論　文　の　要　旨

題　目 Research on Enhancing the Reliability of Neural Network-Based Systems using Testing and Verification
（テストと検証を用いたニューラルネットワークベースシステムの信頼性向上に関する研究）

氏　名　　LIU HAIYI

This dissertation discusses how to use testing and verification methods to enhance the reliability of systems incorporating neural networks. Specifically, this dissertation aims to investigate: 1) How to employ Testing-based methods to identify potential errors that may arise during the training process of neural networks. 2) How to combine testing and verification methods to improve the reliability of trained neural network models. 3) How to use a combination of testing and verification to explore the interpretability of trained neural networks. Particularly, three approaches are proposed to answer the above three questions. We will introduce them successively.

A testing-based method to assess the GPU-memory consumption. During the training process of neural network models, a large amount of GPU computing resources is required, but it is difficult for developers to accurately calculate the GPU resources that the model may consume before running, which brings great inconvenience to the development of neural network-based systems. This is particularly important especially in today's cloud-based model training.　Therefore, it is very important to estimate the GPU memory resources that the neural network model may use in a certain computing framework. Existing work has focused on static analysis methods to assess GPU memory consumption, highly coupled with the framework, and lack of research on low-coupled GPU memory consumption of the framework. In this article, we propose the Testing-Based Estimation Method (TBEM), which is a Testing-based method for estimating the memory usage of the neural network model. First, TBEM generates enough neural network models using an orthogonal array testing strategy and a classical neural network design pattern. Then, TBEM generates neural network model tested in a real environment to obtain the real-time GPU memory usage values corresponding to the model. After obtaining the data of different models and corresponding GPU usage values, the data is analyzed by regression.

A method utilizing Testing-Based Formal Verification for simplifying and verifying neural networks. Although the security of neural networks can be enhanced by verification, verifying neural networks is an NP-hard problem, making the application of verification algorithms to large scale neural networks a challenging task. For this reason, we propose NNTBFV, a framework that utilizes the principles of Testing-Based Formal Verification (TBFV) to simplify neural networks and verify the simplified networks. Unlike conventional neural network pruning techniques, this approach is based on specifications, with the goal of deriving approximate execution paths under given preconditions. To mitigate the potential issue of unverifiable conditions due to overly broad preconditions, we also propose a precondition partition method. Empirical evidence shows that as the range of preconditions narrows, the size of the execution paths also reduces accordingly. The execution path generated by NNTBFV is still a neural network, so it can be verified by verification tools. In response to the results from the verification tool, we provide a theoretical method for analysis.

We evaluate the effectiveness of NNTBFV on the ACAS Xu model project, choosing Verification-based and Random-based neural network simplification algorithms as the baselines for NNTBFV. Experiment results show

that NNTBFV can effectively approximate the baseline in terms of simplification capability, and it surpasses the efficiency of the Random-based method.

An approach to provide localized interpretation of neural networks using the principle of Testing-Based Formal Verification. Although neural networks have been widely used in many fields such as NLP (natural language processing), image processing and even MMML (Multi-modal Machine Learning), their weak interpretability and poor reliability have been criticized by many users for a long time. Specifically, there are two aspects. One is that neural networks are difficult to be verified. The reason is that the architecture of neural networks is based on experience, and the parameter are constructed by back-propagation of the training data. It is difficult to give a formal specification like traditional software, and the verification of neural networks is an NP-hard problem, which makes it difficult to achieve complete verification of the large models. The other is that neural networks are difficult to be explained, that is, it is difficult for us to figure out what features the result of neural network reasoning is based on. For instance, although a neural network correctly recognizes the cat in the picture, we cannot determine whether the neural network correctly recognizes the cat through its features or the watermark in the picture.

These three aspects of research all revolve around the theme of enhancing the reliability of neural network-based systems. They focus on improving the reliability of neural networks during the training process, the reliability of trained neural networks, and their interpretability, respectively. Additionally, applying the theory of TBFV to neural networks also provides theoretical support for generating formal local interpretations of neural networks in the future.