

Research on Software Development Mode and Application of Reusable Framework Based on Neural Network Algorithm

Yingxi Wang

Intelligence Engineering School, Nanjing Institute of Railway Technology, Nanjing, Jiangsu, 210017, China

email: wangyxi@126.com

Keywords: Object Oriented, Software Reuse, Software Component, Software Reuse Waterfall Model, Neural Network

Abstract: The traditional software development method is difficult to support the early verification of requests, so it is easy to cause out of control projects. Among them, the idea of reusing the combination of object-oriented technology and software, as the design idea and method of domain name in engineering, the suggestion of reusing the object-oriented software development method. On this basis, the existing software development models are compared and analyzed, the object-oriented software reuse waterfall model is proposed, and several problems in the development process of application system based on the development model are discussed. This development mode can effectively shorten the software development cycle and reduce the burden of developers.

1. Introduction

Therefore, in the development of new generation software system, the biggest challenge we are facing is how to develop software quickly and high quality to meet various needs[1]. Traditional "freestyle" software development methods for specific needs have become weak at this time. The way to solve this problem is to establish a better software development environment, improve the possibility of system reconfiguration, and develop and use reusable software components. The best method, component-based architecture and software reuse method are effective methods to improve the stability, reliability and development efficiency of software system.

2. Processing Ability of Neural Network

"Artificial neural network" or "neural network", the information processing, memory and search functions of human neural system, to a certain extent, imitate the linear type composed of simple operation units (such as neurons), and the learning system has the function of knowledge processing. The storage and computation of N 51 neural networks have several important characteristics: they have nonlinear mapping ability; they do not need accurate mathematical model; they are good at learning useful knowledge from input and output data; they are easy to execute in parallel; because neural networks are composed of many computing units, they are easy to use[2]. Hardware implementation and more. Because neural network is a new processing mode which imitates the neural system structure of biology and has a unique structure, people expect to be able to solve problems that cannot be solved by existing methods. At present, the theory and application of neural network have been studied. Great development has penetrated almost all engineering applications. At present, the main application fields of artificial neural network are pattern recognition, fault detection, intelligent robot, linear system identification and control, market analysis, optimization, decision-making material movement, knowledge processing, cognitive science.

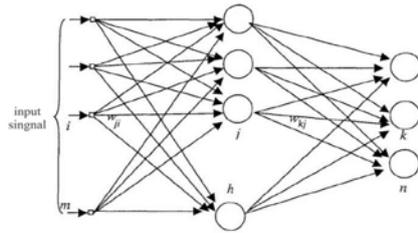


Figure 1 Schematic frame of neural network

3. Traditional Software System Development Methods

3.1 Prototype Method

The main idea of prototype method is to use prototype to support software development[3]. However, because the prototype method needs to carry out system analysis and function design for the whole system development process, the overall structure of the system is unclear, and the network design is not satisfactory. In the process of analysis and design of prototype method, the top-down and layer by layer decomposition of structural method is still going on, and the function and data have not been encapsulated.

3.2 Structured Approach

A structured approach centers on processes and functions. The system is divided into several modules. The connection between the modules is relatively close and the software can be reused[4]. The limitation of data flow oriented structural analysis is the lack of the ability to express the time relationship of data, which reflects that the data can not be expressed in specific data processing, that is, the interaction of system boundary cannot be expressed. The structural method is based on the concept of "process" rather than "object". Therefore, it is obviously unrealistic to develop a system using object-oriented design language. In the real world, "process" and "function" are often unstable. Once you change the functional requirements of the system, you must understand the structure of the system in order to solve the problems that occur to you.

