International Journal of Knowledge-Based Development

ISSN online: 2040-4476 - ISSN print: 2040-4468
https://www.inderscience.com/ijkbd

Quality evaluation of software engineering professional talent training under the background of new engineering

Lijuan Liu, Yong Bai, Shaowei Zhang

DOI: 10.1504/IJKBD.2025.10068649

Article History:

Received:	21 April 2023
Last revised:	23 September 2024
Accepted:	06 November 2024
Published online:	01 April 2025

Quality evaluation of software engineering professional talent training under the background of new engineering

Lijuan Liu* and Yong Bai

School of Artificial Intelligence,
Neijiang Normal University,
Neijiang, 641112, China
Email: liulijuan@njtc.edu.cn
Email: cqbaiyong@163.com
*Corresponding author

Shaowei Zhang

Department of Computer Engineering,
Anhui Wenda University of Information Engineering,
Hefei, 231201, China
Email: swzhang189@163.com

Abstract: In order to address the issues of low recall rate, long clustering time, and low accuracy in the quality assessment of traditional software engineering talent cultivation methods, a new quality evaluation method of software engineering professional talent training under the background of new engineering is proposed. The intrinsic dependency relationship among the evaluation indicators of software engineering talent cultivation quality is analysed in depth using factor analysis, and a talent cultivation quality assessment indicator system is constructed. Indicator data is collected. The ant colony clustering algorithm is used to cluster the collected data, and the processed data is inputted into the talent cultivation quality assessment model based on fuzzy comprehensive evaluation to obtain relevant assessment results. The experimental results showed that the recall rate of this method is between 95% and 99%, the average clustering time of indicators is 7.75 s, and the maximum accuracy rate is 97%.

Keywords: new engineering; software engineering professional; talent training; quality evaluation; collect index data; ant colony clustering algorithm; fuzzy comprehensive evaluation.

Reference to this paper should be made as follows: Liu, L., Bai, Y. and Zhang, S. (2025) 'Quality evaluation of software engineering professional talent training under the background of new engineering', *Int. J. Knowledge-Based Development*, Vol. 15, No. 1, pp.24–41.

Biographical notes: Lijuan Liu received her Master's degree in Software Engineering from University of Electronic Science and Technology; received her PhD in Intelligent Information Integration from Makuwon University in South Korea. She is a Professor in the School of Artificial Intelligence of Neijiang Normal University. Her research interests include data mining, intelligent fusion and big data analysis.

Yong Bai is an Associate Professor. He graduated from the Mokwon University, Korea in 2022, major in Intelligent Information Technology, with a PhD. Now, he is working in Neijiang Normal University. He mainly engages in the research of big data, cloud computing and intelligent information processing. So far, he has published 13 papers and participated in 19 projects.

ShaoWei Zhang received her MS in Computer Application from Dalian University of Technology, DaLian, China, in 2013. She is currently a Teacher at Anhui Wenda University of Information Engineering, Hefei, China. She has authored or co-authored over 20 refereed journal and the book. Her current research interests include sensor network and intelligent algorithm.

1 Introduction

With the continuous improvement of technology and internet technology, the demand for software engineering professionals has skyrocketed. Therefore, it is necessary to innovate the training mode of software engineering professionals in this context, in order to cultivate more high-quality software engineering talents for the country and society (Jian, 2018). In order to effectively achieve this goal, relevant universities need to strengthen the in-depth analysis of new engineering concepts, in order to change talent cultivation concepts, layout in advance, innovate software engineering talent cultivation models, and cultivate high-quality skilled talents that society urgently needs (Jiang, 2023; Ciancarini et al., 2021). The evaluation of talent cultivation quality refers to the process of monitoring or evaluating higher education institutions, vocational education institutions, or other talent cultivation institutions to evaluate their talent cultivation quality, aiming to promote the continuous improvement of the quality and effectiveness of talent cultivation. The main research content of the quality evaluation of software engineering professional talent cultivation includes the evaluation of teaching, practical teaching, student quality, teaching staff, and practical ability cultivation, measures and methods to improve teaching quality, research on indicators, standards, and evaluation systems for talent cultivation quality evaluation, etc. The research significance of evaluating the quality of talent cultivation lies in promoting the close integration of universities and society, optimising talent cultivation goals, plans, teaching methods, and educational resource allocation, improving the competitiveness of graduates in employment, enhancing the international competitiveness of talent cultivation institutions, and promoting the reform and development of higher education. In addition, the evaluation of talent cultivation quality can also provide scientific basis for government decision-making and play an important role in improving the comprehensive strength and international competitiveness of a country. Moreover, the quality evaluation of talent cultivation is an important component of education management and an important means for society to supervise and macro regulate the quality of talent cultivation in universities. Therefore, conducting in-depth research on the quality evaluation of talent cultivation in software engineering is of great significance.

