [3] for automated induction, tightly integrated with novel techniques for SMT modulo recursive functions. In short, Imandra is both a functional programming language in which programs can be implemented and executed, and a reasoning engine which can apply formal verification techniques to these programs.

The biggest shortcoming of NNs is their sensitivity to small variations in the input; this sensitivity can be actively exploited by *adversarial attacks* [4]. Thus, verifying ML components (especially NNs) has been the focus of recent research, essentially divided into SMT-based tools [8] and those based on abstract interpretation [12, 7]. A large part of verification efforts focus on proving that NNs are resistant to adversarial attacks, i.e. they classify consistently examples that are close within the input space. This property is called robustness; multiple

formal definitions of it are used [5]. Even though it is the primary focus of NN verification its scope is limited. Other properties like fairness or domain-specific property are also desirable and worthy of investigation.

2 Research Question

Currently, verification of ML algorithms such as neural networks requires dedicated tools. These tools are often limited to narrow use cases: a single family of NNs (e.g. MLP with Relu activation, RNN) or a single type of properties (e.g. a single definition of robustness). In addition, the integration into more generic and verification frameworks that are able to certify the larger systems containing ML.

There is a need for a flexible and generic NN verification framework that allows building systems with ML components with strong safety and security guarantees.

Thus, we will try to investigate the question: is Imandra the right tool to enable a wider range of verification tasks than is currently available?

In order to answer this research question, we will investigate:

1. The role of ML-components in complex systems (e.g. simple inputs to controllers, or controllers themselves);
2. The verification of NN components;
3. The integration of ML components' verification in larger verification frameworks

We will remain open to new questions as they surely will arise from these initial explorations.

3 Aims

Our aim is to make designing, implementing and deploying trustworthy software easier by building a comprehensive NN verification library in Imandra.

To that end, our objectives are the following:

1. integrate third-party NN verifiers into a generic verification tool like Imandra
2. support verification of advanced NN architectures such as convolutional NNs
3. facilitate interoperability with other machine learning tools by implementing the ONNX standard in Imandra
4. demonstrate the integration of ML components verification to larger systems verification.

4 Methodology

Initial research has been conducted as part of my Master's thesis. So far, we have managed to implement different formalisation of NNs in Imandra and multiple specifications of desirable properties for NNs. Thanks to these implementations, we were able to show that the expressivity of Imandra's programming language and its tight coupling with a range of automated reasoning techniques enables the verification of a range of properties. Notably, we have verified properties on compressed versions of some networks from the the ACAS Xu benchmark [8].

Our plan is to pursue this effort, by further developing the library. Possible directions include creating a "library" of proved lemmas to help Imandra to reason about more complex or larger networks, and interfacing our library with external reasoning tools.

In the medium term and depending on results, it will be interesting to evaluate our library against standard NN Verification benchmarks such as the ones proposed by the VNN-Comp [1].

5 Research Environment and Supervision

Imandra will provide infrastructural support for this research, in the form of funding and guidance throughout of the project. Prof. Ekaterina Komendentskaya, Prof. Kathrin Stark, and Dr. Grant Passmore will co-supervise this research. Passmore is the creator of Imandra and the co-founder and co-CEO of the company that comercialises it.

The research will benefit from the environment of the lab for AI Verification (LAIV). It will be enriched by attending the lab's seminars and schools and discussions with the lab's members.

The research will also be conducted as part of the Edinburgh Center for Robotics. Its ecosystem will allow us to reach a broader range of academics and industry experts in the field of AI, as well as provide additional training opportunities.

6 Feasibility

We have already started the implementation of a library for verifying convolutional NNs. With this library were able to verify robustness on quantised convolutional neural networks, to reason by induction on individual layers and to use the same formalisation to improve network explainability.

We are confident that we will be able to extend this library to improve its performance and the range of supported verification properties and network architectures. One challenging task will be to implement the ONNX (open neural network exchange format) standard; but if we manage to do it, it will open the way to a smooth integration with other tools implementing this standard.

