EnnCore: End-to-End Conceptual Guarding of Neural Architectures

Edoardo Manino, [1] Danilo Carvalho, [1] Yi Dong, [2] Julia Rozanova, [1] Xidan Song, [1] Mustafa A. Mustafa, [1,3] Andre Freitas, [1,4] Gavin Brown, [1] Mikel Luján, [1] Xiaowei Huang, [2] Lucas Cordeiro [1]

[1]Department of Computer Science, The University of Manchester, Manchester, M13 9PL, U.K.
[2]Department of Computer Science, University of Liverpool, Liverpool, L69 3BX, U.K.
[3]imec-COSIC, KU Leuven, Leuven-Heverlee, B-3001, Belgium
[4]Idiap Research Institute, Martigny, 1920, Switzerland
{lucas.cordeiro, andres.freitas, mustafa.mustafa}@manchester.ac.uk, {yi.dong, xiaowei}@liverpool.ac.uk

Abstract

EnnCore addresses the fundamental security problem of guaranteeing safety, transparency, and robustness in neural-based architectures. Specifically, *EnnCore* aims at enabling system designers to specify essential conceptual/behavioral properties of neural-based systems, verify them, and thus safeguard the system against unpredictable behavior and attacks. In this respect, *EnnCore* will pioneer the dialogue between contemporary explainable neural models and full-stack neural software verification. This paper describes existing studies' limitations, our research objectives, current achievements, and future trends towards this goal. In particular, we describe the development and evaluation of new methods, algorithms, and tools to achieve fully-verifiable intelligent systems, which are *explainable*, whose correct behavior is guaranteed, and *robust* against attacks. We also describe how *EnnCore* will be validated on two diverse and high-impact application scenarios: securing an AI system for (i) cancer diagnosis and (ii) energy demand response.

1 Introduction

Deep neural networks (DNNs) are computing models typically deployed for classification, decision-making, and pattern recognition problems (Bishop 2006). Recently, various safety-critical tasks deployed DNNs, e.g., Covid-19 diagnosis (Nour, Cömert, and Polat 2020) and steering control in self-driving cars (Wu et al. 2021). In these contexts, however, incorrect classifications can cause severe damages. It is well-known in the literature that adversarial disturbances can make DNNs misclassify objects, thus causing severe damage to users of safety-critical systems. For example, Eykholt et al. (Eykholt et al. 2018) described that noise and disturbances, such as graffiti on traffic signals, could result in target misclassification during operation. Moreover, as DNNs are difficult to interpret and debug, the whole scenario becomes even more problematic (Lundberg and Lee 2017). Hence, there is a need for techniques to assess their structures and verify their results and behavior. Consequently, there is a growing interest in verification and interpretability methods for ensuring and explaining safety, accuracy, and robustness for DNNs.

According to a recent survey on the state-of-the-art of verification and synthesis methods for cyber-physical systems (Cordeiro, de Lima Filho, and Bessa 2020), most papers published in the area in the past ten years only study the verification of safety properties over mathematical representations of DNNs. However, a top-to-bottom verification process of DNNs will need to cover various aspects, including, for example, the external phenomena with which the DNN models interact and evolve. Thus, there is a considerable gap between low- and high-level models and between engineering and theoretical research efforts.

The *EnnCore* project[1], which stands for *"End-to-End Conceptual Guarding of Neural Architectures"*, aims to fill this gap. It has ambitious cross-cutting and far-reaching goals to provide a coherent and self-containing framework for specifying a conceptual safeguard core to neural-based (NB) Artificial Intelligence (AI) systems and verifying their actual implementations considering security aspects. Our setting draws on all of the above aspects: it covers the full range, from engineering details to abstraction and verification, and reasoning and explainability about model evolution and learning.

As a result, *EnnCore* addresses a fundamental research problem to ensure the security of neural-enabled components by taking into account their entire lifecycle from development to deployment. Solving this problem has a far-reaching impact on areas such as *health* and *energy*, which heavily depend on *secure* and *trusted* software components to meet safety-critical requirements. Hence, our overall research objective is to have a long-term impact on writing secure and trusted AI-based software components, thus contributing to a shared vision of *fully-verifiable software,* where underlying neural-based architectures are built with strong symbolic and mathematical guarantees.

To achieve this objective, *EnnCore* will design and validate a full-stack symbolic safeguarding system for NB architectures. It will advance the state-of-the-art in the development of secure DNN models by mapping, using, and extending explainability properties of existing neuro-symbolic DNN architectures (e.g., Graph Networks, Differentiable Inductive Logic Programming), thus safeguarding them with symbolic verification, abstract interpretation, and program

[1] https://enncore.github.io/

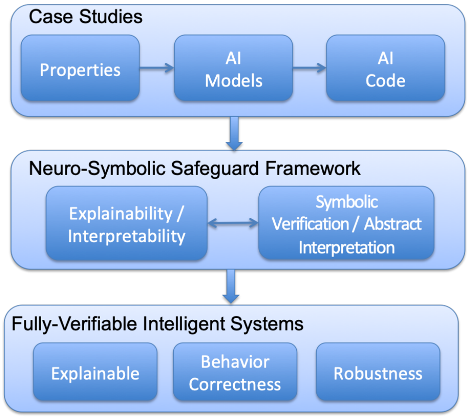
Case Studies
Properties
AI Models
AI Code
Neuro-Symbolic Safeguard Framework
Explainability / Interpretability
Symbolic Verification / Abstract Interpretation
Fully-Verifiable Intelligent Systems
Explainable
Behavior Correctness
Robustness