The issue of evaluating the quality of software engineering talent cultivation has also seen many excellent research results in the current field. For example, Zeng (2022) proposed a multi index fuzzy judgement method for evaluating the quality of software engineering talent cultivation. This method mainly uses questionnaire surveys and visiting surveys to obtain quality evaluation indicators for software engineering talent cultivation, and screens the evaluation indicators to establish a quality evaluation system for software engineering talent cultivation. On this basis, calculate the credibility and weight of the quality evaluation of software engineering talent cultivation, and use relevant evaluation models to obtain relevant evaluation results. However, in practical application, it is found that the recall rate of cluster analysis results, the evaluation index of this method, is low. Xie and Song (2021) proposed a method for evaluating the quality of software engineering talent cultivation based on factor analysis. This method is guided by multiple principles such as scientificity, comprehensiveness, and representativeness. It uses a combination of reference material method and questionnaire survey method to select appropriate evaluation indicators, and uses educational evaluation theory to build a quality evaluation system for software engineering professional talent cultivation. The analytic hierarchy process is used to determine the weight of the indicators, and the indicator data is input into the evaluation model constructed by combining the weight of the indicators, obtain relevant evaluation results. However, this method has the problem of time-consuming clustering of evaluation indicators, and there is still a certain gap between it and the expected goals. Hu and Xiong (2021) proposed a quality evaluation method for software engineering talent cultivation based on cloud databases. This method screens the data of software engineering professional talent cultivation quality evaluation indicators in the cloud database of the ministry of education, assigns weights and scores to different evaluation indicators, constructs a software engineering professional talent cultivation quality evaluation model, and inputs relevant data into the model to achieve talent cultivation quality evaluation. However, in practical applications, it has been found that this method has the problem of low evaluation accuracy, and there is still a certain gap between it and the expected goals. Hu (2020) proposed a talent training quality evaluation method using fuzzy comprehensive evaluation method. The evaluation index system for the quality of innovative and entrepreneurial talents training in Colleges and universities is constructed, which is composed of four first-class indicators, namely, environment, teaching links, teachers and students. The factor set of evaluation objects and the grade domain of comprehensive evaluation are determined. Then AHP is used to determine the weight of evaluation index, and expert scoring method is used to determine the single factor fuzzy comprehensive evaluation matrix of each level. According to the evaluation matrix, the fuzzy relationship between the evaluation object and the evaluation set is calculated. According to the maximum membership principle, the evaluation grade corresponding to the maximum value in the fuzzy relation set is calculated as the final evaluation result of the evaluation object. However, this method has the problem of low evaluation accuracy, and there is still a certain gap with the expected goal. Hua and Tao (2022) proposed a skill based talent quality evaluation algorithm based on a deep belief network model. Establish an evaluation set consisting of 4 primary indicators and 14 secondary indicators, calculate corresponding weights, and complete the construction of the evaluation indicator system. A DBN structure consisting of multiple RBM and BP networks was constructed. A quality evaluation algorithm based on DBN has been designed. Algorithm training is used to evaluate test data and output evaluation levels.

However, during testing, it was found that this method has a low recall rate, which affects the accuracy of subsequent evaluations.

Under the background of new engineering, the purpose of evaluating the quality of talent cultivation in software engineering is to improve the quality and effectiveness of talent cultivation. Due to the rapid development and continuous technological updates in the field of software engineering, the traditional talent training model faces problems such as a disconnect between teaching content and industry needs, a lack of practical opportunities and environments, a lack of diverse training approaches, a single assessment method, a lack of industry cooperation and practical cases, and an imperfect evaluation system. Therefore, the importance of evaluating the quality of talent cultivation in software engineering is increasingly prominent. The relevant research results can help schools and training institutions understand whether their current training programs are effective, and adjust course content and teaching methods in a timely manner to better cultivate software engineering professionals with practical abilities and innovative spirit. However, traditional methods for evaluating the quality of talent cultivation in software engineering have problems such as low recall rate, long clustering time, and low evaluation accuracy, a quality evaluation method of software engineering professional talent training under the background of new engineering has been designed with the research goal of solving various problems existing in the above methods. The main contents of this method are as follows:

1 Based on the principles of comprehensiveness, simplicity, operability, and combination of commonness and individuality, the quality evaluation index of software engineering talent training is preliminarily determined. The internal dependence relationship between each evaluation index is deeply analysed by factor analysis method, so as to achieve the index data screening, so as to build the relevant talent training quality evaluation index system and collect the evaluation index data.

2 The ant colony algorithm was used to improve the k-means clustering algorithm, and the ant colony clustering algorithm was used to cluster the collected evaluation index data. The fuzzy comprehensive evaluation is used to build a software engineering talent training quality evaluation model. The evaluation index data is taken as the model input and the talent training quality evaluation results are taken as the output to get the relevant evaluation results.