In terms of infrastructure, we will benefit from Imandra's financial and technical backing. Imandra's support has already proved to be of great help in the first steps of this project and it will doubtlessly continue to do so.

7 Potential Impact

Doing this research within the LAIV would allow to contribute to the discussion and research on NN verification, in particular by presenting early results to seminars organised by the LAIV and to specialised conferences like FoMLAS.

The partnership with Imandra means that successful results would impact directly their users by allowing them to formally verify their ML-based systems.

References

[1] VNN 2022. https://sites.google.com/view/vnn2022. Accessed:2022-05-16.

[2] Bao, W., Yue, J., and Rao, Y. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLOS ONE 12*, 7 (2017), e0180944. Publisher: Public Library of Science.

[3] Boyer, R. S., and Moore, J. S. *A Computational Logic*. ACM Monograph Series. Academic Press, New York, 1979.

[4] Carlini, N., and Wagner, D. Towards Evaluating the Robustness of Neural Networks. In *2017 IEEE Symposium on Security and Privacy (SP)* (San Jose, CA, USA, May 2017), IEEE, pp. 39–57.

[5] Casadio, M., Daggitt, M. L., Komendantskaya, E., Kokke, W., Kienitz, D., and Stewart, R. Property-driven Training: All You (N)Ever Wanted to Know About. *CoRR abs/2104.01396* (2021). _eprint: 2104.01396.

[6] Dreossi, T., Donzé, A., and Seshia, S. A. Compositional Falsification of Cyber-Physical Systems with Machine Learning Components. In *NASA Formal Methods* (Cham, 2017), C. Barrett, M. Davies, and T. Kahsai, Eds., Lecture Notes in Computer Science, Springer International Publishing, pp. 357–372.

[7] Elboher, Y. Y., Gottschlich, J., and Katz, G. An Abstraction-Based Framework for Neural Network Verification. In *Computer Aided Verification - 32nd International Conference, CAV 2020, Los Angeles, CA, USA, July 21-24, 2020, Proceedings, Part I* (2020), S. K. Lahiri and C. Wang, Eds., vol. 12224 of *Lecture Notes in Computer Science*, Springer, pp. 43–65.

[8] Katz, G., Barrett, C. W., Dill, D. L., Julian, K., and Kochenderfer, M. J. Reluplex: An Efficient SMT Solver for Verifying Deep

Neural Networks. In *Computer Aided Verification - 29th International Conference, CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part I* (2017), R. Majumdar and V. Kuncak, Eds., vol. 10426 of *Lecture Notes in Computer Science*, Springer, pp. 97–117.

[9] Newcombe, C., Rath, T., Zhang, F., Munteanu, B., Brooker, M., and Deardeuff, M. How Amazon Web Services Uses Formal Methods. *Commun. ACM 58*, 4 (Mar. 2015), 66–73. Place: New York, NY, USA Publisher: Association for Computing Machinery.

[10] Passmore, G., Cruanes, S., Ignatovich, D., Aitken, D., Bray, M., Kagan, E., Kanishev, K., Maclean, E., and Mometto, N. The Imandra Automated Reasoning System (System Description). In *Automated Reasoning* (Cham, 2020), N. Peltier and V. Sofronie-Stokkermans, Eds., Lecture Notes in Computer Science, Springer International Publishing, pp. 464–471.

[11] Passmore, G. O. Some Lessons Learned in the Industrialization of Formal Methods for Financial Algorithms. In *Formal Methods* (Cham, 2021), M. Huisman, C. Păsăreanu, and N. Zhan, Eds., Lecture Notes in Computer Science, Springer International Publishing, pp. 717–721.

[12] Singh, G., Gehr, T., Püschel, M., and Vechev, M. An abstract domain for certifying neural networks. *Proceedings of the ACM on Programming Languages 3*, POPL (Jan. 2019), 1–30.