An Industrial Software Model Checking Method Based on Machine Learning and Its Application in Education

Ting Zhang[1,2] and Yong Wang[1,2(✉)]

[1] School of Artificial Intelligence, Wuhan Technology and Business University, Wuhan 430065, Hubei, China
zhangting@wtbu.edu.cn, 506532097@qq.com

[2] Institute of Information and Intelligent Engineering Applications, Wuhan Technology and Business University, Wuhan 430065, Hubei, China

Abstract. In the era of big data, the traditional industrial software model detection methods can not meet the actual needs in the massive data. This paper studies an industrial software system, which is widely used in many industrial fields. A comprehensive analysis of its behavior and correctness has practical needs and significance. At present, there are symbolic, combinatorial and statistical model detection methods to verify the correctness of the system interaction and random behavior. Because there are many variables in the detection system, it is easy to cause the state space explosion of verification; When using abstract technology, verification is incomplete. To solve the above problems, we focus on the processing of variables in the detection system.

Keywords: Machine learning · Software verification · Model detection

1 Introduction

With the continuous expansion of software scale, using traditional methods to eliminate errors in software system has become more and more difficult to meet the actual needs. We must pay attention to new software verification theories and methods. Among the many theories and methods proposed for this purpose, model detection is remarkable for its simplicity and high degree of automation. With great success in the field of hardware and protocol verification, model checking is gradually applied to software verification.

Detection system design is widely used in many industrial fields and is suitable for complex system development. The whole system is regarded as composed of a group of detection (modules) connected. Detection design has many advantages: detection reusability, designed detection can be used by other detection, reducing repeated development; Detection flexibility, modification, addition and deletion of detection will not affect other detection of the system; Detection abstraction. Detection only provides interfaces for interaction with other detection, and does not need to provide implementation details. In addition, the detection system has a clear structure and is easy to verify. The

M. A. Jan and F. Khan (Eds.): BigIoT-EDU 2022, LNICST 465, pp. 53–61, 2023.
https://doi.org/10.1007/978-3-031-23950-2_7

system has a short cycle from design and development to putting on the market. Detection system design has many advantages. However, designing a correct detection system that meets the requirements is full of challenges. Joseph Sifakis, winner of Turing prize, pointed out in his research work that the design of detection system needs rigorous detection design theory, detection model and auxiliary tools. At present, there are some model-based detection system design methods with strong description ability, and the models describing the detection system have formal semantics, such as BiP detection model. However, it is still very difficult to prove the correctness of the detection system. Whether in the system development and design stage or after completion, people can use testing to find out the defects of the system. Although testing can quickly and directly find out system errors, it is difficult to cover all possible operating states of the system. There are many operating states of the system, and the use test can only check a small part of them. Therefore, the test cannot guarantee the correctness of the system. Different from the test method, the formal verification method can cover all possible running states of the system.

2 Related Technologies

Ref [5] demonstrate a method for the reduction of testing effort in safety-critical software development using DO-178 guidance. The key objective of Ref [6] first, is to determine the potential local behaviors using sequential rule mining considering time constraint. Drawing from research traditions in mathematical logic, programming languages, hardware design, and theoretical computer science, model checking is now widely used for the verification of hardware and software in industry. The editors and authors of this handbook are among the world's leading researchers in this domain, and the 32 contributed chapters present a thorough view of the origin, theory, and application of model checking [7]. Theoretical research and experience show that when using conventional design approaches it is impossible to guarantee high confidence to those systems [8]. Ref [9] propose a description language for envisioned scenarios in the problem domain of consistency management, as well as a complementary description language for solution strategies in terms of method fragments and method patterns in the solution domain of Model-Driven Engineering (MDE). A watershed segmentation algorithm with an optimal marker is proposed [10]. Ref [11] propose a method for testing design models using EAST-ADL architecture mutations. Taking a whole plant steam pipeline as an example Ref [12] introduce the stress check criterion of pipeline in ASMEB31.3. Ref [13] present a novel method, using semantic web technologies, to model and validate complex scheduling constraints. Ref [14] propose a mutation-based framework for effective and efficient conformance checking between virtual/silicon device implementations and their specifications.

2.1 Overview of Machine Learning

At the online Amazon cloud technology China Summit on September 14, 2021, Swami subramasan, vice president of artificial intelligence and machine learning of Amazon cloud technology, talked to deep learning, an artificial intelligence education company

Al founder Wu Enda Anders, China n, discussed the future of machine learning and the basic skills that the next generation of computer researchers need to master [1].