3 The recall ratio, clustering time and evaluation accuracy were used as evaluation indexes to verify the practical application effect of the method.

4 Summarise the entire paper, analyse the limitations of the research and future prospects.

2 Design of quality evaluation method for software engineering talent training

2.1 Building a quality evaluation system for software engineering professional talent training

The principles to be followed in the construction of the quality evaluation system for software engineering talent cultivation are as follows:

1 *The principle of comprehensiveness*. Comprehensiveness mainly has the following two meanings: Firstly, in the design of the indicator system, not only should it consider reflecting the level of talent cultivation work in the software engineering profession, but also comprehensively reflecting various aspects of talent cultivation concepts, models, teaching, scientific research, management, hardware construction, etc. in the software engineering profession; The second point is that in the process of designing quality evaluation indicators for software engineering talent cultivation, both qualitative and quantitative indicators should be considered.

2 *The principles of simplicity and operability*. On the one hand, the quality of talent cultivation in software engineering is influenced and constrained by various factors; On the other hand, the uncertainty and complexity of education. If all factors are taken into account, it will inevitably make the indicator system too complex, thereby affecting its operability. Therefore, in the design of the quality evaluation index system for software engineering talent cultivation, it is necessary to ensure that the quality evaluation index for talent cultivation has operability.

3 *The principle of combining commonality and individuality*. Regardless of the type of education, there must be certain commonalities. Therefore, various types of universities have common standards for the quality of talent cultivation in software engineering. However, due to the different types of talents cultivated, the quality standards for software engineering talent cultivation in different universities also have their own characteristics. Therefore, in the process of constructing an evaluation index system, it is necessary to consider the design of common indicators for higher education and highlight the characteristic indicators of this type of university.

Based on the above principles, a preliminary evaluation index for the quality of software engineering talent cultivation needs to be determined. Next, factor analysis (Wu and Luo, 2022; Chou et al., 2022) needs to be used to conduct in-depth analysis of the internal dependencies between each evaluation index, while ensuring the loss of as little original variable information as possible. Starting from the correlation matrix of the original variables, the truly relevant variables should be identified and classified into a category, ensure that the evaluation indicators have a close correlation with the quality of software engineering professional talent cultivation in the end. The general model for factor analysis is as follows:

$$\begin{cases} x_1 = a_{11}f_1 + a_{12}f_2 + \cdots + a_{1m}f_m + \varepsilon_1 \\ x_2 = a_{21}f_1 + a_{22}f_2 + \cdots + a_{2m}f_m + \varepsilon_2 \\ \vdots \\ x_p = a_{p1}f_1 + a_{p2}f_2 + \cdots + a_{pm}f_m + \varepsilon_p \end{cases} \tag{1}$$

In the above formula, $x_1, x_2, \ldots, x_p$ is the standardised sample observation variable for software engineering talent training evaluation indicators, $a_{11}, \ldots, a_{pm}$ represents factor load, $f_1, f_2, \ldots, f_m$ represents common factor, and $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_p$ represents a special factor.

The commonality of evaluation indicator variables represents the ratio of the total variance expressed by common factors to the original variable (Aghamolaei and Shojaee, 2022; Fu et al., 2022), and its calculation formula is as follows:

$$h_i^2 = \sum_{j=1}^{m} a_{ij}^2 x_p \tag{2}$$

The variance contribution rate of common factors is an indicator to verify the relative importance of common factors, and the higher this value, the greater the relative importance of common factors. The calculation formula is as follows:

$$g_i^2 = \sum_{i=1}^{p} a_{ij}^2 \tag{3}$$

The rationality analysis results of the quality evaluation indicators for software engineering talent cultivation are as follows:

$$K = \frac{\sum\sum_{\mathrm{j}\neq\mathrm{i}} g_i^2}{\sum\sum_{\mathrm{j}\neq\mathrm{i}} h_i^2 + \sum\sum_{\mathrm{j}\neq\mathrm{i}} a_{ij}^2} \tag{4}$$

In the above formula, g_i represents the common factor weight coefficient, and $K >$ represents that the evaluation index is very reasonable; $0.8 < K < 0.9$ is reasonable; $0.7 < K < 0.8$ is general; $0.5 < K < 0.7$ is not very reasonable and quite reluctant; $K < 0.5$ is unreasonable.

Establish a relevant evaluation index system based on the screening results of software engineering talent cultivation quality evaluation indicators using factor analysis method, as shown in Table 1.