Figure 1: *EnnCore* methodology.

synthesis methods. *EnnCore* will pioneer the interdisciplinary dialogue between explainable AI and formal verification. In particular, it will deliver safeguarding for NB architectures with the following properties:

1. *Full-stack symbolic software verification*: *EnnCore* will develop the first bit-precise and scalable symbolic verification framework to reason over implementations of DNNs, thereby providing additional guarantees of security properties concerning the underlying hardware. We will exploit state-of-the-art abstract interpretation and synthesis techniques to synthesize invariants to prune the state-space exploration and thus verify intricate security properties to ensure confidentiality, integrity, and availability.
2. *Explainability/Interpretability*: *EnnCore* will pioneer the integration of knowledge-based and neural explainability methods to support end-users specifying security constraints and diagnosing security risks in order to provide security assurances as NB models evolve. Attention will be given to the quantitative and qualitative characterization of semantic-drift phenomena in security scenarios.
3. *Scalable*: *EnnCore* will systematically combine contemporary symbolic methods for explaining, interpreting and verifying neural representations. In particular, we will develop a neuro-symbolic safeguard framework by linking the structural knowledge-based representation elements to the attentional architecture elements to achieve scalability and precision in an unprecedented manner.

EnnCore will systematically validate the system using two different case studies from different domains: healthcare and energy, in order to achieve fully-verifiable intelligent systems that are explainable, ensure behavior correctness and are robust against unanticipated behaviors and attacks.

The remainder of the paper is organized as follows. In Section 2, we discuss the related work, including the limitation of existing studies. Section 3 describes a logical basis for proposing our approach as part of the *EnnCore* project, while Section 4 outlines our research objectives. Section 5 describe our current achievements broken down into four research areas to tackle the safety and security of NB architectures. Finally, we conclude and describe future work in Section 6.

2 Limitations of Existing Work

Explainable/Interpretable ML models. Doshi-Velez and Kim (Doshi-Velez and Kim 2017) define interpretability as “the ability to explain or to present in understandable terms to a human”. Interpretability is an active area of machine learning. The recent Neuro-Symbolic (NS) architectures (Garcez et al. 2019) inherit the strengths of deep learning models, while extending it with explainability and fine-grained/abstractive reasoning capabilities. NS models such as Graph Networks (GNs) (Alshahrani et al. 2017) and Differentiable ILP (Manhaeve et al. 2018) operate over and depend upon knowledge bases and focus on addressing inference problems which require relational reasoning and combinatorial generalization. No prior work has exploited: (i) the use of knowledge-based neuro-symbolic architectures for supporting end-users communicating their security constraints and (ii) the combination of explainability with symbolic verification to assure security properties.

Verification of DNN Models. Verification of DNN models has attracted lots of attention recently, including unique approaches from formal verification (Huang et al. 2017; Katz et al. 2017; Lomuscio and Maganti 2017; Wu et al. 2020), which deals with the problem through exhaustive search, SMT constraint solving, MILP constraint solving, and reduction to two-player game, respectively. A key problem remains on the scalability – the theoretical complexity of the verification problem is NP-complete either on the number of hidden neurons (Katz et al. 2017) or the input dimensions (Ruan, Huang, and Kwiatkowska 2018). This pessimistic result has led to the consideration of approximation methods, such as abstract interpretation (Gehr et al. 2018), interval analysis (Li et al. 2019), and polynomial approximation (Huang et al. 2019). These methods provide soundness guarantees to the result but cannot ensure completeness. Such a relaxation on the guarantees can improve the scale of the network models that the methods can work with but still cannot reach the industrial-scale network models, even when GPU-based Parallelisation is applied (Ruan et al. 2019). Besides, they are often restricted by the types of layers or activation functions they can work with.

The above observation has led to the development of the other thread of works called testing methods, which generate a large number of test cases to intensively test the existence of errors, such as (Wicker, Huang, and Kwiatkowska 2018; Sun et al. 2018). Furthermore, the generation of test cases may often be guided by the coverage metrics such as neuron coverage (Pei et al. 2017) or MC/DC (Sun et al. 2019). While it is arguable whether the generated test cases are representative for the property to be verified, the testing results can be utilized to either understand the internal working mechanism (Huang et al. 2021b) of neural network models or support safety argument (Zhao et al. 2020a) together with the verification results. Please refer to a recent survey (Huang et al. 2020), or tutorial (Ruan, Yi, and Huang

2021) for more discussions on the verification and testing techniques for neural network models.

Verification of Actual Implementations of DNNs. While existing verification methods work with DNN models and adversarial examples (i.e., a small perturbation on a correctly-labeled input leads to a different classification), it has been pointed out in (Odena et al. 2019) that there are errors in the Tensorflow graph representation of DNNs, a lower-level implementation of DNNs, such as numerical errors and disagreements between DNN implementations and their quantized versions. It is reasonable to believe that, when working with code-level implementations, e.g., on the Compute Unified Device Architecture (CUDA) and GPU hardware, there will be other errors, including security loopholes, that are more difficult to detect and mitigate than on CPU implementations (Miele 2016; Di et al. 2020).

