

# Exploration of Innovative New Paths for Intelligent Empowerment in Software Testing and Evaluation

Dinggui Jiang<sup>1\*</sup>

<sup>1</sup> Zhongke Software Testing (Guangzhou) Co., Ltd., Guangzhou, 510075, China

\* 2320187062@qq.com

<https://doi.org/10.70695/IAAI202501A5>

---

## Abstract

With the vigorous development of the software industry, the complexity of software systems is increasing day by day, and the shortcomings of traditional software testing and evaluation methods in terms of efficiency and accuracy have become more apparent. This paper focuses on the field of intelligent empowerment in software testing and evaluation, delving into innovative paths. By constructing an intelligent empowerment framework for software testing and evaluation, it elaborates on its core modules and the principles of embedding intelligent technologies, and conducts argumentation and analysis with actual cases. The research indicates that intelligent empowerment can significantly enhance the efficiency and quality of software testing and evaluation, injecting new momentum into the development of the software industry, while also pointing the way for future research and practice in the field.

**Keywords** Intelligent Empowerment; Software Testing; Innovation

---

## 1 Introduction

In the surge of digitalization, software has permeated every corner of society, from the mobile apps we use daily to intricate industrial control systems, making its significance self-evident [1]. However, the relentless expansion of software system size and the escalating complexity of features have posed unprecedented challenges to software testing. Traditional software testing and evaluation rely on manually designed test cases, test execution, and result analysis, which not only consume substantial human and time resources but are also prone to oversights and misjudgments when dealing with vast codebases and complex business logic [2][3]. This approach struggles to align with the demands of modern software for rapid iteration and high-quality delivery. At the same time, intelligent technologies such as artificial intelligence, machine learning, and big data analytics have made significant breakthroughs in recent years, offering new opportunities for the transformation of software testing and evaluation. The integration of intelligence into software testing and evaluation has become an inevitable trend in the industry's progression.

This study aims to explore innovative approaches to intelligent-enabled software testing and evaluation, and to construct a scientific and efficient intelligent software testing and evaluation system. Specifically, by introducing intelligent technologies, the study aims to achieve the automatic generation and optimization of test cases, real-time monitoring and intelligent decision-making during the testing process, and precise analysis and in-depth mining of test results, thereby enhancing the efficiency and accuracy of software testing and evaluation, reducing testing costs, and ensuring the quality of software products. This research holds significant practical importance for promoting technological innovation and development in the software industry. On one hand, it aids software enterprises in launching high-quality software products more quickly in the fierce market competition, thereby enhancing their core competitiveness. On the other hand, it provides new insights and methods for theoretical research in the field of software testing, promoting further development of the discipline.

## 2 Framework Construction for Intelligent Empowerment in Software Testing and Evaluation

The framework should be able to accurately utilize intelligent technologies for software testing and evaluation, ensuring that the test results truly reflect the software quality status, and reducing the rates of false positives and false negatives [4]. From a mathematical perspective, assuming the set of test results is  $T$  and the set of actual software quality states is  $S$ , the ideal case is  $T=S$ , meaning the test results completely match the real quality states. The principle of accuracy is to make the test results as close to the real states as possible, which can be expressed by the formula, indicates a measure of distance between the test result set  $T$  and the real quality state set  $S$ , such as Hamming distance, etc., by optimizing the framework to minimize this distance.

By leveraging automation and intelligent methods, the efficiency of test evaluation is enhanced, manual intervention is reduced, the testing cycle is shortened, and the requirements for rapid software iteration are met. Let the time required to complete a test using traditional testing methods be denoted as, In terms of resource utilization, let the traditional testing resource consumption be denoted as.

Given the continuous development of software technology and the changing business requirements, the framework should have good scalability, allowing for the convenient integration of new intelligent technologies and testing methods to adapt to the testing needs of different types of software projects [5][6]. From the perspective of system architecture, the framework can be viewed as a function  $F$ , where the input consists of various testing requirements and technical modules  $x$ , and the output is the test evaluation results that meet the requirements, denoted as  $y = F(x)$ . When new intelligent technologies or testing methods, represented as, are introduced, the framework should ensure that remains effective without the need for significant restructuring of the framework.

The application of intelligent technologies should ensure the stability and reliability of the framework, preventing errors or unreliable results in testing evaluation due to technical failures or algorithmic biases. Taking machine learning algorithms as an example, let the accuracy rate of the algorithm be  $P$ , and the recall rate be  $R$ . The reliability principle requires that both  $P$  and  $R$  are maintained at high levels and that they fluctuate minimally across different datasets and operating environments. Typically, the  $F$  score is used to comprehensively measure the reliability of the algorithm,

$$F = 2 \times \frac{P \times R}{P + R} \quad (1)$$

The overall framework is shown in Figure 1.

