www.cadanda.com Computer-Aided Design AND APPLICATIONS™

E-learning in Software Engineering Education for Computer Software Development in the Big Data Context

Yifan Huang[1],Yaqiong Wang[2*] and Pengcheng Wang[3]

[1,2,3]School of Basic Education, Hunan Vocational College for Nationalities, YueYang414000, Hunan Province, China

[1]hyfrain@foxmail.com,[2]rainhermain@163.com,[3]wpc_007@qq.com

Corresponding author: Yaqiong Wang, rainhermain@163.com

Abstract. In the context of wide application of information technology, fully mastering the technical methods and technologies of software development and understanding the essential characteristics of software development can fully improve the technical capacity of application software, and it also plays a supporting role in solving technical problems related to software development. In the era of big data, economic development and technological innovation are prerequisites for the development of the era; most information technology applications require software as a carrier. We must develop appropriate software according to our needs in the application process. Application software technology can improve the efficiency of software development. The developed software can effectively improve users' understanding and application ability. This paper introduces the importance of the software development method and analyzes its application practice in software development. The case study shows that the method in this paper has obtained 87% utility approval in the practical application process.

Keywords: E-Learning; software; big data; application ability; Software Engineering.
DOI: https://doi.org/10.14733/cadaps.2024.S22.1-17

1 INTRODUCTION

In the information age, people's lives and work will produce a lot of data, which contains valuable information and knowledge and will help enterprise managers make scientific and practical management decisions[6]. Data analysis and mining are information technologies in the age of big data. Its driving force is the increase in data and the demand for data analysis [8]. The amount of information carried by words doubled every ten years after the Industrial Revolution. After 1970, the amount of information began to double every three years. Up to now, the amount of information

has doubled every two years. In 2020, the global data usage will be about 35ZB, achieving explosive growth. The main areas where data exist are geographic information, medical and health care, and others. The amount of data has increased rapidly in the process of decreasing the cost of data collection, and the data types have significantly increased with the emergence of new data sources and data collection technologies, which also increases the spatial dimension of data and the complexity of big data [12]. Take Taobao as an example. The number of members is about 370 million, and the number of online goods exceeds 880 million. A large amount of data will be generated every day. If these data are not used scientifically, they will be wasted, and the value of big data cannot be realized[9].

SE methods have also undergone constant innovation and upgrading, and they play an indispensable role in computer software development. Modern technology also makes software development more concise, reducing the time and workforce required for software development[15]. Whether in the medical, financial, tax, or education fields, we need support for science and technology to achieve our innovative development. Applying modern SE methods to the development of computer software can make the systems and software used in various industries have more intelligent functions, help the industry carry out information reform, and promote the industry's sustainable development, which is of great practical significance [4].

SE methods have profound application significance in computer software development. Some software development difficulties involve a wide range of people, and having many applications is relatively high. More software engineers must cooperate closely in the development process, which takes a long time [14]. Once the development quality is unqualified or cannot meet users' needs, new software will be developed, and the demand for human resources, economy, and material resources is high.

The development efficiency can be improved after the SE method is applied. The developed things can be detected promptly during development, and each port can be effectively monitored. Once problems occur, adjustments can be made promptly, ensuring the developed software's overall performance and avoiding multiple repairs later [19]. The SE method provides an automated or semi-automatic development environment in the development process. It directly constructs a series of task frameworks, sets corresponding tasks according to the development goals, divides the tasks into different stages, and completes the software development stage.

The design and development of software is to meet the application of established groups. Only a good user experience of software can achieve the sustainable development of the software itself. For software development engineers, only by designing and developing software satisfactory to the public and users with good experience can the software be valuable and its value be reflected. Applying SE methods can effectively assist software engineers in developing software products, more closely connecting data and data in the development process, repeatedly conducting independent tests, and ensuring smooth transformation between various software functions [23].

Nowadays, most computer software needs to use the network as the basis of its application. Some large-scale software has high requirements for computer hardware and networks. Once the network signal is terrible, it will lead to software stuttering, frame dropping, and other situations that affect the user experience; it can optimize the network application system, have a more vital network capture ability, and reduce the impact of the network on software applications[13],[16].

The application of SE allows computer software development access to more intelligent functions. Some computer programs are challenging to achieve application by relying on pure manual programming, which consumes more time and human resources and is difficult to maintain later.

Therefore, only some computer software will be put into application. With the support of SE, computer software has achieved semi-automatic development. With the support of SE, many links have reduced human participation, and the accuracy of software functions has been improved through modern technology.

2 RELATED WORKS

Developing computer system software is complex and tedious work involving a wide range of professional knowledge. Carelessness may lead to various problems and may have a particular impact on the performance and stability of the computer software system [11]. Applying SE technology can provide a suitable environment for the development work. In the actual development work, the relevant staff can timely find out the problems and deficiencies in all aspects of software development, reduce the influence and restriction of external factors on software system development as much as possible, and make the computer software development system and program more and more perfect. In the 1960s, various SE methods emerged, such as structured, formal, and object-oriented methods [7]. The structured method mainly divides the life cycle of software into different stages according to the actual situation of software development and gradually realizes the development goals of different stages through structured technology. The formal method is mainly based on traditional mathematical transformation, which can transform the system description into an executable program. Object-oriented methods can closely relate data and related operation steps, making the software development process more stable and orderly. Only by strictly controlling each step and doing well in every detail can the limitations and shortcomings of computer software development be reduced as much as possible and the development task be ensured orderly [5],[1]. In addition, SE technology can further expand the storage space of computers, reduce the pressure on hardware, and avoid excessive resource consumption in the actual operation process. When any problem occurs in the development process, the relevant staff can also give full play to the advantages of SE technology, quickly find the problem, and take corresponding measures to solve it in time to avoid becoming more serious.