Prior work focused on the verification of the robustness of the neural net with respect to its models (Huang et al. 2017; Katz et al. 2017; Sun et al. 2018; Zheng et al. 2016). In these approaches, off-the-shelf Satisfiability Modulo Theories (SMT) solvers are used to find robustness violations. However, this verification scheme cannot precisely capture issues that can be introduced in the implementations of DNNs. There exist four reasons: *(i)* one cannot model bit-level operations using the theory of integers and reals (Cordeiro, Fischer, and Marques-Silva 2011); *(ii)* libraries, such as TensorFlow, often take advantage of available Graphical Processing Units (GPUs) to explore the inherent parallelism of DNNs, so the translation to GPUs can be problematic; *(iii)* some security vulnerabilities cannot be detected in high-level models since they depend on implementation aspects (e.g., finite word-length); lastly *(iv)* there exists no connection between automated verification and explainability approaches, making it difficult to interrogate a system if something goes wrong.

Towards this, Pereira et al. (Pereira et al. 2017) propose to verify CUDA programs written for GPU platforms with an SMT-based context-bounded checking technique. They developed ESBMC-GPU, which is the first verifier to discover adversarial cases and validate coverage methods in DNNs using the cuBLAS and cuDNN libraries (Sena et al. 2019). However, Pereira et al. (Pereira et al. 2017) do not exploit invariant inference to prune the state-space exploration for greater scalability. Also, their approach cannot explain the parameters of the DNN implementation to understand the root cause of errors.

3 Rationale and Approach

We believe that a holistic approach is necessary to overcome the challenges and limitations listed in Section 2. To this end, *EnnCore* will pioneer the dialogue between all the very different components of the contemporary AI safety stack (see Figure 2).

On the one hand, we will draw inspiration and support from the diverse industrial experiences of our partners. For healthcare, digital Experimental Cancer Medicine Team (dECMT)[2] requires a provably correct, trusted, explainable decision making for medical diagnosis. For energy, Urbanchain[3] requires a fair, explainable, and trusted decision making system to ensure the security and privacy of clients' data.

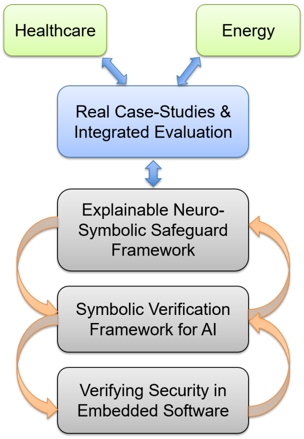
Healthcare
Energy
Real Case-Studies & Integrated Evaluation
Explainable Neuro-Symbolic Safeguard Framework
Symbolic Verification Framework for AI
Verifying Security in Embedded Software

Figure 2: *EnnCore* holistic approach.

On the other hand, *EnnCore* will bridge the gap between the user's need to communicate their security constraints, and the technical challenges involved in formalising these constraints and checking whether neural-based system satisfy them. In this respect, we consider explainability/interpretability techniques as a fertile common ground for translating the user's requirements to rigorous mathematical constraints. Furthermore, we believe that exploiting the structure of such constraints, and the neural-based architecture that is required to satisfy them, is the key towards a truly scalable full-stack verification approach.

4 Research Objectives

EnnCore aims to fundamentally shift the state-of-the-art of what is achievable in formal verification of AI-based software systems to make them secure and trusted against unanticipated behavior and attacks. We are convinced that this cannot be achieved by a "proof-of-concept implementation" with an artificial case study. This particular approach will not have much credibility – and thus impact – with systems and software engineers. We will work in close collaboration with industrial partners to tackle real-world case studies in healthcare and energy domains. We will also use real data and work with domain experts to develop and validate our algorithms, methods and tools. In a multidisciplinary fashion, *EnnCore* will link two areas, which include neuro-symbolic and explainable machine learning and software verification, to deliver a full-stack security mechanism for DNNs operating in safety-critical scenarios. Our core objectives are:

O1: Develop a novel conceptual/symbolic safeguard mechanism for neuro-symbolic platforms

[2] dECMT is a clinical digital research group based in the Cancer Research UK Manchester Institute (https://digitalecmt.org/).

[3] Urbanchain develops a world-leading platform for energy generators in the wholesale market (https://www.urbanchain.co.uk/).

EnnCore will pioneer the use of neuro-symbolic architectures and explainability/interpretability mechanisms to support end-users specifying a conceptual safeguard core to neural-based AI systems. The project will also contribute to a broad and in-depth systematic analysis of the impact of existing explainability/interpretability mechanisms in security scenarios. These mechanisms include the interpretation of high-dimensional embeddings, attentional mechanisms, decoding from intermediate representations and black-box debugging methods using artificially generated datasets.

O2: Develop scalable SMT theories and invariant inference methods for DNNs

EnnCore will develop new SMT theories to reason about the safety and security of actual implementations of DNNs. Our ultimate goal is to mitigate security vulnerabilities and incorrect predictions, which make AI-based applications susceptible to errors and mischance. Additionally, *EnnCore* will develop a new invariant inference method based on the structure of the DNNs. We aim to simplify the DNN output computation for some input intervals using abstract interpretation and program synthesis. In particular, we will exploit invariant inference to prune the state-space exploration for verifying security properties in real implementations of DNNs.

O3: Grounding, deploying and evaluating high-impact real-world use cases

EnnCore will be co-designed with industrial and clinical partners around exemplary use-case scenarios. The selected use cases reflect standard security requirements for DNNs, which are transferable to other sectors such as automotive and consumer electronics. Additionally, usability is at the center of the unique value proposition of *EnnCore*, where the model can interface with end-users (system designers and security experts). We will allow users to state areas within the model that should be safeguarded.

5 Current Achievements and Future Trends

The proposed research is broken down into four research areas: *Real Case-Studies & Integrated Evaluation*, *Explainable Neuro-Symbolic Safeguard Framework*, *Symbolic Verification Framework for AI*, and *Verifying Security in Embedded Software running in GPUs*. In the following, we describe the research contents of each area. In particular, we provide details of what has been achieved to date and what we intend to tackle as future work.