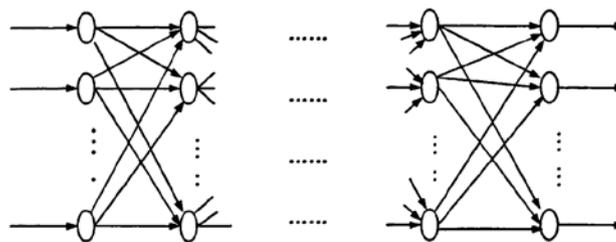


Figure 2 Structure model of BP neural network

3.3 Object Oriented Method

Compared with structured method and prototype method, object-oriented method is helpful for software reuse, and class and object are ideal reusable components. To some extent, the pure object-oriented method can cope with the change of user requirements, but it can not solve the following problems: the encapsulation part of the object is modified, then the relevant modules are recompiled in the network environment, and it is necessary to be recompiled[5]. Install each subsystem. OOP can only achieve class level reuse, which is too small to solve the problem of more efficient reuse and rapid system reconfiguration. OOP is limited to the fields of program source code, and its components are in the form of source code. Due to the lack of subordination standards, it is difficult to integrate rendering applications with different object-oriented languages and different types of libraries.

4. Object Oriented Software Reuse Development Method

From the point of view of reuse, the module of structured method is the functional unit of system specific service-oriented system [6]. Combined with process based instability, there is almost no possibility of reuse. The class and object of object-oriented method, the unit of reuse as too fine granularity, has the disadvantage of inheritance and the dependency between the objects. Therefore, it is restricted by independent conditions and reused. The key point is that the pure object-oriented approach is to take development as a new task that must be started completely from scratch, and not fully use the existing software resources. Therefore, the design and installation results are difficult to reuse in similar systems. Components are object-oriented, beyond the thinking of object-oriented. Components can be implemented in an object-oriented way to complete independent business logic. Implement object reuse. Component is the most important component in software reuse thinking. Therefore, the development trend of software engineering design thinking must be the organic combination of object-oriented method and software reuse thinking. It is a proposed object-oriented software reuse development method. The core content of this development method is based on the object-oriented development method. In system development, the whole construction process is guided by the idea of utilization. That is, on the other hand, it is relatively independent or has common elements. The logic with these functions and the constituent elements are packaged [7]. Finally, there is a large and small component library. In the future, the materials provided by the development of similar application systems in the same arena can be reused in the construction of the system. It is necessary to make full use of the design and development of the system. The object-oriented software reuse development method combines the advantages of the object-oriented method and the idea of software reuse. This clarifies the design ideas and methods of software system development in domain engineering and application engineering, which can effectively guide domain engineering. Domain information system development and other practical activities.

5. Object Oriented Software Reuse Waterfall Model

5.1 Analysis of Software Development Model

The software development model provides the relationship between each stage of software development activities. There are three types of software development models [8]. This is a waterfall model, completely based on software requirements. That is, in the initial stage of software development of prototype model and spiral fraction model, when basic conditions are provided, it can be used, for the basic development mode that does not support line by line development model and formal transformation method. At this point, the waterfall model is used as an integration of a new software development model for two reasons. Fully define the software requirements of the field to extract components with common factors from multiple different user requirements. As a result, the designed components and component libraries can build flexible software systems. On the other hand, waterfall model provides the basic framework of software development for personnel organization and management in the process of large-scale software development, which is helpful for the effective use of software development methods and tools. This improves the quality of large-scale software project development.

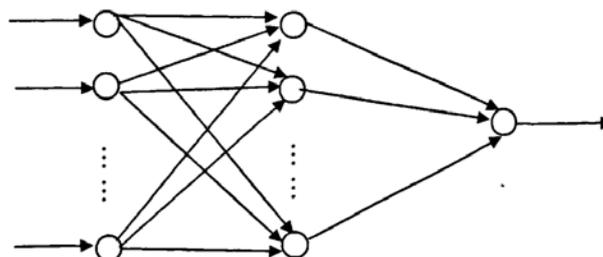


Figure 3 Structure of neural network for comprehensive evaluation index calculation

