

Software Risk Assessment Method based on Fuzzy Neural Network

Wang Hua

Human Resource Operation Department
China-Power Information Technology Limited Company
Beijing, China
wanghua_mailbox@163.com

Wang Ben

Human Resource Operation Department
China-Power Information Technology Limited Company
Beijing, China
wangben@sgitg.sgcc.com.cn

You Longyong

Human Resource Operation Department
China-Power Information Technology Limited Company
Beijing, China
youlongyong@sgitg.sgcc.com.cn

Zhang Wentong

Human Resource Operation Department
China-Power Information Technology Limited Company
Beijing, China
zhangwentong@sgitg.sgcc.com.cn

Abstract—With the development and improvement of technology and the progress of the theory of project management, software risk management plays a more and more important role in modern project management. In order to enhance the accuracy and objectivity of the software risk assessment, we propose a software project risk assessment method in this paper. By building a software risk assessment model, using principal components analysis for dimensionality reduction of multiple risk factors, and the fuzzy neural network is used to predict the total risk of the project. The simulation results show that, the method can effectively evaluate the risk of software project, has strong objectivity in combining the expertise at the same time, it can provide effective support for the risk management of the project, improve the success rate of software projects.

Keywords—Software risk assessment; Principal components analysis; Assessment index; Fuzzy neural network

I. INTRODUCTION

In software development project, risk is the objective existence, which does not transfer with the will. As modern project management theory says, software risk management can be divided into 5 aspects, namely, risk identification, risk analysis, risk plan, risk tracking, risk control. Software risk assessment contains the following three processes: first, identify the known risks of software project, and then build a risk assessment model, finally completed risk analysis. A successful risk assessment can make organization avoid the risk factors, establish management plan, so the project can be completed successfully.

With the continuous development of software industry, the experts and scholars have gradually established a complete software project management theory. At the same time, the software risk assessment has also made a lot of research. Common software project risk assessment methods include expert scoring method, the analytic hierarchy process (AHP), fuzzy mathematics method and neural network method. As the traditional method for software risk assessment, expert scoring method has advantages to make full use of expert experience,

but the objectivity of scores depend on the ability and level of the experts[1]. At the same time, this method in the actual operation is time-consuming, so the field of software risk assessment urgently needs a method which can make full use of the experience of experts, and can objectively reflect the size of the risk. A method based on the combination of AHP and fuzzy logical method is put forward, which makes an improvement of them according to the actual situation of the software management. Fuzzy comprehensive assessment method bases on fuzzy mathematics, adopts fuzzy relation synthesis principle, gives a detailed description of their properties and possible consequences using the historical experience and expert knowledge, so as to quantify and rank various risk factors[2]. There is also a method based on neural network, which takes advantage of its nonlinear prediction ability to accomplish software risk assessment[5].

Neural network method cannot make use of expert experience, so we put forward a software risk assessment method using fuzzy neural network. Fuzzy neural network is a system based on the combination of fuzzy logic and neural network, which has the uncertain information processing capacity of fuzzy logic as well as the self-learning ability of neural network.

II. SOFTWARE RISK ASSESSMENT INDEX SYSTEM

The purpose of software risk management is to find out the possible risks in advance, and control the risk timely. Meanwhile, the purpose of software risk assessment is to predict the probability and impact of risks, to enable an organization to control risks more effectively.

According to the definition of Boehm, who is a risk management expert, risk refers to the possibility of loss or injury in the software risk management, commonly known as risk impact or risk exposure. Risk exposure can be defined by the following formula:

$$RE = P(UO) * L(UO) \quad (1)$$

RE represents the risk exposure value, P (UO) indicates the possibility of non expected result, L(UO) represents the extent of loss of software project caused by the occurrence of risks[3].

Software risk assessment refers to the assessment of risk caused by all the factors involved in software development. By risk identification and risk analysis, risk value of the project is achieved through comprehensive assessment. Software risk identification activities should be throughout the entire project. Software project risk assessment consists of 5 basic factors, respectively, for software process risk, software management risk, software technology risk, software environmental risk and software personnel risk. So we select 18 indexes in the 5 basic factors to establish the following model:

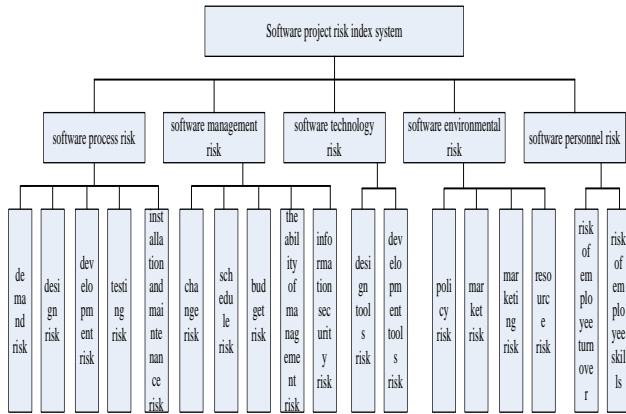


Fig.1. Software project risk index system

Among them, the software process risk include: demand risk, design risk, development risk, testing risk, installation and maintenance risk; the software management risk include: change risk, schedule risk, budget risk, the ability of management risk and information security risk; the software technology risk include: design tools risk and development tools risk; the software environmental risk include: policy risk, market risk, marketing risk and resource risk; the software personnel risk include: risk of employee turnover and risk of employee skills.