5.1 Real Case-Studies & Integrated Evaluation

EnnCore aims to tackle two real-world use cases in two distinct domains: health and energy. In the health domain, the use case is *cancer diagnoses* (Lee et al. 2021), where a medical institution (e.g., hospital) aims to determine if suspect patients have cancer or not based on analyzing a set of biomarkers. To achieve this, the medical institution deploys an AI model that uses the patients' biomarkers to predict the likelihood of a patient having (or developing) cancer. In the energy domain, the use case is *demand response* (Albadi and El-Saadany 2008), where an energy supplier company aims to match their customers' energy consumption with the energy supply available, in order to facilitate peer-to-peer energy trading (Capper et al. 2021) without violations of the grid constraints (Dudjak et al. 2021). To complete this efficiently and effectively, the supplier needs to predict the half-hourly electricity consumption of each of their customers. To achieve this, the energy supplier deploys an AI model that uses their customers' historical consumption data to predict their consumption data for the next half-hourly time slot. Unfortunately, this approach allows the supplier to have access to households' fine-grained consumption data, which poses a high risk to users' privacy (Mustafa, Cleemput, and Abidin 2016) as well as hinders the adoption of smart meters (Briggs, Fan, and Andras 2020).

Up to now, we have performed security analyses of both use cases to identify potential threats to the AI models, hence specifying concrete security requirements/properties that these AI models should satisfy. Apart from the 'standard' confidentiality, integrity, and availability requirements, we have identified the following properties relevant to AI models: robustness, transparency, auditability (traceability), accountability, and privacy.

Robustness ensures that AI models are resilient against malicious input and corner cases (a.k.a adversarial examples). *Transparency* ensures that all phases of an AI model processing chain (including the technical details of the models and the training data used) are well documented. *Auditability* (traceability) ensures that all the processing steps of the AI models (i.e., cause-effect) can be traced by third parties if needed. Finally, *accountability* ensures evidence of who has developed/managed/maintained every component/step of the AI model. These four properties are closely related to each other and contribute to the explainability of AI models. On the other hand, privacy ensures that sensitive user data and sensitive AI models are protected from unauthorized entities, sometimes even from the companies that have developed and managed the AI models.

To ensure AI models' robustness against malfunction and attacks, one promising approach is to adopt robust training for AI models (Gehr et al. 2018). This ensures that the AI models are already fed with data representing potential corner cases in the training phases. To achieve this, the training data is usually augmented by adding a certain degree of randomness. The DiffAI framework (Mirman, Gehr, and Vechev 2018) has successfully applied this approach to develop AI models that are provably robust. This is achieved by deploying abstract interpretation techniques by overapproximating the AI system's behavior. However, the DiffAI framework has been designed to process images. As a next step, we plan to adapt the DiffAI framework to process other types of input data, e.g., biomarkers.

To ensure that AI systems protect user-sensitive data, deploying Federated Learning (FL) (Bonawitz et al. 2019) is a promising approach. FL, by design, allows end-users to train their models locally, never share their sensitive raw data, yet benefit from the data of others. This is achieved by sharing only the gradients of the locally trained and deployed AI models, which are then aggregated to build a global model, distributed to the end-users. Although it already provides a good level of user privacy protection, this approach has some limitations. For example, the gradients of the locally trained AI models can reveal information about the model itself and/or the data used for the local training of the models (Melis et al. 2019; De Cristofaro 2021). In addition, a single global AI model does not always provide the best possible outcome for all the end-users. To address these limitations, we plan to deploy advanced cryptographic techniques for secure computation (homomorphic encryption and multiparty computation) to perform the gradient aggregation and devise the global model in a secure way such that no entity has access to the gradients provided by the individual end-users. In addition, to improve performance, we plan to adopt a clustering method (Sattler, Müller, and Samek 2021), which would classify the end-users based on their data into several clusters, creating variants of the global model, which will contribute differently to the final model used by each of the individual end-users. Our approach will be tested on the energy use case to predict individual users' household electricity consumption data.

5.2 Safeguards for Explainable Neuro-Symbolic Inference

EnnCore sets the vision of delivering neural representation models that are highly controlled regarding their inference properties. The key concept is to allow model developers and domain experts to encode complex symbolic and geometric constraints within the models (safeguards), allowing for more controlled inference and better disentanglement. Additionally, the project expands emerging probing and metamorphic testing methodologies to measure and qualify the internal properties of the latent representation. For this work-stream, we focus on designing controlled embeddings for complex tasks in Natural Language Processing (NLP), emphasizing textual entailment and question answering. These tasks allow for the design of models which require the encoding of complex (i.e., requiring multiple semantic operations) and multi-hop natural language inference in an explainable manner.

The *inference control methods* are represented as explicit linguistic and inference constraints, which elicit abductive inference biases that are integrated into the latent model. The intuition is that universal patterns of abstract inference, such as abstraction and fact unification (Valentino, Thayaparan, and Freitas 2021; Valentino, Pratt-Hartman, and Freitas 2021) can be programmed into the model, prescribing an expected inference pattern, which can facilitate generalization but also enforce more consistent inferences. Following the results achieved by encoding these constraints using Integer Linear Programming (ILP) (Thayaparan, Valentino, and Freitas 2020), which demonstrate its positive impact on inference control and explainability, we proposed ∂-Explainer (Thayaparan et al. 2021), an end-to-end differentiable architecture that integrates Convex Optimization such as Linear Programming with neural representations for abductive natural language inference. Specifically, we demonstrated that these models could integrate explicit inference constraints with Transformers-based sentence representations and train the architecture end-to-end to improve explanation generation and accuracy in multi-hop and abstractive reasoning tasks.