Table 1 Quality evaluation index system for software engineering talent training

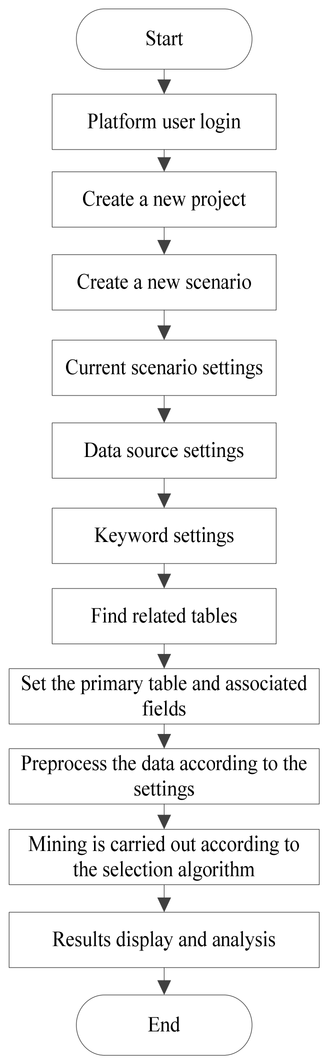
Primary indicators	*Secondary indicators*
Discipline concept	Discipline positioning
	Educational philosophy
Educational mode	School enterprise cooperation
Talent cultivation mode	Work-integrated learning
Construction of teaching staff	Quantity and structure
	Quality and construction
Teaching conditions and utilisation	Basic teaching facilities
	On campus experimental and practical training conditions
	Off campus training base
	Teaching expenses
Teaching construction and reform	Professional courses
	Technical application ability training
Teaching management	Management team
	Quality assurance system
Scientific research work	Discipline construction
	Research team
	Scientific research projects and achievements

Table 1 Quality evaluation index system for software engineering talent training (continued)

Primary indicators	*Secondary indicators*
Educational effect	Basic theory and skills
	Comprehensive quality
	Graduation thesis or graduation project
	Employment and social reputation

Based on the evaluation index system constructed above, collect data on the quality evaluation index of software engineering professional talent cultivation. The specific process is shown in Figure 1.

Figure 1 Data collection process for quality evaluation indicators of software engineering professional talent training



The process for collecting quality evaluation indicators for software engineering talent cultivation is as follows:

1 Users log in to the platform and create a new collection project and plan for the quality evaluation indicators of software engineering professional talent cultivation.

2 Then, set data preprocessing parameters and data mining algorithms based on the characteristics of the quality evaluation index data for software engineering talent cultivation.

3 To find data related to data mining, users also need to set keywords, such as table and field names, or users can customise keywords. Based on keywords, the platform searches and lists all tables related to keywords. Users can choose from these tables and delete tables that they consider irrelevant. The platform merges the tables selected by the user into a new real table based on foreign keys and other relationships.

4 According to the parameters set by the user, the platform preprocesses the data for the new table. If necessary, users can manually preprocess the data on the table. After the preprocessing is completed, the platform conducts data mining on the processed software engineering professional talent training quality evaluation index data based on the algorithm selected by the user, and displays the mining results in text form.

In summary, this paper is mainly based on the principles of comprehensiveness, simplicity, operability, and the combination of commonality and individuality to preliminarily determine the quality evaluation indicators for software engineering talent cultivation. Factor analysis is used to deeply analyse the internal dependencies between each evaluation indicator, establish a relevant talent cultivation quality evaluation indicator system, and design a data collection process for software engineering talent cultivation quality evaluation indicators, collect relevant evaluation indicator data to lay a solid foundation for subsequent analysis.

2.2 Cluster software engineering professional talent training quality evaluation data

Based on the evaluation index system and data collection results constructed in Section 2.1, the ant colony clustering algorithm is used to cluster the quality evaluation data of software engineering professional talent cultivation. The clustering results are used as an important basis for the quality evaluation of software engineering professional talent cultivation.

The k-means clustering algorithm is named after the fact that each cluster requires the calculation of the average value of the objects within that cluster. It is one of the most classic partition based clustering algorithms (Wang et al., 2022; Prakash et al., 2022). The increasing scale of data in the age of big data highlights the insufficient computing power of k-means clustering algorithm, which brings a research upsurge of parallel clustering algorithm represented by swarm intelligence clustering algorithm. Ant colony clustering algorithm is one of the promising swarm intelligence clustering algorithms. Therefore, this paper uses this algorithm to cluster the quality evaluation data of training software engineering professionals.

Under the application of k-means clustering, the square error of the clustering results is calculated using the following formula:

$$E = \sum_{i=1}^{k} \left| x - m_i \right|^2 \tag{5}$$

In the above formula, x represents the sample point and m_i represents the mean of cluster c_i.

Select k objects from the software engineering talent training quality evaluation dataset as the initial cluster center, set the iteration counter $t = 0$, so as to allocate the sample points to the cluster nearest to the cluster center, and calculate the cluster analysis objective function $J(t)$ of the software engineering talent training quality evaluation (Li et al., 2021; Shrifan et al., 2021). When the following formula holds, recalculate each cluster center:

$$r = J(t) - J(t-1) \tag{6}$$

In the above formula, $J(t-1)$ and $J(t)$ represent the $t-1$th and *t*th clustering objective functions, respectively. In the case of $t = t+1$, output the clustering center to obtain the clustering result.