Haoenda shared the skills of bringing machine learning from proof of concept to production on site, and quickly won the first project to obtain driving force, so as to ensure that senior managers responsible for developing and implementing machinery research strategies can fully educate the technology. You only need to quickly build the "initial components" of the first prototype system to understand what is possible. By recording the speed, acceleration and position of the working parts for many times, then establishing the data model and continuously optimizing the "training" model, and finally obtaining the best motion curve. Simply understand, machine learning is to establish a mathematical model through various algorithms, and then continuously use data to train the model to improve the accuracy of the model. Therefore, it can be summarized that the introduction of machine learning into industrial automation requires three steps: collecting industrial field data, establishing models and training models, and downloading them to practical applications. This link will be used for tools such as many products of our automation control (such as twincat3scope, twincat3mysql server, twincat3dataagent and twincat3analyticslogger of Beifu). These tools can use this tool to collect them into local database or cloud storage and display the data, so as to build the model and training in the next step. After model training, generate and export description files that can provide twincat3 and other mode operation environments: XML files or onnx files. When it comes to model building and model training, the last step is to load the model into an industrial computer or controller for operational calculation. Figure 1 shows the machine learning research model.

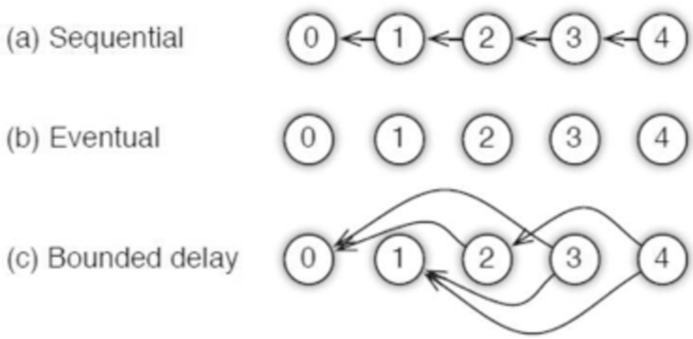

Fig. 1. Machine learning research model

The correct machine learning model should be based on the understanding of business problems, and data and machine learning algorithms must be applied to solve problems to build a machine learning model that can meet the needs of the project. It can make it easy to build and train machine learning models. To succeed in the construction of machine learning model, the most important thing is to be prepared to avoid mistakes in the early stage, and constantly find improvements and better ways to meet the business needs of the sustainable development of the organization.

Industrial software (English: industrial software) refers to the software applied in the industrial field, including system, application, middleware, embedded, etc. Generally speaking, industrial software is divided into programming language, system software, application software and middleware between the two. The system software provides the most basic functions for computer use, but it is not specific to a specific application field. Application software is just the opposite. Different application software provides different functions according to users and the fields they serve. Industrial software is generally divided into two types: embedded software and non embedded software. Embedded software is the acquisition, control, communication and other software embedded in the controller, communication and sensor devices. Non embedded software is the design, programming, process, monitoring, management and other software installed in the general-purpose computer or industrial control computer.

In particular, embedded software, which is applied in the fields of military electronics and industrial control, has high requirements for reliability, security and real-time, and must undergo strict inspection and evaluation. Special emphasis should also be placed on the design related software, such as CAD, CAE, etc. In addition to the nature of software, industrial software also has distinct industry characteristics. With the continuous development of the automation industry, through the continuous accumulation of industry knowledge, the industry application knowledge as a key element in the development of the automation industry has gradually become the main factor for enterprises to adjust the economic structure and change the mode of economic growth.

1. Industrial software is inseparable from process support
   The industrial control software in different industries serves different objects. The iron and steel industry is aimed at the metallurgical industry, and its control software is difficult to apply to the machinery industry, and vice versa. A set of good industrial control software can not only meet the needs of the current process, but also have a certain sense of advance in the control idea, and will not fall behind in a certain period of time.
2. Industrial software should be supported by industry data knowledge base
   The industry data knowledge base refers to the collection of experience accumulated in the industry production process that plays a supporting role in the industry control software. In particular, it should be pointed out that the collection of key knowledge, software, know-how and data in the production process of the industry is also the basis for the installation of China's automatic control system. Its main contents include: experience calculation formula, technical know-how, various accident handling experience and various operation experience, operation manual, technical specification, process model, algorithm parameters, coefficient and weight proportion distribution after various data are collected in the production process. It includes not only technical specifications, operation specifications, national standards, etc. in the form of documents, but also core software contents and solution tools such as empirical formulas and model algorithms. The development and production of industrial control software products for different industries are rising. The data knowledge base of various industries is becoming the core of industrial control software and the development and expansion of basic elements. This is also our advantage. Building a good industrial data knowledge base will turn industrial knowledge into the driving

force for the development of industrial control software, And promote the technical level of China's industrial control software to better complete the transformation from low-end to high-end.