Part of the inference control mechanisms is expressed in the design of generative models for natural language inference with *better disentanglement* of latent factors. While representing the meaning of a sentence or an inference step in a continuous latent sentence space, models will aim for specializing latent dimensions to capture consistent linguistic and inference phenomena (e.g., tense variations for verbs), allowing for both interpretability and control. In (Mercatali and Freitas 2021), we proposed a variational autoencoder (VAE) model which better disentangles sentence discrete generative language factors. Recent work is expanding the same level of linguistic control via disentanglement for abstract sentences and multi-hop inference.

The level of additional control needs to be accompanied by methodologies that can measure and qualify the internal properties of these embedding spaces. For example, probing or diagnostic classification (Hewitt and Liang 2019; Ferreira et al. 2021) is a method for investigating whether a set of intermediate (e.g., semantic) features are present in latent spaces. In *EnnCore*, we extend emerging methodologies such as metamorphic testing, geometric probing, and abstract inference to systematize the internal properties and consistency of controlled embedding spaces. Examples include the verification of abstract properties highly relevant to controlled inference such as monotonicity (Rozanova et al. 2021a,b) or variable substitution (Ferreira et al. 2021).

We also mention BayLIME (Zhao et al. 2020b), which is a novel explainable AI tool enhancing the well-known LIME tool with Bayesian reasoning to achieve better consistency in repeated explanations of a single prediction and better robustness to the hyper-parameters.

5.3 Symbolic Verification Framework for AI

Verification refers to algorithms that determine whether or not a model satisfies some pre-specified property. Symbolic verification algorithms compute the intermediate results using a symbolic representation – such as BDD, SAT, and SMT. Usually, symbolic verification scales better than explicit verification, thanks to its memory efficiency and efficient computation. In the first year of *EnnCore*, we have explored a few directions on the symbolic verification techniques for AI, including working directly with the machine learning models. The other two directions aim to deal with the scalability issues through abstract models and acceptable solutions safety, respectively.

We considered two classes of machine learning models when working directly with the models. For convolutional neural networks (CNNs), a symbolic verification algorithm based on interval analysis and symbolic layer-by-layer propagation was developed in (Yang et al. 2021; Li et al. 2020), together with a global optimisation based method (Xu, Ruan, and Huang 2021). Second, for the random forest, an SMT-based method was considered to determine whether a model has been data poisoned by a backdoor attack (Huang, Zhao, and Huang 2020).

We also develop methods when scalability is an obstacle to the verification algorithms. For example, for deep reinforcement learning, we abstract its interactive behavior with the environment into a discrete-time Markov chain and then apply an off-the-shelf probabilistic model checker to do verification (Dong, Zhao, and Huang 2021). For CNNs, we abstracted a model into a Bayesian network and then conducted probabilistic inference as the verification algorithm (Berthier et al. 2021a,b).

We also deal with scalability from the perspective of acceptable safety. In (Huang et al. 2021a), we developed a statistical certification algorithm for the robustness of CNNs, and in (Zhao et al. 2021a,b), we considered an acceptable level of reliability of CNNs. Moreover, coverage-guided testing is proven an effective way to quantify the quality of a recurrent neural network (Huang et al. 2021b).

In addition to the above directions, we also developed our views in (Ruan, Yi, and Huang 2021; Huang 2021), which includes potential directions for exploration.

5.4 Verifying Security in Embedded Software running in GPUs

We have developed and evaluated various verification strategies to detect errors in learning and classifications performed by DNNs. In particular, we analyzed potential failures of DNNs due to bugs in the implementation of the embedded software of the DNNs. Here, we distinguish two classes of bugs: 1) generic implementation errors, for instance, memory safety, arithmetic overflow, and division-by-zero; they can cause the implementation of the DNN to crash; 2) failure of the implementation to behave according to the high-level rules, which may cause miss-classifications.

In (Sena et al. 2019, 2021), we develop and evaluate a novel symbolic verification framework using software model checking (SMC) and satisfiability modulo theories (SMT) to check for safety properties in quantized neural networks (QNNs). More specifically, we propose several QNN-related optimizations for SMC, including invariant inference via interval analysis, slicing, expression simplifications, and discretization of non-linear activation functions. We also quantified each technique's impact using different SMT solvers. We observed a significant performance improvement if we enabled slicing, interval analysis, and expression simplifications with the SMT solver Yices (Sena et al. 2021).

With this verification framework, we also provide formal guarantees on the safe behavior of QNNs implemented both in floating- and fixed-point arithmetic. In particular, we have observed that the verification time correlates with the number of bits used for ANN quantization. Interestingly, this correlation disappears for the number of bits above 14 due to the increasing state-space exploration. In this regard, our verification approach was able to verify and produce adversarial examples for 52 test cases spanning image classification and general machine learning applications. Furthermore, for small- to medium-sized QNNs, our approach completes most of its verification runs in minutes. In contrast to most state-of-the-art methods, our approach is not restricted to specific choices regarding activation functions and non-quantized representations. Finally, our experiments show that our approach can analyze larger ANN implementations and substantially reduce the verification time compared to state-of-the-art techniques that use SMT solving, e.g., Marabou (Kim et al. 2016). It is also competitive to verification approaches that employ symbolic interval, e.g., Neurify (Wang et al. 2018).