The structural analysis principle is to form a complete structural framework based on the content association between various system elements so that the use of computer software functions can be fully expanded [17],[21],[10]. Most foreign object-oriented methods are used for the functional advantages of computer systems, and the system functions are adjusted based on the use of different system languages. However, in terms of the structural analysis method, it is impossible to match the existing computer language system due to the difficulty of technical development and the outdated technical system [18]. Therefore, it is mainly used for the function development of server computer equipment [20].

Applying domestic SE methods also adopts the object-oriented method module [2]. In the early stages, computer software development in China mainly relied on formal methods. The development mechanism used in the underlying architecture was the same, but it was only optimized in terms of software content and functional assistance [24]. In recent years, the computer technology level in China has improved yearly, and SE technology methods have gradually shifted to object-oriented methods [22]. The domestic object-oriented method system development logic application is extended around using foreign Object-c, C++, Eiffel, Object Pascal, Java, and other languages [3]. However, the difference is that most of the domestic object-oriented methods make use of the polymorphic characteristics of this method and use the computer system clock to operate instructions on different objects of the same message to maximize the ability of computer software to synchronize information processing so that the functions of computer system software can be

further adjusted based on user needs. Therefore, it has advantages over foreign countries regarding method use and technical convenience.

3 METHODS

3.1 SE Ecology

Through comprehensive sorting and analysis of software ecology, this paper divides the software community into the development and application communities, covering different stages of software development, release, and application, and conducts research and analysis on SE big data. SE big data is based on code, documents, development records, and other texts with rich semantics. To this end, this paper has built a systematic SE open-source ecological extensive data system (as in Figure 1), which involves development products, development processes, software products, software images, consultation discussions, application Q&A, and other aspects, covering GitHub, Apache, Topcoder, Docker Hub, OSCHINA, Stack Overflow and different types of mainstream open source communities, providing a more complete global view for SE research and experiments.

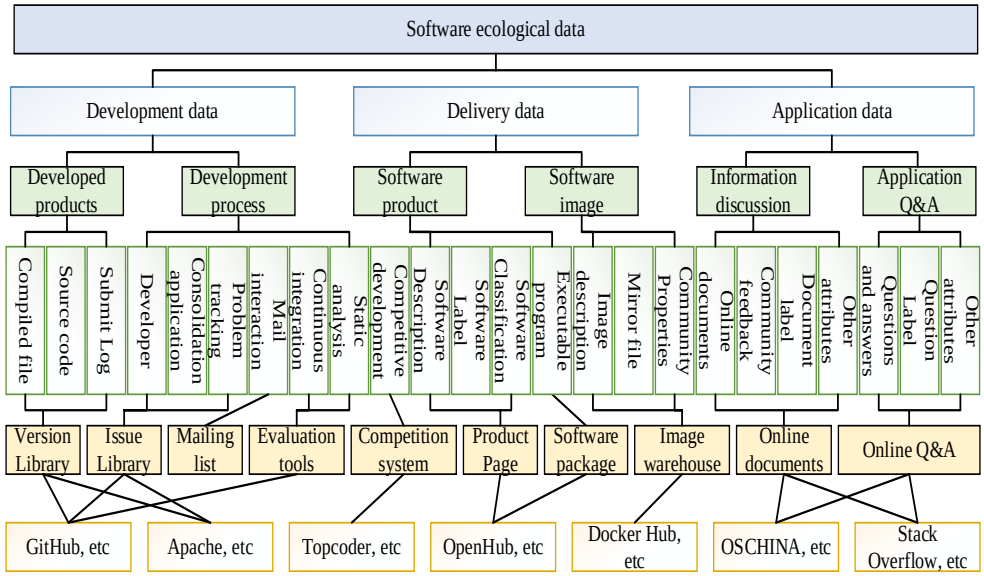

Figure 1: Open-source ecological extensive data system of SE.

The system mainly includes three categories: development data, delivery data, and application data, and each category is subdivided into multiple subcategories. Finally, the classification system has established a mapping with the specific data formats of various current software warehouses, communities, and forums. The details are as follows: (1) Development data: The development data takes the products and processes of software development and design in SE as the core. The development products include source code, compiled files, and submission logs. The development process involves developers, merger applications, problem tracking, email interaction, continuous integration, and competitive development. Specifically, the data sources corresponding to the development data include version library, Issue library, mailing list, evaluation tool, and competition

system, and the data source instances involved include GitHub, Apache, and Topcoder. (2) Delivery data: Delivery data mainly refers to software products delivered to the production environment. At the same time, with the development of virtualization technology and container technology, software image, as a particular delivery method of software products, is gradually emerging and exists in large numbers. Therefore, it is also a separate subcategory of delivery data. Software product data involves all aspects of software, including software description, software label, software classification, and executable program itself; software images are similar, including image descriptions and image files. Delivery data mainly comes from software product pages, software packages, and image warehouses. Specific examples include OpenHub and Docker Hub. (3) Application data: application data mainly includes information discussion and application Q&A, in which information discussion includes online documents, community feedback, document tags, and other attributes. Application Q&A is mainly aimed at online community activities through questions and answers. The data includes questions and answers, question tags, and other attributes. The primary data sources of application data are online documents, and Q&A. Specific examples include OSCHINA and Stack Overflow.

This paper proposes an "incremental, multi-mode" self-growing data acquisition and processing framework (as in Figure 2), which can aggregate, collect, and organize different software data types. Specifically, for different kinds of data such as web page data, version library data, and defect library data, this paper studies the key technologies such as active awareness, directional collection, multi-source association, and incremental detection, designs and deploys distributed crawlers, and implements a variety of collection methods such as web page crawlers, data acquisition based on application programming interface (API), and direct download of data packets. The details are as follows.