2.2 Model Inspection

Probabilistic model detection can make qualitative and quantitative analysis of attributes on the stochastic system model, verify whether the probability of attributes meets the limited threshold, or calculate the probability of attributes to be verified. The attributes to be verified are expressed by probabilistic temporal logic. Specifically, the model detection logic pctl and pctl * 1 are used to express the probability attribute on the discrete-time Markov chain model, and can express whether the probability of random behavior is within the limited threshold. For example, "in a network protocol, the probability that information can finally be sent is greater than 0.9", or "95% of the probability that information transmission is completed within 10 ms", etc. Csli60l is the expression of probability attributes on continuous time Markov chains. It has an operator s that describes the stability probability of the system. For example, "after a long period of time, the probability that the message queue in the network is full is 0.5". In addition, probabilistic logic can express the desired properties of the system. The random component system is modeled as Markov reward model, and the system expectation to be verified can be described by pctl, pctl * or CSL. For example, "given the current probability distribution, the energy consumption in the network is in the interval [30MW, 50MW] for a given period of time". When the attributes are not satisfied, the probabilistic model detection method can not directly provide the counterexample leading to the result, and additional methods are needed to analyze the system counterexample [2].

Prism platform integrates probabilistic model detection and statistical model detection tools, which can conduct a more comprehensive probability analysis of modular (component) systems. In particular, an effective statistical model detection algorithm is used. Compared with SBIP verification tool, prism's statistical model detection algorithm is more efficient, and can verify all bounded temporal logic and some non bounded logic attributes. Andova et al. Extended pctl to describe the temporal logic on Markov reward model, and implemented an algorithm to verify it in prism. Prism tool uses modular language L64 to describe the system model. For component system, the description ability of the model is limited, and it is difficult to describe the behavior and interaction of complex component system.

3 An Industrial Software Model Detection Method Based on Machine Learning

Industrial software model checking is an effective method to alleviate the explosion of system verification state. It deduces the attributes satisfied by the whole complex component system from the attributes satisfied by each sub component. The combined model detection method based on invariants is to verify whether the system attributes (invariants) can be derived from the invariants calculated by each sub component. The effectiveness of industrial software verification depends on the system infrastructure or

the attributes to be verified, or both. In the detection system, in addition to the behavior of each sub component, the interaction behavior between components should also be considered.

Machine learning "ml" is driving the explosive growth of artificial intelligence "Ai" applications to help software understand the uncertain and unpredictable real world. The main difference between machine learning and traditional computer software is that human developers have not written code to instruct the system how to distinguish between bananas and apples. In addition to machine learning, various methods are used to build al systems, including evolutionary computing and other methods of expert systems, which is shown in Fig. 2.