As future work, we plan to work in two directions. First, we aim to evaluate security properties in various real case studies. Second, we will lead extensive experiments to validate the implementations of DNNs for a set of case studies from our industrial partners. This procedure requires us to set an environment for running the implementations of DNNs in typical GPUs, which will rely on our prior work on the verification of GPU programs (Pereira et al. 2017). Additionally, creating the benchmarks for the experiments is a continuous and iterative task, consisting of two main steps: (i) creating benchmarks using real applications of DNNs; and (ii) using industry-standard benchmarks in close collaboration with our partners. Lastly, we will interpret and validate the results obtained during these experiments and then compare our approach using other state-of-the-art verification tools, similar to our recent work (Sena et al. 2021).

6 Conclusions

EnnCore contributes to the development of trustworthy neural-based systems, which are highly applicable to areas of high societal impact, such as reliable infrastructure management, defense, medical diagnosis and treatment, and fair/unbiased decision making. In particular, *EnnCore* emphasizes safety for medical diagnosis and treatment, with a use case targeting cancer. Personalized medicine requires the increasing use of automated data-driven methods. The *EnnCore* project can directly impact the reduction of the barriers to adopting AI-based methods in clinical settings, thereby democratizing personalized cancer diagnosis and treatment. Additionally, one of our industrial partners is currently acting as a blockchain-based supplier in the energy market. *EnnCore* tools will ensure the privacy and security of the clients' data in the energy sector. In particular, *EnnCore* will assist this industrial partner by providing innovative methods to protect their customers' data and applied algorithms. As a result, we expect the *EnnCore* tools to push the state-of-the-art on formal verification and explainability techniques to provide assurances about AI applications' security and explain their security properties.

Acknowledgment

The work is funded by EPSRC grant EP/T026995/1 entitled "EnnCore: End-to-End Conceptual Guarding of Neural Architectures" under *Security for all in an AI enabled society*.

References

Albadi, M. H.; and El-Saadany, E. F. 2008. A summary of demand response in electricity markets. *Electric power systems research*, 78(11): 1989–1996.

Alshahrani, M.; Khan, M. A.; Maddouri, O.; Kinjo, A. R.; Queralt-Rosinach, N.; and Hoehndorf, R. 2017. Neurosymbolic representation learning on biological knowledge graphs. *Bioinformatics*, 33(17): 2723–2730.

Berthier, N.; Alshareef, A.; Sharp, J.; Schewe, S.; and Huang, X. 2021a. Abstraction and Symbolic Execution of Deep Neural Networks with Bayesian Approximation of Hidden Features. *CoRR*, abs/2103.03704.

Berthier, N.; Sun, Y.; Huang, W.; Zhang, Y.; Ruan, W.; and Huang, X. 2021b. Tutorials on Testing Neural Networks. *CoRR*, abs/2108.01734.

Bishop, C. M. 2006. Pattern recognition. *Machine learning*, 128(9).

Bonawitz, K.; Eichner, H.; Grieskamp, W.; Huba, D.; Ingerman, A.; Ivanov, V.; Kiddon, C.; Konečnỳ, J.; Mazzocchi, S.; McMahan, H. B.; et al. 2019. Towards federated learning at scale: System design. *CoRR*, abs/1902.01046.

Briggs, C.; Fan, Z.; and Andras, P. 2020. Privacy Preserving Demand Forecasting to Encourage Consumer Acceptance of Smart Energy Meters. *CoRR*, abs/2012.07449.

Capper, T.; Gorbatcheva, A.; Mustafa, M. A.; Bahloul, M.; Schwidtal, J. M.; et al. 2021. A Systematic Literature Review of Peer-to-Peer, Community Self-Consumption, and Transactive Energy Market Models. *Available at SSRN: https://ssrn.com/abstract=3959620*.

Cordeiro, L.; Fischer, B.; and Marques-Silva, J. 2011. SMT-based bounded model checking for embedded ANSI-C software. *IEEE TSE*, 38(4): 957–974.

Cordeiro, L. C.; de Lima Filho, E. B.; and Bessa, I. V. 2020. Survey on automated symbolic verification and its application for synthesising cyber-physical systems. *IET Cyber-Physical Systems: Theory and Applications*, 5(1): 1–24.

De Cristofaro, E. 2021. A critical overview of privacy in machine learning. *IEEE S&P*, 19(4): 19–27.

Di, B.; Sun, J.; Chen, H.; and Li, D. 2020. Efficient Buffer Overflow Detection on GPU. *IEEE TPDS*, 32(5): 1161–1177.

Dong, Y.; Zhao, X.; and Huang, X. 2021. Dependability Analysis of Deep Reinforcement Learning based Robotics and Autonomous Systems. *CoRR*, abs/2109.06523.

Doshi-Velez, F.; and Kim, B. 2017. Towards a rigorous science of interpretable machine learning. *CoRR*, abs/1702.08608.

Dudjak, V.; Neves, D.; Alskaif, T.; Khadem, S.; Pena-Bello, A.; Saggese, P.; Bowler, B.; Andoni, M.; Bertolini, M.; Zhou, Y.; et al. 2021. Impact of local energy markets integration in power systems layer: A comprehensive review. *Applied Energy*, 301: 117434.

Eykholt, K.; Evtimov, I.; Fernandes, E.; Li, B.; Rahmati, A.; Xiao, C.; Prakash, A.; Kohno, T.; and Song, D. 2018. Robust physical-world attacks on deep learning visual classification. In *CVPR*, 1625–1634.