III. THE ASSESSMENT MODEL BASED ON FUZZY NEURAL NETWORK

Fuzzy neural network realizes the combination of fuzzy logic and neural network, the system has both the advantage of the ease of expertise expressing of fuzzy logic, and the advantage of distributed information memory and learning ability of neural network. So it is a milestone in the field of intelligent science, and it provides an effective way for the intelligent prediction and classification. An assessment model presented in this paper as follows: First, gain the index of risk factors of software risk assessment according to expertise. Then analyze the contribution of the risk factors for the whole project risk using principal components analysis. Finally has the results of factor analysis as the input of the fuzzy neural network, obtain generalization model by training, achieve the overall risk of the project by input risks need assessed[4]. The

flow chart of software risk assessment method based on fuzzy neural network as follows:

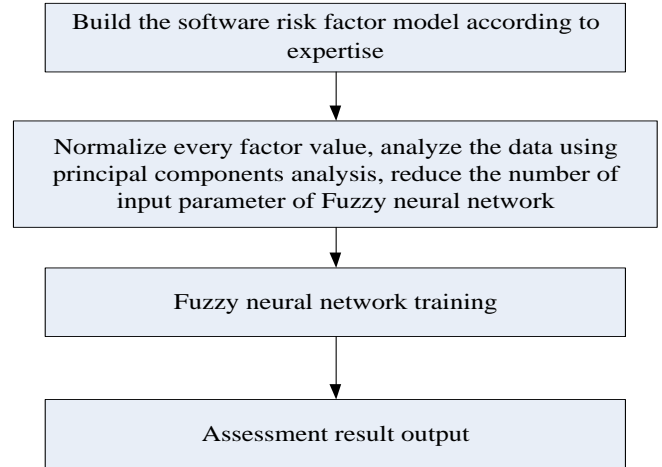


Fig.2. Flow chart of software risk assessment method based on fuzzy neural network

A. The principal components analysis

The main principle of the principal components analysis is to find a suitable linear transformation, which can transform related variables into independent variables. The new variables each with unique professional meaning, have larger variance can comprehensive response the main information of the original variables. In this paper, the principal components analysis is used for dimensionality reduction of data processing, and the meaning of original data can be represented by less dimension, so the input data dimensions of fuzzy neural network are reduced, which can enhance the performance of fuzzy neural network[7].

B. Adaptive neural fuzzy inference system

The typical framework of adaptive neural fuzzy inference system(ANFIS) as follows. In order to achieve the learning process of T-S fuzzy model, we transform it into an adaptive network, namely adaptive neural fuzzy inference system[6].

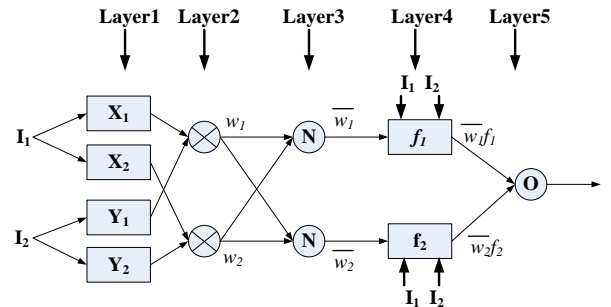


Fig.3. Adaptive neural fuzzy inference system

The nodes in the first and fourth layers of fuzzy neural network adjust their parameters through learning and training.

Layer1 is the membership function layer, nodes in this layer are responsible for the fuzzy of input signal.

$$O_i^1 = \mu_{xi}(I_1) \quad i = 1, 2 \quad (2)$$

or

$$O_i^1 = \mu_{yi}(I_2) \quad i = 1, 2$$

I_1 , I_2 are the input of node i , X_i , Y_i are the fuzzy rules set. O_i^1 is the output membership function values of the first layer, which represents the degree of I_1 , I_2 belongs to X_i , Y_i . The shape of membership function μ_{xi} and μ_{yi} is determined by the antecedent parameters.