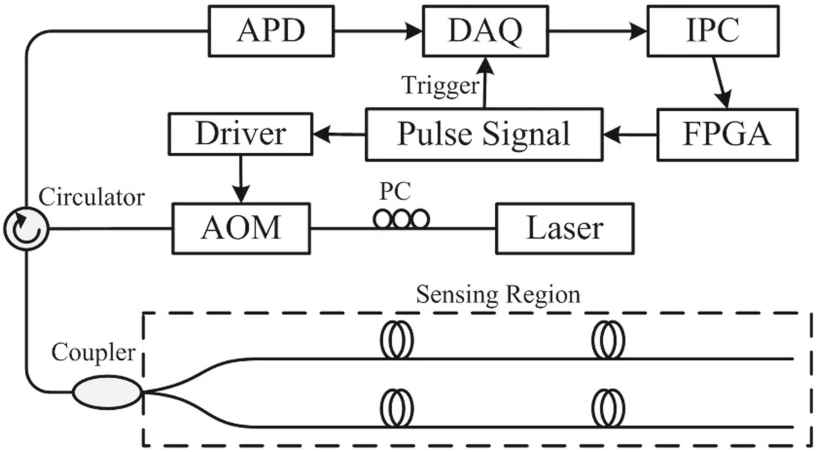

Fig. 2. Industrial software model detection method

However, from a practical point of view, machine learning is the research of improving algorithms through experience or data. Since the 1950s, the research on machine learning has played a leading role in the practical application of related disciplines, from the initial research on the methods based on neuron model and function proximity theory to the emergence of symbolic calculus rule learning and decision tree learning, and the subsequent introduction of concepts such as summary, interpretation and analogy in cognitive psychology. Programmers develop algorithms to guide computers in new tasks, which is the basis of the advanced digital world we see today [3]. The basic process of machine learning is to provide training data for learning algorithms. By using different training data, the same learning algorithm can generate different models. Non experts tend to use automl to automate as many pla steps as possible, and they can't maintain model performance only by manpower · the three advantages of automl 1 It also helps to avoid manual operation; Compared with traditional data analysis, data mining reveals the unknown relationship between data. The random detection model of industrial software is shown in Fig. 3 below.

Machine learning software uses data to train models. Among many statistical algorithms, the more popular ones are: simple Bayesian algorithm for emotion analysis, spam

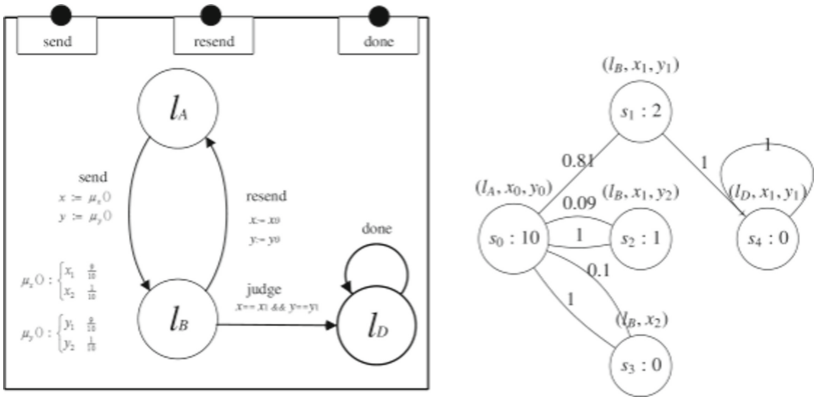

Fig. 3. Random detection model of industrial software

detection and recommendation, decision tree for result prediction; Multiple deterministic forests can be combined to improve the prediction of random forests; Logistic regression using binary classification "a or B"; AdaBoost, gas, bayuan, materials, leader and k-mease clustering reorganize data, such as market segmentation. Training in artificial intelligence and mechanical research models. However, converting data into order may be a time-consuming and detail oriented process, which may require manual processing. Model parameters are automatically learned by machine learning software from artificial intelligence input data in the training process, although users can manually change parameter values in the training process.

4 Machine Learning Improves the Model Detection Rate

From a horizontal perspective, from the free computing pool prepared for individual developers to the al model optimization tools required by large factory professionals, the corresponding releases are also available. Awsmaker canvas treats many steps of the machine learning model as interactive u to solve their business problems [4]. As a supplement to his machine learning toolkit, the head of the data science group of the application department said: Canvas enables us to share and collaborate with the data science team to help produce more machine learning models and ensure that the models meet quality standards and specifications.

This is a new type of example designed specifically for training machine learning models. With the example of Amazon ec2dl1, customers can optimize workloads such as natural language processing, object detection and classification, fraud testing, recommendation and personalization engine, intelligent document processing and business forecasting. In order to improve the prediction accuracy of the model, data scientists and machine learning engineers are building more and more complex models.

Developers and data scientists can also optimize the Gaudi accelerator from the reference model provided in the Havannah hackhub repository, including popular patterns for various applications, including image classification, object detection, natural language processing and recommendation systems. Accessing the D1 instance through Amazon

sagemaker, customers' experience based on managed services makes it easier and faster for developers and data scientists to build, train machine learning models and deploy cloud and edge operations in the cloud. Benefiting from the Amazon unar system, many traditional virtualization functions can be unloaded to dedicated hardware and software to provide high performance, high availability and high security and reduce the sales of virtualization.

Seagate data science and machine learning engineers have established an advanced deep learning "DAAD" defect detection system and deployed it in the company's global manufacturing facilities. In a recent proof of concept project, Havannah Gadi surpassed Seagate's current performance goal of training to produce BL semantic segmentation models.

In addition, users can also use Havannah synphosal integrated with mainstream machine learning frameworks such as tlaxflow and pytorch. With only a small amount of code change, the current machine learning model based on GPU or CPU operation can be seamlessly migrated to D1 instance for training, which is shown in Fig. 4.