Combining ant colony algorithm to optimise k-means clustering algorithm, an ant colony clustering algorithm was designed, in which the probability of one ant picking up an object is calculated using the following formula:

$$P_P = \left(\frac{k_1}{f + k_1} \right)^2 \tag{7}$$

In the above formula, f represents the perception factor of objects around ants, and k_1 represents the threshold parameter. The probability of ants dropping objects is calculated using the following formula:

$$P_d = \left(\frac{f}{f + k_2} \right)^2 \tag{8}$$

In the above formula, k_2 represents a threshold parameter that is completely different from k_1. When $f \ll k_1$ and P_d approaches 0, the probability of ants dropping an object is very low. When $f \gg k_2$ and P_P is close to 1, the probability of ants dropping the loaded object is high (Wang et al., 2022; Dang et al., 2021).

Assuming that the weighted Euclidean distance between the data objects X_i to X_j of the training quality evaluation of software engineering professionals is expressed as d_{ij}, and the pheromone and cluster radius on this path are expressed as τ_{ij} and r respectively, the following formula holds:

$$\tau_{ij}(t) = \begin{cases} 1 d_{ij} \leq r \\ 0 d_{ij} > r \end{cases} \tag{9}$$

The probability of merging X_i into X_j is calculated using the following formula:

$$p_{ij}(t)=\frac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{seS}\tau_{sj}^{\alpha}(t)\eta_{sj}^{\beta}(t)} \tag{10}$$

In the above formula, α and β represent different combination probabilities, η_{ij} represent visibility, and τ_{sj} and η_{sj} represent different pheromone concentration characteristic parameters. When and only when $p_{ij}(t)$ is greater than the threshold of p_0, X_i is merged into X_j.

The BM model simulates the behaviour of two ants meeting for clustering. After the two ants meet, they accept each other and merge their nests to meet the following requirements:

$$Acceptance(i,j)\Leftrightarrow\left(Sim(i,j)>Template_i\right)\Lambda\left(Sim(i,j)>Template_j\right) \tag{11}$$

In the above formula, $Sim(i,j)$ is the similarity between object *i* and object *j* in the quality evaluation data of software engineering talent cultivation. $\mathrm{Template_i}$ and $\mathrm{Template_j}$ represent different coefficient of difference. The calculation formula for $\mathrm{Template_i}$ is as follows:

$$Template_i\leftarrow\frac{Sim_{avg}(i,.)+Max\left(Sim(i,.)\right)}{2} \tag{12}$$

In the above formula, $Max\left(Sim(i,.)\right)$ and $Sim_{avg}(i,.$ represent the maximum and average similarity between ant *i* and other ants.

The cluster analysis process of software engineering talent training quality evaluation indicators is as follows:

Step 1: When two nest free ants meet, create a new nest.

Step 2: Nestless ants and nesting ants meet and return the nestless ants to the other nest.

Step 3: When ants a and b meet in the same nest, if they can accept each other, increase the values of M_i and M_i^+. If they can accept each other, decrease the values of M_i and M_i^+, and move the ants with smaller values of M_i^+ out of the nest.

Step 4: Two ants from different nests meet and accept each other, merging their nests.

Step 5: If none of the above situations occur, no action will be taken.

The final cluster analysis result of software engineering professional training quality evaluation can be obtained after iterative algorithm according to the above rules.

The dataset validation process for the clustering results of quality evaluation indicators for software engineering talent cultivation and the replication of clustering results is as follows:

Step 1: Prepare the validation dataset: Randomly select a portion of the actual software engineering talent training data as the validation dataset.

Step 2: Apply the same clustering algorithm: Input the validation dataset into the same ant colony clustering algorithm, and use the same parameters and settings for clustering processing.

Step 3: Compare clustering results: Compare the clustering results of the validation dataset with the original clustering results to evaluate the consistency and reliability of the clustering. Some evaluation indicators such as contour coefficient and Davies Bouldin index can be used to measure the quality of clustering results.

The verification process of replicating clustering results is as follows

Step 1: Obtaining Existing Clustering Results: Obtain the dataset that has already been clustered and the corresponding clustering results, ensuring that the copied clustering results are based on existing clustering operations.

Step 2: Cluster using the same method: Using the same clustering algorithm, parameters, and settings, input the existing dataset into the algorithm for clustering.

Step 3: Compare replicated clustering results: Compare the replicated clustering results with existing clustering results to evaluate their consistency and similarity. The same evaluation metrics can be used to measure the quality and consistency of clustering results.