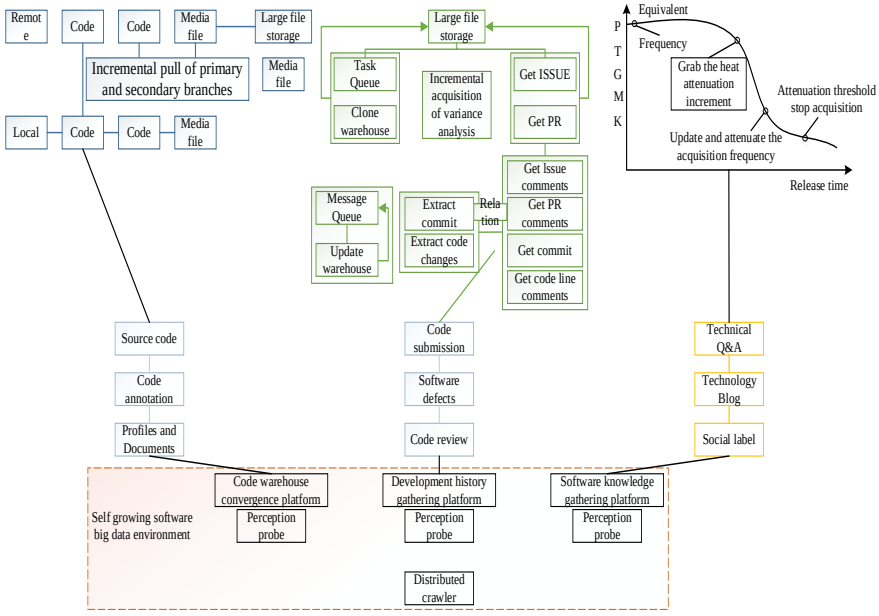

Figure 2: Open-source ecological extensive data collection and processing framework of SE.

Data collection method based on web crawler: the technique of fixed-point crawling is adapted for a specific software library, and the corresponding data information is obtained based on the strategies of tag matching and regular expression matching commonly used by the crawler by analyzing the data characteristics and schema format in specific data source web pages. In addition, periodic and incremental crawling is realized based on information such as timestamps to avoid repeated data acquisition. Among them, most software data are obtained based on the crawler technology, including the source code, email, web page, and version control of the Apache Foundation project, the defect report and code of the Eclipse community project, the metadata information in the funded code base and Docker Hub, the Dockerfile of code products and containers, the blog, question and answer and forum of CSDN, etc.

API-based data collection method: Besides web crawlers, some open-source software libraries provide open APIs for obtaining and downloading data information. Therefore, corresponding data information can be obtained by calling APIs, including Topcoder crowdsourcing development data, Apache Foundation project defect reports, etc.

Direct download method of data package: some communities compress and archive historical data and directly provide external data download addresses, such as document data of the Stack Overflow community. On this basis, SE's open-source ecological extensive data collection and processing framework can be divided into three layers: data acquisition, data analysis, and persistence. The initial data collection, analysis, processing, and final data display can be completed by collaborating the three layers. The data analysis layer extracts the page information obtained by the data acquisition layer, extracts the critical information on each page, and verifies the extraction results. The verified page data will be stored in the database to prepare for data analysis. There are also some data mining algorithms in this layer to analyze the extracted data, including community association, software evaluation, etc. The persistence layer processes the data processing results according to the final data format to be displayed and stores the processing results in the data buffer pool. Finally, the persistence layer continuously transmits the data to the display platform to provide data support for the platform's display.

3.2 Modeling based on SE

The main structure of the system is divided into three parts: graphical modeling end, background server, and cloud computing platform. The interaction between the graphical modeling end and the background server is mainly about uploading and downloading documents. The specific process is that modelers click to upload documents to the background server and click to download documents to download documents to the local. The interaction between the background server is mainly about the upload of documents and the reception of feedback.

As in Figure 3, the work on this topic is in the red box. The output of the modeling end includes the modeling operation of the modeler, the model description document, and the data document to be collected in the process management. The output of the cloud computing platform is two types of analysis results, one for models and one for modelers, which are managed by the back-end. The model description generation document module is used to collect and store the model description document, the model validation record document generation module is used to collect and store the model validation information during the modeling process, and the modeling operation sequence document generation module is used to collect and store the operation sequence of the modeler during modeling, The function modules and function point change documents are used to record business management data, the software process management documents are used to record measurable indicators in the software management process that need to be collected, and the

background server is responsible for receiving and sending various documents with the modeling client, and shipping and receiving documents with the cloud computing platform, as well as managing the details of the modeling personnel, The document upload and download module is responsible for the upload of client documents and the receipt of background documents. The software reliability prediction module is responsible for the reliability prediction of various metrics extracted by the cloud computing platform.

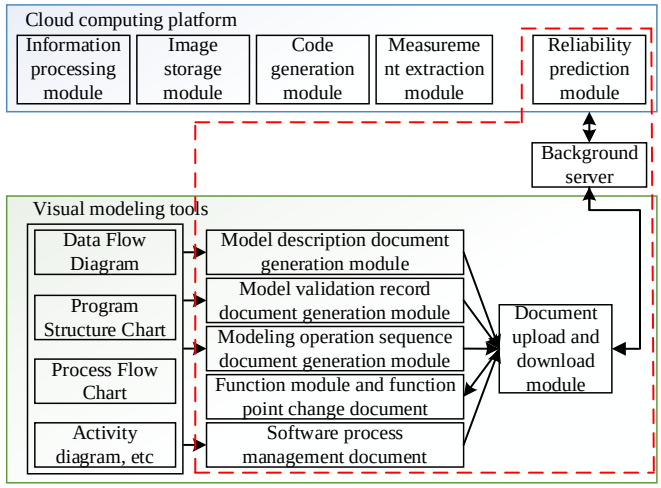

Figure 3: System architecture.