Ferreira, D.; Rozanova, J.; Thayaparan, M.; Valentino, M.; and Freitas, A. 2021. Does My Representation Capture X? Probe-Ably. In *ACL-IJCNLP*, 194–201.

Garcez, A. d.; Gori, M.; Lamb, L. C.; Serafini, L.; Spranger, M.; and Tran, S. N. 2019. Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning. *CoRR*, abs/1905.06088.

Gehr, T.; Mirman, M.; Drachsler-Cohen, D.; Tsankov, P.; Chaudhuri, S.; and Vechev, M. 2018. AI2: Safety and robustness certification of neural networks with abstract interpretation. In *IEEE S&P*, 3–18.

Hewitt, J.; and Liang, P. 2019. Designing and Interpreting Probes with Control Tasks. In *EMNLP*, 2733–2743.

Huang, C.; Fan, J.; Li, W.; Chen, X.; and Zhu, Q. 2019. ReachNN: Reachability Analysis of Neural-Network Controlled Systems. *ACM TECS*, 18(5s).

Huang, C.; Hu, Z.; Huang, X.; and Pei, K. 2021a. Statistical Certification of Acceptable Robustness for Neural Networks. In *ICANN*, 79–90.

Huang, W.; Sun, Y.; Zhao, X.; Sharp, J.; Ruan, W.; Meng, J.; and Huang, X. 2021b. Coverage-Guided Testing for Recurrent Neural Networks. *IEEE Tran. on Reliability*, 1–16.

Huang, W.; Zhao, X.; and Huang, X. 2020. Embedding and Extraction of Knowledge in Tree Ensemble Classifiers. *CoRR*, abs/2010.08281.

Huang, X. 2021. Safety and reliability of deep learning: (brief overview). In *VARS*, 1–2.

Huang, X.; Kroening, D.; Ruan, W.; Sharp, J.; Sun, Y.; Thamo, E.; Wu, M.; and Yi, X. 2020. A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review*, 37: 100270.

Huang, X.; Kwiatkowska, M.; Wang, S.; and Wu, M. 2017. Safety verification of deep neural networks. In *CAV*, 3–29.

Katz, G.; Barrett, C.; Dill, D. L.; Julian, K.; and Kochenderfer, M. J. 2017. Reluplex: An efficient SMT solver for verifying deep neural networks. In *CAV*, 97–117.

Kim, Y.; Park, E.; Yoo, S.; Choi, T.; Yang, L.; and Shin, D. 2016. Compression of Deep Convolutional Neural Networks for Fast and Low Power Mobile Applications. In *ICLR*.

Lee, R.; Wysocki, O.; Zhou, C.; Calles, A.; Eastlake, L.; Ganatra, S.; Harrison, M.; et al. 2021. CORONET; COVID-19 in Oncology evaluatiON Tool: Use of machine learning to inform management of COVID-19 in patients with cancer.

Li, J.; Liu, J.; Yang, P.; Chen, L.; Huang, X.; and Zhang, L. 2019. Analyzing Deep Neural Networks with Symbolic Propagation: Towards Higher Precision and Faster Verification. In *Static Analysis*, 296–319.

Li, R.; Li, J.; Huang, C.-C.; Yang, P.; Huang, X.; Zhang, L.; Xue, B.; and Hermanns, H. 2020. Prodeep: a platform for robustness verification of deep neural networks. In *ESEC/FSE*, 1630–1634.

Lomuscio, A.; and Maganti, L. 2017. An approach to reachability analysis for feed-forward ReLU neural networks. *CoRR*, abs/1706.07351.

Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. In *NIPS*, 4768–4777.

Manhaeve, R.; Dumančić, S.; Kimmig, A.; Demeester, T.; and De Raedt, L. 2018. Deepproblog: Neural probabilistic logic programming. *CoRR*, abs/1805.10872.

Melis, L.; Song, C.; De Cristofaro, E.; and Shmatikov, V. 2019. Exploiting unintended feature leakage in collaborative learning. In *IEEE S&P*, 691–706.

Mercatali, G.; and Freitas, A. 2021. Disentangling Generative Factors in Natural Language with Discrete Variational Autoencoders. *CoRR*, abs/2109.07169.

Miele, A. 2016. Buffer overflow vulnerabilities in CUDA: a preliminary analysis. *Journal of Computer Virology and Hacking Techniques*, 12(2): 113–120.

Mirman, M.; Gehr, T.; and Vechev, M. 2018. Differentiable abstract interpretation for provably robust neural networks. In *ICML*, 3578–3586.

Mustafa, M. A.; Cleemput, S.; and Abidin, A. 2016. A local electricity trading market: Security analysis. In *ISGT Europe*, 1–6.

Nour, M.; Cömert, Z.; and Polat, K. 2020. A novel medical diagnosis model for COVID-19 infection detection based on deep features and Bayesian optimization. *Applied Soft Computing*, 97: 106580.

Odena, A.; Olsson, C.; Andersen, D.; and Goodfellow, I. 2019. Tensorfuzz: Debugging neural networks with coverage-guided fuzzing. In *ICML*, 4901–4911.

Pei, K.; Cao, Y.; Yang, J.; and Jana, S. 2017. DeepXplore: Automated whitebox testing of deep learning systems. In *SOSP*, 1–18. ACM.

Pereira, P.; Albuquerque, H.; da Silva, I.; Marques, H.; Monteiro, F.; Ferreira, R.; and Cordeiro, L. 2017. SMT-based context-bounded model checking for CUDA programs. *Concurrency and Computation: Practice and Experience*, 29(22): e3934.