The purpose of these validation processes is to ensure the credibility and stability of clustering results. By verifying different datasets and replicating clustering results, the effectiveness of clustering algorithms can be more reliably evaluated, and a reasonable explanation of the quality evaluation index data for software engineering talent cultivation can be obtained.

In summary, by using ant colony algorithm to improve the traditional k-means clustering algorithm, the ant colony clustering algorithm is used to cluster the quality evaluation data of talent cultivation in software engineering, providing an important data foundation for subsequent analysis.

2.3 Implementation process of quality evaluation for software engineering talent cultivation

Based on the above cluster analysis results of software engineering talent training quality evaluation, a software engineering talent training quality evaluation model is built using fuzzy comprehensive evaluation. The evaluation index data is used as the model input, and the talent training quality evaluation results are used as the output to obtain the relevant evaluation results.

This paper uses the fuzzy comprehensive evaluation method to evaluate the quality of software engineering talent cultivation. The specific implementation steps are as follows:

1 Determine the set of fuzzy comprehensive evaluation factors

$$U = \{u_1, u_2, ...u_i...u_m\}(i = 1, 2, ..., m) \tag{13}$$

2 Establish a comprehensive evaluation set

$$V = \{v_1, v_2, ...v_j...v_n\}(j = 1, 2, ..., n) \tag{14}$$

3 Construct evaluation matrix

Assuming that the single factor for evaluating the quality of talent cultivation in software engineering is represented by $u_1\left(i=1,2,\ldots,m\right)$, and the membership of factor u_i to decision level $v_j\left(j=1,2,\ldots,n\right)$ is r_{ij}, the single factor evaluation set of factor $\mathrm{u_i}$ is represented by the following formula:

$$r_i=\left(r_{i1},r_{i2},\ldots,r_{in}\right) \tag{15}$$

If there is a set of factors for evaluating the quality of software engineering talent cultivation, $U=\left\{u_1,u_2,u_3,L,u_m\right\}$, $V=\left\{v_1,v_2,v_3,L,v_n\right\}$ is a set of decision comments, R is a fuzzy mapping between U and V, and $r_{ij}\left(i=1,2,...,m;j=1,2,...,n\right)$ describes the results of the *j*th factor of the evaluated teacher starting from the *i*th factor. The following fuzzy evaluation matrix is established:

$$R=\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{bmatrix} \tag{16}$$

4 Determine weight

Taking the weighted arithmetic mean value of *R* as the comprehensive ranking vector $w=\left(w_1,w_2,...,w_n\right)^T$, the following formula holds:

$$\begin{cases} w_j=\lambda_1 w_{j1}+\lambda_2 w_{j2}+\cdots+\lambda_s w_{js}\, j=1,2,...,n \\ \sum_{k=1}^{s}\lambda_k=1 \end{cases} \tag{17}$$

When $\lambda_1=\lambda_2=...\lambda_s$, the weight calculation of the quality evaluation index for software engineering talent cultivation is as follows:

$$w_j=\frac{1}{s}\left(w_{j1}+w_{j2}+w_{js}\right) j=1,2,...,n \tag{18}$$

5 Perform fuzzy synthesis and make decisions

$$B=A\cdot R \tag{19}$$

In the above formula, '·' is a fuzzy operator, and *1* represents the decision matrix. The fuzzy decision function for evaluating the quality of talent cultivation in software engineering is as follows:

$$b_j=\sum_{i=1}^{m}a_i r_{ij} \tag{20}$$

In the above formula, a_i represents the decision parameter and r_{ij} represents the ambiguity.

Based on the above analysis, the quality evaluation model for software engineering talent cultivation is as follows:

$$S = V \times B = v_1^* b_1 + v_2^* b_2 + v_3^* b_3 + \cdots + v_j^* b_j \tag{21}$$

In summary, this paper mainly establishes a software engineering professional talent training quality evaluation model using fuzzy comprehensive evaluation by determining the set of fuzzy comprehensive evaluation factors, establishing an evaluation set for comprehensive evaluation, constructing a evaluation matrix, determining weights, conducting fuzzy synthesis, and making decisions. The evaluation index data is used as the model input, and the talent training quality evaluation results are used as the output, implement quality evaluation of software engineering talent cultivation.

3 Experimental design and result analysis

In order to verify the effectiveness of the quality evaluation method of software engineering professional talent training under the background of new engineering designed in this paper, relevant experimental designs were carried out. The specific experimental plan is as follows:

1 *Experimental data*: Taking software engineering students from 10 comprehensive universities in China as the research subjects, with a student population of 40,000 to 70,000, randomly select one class of software engineering students from each university for experimental testing, with a student population of approximately 30–50 in each class. During the testing process, relevant courses will be conducted uniformly. After one year of conducting relevant courses, data related to software engineering students will be collected, and the collected data will be filtered and cleaned. The processed data will be used as experimental sample data.