Layer2 is the release layer of rule strength, which realize the multiplication of input signals.

$$O_i^2 = \mu_{xi}(I_1) * \mu_{yi}(I_2) \quad i = 1, 2 \quad (3)$$

The output of each node represents the credibility of the corresponding rule.

Layer3 normalizes all the rules strength, the i node calculates the credibility of the i rule.

$$\bar{O}_i^3 = \bar{\omega}_i = \omega_i / (\omega_1 + \omega_2) \quad i = 1, 2 \quad (4)$$

Layer 4 calculates the output of fuzzy rules. Each node in this layer is an adaptive node, the output of this layer is:

$$O_i^4 = \bar{\omega}_i f_i = \bar{\omega}_i (p_i x + q_i y + r_i) \quad i = 1, 2 \quad (5)$$

$\bar{\omega}_i$ is the output of the third layer, $\{p_i, q_i, r_i\}$ is the parameter set of nodes.

Layer 5 is a fixed node to calculate the total output of all input signals:

$$O_i^5 = \sum \bar{\omega}_i f_i = \sum \omega_i f_i / \sum \omega_i \quad i = 1, 2 \quad (6)$$

Given antecedent parameters, the output of the adaptive neural fuzzy inference system can be expressed as follows.

$$\begin{aligned} O_i^5 &= \bar{\omega}_1 f_1 + \bar{\omega}_2 f_2 \\ &= (\bar{\omega}_1 x) p_1 + (\bar{\omega}_1 y) q_1 + (\bar{\omega}_1) r_1 + (\bar{\omega}_2 x) p_2 + (\bar{\omega}_2 y) q_2 + (\bar{\omega}_2) r_2 \end{aligned} \quad (7)$$

The learning and training of ANFIS algorithm in this paper using hybrid algorithm of the least square method and BP algorithm, to adjust the parameters of ANFIS system.

In the network forward learning process, N sets of training data used to get parameter $\{p_i, q_i, r_i\}$ and output O_i^5 , the least square method used to calculate the error value between the calculated value and the expected value of training data, and the error value is reversed back. The antecedent parameters are modified according to the maximum gradient method, at the same time adjust the membership function figure, to achieve the minimum error value.

IV. SIMULATION AND ANALYSIS

In this paper, we use software project risk management data in the project document library, to form risk control table as follows:

TABLE I. RISK CONTROL TABLE

Project	P1	P2	P3	P4	P5	P6	P7	P8
R1	0.2	0.8	0.7	0.2	0.3	0.2	0.2	0.3
R2	0.2	0.6	0.3	0.3	0.1	0.4	0.7	0.5
R3	0.2	0.1	0.2	0.1	0.1	0.2	0.4	0.8
R4	0.2	0.4	0.1	0.5	0.2	0.4	0.7	0.3
R5	0.3	0.6	0.3	0.1	0.2	0.8	0.2	0.6
R6	0.1	0.3	0.1	0.5	0.1	0.7	0.7	0.4
R7	0.2	0.8	0.4	0.5	0.3	0.7	0.3	0.4
R8	0.2	0.6	0.6	0.2	0.1	0.3	0.1	0.1
R9	0.5	0.7	0.1	0.3	0.1	0.2	0.8	0.3
R10	0.1	0.2	0.2	0.3	0.2	0.3	0.7	0.1
R11	0.2	0.1	0.4	0.2	0.3	0.4	0.7	0.6
R12	0.1	0.4	0.1	0.3	0.1	0.3	0.4	0.4
R13	0.1	0.7	0.1	0.3	0.3	0.5	0.8	0.5
R14	0.1	0.1	0.3	0.6	0.2	0.2	0.6	0.2
R15	0.3	0.4	0.3	0.2	0.3	0.3	0.4	0.2
R16	0.2	0.3	0.1	0.3	0.2	0.3	0.1	0.3
R17	0.1	0.2	0.2	0.1	0.4	0.2	0.2	0.5
R18	0.3	0.1	0.3	0.3	0.2	0.5	0.4	0.2
Project Risk	0.12	0.72	0.37	0.44	0.23	0.56	0.73	0.55

Using SPSS(Statistical Product and Service Solutions) for principal components analysis, we get 4 dimension input fuzzy neural network. The component matrix of noise reduction and dimension reduction as follows:

TABLE II. COMPONENT MATRIX

Component matrix				
	Component			
	1	2	3	4
R1	-.265	.741	-.269	.415
R2	.862	.379	.019	.261
R3	.354	-.168	.754	.406
R4	.935	-.058	-.161	-.201
R5	.133	.653	.500	-.149
R6	.886	-.041	.126	-.405
R7	.218	.840	.021	-.414
R8	-.266	.722	-.424	.070
R9	.660	.242	-.336	.262
R10	.823	-.304	-.418	.056
R11	.579	-.458	.357	.409
R12	.836	.383	.297	-.042
R13	.875	.359	.088	.148
R14	.560	-.531	-.301	-.159
R15	.350	.362	-.610	.404
R16	.001	.487	.535	-.607
R17	-.046	-.016	.790	.457
R18	.366	-.460	-.133	-.445

Then, we establish fuzzy neural network model using ANFIS in MATLAB toolbox according to risk factors of the four dimension. The membership function uses Gaussian function, the training rate of the fuzzy neural network set to 0.01, the number of training set to 100, the training goal set to 1×10^{-6} . Following figure shows the structure of fuzzy neural network.