Rozanova, J.; Ferreira, D.; Thayaparan, M.; ; Valentino, M.; and Freitas, A. 2021a. Probing Context Monotonicity and Relational Knowledge in Neural Natural Language Inference. *CoRR*.

Rozanova, J.; Ferreira, D.; Thayaparan, M.; ; Valentino, M.; and Freitas, A. 2021b. Supporting Context Monotonicity Abstractions in Neural NLI Models. *CoRR*, abs/2105.08008.

Ruan, W.; Huang, X.; and Kwiatkowska, M. 2018. Reachability Analysis of Deep Neural Networks with Provable Guarantees. In *IJCAI*, 2651–2659.

Ruan, W.; Wu, M.; Sun, Y.; Huang, X.; Kroening, D.; and Kwiatkowska, M. 2019. Global Robustness Evaluation of Deep Neural Networks with Provable Guarantees for the Hamming Distance. In *IJCAI*, 5944–5952.

Ruan, W.; Yi, X.; and Huang, X. 2021. Adversarial Robustness of Deep Learning: Theory, Algorithms, and Applications. In *ACM CIKM*, 4866–4869.

Sattler, F.; Müller, K.-R.; and Samek, W. 2021. Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization Under Privacy Constraints. *IEEE TNNLS*, 32(8): 3710–3722.

Sena, L.; Song, X.; Alves, E.; Bessa, I.; Manino, E.; and Cordeiro, L. 2021. Verifying Quantized Neural Networks using SMT-Based Model Checking. *CoRR*, abs/2106.05997.

Sena, L. H.; Bessa, I. V.; Gadelha, M. R.; Cordeiro, L. C.; and Mota, E. 2019. Incremental Bounded Model Checking of Artificial Neural Networks in CUDA. In *SBESC*, 1–8.

Sun, Y.; Huang, X.; Kroening, D.; Sharp, J.; Hill, M.; and Ashmore, R. 2019. Structural Test Coverage Criteria for Deep Neural Networks. *ACM TECS*, 18(5s): 1–23.

Sun, Y.; Wu, M.; Ruan, W.; Huang, X.; Kwiatkowska, M.; and Kroening, D. 2018. Concolic testing for deep neural networks. In *ASE*, 109–119.

Thayaparan, M.; Valentino, M.; Ferreira, D.; Rozanova, J.; and Freitas, A. 2021. ∂ -Explainer: Abductive Natural Language Inference via Differentiable Convex Optimization. *CoRR*, abs/2105.03417.

Thayaparan, M.; Valentino, M.; and Freitas, A. 2020. ExplanationLP: Abductive Reasoning for Explainable Science Question Answering. *CoRR*, abs/2010.13128.

Valentino, M.; Pratt-Hartman, I.; and Freitas, A. 2021. Do Natural Language Explanations Represent Valid Logical Arguments? Verifying Entailment in Explainable NLI Gold Standards. *CoRR*, abs/2105.01974.

Valentino, M.; Thayaparan, M.; and Freitas, A. 2021. Unification-based Reconstruction of Multi-hop Explanations for Science Questions. In *EACL*, 200–211.

Wang, S.; Pei, K.; Whitehouse, J.; Yang, J.; and Jana, S. 2018. Efficient Formal Safety Analysis of Neural Networks. In *NeurIPS*, 6369–6379.

Wicker, M.; Huang, X.; and Kwiatkowska, M. 2018. Feature-guided black-box safety testing of deep neural networks. In *TACAS*, 408–426.

Wu, H.; Lv, D.; Cui, T.; Hou, G.; Watanabe, M.; and Kong, W. 2021. SDLV: Verification of Steering Angle Safety for Self-Driving Cars. *Formal Aspects of Computing*, 33(3): 325–341.

Wu, M.; Wicker, M.; Ruan, W.; Huang, X.; and Kwiatkowska, M. 2020. A game-based approximate verification of deep neural networks with provable guarantees. *Theoretical Computer Science*, 807: 298–329.

Xu, P.; Ruan, W.; and Huang, X. 2021. Towards the Quantification of Safety Risks in Deep Neural Networks. *Complex and Intelligent Systems*.

Yang, P.; Li, J.; Liu, J.; Huang, C.-C.; Li, R.; Chen, L.; Huang, X.; and Zhang, L. 2021. Enhancing robustness verification for deep neural networks via symbolic propagation. *Formal Aspects of Computing*, 1–29.

Zhao, X.; Banks, A.; Sharp, J.; Robu, V.; Flynn, D.; Fisher, M.; and Huang, X. 2020a. A Safety Framework for Critical Systems Utilising Deep Neural Networks. In *SafeComp2020*, 244–259.

Zhao, X.; Huang, W.; Banks, A.; Cox, V.; Flynn, D.; Schewe, S.; and Huang, X. 2021a. Assessing the Reliability of Deep Learning Classifiers Through Robustness Evaluation and Operational Profiles. *CoRR*, abs/2106.01258.

Zhao, X.; Huang, W.; Huang, X.; Robu, V.; and Flynn, D. 2020b. Baylime: Bayesian local interpretable model-agnostic explanations. *CoRR*, abs/2012.03058.

Zhao, X.; Huang, W.; Schewe, S.; Dong, Y.; and Huang, X. 2021b. Detecting Operational Adversarial Examples for Reliable Deep Learning. *CoRR*, abs/2104.06015.

Zheng, S.; Song, Y.; Leung, T.; and Goodfellow, I. 2016. Improving the robustness of deep neural networks via stability training. In *CVPR*, 4480–4488.