The data storage of this system is divided into modeling tool client and background server. Most of the data is stored in the background server, and a small part of the data needs to be stored locally in the modeling tool client. This system's vast majority of data are stored, uploaded, and downloaded as documents. Only the user details table at the background server is stored in the database. Therefore, only one database table exists in this system to keep the detailed information of modeling staff. The background server uses the MySQL database. According to the requirements of the system, the detailed information table of modeling staff is described as follows: the detailed information table of modeling staff stores the basic information of users, including user ID, user education, project development experience, learning ability, modeling ability, modeling times and historical modeling scoring. Its physical structure is in Table 1.

Data item	*Data type*	*Default*	*Can it be blank*
User ID	*TINYINT*	*1*	*Must not*
Education background	*TINYINT*	*1*	*Must not*
Project development experience	*TINYINT*	*0*	*Must not*
Learning ability	*TINYINT*	*1*	*Must not*
Modeling capability	*TINYINT*	*1*	*Must not*
Modeling times	*TINYINT*	*0*	*Must not*
Historical modeling scoring	*TINYINT*	*0*	*Must not*

Table 1: User table structure.

In addition, since the background server also provides FTP services, documents need to be stored. It includes the software process management documents to be collected from the modeling staff and the function module and function point change documents to be sent to the modeling staff. In addition, the modeling operation sequence document will be uploaded to the opSeq folder under the folder named by the member ID, the model verification record document will be uploaded to the checkers folder under the folder named by the member ID, and the model description document folder will be directly uploaded to the folder named by the member ID. The locally stored documents include the model description document to be uploaded, the modeling operation sequence document, the model verification record document, the downloaded function module and function point change document, and the data document to be collected in the software process management process. Under the local folder named by user ID, a total of 5 subfolders are created, including the model description document folder, modeling operation sequence document folder, model verification record document folder, function module and function point change document folder, software process management document folder, etc., The model description document exists in the corresponding folder in XML form. The naming format of a single document is ID+ model name. The naming of the remaining files is determined by their respective personnel IDs and file types. The format is ID+ document type.

There is only a mutual transfer of documents between the modeling tool client and the background service module. The FTP protocol is used to transfer documents here. The background service module enables the FTP service. When the modeling tool client needs to upload and download documents, it creates an FTP connection with the background service module and selects the documents to upload or download. The background server and cloud computing platform also use the FTP protocol to realize the document upload function.

3.3 Design and Implementation of Reliability Prediction Module

This paper optimizes the reliability prediction model previously used in the system and adds a reliability prediction method based on the activity diagram. The system has once achieved and adjusted the parameters of support vector machine, logical regression, AdaBoost, random forest method, neural network, and K-nearest neighbor algorithm through experimental comparison and obtained the prediction model with the best prediction effect. This model is based on the AdaBoost algorithm; its values are in Figure 4.

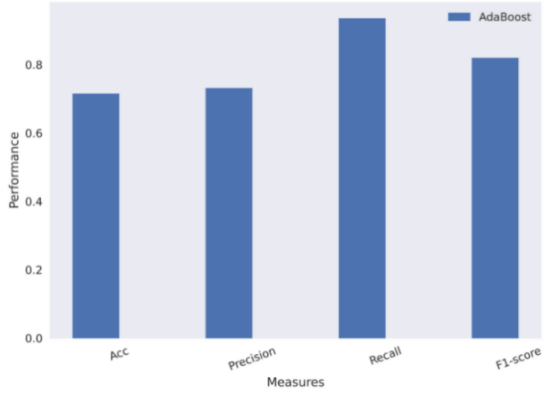

Figure 4: Model prediction effect based on the Adaboost algorithm.

The accurate Adaboost algorithm is in Table 2 below.

Input: training set $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}, y_i \in \{-1, +1\}$
Output: Strong classifier H (x)

Table 2: Real AdaBoost algorithm.

1. The initial distribution $D_1(i) = \frac{1}{N}$,

2. For t=1,2,3, …, T, T is the number of weak classifiers

1. Divide the training set $X_1, X_2, \ldots, X_N$,

2. Calculate the weight:

$$W_l^j = P(x_i \in X_j, y_i = l) = \sum_{x_i \in X_j, y_i = l} D_l(i), l \in \{-1, +1\} \quad (1)$$

3. Calculate t each weak classifier, $\forall x \in X_j, h(x) = \frac{1}{2} 1n \left(\frac{W_{+1}^j + \gamma}{W_{-1}^j + \gamma} \right)$, where γ is the smoothing factor, $\gamma \in (0,1)$.

4. Calculate the normalization factor $Z_t = 2 \sum_j \sqrt{W_{+1}^j W_{-1}^j}$,

5. The weak classifier with the smallest Z is selected $Z_t = min\, Z, h_t = arg\, min\, Z$selected in this iteration,

6. Update the distribution $D_{t+1}(i) = D_t(i) exp[-y_i h_i(x_i)]$,

7. Strong classifier $H(x) = sign[\sum_{t=1}^{T} h_t(x) - b]$.

Through the implementation and parameter adjustment of the Real Adaboost algorithm, the prediction model obtained is compared with different values of the prediction model obtained based on the AdaBoost algorithm. It can be found that Real Adaboost will have a better prediction effect. See Figure 5.