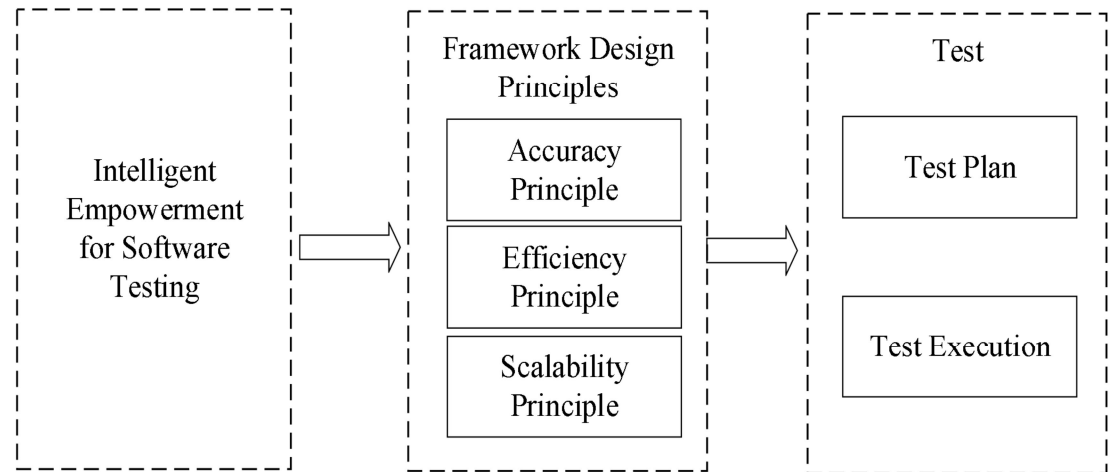


Fig. 1. Framework Overview

## 3 Framework Construction for Intelligent Empowerment in Software Testing and Evaluation

To verify the effectiveness of the innovative path for intelligent empowerment in software testing evaluation, multiple actual cases were selected for in-depth analysis. Taking an e-commerce platform software project of an Internet company as an example, before the introduction of the intelligent

empowerment testing evaluation system, the project's testing team used traditional testing methods, which had a long testing cycle and a low defect detection rate. During the project development process, an average of 2 weeks was spent on testing for each iteration, and some defects were still reported by users after the software went live.

After the introduction of the intelligent empowerment software testing evaluation framework, the natural language processing technology in the requirement analysis module was used to quickly and accurately understand and analyze software requirements, reducing the requirement analysis time by 30%. Through the intelligent algorithms in the test plan formulation module, test resources were allocated reasonably, reducing the number of test cases by 20% while ensuring coverage, and making the focus of testing more prominent. In the test execution phase, the application of intelligent automated testing tools reduced the test execution time by 50% and was able to automatically handle most exceptions during the testing process. In the results evaluation and feedback module, the use of big data analysis and machine learning algorithms increased the defect detection rate by 40%, made defect localization more accurate, and reduced the average defect repair time by 3 days.

Through the statistical analysis of data from multiple similar cases, it was found that intelligent empowerment in software testing evaluation has significant effects in improving testing efficiency, reducing testing costs, and enhancing software quality. The average test cycle was shortened by 40%-50%, the defect detection rate increased by 30%-50%, and the testing costs were reduced by about 30%. These data fully verify the feasibility and superiority of the innovative path for intelligent empowerment in software testing evaluation.

## 4 Conclusion

In the implementation process of intelligent empowerment for software testing evaluation, some challenges are also encountered. The primary challenge is data security. Since intelligent technology relies on a large amount of test data for training and analysis, this data may contain the software company's business secrets and user privacy information. To address this challenge, companies should establish strict data security management systems, employ data encryption technology to encrypt the storage and transmission of sensitive data, and at the same time, strengthen access control to ensure that only authorized personnel can access the relevant data. For example, in terms of data encryption, the AES (Advanced Encryption Standard) algorithm can be used. Based on this, further research can be deepened.

## Conflicts of Interest

The authors declare no conflicts of interest.

## References

1. Qiu, W. S. (2021). A discussion on the automation testing of intelligent connected systems. *Time Auto*, (18), 10-11.
2. Wei, D., Xia, B., & Wang, Z. K. (2021). Design of a voice-activated intelligent evacuation system based on a microcontroller. *Precision Manufacturing and Automation*, (03), 11-16.
3. Shen, Y. (2021). Research on software engineering methods based on swarm intelligence. *Electronic Technology and Software Engineering*, (16), 35-38.
4. Huang, Y. H. (2021, 07-30). Building an AI automation testing software toolchain. *Guangdong Science and Technology Daily*, p. 006.
5. Wang, Y. (2021). Research on GUI software test data generation method based on visual mining. *Nanjing University of Aeronautics and Astronautics*.
6. Cheng, K. Q., Lin, J. Q., Yang, D. Y., et al. (2021). Accelerated testing method for intelligent workshop system simulation based on digital twin. *Computer Measurement and Control*, 29(01), 39-44+49.

## Biographies

1. **Dinggui Jiang** graduated from Jishou University with a B.S. in Computer Science and Technology, and worked in software testing for 5 years.

## 探索軟件測試與評估中智能賦權的創新新路徑

蔣定桂

---

摘要：隨著軟件產業的蓬勃發展，軟件系統的復雜性日益增加，傳統軟件測試和評估方法在效率和準確性方面的不足日益明顯。本文重點研究了軟件測試和評估中的智能賦權領域，深入探討了創新路徑。通過構建軟件測試與評估的智能授權框架，闡述了其核心模塊和嵌入智能技術的原則，並結合實際案例進行了論證和分析。研究表明，智能賦能可以顯著提高軟件測試和評估的效率和質量，為軟件產業的發展注入新的動力，同時也為該領域的未來研究和實踐指明了方向。

關鍵詞：智能賦能；軟件測試；創新

---