2 Evaluation indicators: Zeng (2022) method, Xie and Song (2021) method and this method are used as experimental comparison methods, and the actual application effects of different methods are tested by comparing the cluster analysis recall rate, clustering time consumption and evaluation accuracy of different methods of software engineering professional training quality evaluation indicators. Among them, the index cluster analysis recall refers to the ratio of clustering results to all experimental samples. The higher the index, the more comprehensive the cluster analysis results will be; Index cluster analysis time is an important index to test the clustering efficiency. The shorter the clustering time is, the higher the efficiency is; The higher the evaluation accuracy value, the higher the evaluation accuracy.

The comparison results of the cluster analysis recall rate of the three methods for software engineering professional training quality evaluation indicators are shown in Figure 2.

An analysis of the results in Figure 2 shows that the clustering recall rate of evaluation index data of the method in this paper varies between 95–99%, that of Zeng (2022) method varies between 75–87%, and that of Xie and Song (2021) method varies between 67–84%. In summary, the clustering recall rate of software engineering professional talent cultivation quality evaluation index data of the method in this paper is higher, and the change of the recall rate is more stable, indicating that the data clustering results of the method in this paper are more comprehensive, the clustering process is more stable, and the clustering performance is due to the experimental comparison method.

Figure 2 Comparison results of cluster analysis recall rate of evaluation indicators (see online version for colours)

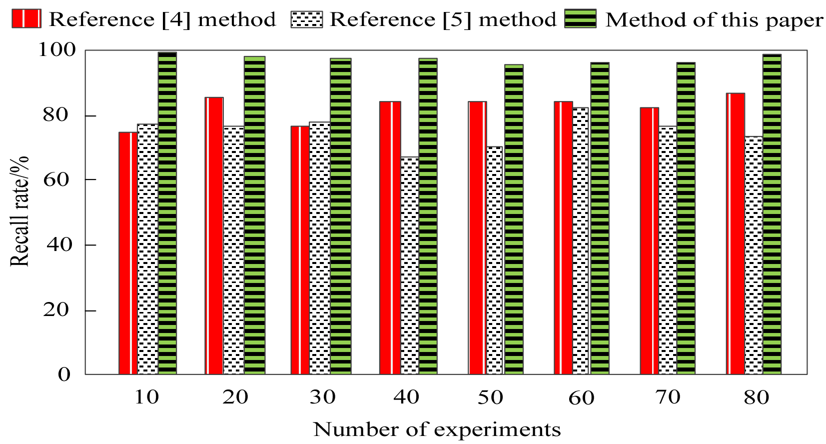


Table 2 shows the time consuming comparison results of cluster analysis of the three methods for evaluating the training quality of software engineering professionals.

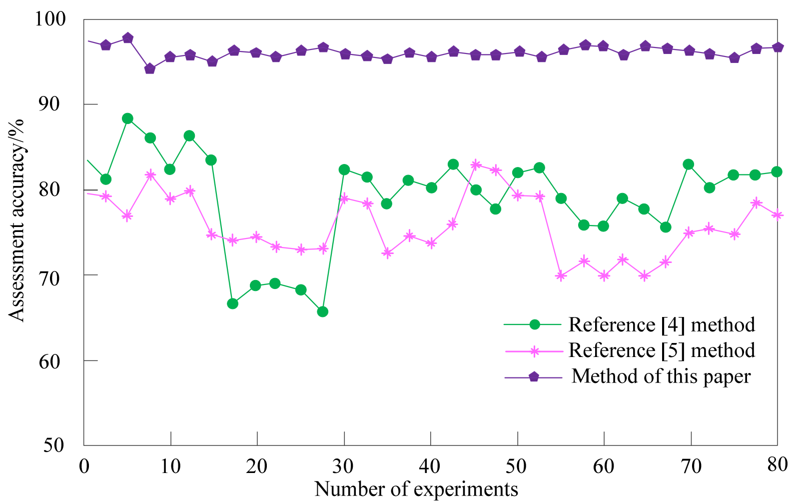
Table 2 Time consuming comparison results of cluster analysis of evaluation indicators

	Clustering time consumption/s		
Number of experiments	*Zeng (2022) method*	*Xie and Song (2021) method*	*Method of this paper*
10	86.32	35.68	9.63
20	85.47	35.74	9.58
30	84.61	32.47	6.33
40	81.43	31.49	7.54
50	82.66	36.99	7.66
60	85.24	35.84	7.21
70	84.78	30.17	7.45
80	86.32	35.87	7.36
90	79.89	34.71	7.14
100	81.63	39.88	7.55
Average value	83.84	34.88	7.75

After analysis and calculation, it can be seen that the maximum, average and minimum clustering time of evaluation index data in this paper is 9.63 s, 7.75 s and 6.33 s. The maximum, average and minimum clustering time of evaluation index data in Zeng (2022) method was 39.88 s, 34.88 s and 30.17 s. The maximum, average and minimum clustering time of evaluation index data in Xie and Song (2021) method was 86.32s, 83.84 s and 79.89 s. In terms of the maximum, average and minimum time of evaluating index data clustering, the clustering time of the method in this paper is the shortest, indicating that the clustering efficiency of index data in this method is higher.