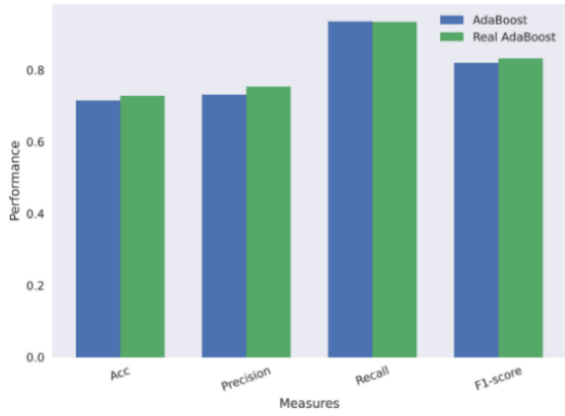

Figure 5: Comparison of reliability prediction model experiments.

The steps of reliability prediction based on the activity diagram are as follows: first, judge its basic control structure, mark the reliability information of each node in the activity diagram, simplify the activity diagram to make each path into a chain sequential structure, and generate test paths, mark the transition probability of each branch, and then calculate the reliability of each path through the reliability information of each node in the path. Finally, the system's reliability is further predicted and analyzed according to the branch transition probability. It can be seen from the definition of the UML activity diagram that the graphic elements in the activity diagram include control node elements such as branch node, merge node, fork node, and junction node in addition to activities, activity migration (action flow), start node, and end node. The standard basic control structures formed by these control nodes are as follows. See Figure 6.

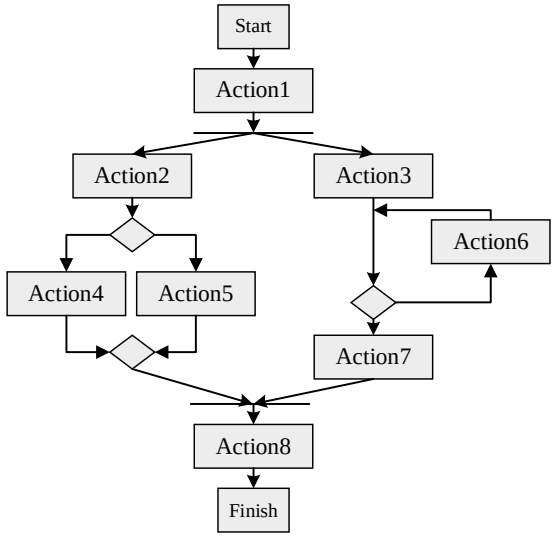

Figure 6: Typical activity chart.

Sequence structure: the execution of activity nodes in the activity diagram occurs in chronological order, and these nodes form an orderly sequence in chronological order. As in the above Action_ 1 to Action_ 8 (the bifurcated node to the confluent node are regarded as a whole). Selection structure: when an action is executed to a branch node in the activity diagram, one of the branches will be selected for execution. Each action flow leaving the branch node will have a monitoring condition, which indicates what conditions are met when the branch is executed. Predicting which branch the software will perform at a particular time in the running process is complex. Still, the probability of the program executing each branch can be set through investigation, calculation, or hypothesis, and the system's reliability can also be evaluated through these probabilities. As in the figure above, there is a selection structure between the branch node and the merge node in the left half.

Loop structure: A loop structure is a structure that contains a series of repeatedly executed activity sequences. Its modeling is implemented through a branch node. In the branches of this branch node, one branch continues to loop by connecting with the nodes in the region, and the other branches connect with the active nodes outside the area and exit the loop. Similarly, we cannot predict whether the program will enter the loop body or how many times the program will be executed in the loop body. Therefore, the execution probability of each branch needs to be marked

when predicting the system's reliability. As mentioned above, Action_ 3 to Action_ 7 is a circular knot.

Bifurcation structure: The appearance of a bifurcation node in the UML diagram indicates a concurrent control flow in the control flow. It divides the action flow into two or more concurrent action flows. Each bifurcated action flow is independent and running, and synchronization is achieved at the merging node. As in the figure above, a bifurcated structure exists between the bifurcated and confluent nodes (the intermediate selection and circular structure are considered a whole).

The probability of the activity transfer involved in the above structure can be set according to the following three conditions: (1) Assume that there are n action branches under the branch node, assuming that the probability of occurrence of these n events is the same, then the execution probability of each branch action flow is 1/n. (2) Historical data: collect historical data of old versions or similar functional software and calculate the average execution probability of each branch action flow. (3) Evaluation: By collecting user experience and suggestions or consulting experts in relevant fields with high academic levels and rich practical experience in relevant systems, we can infer and predict the transfer probability of each branch.

4 SE APPLICATION ANALYSIS

4.1 Specific Application

The management system is called MIS for short. It is mainly used to acquire and organize information in different fields and scientifically process and apply data information. It is also an essential modern tool. In MIS development, if there are defects in the system, it will have a significant impact on the system's efficiency. System analysis and logical model construction are generally related to data collection and business processes. By constructing a data flow chart and organization structure, the establishment of the model also lays a good foundation for software system development. The user business data processing process can be further simplified using data flow diagrams, and structural analysis plays an important role. At present, the internal organizational structure of some large institutions and departments is relatively complex, including different branches and business processes, which increases the difficulty of MIS system development. Relevant staff can make full use of SE technology methods to define the entity goals in business processes, establish structure diagrams, message diagrams, etc., in combination with specific characteristics, states, and relationships, create logical models based on the actual situation, view business processes related to the organization more intuitively, obtain data information, improve processing efficiency, and effectively remedy and improve the defects and deficiencies of structural methods. The guarantee system meets the actual needs of users to the maximum extent and also creates reliable and favorable data environment conditions for users' essential operations. See Figure 7.