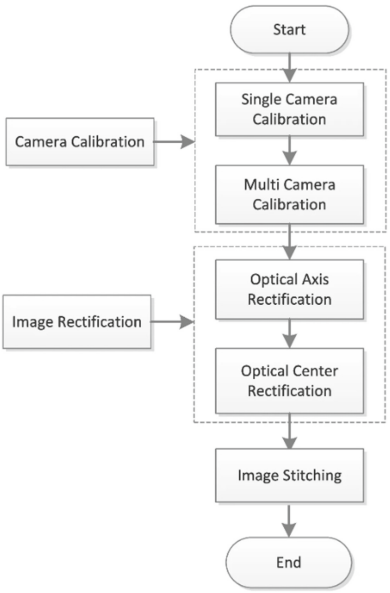

Fig. 4. Software flow chart using image proceed

5 Conclusion

With the deepening of the research on model checking technology and software formal verification technology, the software verification technology based on model checking will be gradually improved and become a necessary method to ensure software quality in the process of software development. This paper uses traditional methods to eliminate

errors in software systems, which has become more and more difficult to meet the actual needs. We must pay attention to new software verification theories and methods. Among the many theories and methods proposed for this purpose, model detection is remarkable for its simplicity and high degree of automation. With great success in the field of hardware and protocol verification, model checking is gradually applied to software verification.

Acknowledgements. This work was supported by the Scientific Research Plan Guidance Project of Hubei Education Department (Project No. B2021310).

References

1. Deb, S.D., Jha, R.K., Jha, K., Tripathi, P.S.: A multi model ensemble based deep convolution neural network structure for detection ofcOVID19. Biomed. Signal Process. Control. **71**(Part), 103126 (2022)
2. Mohebbanaaz, Sai, Y.P., Rajani Kumari, L.V.: Cognitive assistant DeepNet model for detection of cardiac arrhythmia.Biomed. Signal Process. Control. **71**(Part), 103221 (2022)
3. Togacar, M., Muzoglu, N., Ergen, B., Yarman, B.S.B., Halefoglu, A.M.: Detection of COVID-19 findings by the local interpretable model-agnostic explanations method oftypes-based activations extracted from CNNs.Biomed. Signal Process. Control. **71**(Part), 103128 (2022)
4. Verma, S.S., Prasad, A., Kumar, A.: CovXmlc: high performance CoVID-19detection on X-ray images using Multi-Model classification. Biomed. Signal Process. Control. **71**(Part), 103272 (2022)
5. Sun, Y., et al.:Functional Requirements-Based Automated Testing For Avionics, ARXIV (2017)
6. Setiawan, F., Yahya, B.N.: Improved behavior model based on sequential rule mining. Appl. Soft Comput. (2018)
7. Clarke, E.M., Henzinger, T.A., Veith, H., Bloem, R.: Handbook of Model Checking (IF: 5) (2018)
8. Todorov, V., Boulanger, F., Taha, S.: Formal verification of automotive embedded software. In: 2018 IEEE/ACM 6TH International Fme Workshop on Formal Methods in Software Engineering (FORMALISE) (2018). (IF: 3)
9. Anjorin, A., Yigitbas, E., Leblebici, E., Schürr, A., Lauder, M., Witte, M.: Description languages for consistency management scenarios based on examples from the industry automation domain, ARXIV (2018). (IF: 3)
10. Zhang, H., Tang, Z., Xie, Y., Gao, X., Chen, Q.: A watershed segmentation algorithm based on an optimal marker for bubble size measurement. Measurement (2019). (IF: 3)
11. Enoiu, E.P., Seceleanu, C.: Model Testing of Complex Embedded Systems Using EAST-ADL and Energy-Aware Mutations (2020)
12. Pu, Z., Wang, J., Ren, B., Song, P., Zhao, F.: The Research on The Stress Analysis of Overhead Steam Pipeline (2020)
13. Soman, R.K., Molina-Solana, M., Whyte, J.K.: Linked-Data Based Constraint-Checking (LDCC) to Support Look-ahead Planning in Construction, AUTOMATION IN CONSTRUCTION (2020). (IF: 3)
14. Gu, H., Zhang, J., Chen, M., Wei, T., Lei, L., Xie, F.: Specification-driven conformance checking for virtual/silicon devices using mutation testing. IEEE Trans. Comput. (2021)