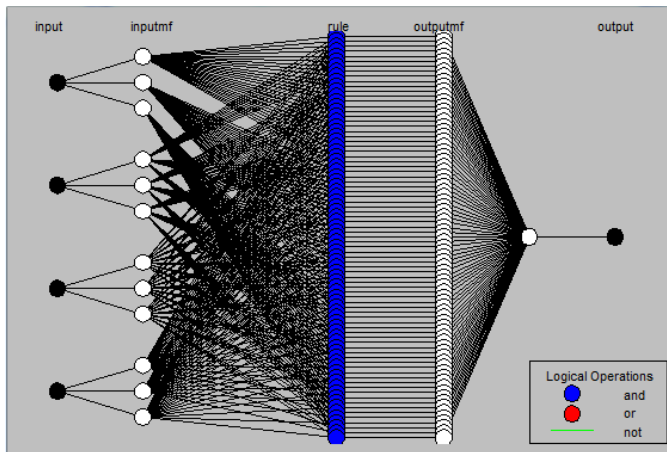


Fig.4. The structure of fuzzy neural network

After training for 100 times, curve graph is achieved as follows:

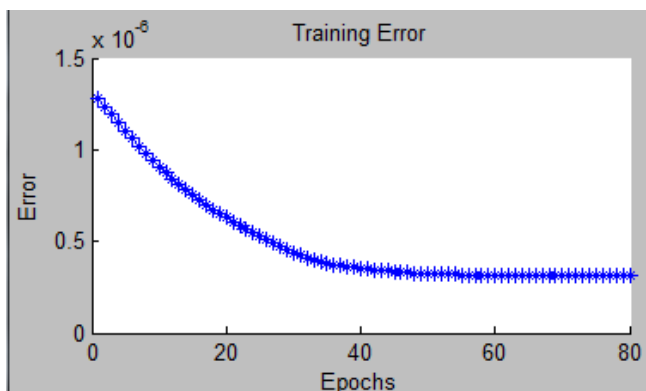


Fig.5. Training error curve

According to the forecast of the 7 groups test data, we get the average prediction error of this software project, which is 0.0189. The result indicates that the software risk assessment method based on fuzzy neural network has a good

generalization ability, and it is a useful attempt in the field of software risk assessment.

V. TAG

Software risk is one of the important problems of enterprises or organizations in the software project management to face. Scientific assessment of software risk will help enterprises to avoid the risk, increase the success rate of software projects. First, we analyze the results of previous studies, select 5 basic risk factors as software process risk, software management risk, software technology risk, software environmental risk and software personnel risk, and propose the software risk assessment index system. Then analyze the contribution of the risk factors for the whole project risk using principal components analysis. Finally has the results of factor analysis as the input of the fuzzy neural network, obtain generalization model by training, achieve the overall risk of the project by input risks need assessed. The fact proves that the model has good adaptability, through the historical data training, can be used in software project risk assessment.

REFERENCES

- [1] Zhang Luoling, Li Shixian, "Comparison and Research on Models of Software Project Risk Management," Computer Engineering, pp. 91-94, March 2003.
- [2] Pan Chunguang, Chen Yingwu, Wang Hao, "Overview of the study on theories and methods of software project risk mnagement," Control and Decision, pp.481-486, May 2007.
- [3] Zhao Dongmei, "A Model of Risk Assessment of Software Project and its Application," Computer Engineering and Applications, pp. 192-194, 2006.
- [4] Tang Aiguo, Wang Rulong, "Software Project Risk Assessmet Model Based on Bayesian Networks," Computer Engineering, pp. 91-93, November 2008.
- [5] Ju Yanbing, Feng Yuncheng, Yao Ligang, "Research on the Measure of Risk in the Course of Software Development," Systems Engineering-Theory Methodology Application, pp. 218-223, September 2004.
- [6] Sun Zengqi, Xu Hongbin, "Fuzzy Neural Network Based on T-S Model," Journal of Tsinghua University(Science and Technology), pp. 76-80, 1997.
- [7] Li Jinghua, Guo Yaohuang, "Principal Components Assessment," Journal of Industrial Engineering and Engineering Management, pp. 39-43, Vol.16, No.1 2002.