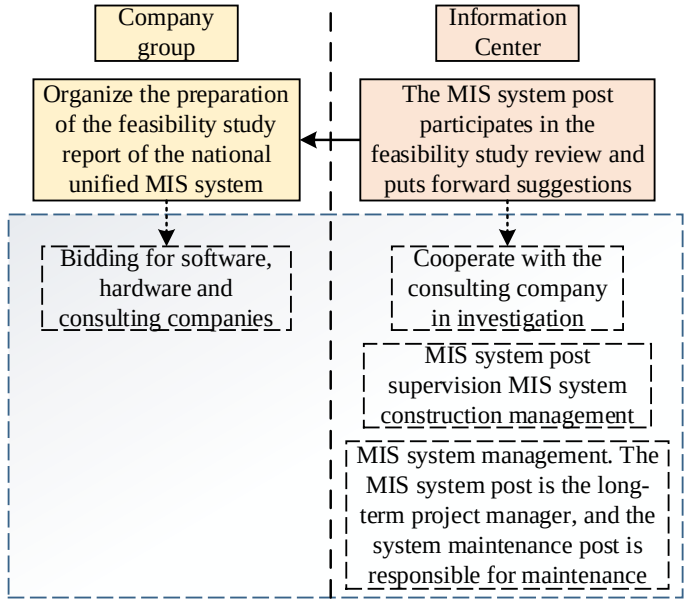

Figure 7: Development process of MIS software system.

4.2 Practical Application in the Development of CAI Software

Computer-aided instruction software is referred to as "CAI" for short. In the actual development work, determine the language and tools and then adopt modular design concepts or structural methods according to the development tasks and basic needs, which requires the staff to have professional experience. If the software requirements change, the development staff should debug and promptly correct the software system and related parameters to update the practical functions. With society's rapid development and progress, people's needs in all aspects are also increasing. This method has a certain complexity and a high bit error rate. Therefore, we must update the software system in time and meet the actual needs of users through constant modification. The software development completed by this method is usually expensive to maintain, so it is unsuitable for repeated use. The previously developed CAI software does not have some essential functions and cannot be effectively connected with the existing functional modules. Therefore, the development staff should reasonably use SE methods and attach great importance to CAI software development. Through various means, they should constantly improve the actual development efficiency and effectively solve and improve the problems and shortcomings of CAI software. See Figure 8.

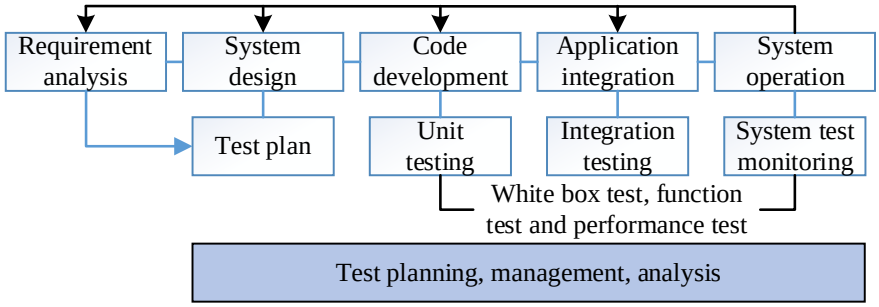

Figure 8: Flow chart of CAI software development.

4.3 Practical Application in the Development of Modern Medical Software

Computer software systems have been popularized in the catering and medical industries, becoming increasingly intelligent and information-based. In such an environment, scientific and reasonable application of computer software systems can significantly improve production and work efficiency and effectively coordinate and optimize the operating standards in the industry. In modern software development, relevant practitioners should gradually become aware of innovative development. For example, many hospitals have steadily increased the development of data management software related to patient pathological information; Doctors can directly obtain the pathological report information and timely transmit the analysis results to the software, effectively saving human resources and time costs. They can keep the patient's information and data for a long time for other clinicians to learn, provide a reliable reference for disease diagnosis and treatment, and observe the patient's condition in real-time. The actual operation of the software is also relatively simple, which can achieve a high-precision division of labor. However, in the actual development work, due to the influence of various factors, some software companies did not combine the basic needs of the hospital before development, the software did not give full play to its role and value, and there was a lack of effective communication and exchange between the software development staff and the hospital. Therefore, such software cannot effectively meet the actual development needs of the hospital, and it will also cause certain economic losses to the hospital.

4.4 Reliability Prediction Result Test

In the test, the activity diagram of the system administrator in a library management system to maintain the Account was selected. The administrator logs in to the account to manage it. There are three management methods to add new accounts, remove or update existing and query accounts, and exit the system after completing the management operation. Its activity diagram is in Figure 9.

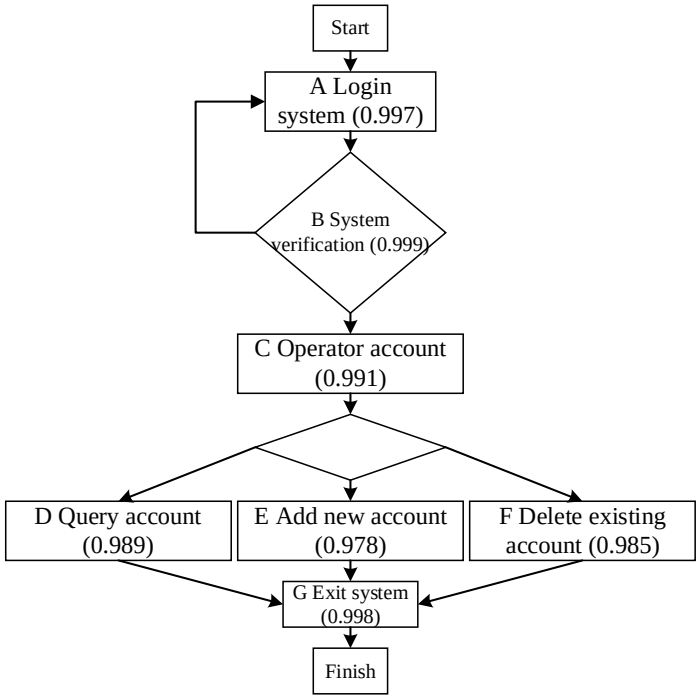

Figure 9: Activity chart of system management product maintenance account.