5.2 Domain System Design

Architecture is an important product in the process of software development and the main asset of system reuse. Component based software development must focus on architecture[9]. According to information system components and their related functions and characteristics, information system components are organized and classified into multiple levels. The configuration and operation of high-level software components depend on the structure and operation of low-level software components. This kind of software reuse can be realized in domain system design stage. Design reuse means that system architecture, software architecture, software design architecture, various regular design patterns and design methods and development specifications can be reused. The architecture provides a set of abstract configurations associated with the problem solving space. If you can reuse the architecture, you don't need to repeat the formula. Reuse of software architecture is the framework of software architecture determined by the software architecture of a specific domain. The design mode of the whole system, the dependence of coordination components, the distribution of responsibility and the control process are described. This is represented as a collection of abstract classes and methods that collaborate between instances. That's the foundation of component components. By using the architecture as the foundation of system construction and development, large-scale and organized software reuse can be realized. The software system design model can also be reused. This reuse is necessary when a software system needs to be moved to a completely different hardware and software environment. Subsystem reuse is a kind of design reuse[10]. This means that the subsystem of an application may be reused by different applications. In an object-oriented approach, you can use a wrapper graph to represent the dependencies between Java packages and software packages in an application. This is an abstract system structure that can be understood.

6. Conclusion

Using the idea of component to develop software is now the focus of research. Based on the research results of software development model based on reusable components, a series of application systems based on srwm model are constructed. The WYSIWYG functions of these systems support functions that are language independent, easy to operate, easy to maintain and reusable, which not only improves the development efficiency, but also greatly reduces the system maintenance cost. This means that developers can focus on business processes and business rules and remove these details. Our next work is to enrich the content of the model, improve its usability, and continue to expand its application.

Acknowledgements

This research has been financed by The Hight-Speed Railway Centralized Traffic Control and Maintenance System Project (KFJ1704) in 2017 of The Railway Transportation Control Engineering and Technology Study Center of JiangSu Province.

References

- [1] Yoshiaki Taniguchi, Koji Suganuma, Takaaki Deguchi,. Tandem Equipment Arranged Architecture with Exhaust Heat Reuse System for Software-Defined Data Center Infrastructure. *IEEE Transactions on Cloud Computing*, no. 99, pp. 182-192, 2017.
- [2] Msirdi S R. Modular Avionics Software Integration on Multi-Core COTS : certification-Compliant Methodology and Timing Analysis Metrics for Legacy Software Reuse in Modern Aerospace Systems, 2017.
- [3] Mirco Franzago, Davide Di Ruscio, Ivano Malavolta,. Collaborative Model-Driven Software Engineering: A Classification Framework and a Research Map. *IEEE Transactions on Software Engineering*, vol. 44, no. 12, pp. 1146-1175, 2018.

- [4] Shi-Kuo Chang, Jinpeng Zhou, Akhil Yendluri,. Research Notes — Software Process Selection Based Upon Abstract Machines for Slow Intelligence Systems. *International Journal of Software Engineering and Knowledge Engineering*, vol. 29, no. 7, pp. 1053-1067, 2019.
- [5] Juan Manuel Carrillo de Gea, Joaquín Nicolás, José L. Fernández I|lemán,. Automated support for reuse - based requirements engineering in global software engineering. *Journal of Software Evolution & Process*, 2017.
- [6] Fabien C. Y. Benureau, Nicolas P. Rougier. Re-run, Repeat, Reproduce, Reuse, Replicate: Transforming Code into Scientific Contributions. *Frontiers in Neuroinformatics*, vol. 11, 2017.
- [7] Magali Ruffier, Stephen M J Searle. Ensembl core software resources: Storage and programmatic access for DNA sequence and genome annotation. *Database the Journal of Biological Databases & Curation*, vol. 2017, no. 1, 2017.
- [8] Hamada Abd El - Wahab. Preparation and characterization of modified reclaimed asphalt using nanoemulsion acrylate terpolymer. *Pigment and Resin Technology*, ahead-of-print(ahead-of-print), 2019.
- [9] Silva M J D, Rodrigues F D Á, Júlio A A. SnF₂ -catalyzed glycerol ketalization: a friendly environmentally process to synthesize solketal at room temperature over on solid and reusable Lewis acid, vol. 307, pp. 828-835, 2017.
- [10] Loes M. J. Kroon-Batenburg, John R. Helliwell, Brian McMahon,. Raw diffraction data preservation and reuse: Overview, update on practicalities and metadata requirements. *Iucrj*, vol. 4, no. 1, 2017.