Automated machine learning for neural network verification

König, H.M.T.

Chapter 7

Conclusions and Outlook

In this chapter, we revisit the research questions presented in Chapter 1 and provide a detailed discussion on how each question has been addressed. Lastly, we discuss directions for future research.

7.1 Answers to Research Questions

In Chapter 1, we presented four research questions that guided the exploration and structure of this thesis. In the following, we will revisit each research question, outlining the methods we used to address them and summarising our key findings.

RQ1: What constitutes the state of the art in neural network verification?

In Chapter 3, we performed a comprehensive performance analysis of various CPU- and GPU-based neural network verification methods and revealed an algorithmic landscape characterised by significant performance diversity across different types of verification problem instances. From this study, we concluded that no single verifier consistently outperforms other methods in every scenario, challenging the notion of a universally superior algorithm within the neural network verification domain. Instead, we observed high levels of complementarity, *i.e.*, instances solved by one verifier that other verifiers could not solve and vice versa, quantified by means of marginal contribution and Shapley values. Moreover, these findings highlight the complex nature of neural network verification problems and suggest the usage of algorithm portfolios for optimal verification outcomes. Notice that these findings are in line with those for

other NP-hard problems; *e.g.*, the work of Leyton-Brown et al. [70].

RQ2: How can we leverage automated algorithm configuration techniques to improve the performance of a MIP-based verification system?

In Chapter 4, we investigated the application of automated algorithm configuration techniques to enhance the operational efficiency of MIP-based verification systems. Notably, we observed strong heterogeneity among different problem instances, which is not handled well by standard configuration approaches. Instead, we employed advanced portfolio construction techniques, which combine different solver configurations with complementary strengths into a parallel portfolio. This portfolio runs the solver configurations in parallel, stopping each configuration as soon as one of them has returned a solution. This implicitly ensures the we always benefit from the best-performing algorithm in the portfolio. Notably, we achieved substantial improvements in terms of running time and an increased number of successfully solved instances, despite the increased overhead demanded by the parallel portfolio. These outcomes highlight the potential of automated configuration and portfolio construction techniques as a fruitful approach for improving the performance of combinatorial solvers employed in the context of neural network verification systems, thereby streamlining the verification procedure overall. This confirms findings from previous work on related problems, notably mixed integer linear programming [46, 44, 45, 76].

RQ3: To which extent can we predict the running time of a given verification algorithm for a specific problem instance?

In Chapter 5, we investigated the possibilities of running time prediction for neural network verification algorithms. While we found that precise running time prediction (*i.e.*, regression) remains a challenging task, we developed a timeout prediction model that anticipates the feasibility of completing a verification query within a predefined time budget. This was enabled by newly defined features of the problem instance and the verification algorithm in use. Using this approach, we achieved a more efficient allocation of computational resources, strongly enhancing the overall efficiency of the verification procedure. Furthermore, our timeout prediction method could be leveraged in the context of parallel algorithm portfolios or, more specifically, per-instance algorithm selection; given several algorithms that run in parallel on a specific instance, our method could terminate those that are not able to solve the instance in the given

time budget. However, the verification algorithms we considered in our study did not display sufficient performance complementarity on the benchmarks we used. This could be explained by the fact that the verification algorithms were employed with configurations specifically tailored to the given benchmarks.

RQ4: How can we efficiently select the neural network model from a given set of models that achieves the highest certified robust accuracy?

In Chapter 6, we introduced a novel racing algorithm designed to guide the selection process of the most robust neural network model from a set of candidate models. In this context, we proposed a novel heuristic that captures the likelihood of a given instance to be robust or non-robust. Using this heuristic, we can guide the search towards neural network models that are most likely to show a high adversarial robustness. We found that our proposed solution significantly reduces the computational costs typically associated with model selection by iteratively eliminating less promising candidates, thereby facilitating a more efficient selection process of the optimal, *i.e.*, most robust model. This approach presents a practical solution to the challenge of robust model selection, ensuring computational resources are utilised judiciously while selecting the model with the highest robustness.

7.2 Directions for Future Research

With the work presented in this thesis, we sought to enable future progress in the field of neural network verification. These methods offer great potential for obtaining safety guarantees for neural-network-based AI systems, which is a crucial requirement for their use in high-risk domains, such as medical diagnosis or advanced driver assistant systems. At the same time, computational complexity remains a major challenge and current methods do not scale to complex architectures, such as large language models.

One general direction involves expanding this work to encompass a broader spectrum of neural network architectures and verification problems beyond local robustness for image classification models. Such an expansion would not only further validate the generalisability of the findings presented in this thesis but also potentially reveal new insights and challenges that could further refine the state of the art in neural network verification. While we considered only local robustness properties in this thesis, mainly due to their prominence in the literature and the availability of suitable benchmarks and solvers, these properties do not capture semantic changes or domain-specific noise

models; these would require different distance metrics that take into account the dependencies among input variables and, possibly, novel approaches for reasoning over the resulting properties.

Furthermore, we focused on the automated configuration of MIP-based verification systems. At the same time, verification algorithms have additional hyperparameters, also unrelated to MIP solvers; *e.g.*, configuring $\alpha\beta$-CROWN [111, 119] gives rise to several choices ranging from the selection of a bounding method to the number of branches for non-linear branching. Automatically configuring these algorithms could lead to substantial performance improvements and enhance usability, given the vast hyperparameter space presented by some of these methods.

Another fruitful direction for future work lies in the enhancement of running time prediction models. Enabling running time regression could provide a more nuanced understanding of the verification process, leading to improved resource allocation and process efficiency. Specifically, it would be desirable to predict the running time needed to solve a specific instance beyond a given cutoff point. This would, firstly, require to study the behaviour of verification algorithms when supplied with large time budgets to see if and when hard instances get solved and, potentially, the definition additional features.

In addition, applying the insights and methodologies developed in this thesis to a variety of real-world scenarios holds considerable promise, such as medical diagnosis. Such applications would not only test the practical implications of the research but also uncover new challenges and opportunities for innovation in the field of neural network verification. For example, assessing the robustness of a neural network model used for the classification of ECG measurements requires the definition of robustness properties with respect to specific noise models, such as baseline wander or power-line interference [81]. Furthermore, the scalability of verification methods to neural network models used in practice would be interesting to study.

Finally, the methods and findings presented in this thesis could be jointly utilised in the context of neural architecture search. Given the task of finding a neural network architecture that achieves a high level of robustness, it becomes necessary to perform verification as efficiently as possible, as multiple network architectures or configurations need to evaluated to guide the search process. This would require well-calibrated verifiers as well as efficient resource allocation, and a joint framework that can handle both the neural architecture search process as well as the verification system.

Overall, these future research directions hold the potential to significantly advance the research are of neural network verification, building on the contributions of this

thesis to explore new frontiers in neural network verification and automated machine learning. This can ultimately foster the employment of neural-network-based AI systems in safety-critical tasks, as neural network verification provides a method to formally prove that the system behaves as intended for a given operational domain. In fact, proving the safety (and, specifically the robustness) of an AI System is demanded by the European AI Act [106], underlining the relevance and importance of the concepts and methods introduced in this thesis.