The transition probability of nodes and the reliability of each activity are given by known statistical data, as in Table 3.

Transfer out node	*Transfer in node*	*Transition probability*
A	*B*	*1.000*
B	*C*	*0.747*
B	*A*	*0.254*
C	*D*	*0.616*
C	*E*	*0.233*
C	*F*	*0.144*
D	*G*	*1.000*
E	*G*	*1.000*
F	*G*	*1.000*

Table 3: Transfer probability.

Table 4 shows the paths that can be generated by converting the loop and selection structures into sequence structures.

Route	*Probability of occurrence*

A-B-A-B-C-D-G	*0.116*
A-B-A-B-C-E-G	*0.043*
A-B-A-B-C-F-G	*0.026*
A-B-C-D-G	*0.462*
A-B-C-E-G	*0.178*
A-B-C-F-G	*0.107*

Table 4: Path occurrence probability.

According to the calculation of $R_s = \sum_{i=1}^{n} R_i P_i$, the reliability prediction result of this part of the system activity diagram is $R_s = 0.114 + 0.043 + 0.027 + 0.448 + 0.170 + 0.106 + 0.908$.

To show the practicability of this method, this paper uses the technique of randomly generating paths to conduct a comparative experiment and randomly simulates the activities of managing the Account. 50, 500, and 5000 test paths were randomly generated, and each case was affected 20 times. The reliability estimates of each simulation were calculated, and the average values of the three cases were 0.917, 0.901, and 0.910, respectively. The more paths are generated, the more negligible the reliability variance and the more significant the overhead. The reliability of the method adopted in this paper is very stable and can significantly improve the reliability evaluation efficiency of the system. We further investigated the extent to which people with different professional backgrounds recognized the effectiveness of software engineering education in computer software development engineering proposed in this paper. The specific results are shown in Figure 10.

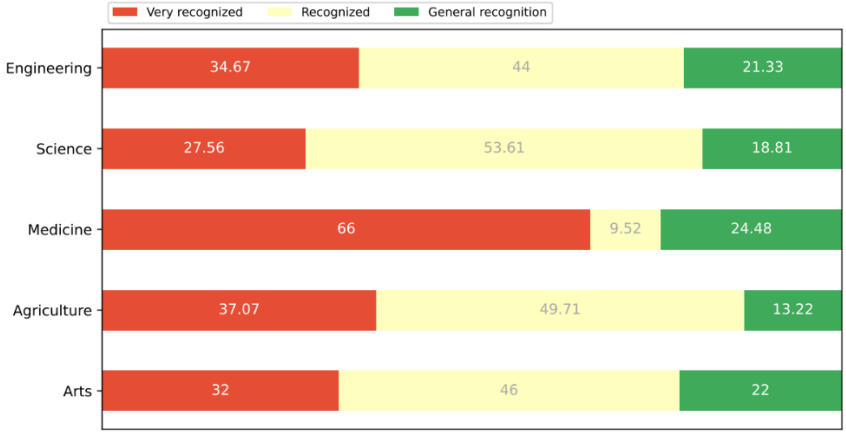

Figure 10: Degree of Utility Recognition of Different Majors.

5 CONCLUSIONS

The application of SE methods in computer software development under the background of big data should make good use of big data, deeply mine and analyze data, and effectively use data to help effectively use SE methods. The SE method can not be stuck in its way. It is necessary to strengthen

the use of big data, innovate application methods, effectively combine big data with software development methods, reasonably apply it in software service engineering, pay attention to the fourth paradigm of data-intensive scientific research, improve the application effect of SE methods in different fields, and improve the overall impact of computer software development. E-learning in software engineering education for big data amalgamates theoretical knowledge with practical application, fostering a new generation of adaptable, skilled professionals equipped to navigate the complexities of modern software development within the expansive realm of big data.

Yifan Huang, https://orcid.org/0009-0009-3717-7807
Yaqiong Wang, https://orcid.org/0009-0002-5743-3053
Pengcheng Wang, https://orcid.org/0009-0008-9829-6829

REFERENCES

[1] Adnan, N. H.; Ritzhaupt, A. D.: SE Design Principles Applied to Instructional Design: What Can We Learn from Our Sister Discipline?, TechTrends, 62(1), 2018, 77-94. https://doi.org/10.1007/s11528-017-0238-5

[2] Akbar, M. A.; Mahmood, S.; Alsalman, H.; Razzaq, A.; Gumaei, A.; Riaz, M. T.: Identification and Prioritization of Cloud Based Global Software Development Best Practices, IEEE Access, 8,2020, 191242-191262. https://doi.org/10.1109/ACCESS.2020.3031365

[3] Ali, Z.; Yaseen, M.; Ahmed, S.: Effective Communication as Critical Success Factor During Requirement Elicitation in Global Software Development, International Journal of Computer Science Engineering, 8(3),2019, 108-115.

[4] Alsubari, S. N.; Deshmukh, S. N.; Alqarni, A. A.; Alsharif, N., H.; T.: et al. Data Analytics for the Identification of Fake Reviews Using Supervised Learning, CMC-Computers, Materials & Continua, 70(2),2022, 3189–3204. https://doi.org/10.32604/cmc.2022.019625

[5] Anu, V.; Hu, W.; Carver, J. C.; Walia, G. S.; Bradshaw, G.: Development of a Human Error Taxonomy for Software Requirements: A Systematic Literature Review, Information and Software Technology, 103, 2018, 112-124. https://doi.org/10.1016/j.infsof.2018.06.011

[6] Barricelli, B. R.; Cassano, F.; Fogli, D.; Piccinno, A.: End-user Development,End-User Programming and End-User SE: A Systematic Mapping Study, Journal of Systems and Software, 149, 2019, 101-137. https://doi.org/10.1016/j.jss.2018.11.041

[7] Blincoe, K.; Springer, O.; Wrobel, M. R.: Perceptions of Gender Diversity's Impact on Mood in Software Development Teams, IEEE Software, 36(5), 2019, 51-56. https://doi.org/10.1109/MS.2019.2917428

[8] Ciancarini, P.; Farina, M.; Masyagin, S.; Succi, G.; Yermolaieva, S.; Zagvozkina, N.: Non Verbal Communication in SE–an Empirical Study, IEEE Access, 9, 2021, 71942-71953. https://doi.org/10.1109/ACCESS.2021.3075983

[9] Garcia, I.; Pacheco, C.; Méndez, F.; Calvo-Manzano, J. A.: The Effects of Game - Based Learning in the Acquisition of Soft Skills on Undergraduate SE Courses: A Systematic Literature Review, Computer Applications in Engineering Education, 28(5), 2020, 1327-1354. https://doi.org/10.1002/cae.22304

[10] Garousi, V.; Giray, G.; Tuzun, E.; Catal, C.; Felderer, M.: Closing the Gap Between SE Education and Industrial Needs, IEEE Software, 37(2), 2019, 68-77. https://doi.org/10.1109/MS.2018.2880823

[11] Gogoll, J.; Zuber, N.; Kacianka, S.; Greger, T.; Pretschner, A.; Nida-Rümelin, J.: Ethics in the Software Development Process: from Codes of Conduct to Ethical Deliberation, Philosophy & Technology, 34(4), 2021, 1085-1108. https://doi.org/10.1007/s13347-021-00451-w
[12] Hoda, R.; Salleh, N.; Grundy, J.: The Rise and Evolution of Agile Software Development, IEEE Software, 35(5), 2018, 58-63. https://doi.org/10.1109/MS.2018.290111318
[13] Kruglyk, V. S.; Osadchyi, V. V.: Developing Competency in Programming Among Future Software Engineers, Integration of Education, 23(4), 2019, 587-606. https://doi.org/10.15507/1991-9468.097.023.201904.587-606
[14] Liu, Qingfeng.; Liu, Chenxuan.; Wang, Yanan.: etc. Integrating External Dictionary Knowledge in Conference Scenarios The Field of Personalized Machine Translation Method, Journal of Chinese Informatics, 33(10),2019, 31-37.
[15] Lo, S. K.; Lu, Q.; Wang, C.; Paik, H. Y.; Zhu, L.: A Systematic Literature Review on Federated Machine Learning: From a SE Perspective, ACM Computing Surveys, 54(5), 2021, 1-39. https://doi.org/10.1145/3450288
[16] Maurício, R. D. A.;Veado, L.; Moreira, R. T.; Figueiredo, E.; Costa, H.: A Systematic Mapping Study on Game-Related Methods for SE Education, Information and Software Technology, 95, 2018, 201-218. https://doi.org/10.1016/j.infsof.2017.09.014
[17] Mezhuyev, V.; Al-Emran, M.; Fatehah, M.; Hong, N. C.: Factors Affecting the Metamodelling Acceptance: a Case Study from Software Development Companies in Malaysia, IEEE Access, 6, 49476-49485. https://doi.org/10.1109/ACCESS.2018.2867559
[18] Ozkaya, I.: Ethics is a Software Design Concern, IEEE Software, 36(3), 2018, 4-8. https://doi.org/10.1109/MS.2019.2902592
[19] Palanisamy, S.; Thangaraju, B.; Khalaf, O.I.; Alotaibi, Y.; Alghamdi, S.;Alassery, F.: A Novel Approach of Design and Analysis of a Hexagonal Fractal Antenna Array (HFAA) for Next-Generation Wireless Communication, Energies, 14, 2021, 6204. https://doi.org/10.3390/en14196204
[20] Raza, M.; Faria, J. P.; Salazar, R.: Assisting SE students in Analyzing their Performance in Software Development, Software Quality Journal, 27(3), 2019, 1209-1237. https://doi.org/10.1007/s11219-018-9433-7
[21] Robb, C. C.; Rahn, D.; Buffardi, K.: Bridging the Gap: A Model for Interdisciplinary Collaboration Between Entrepreneurship and SE Students, Journal of Education for Business, 95(5), 2020, 321-330. https://doi.org/10.1080/08832323.2019.1644275
[22] Saeed, S.; Jhanjhi, N. Z.; Naqvi, M.; Humayun, M.: Analysis of Software Development Methodologies, International Journal of Computing and Digital Systems, 8(5), 2019, 446-460. https://doi.org/10.12785/ijcds/080502
[23] Scaffidi, C.: Employers' Needs for Computer Science, Information Technology and SE Skills Among New Graduates, International Journal of Computer Science, Engineering and Information Technology, 8(1), 2018, 1-12. https://doi.org/10.5121/ijcseit.2018.8101
[24] Stol, K. J.; Fitzgerald, B.: The ABC of SE research, ACM Transactions on SE and Methodology 27(3), 2018, 1-51. https://doi.org/10.1145/3241743