The comparison results of the accuracy of the quality evaluation of software engineering talent cultivation using three methods are shown in Figure 3.

Figure 3 Comparison of accuracy of talent cultivation quality evaluation results (see online version for colours)



The analysis of the results in Figure 3 shows that, with the increase of experiment times, the accuracy curves of the three methods all show a trend of significant fluctuation. Among them, the maximum and minimum accuracy of talent training quality assessment of this method are 97% and 94%, and the accuracy curve shows a stable fluctuation trend. The maximum and minimum accuracy of talent training quality assessment in Zeng (2022) method is 88% and 65%, and the accuracy curve shows a significant fluctuation trend. The maximum and minimum accuracy of talent training quality assessment in Xie and Song (2021) method is 83% and 70%, and the accuracy curve shows a significant fluctuation trend. In summary, the method presented in this paper is more accurate and stable, and can achieve accurate assessment of the training quality of software engineering professionals.

In summary, the recall rate of the quality evaluation index data for software engineering talent cultivation in this paper's method is between 95–99%, the average clustering time of the index is 7.75 s, and the maximum accuracy is 97%. High recall rate, reasonable clustering time, and high accuracy have been demonstrated in the quality evaluation of talent cultivation in software engineering. It has certain effectiveness and reliability in the quality evaluation of talent cultivation in software engineering.

1 *High recall rate*: A recall rate of 95–99% means that this method can cover all aspects of the quality evaluation of software engineering talent cultivation well, without missing important indicators. This indicates that the method has good sensitivity and comprehensiveness in evaluating high-quality talent cultivation.

2 *Reasonable clustering time*: The clustering time with an average of 7.75 s indicates that this method has good efficiency in processing indicator data. This result shows that the method has good performance in quickly obtaining clustering results and is suitable for processing large-scale data.

3 *High accuracy*: The maximum accuracy of 97% indicates that this method can accurately cluster and fuzzy comprehensive evaluate evaluation indicators. High accuracy means that this method can better distinguish different evaluation results and provide objective evaluation basis for talent cultivation.

4 Conclusion

1 At present, there are problems with low recall rate, long clustering time, and low evaluation accuracy in the evaluation methods of talent cultivation quality. To solve the problems of existing methods, a new method for evaluating the quality of software engineering talent cultivation in the context of new engineering is proposed as the research goal.

2 Through experiments, it has been proven that the clustering recall rate of the evaluation index data of this method varies between 95% and 99%. The average clustering time of the evaluation index data is 7.75 s, and the maximum accuracy rate of talent cultivation quality evaluation is 97%. Compared with other experimental methods, this method has multiple characteristics such as good clustering effect and high evaluation accuracy. It can optimise the talent cultivation plan for software engineering majors, aiming to cultivate more cutting-edge talents in the industry.

3 In the context of the new engineering discipline, the quality evaluation method for talent cultivation in software engineering has had a positive impact on deepening educational reform, improving educational quality, and promoting the comprehensive development of students. It encourages educators to pay more attention to practical ability cultivation, comprehensive quality development, and innovative educational methods, while guiding students to actively participate in learning, adapt to the needs of the new era, and cultivate software engineering professionals who can adapt to future social development.

4 However, this paper also has certain limitations, that is, factor analysis assumes that the evaluation indicators are linearly correlated, and this assumption may not be applicable to evaluation indicators in specific situations. If there are non-linear relationships or complex interactions between indicators, factor analysis may not be able to fully describe their intrinsic dependencies. Therefore, the next step is to improve this method to adapt to more complex evaluation indicators.

Acknowledgements

This work was supported by the first batch of new liberal arts research and reform practice projects in Sichuan Province under grant no.127, and the 2023 Education and Teaching Reform Research Project of Neijiang Normal University under grant no. GJPY2317.

References

Aghamolaei, A.S.H. and Shojaee, S. (2022) 'Complex Fourier-based stress intensity factor analysis of plane elasticity using boundary element theories', *International Journal for Numerical Methods in Engineering*, Vol. 123, No. 5, pp.1360–1380.

Chou, H., Lin, D. and Ishida, T. (2022) 'Understanding open collaboration of Wikipedia good articles with factor analysis', *Journal of Information and Knowledge Management*, Vol. 23, No. 6, pp.223–235.

Ciancarini, P., Farina, M., Masyagin, S., Shojaee, S. and Isa, N. (2021) 'Non verbal communication in software engineering – an empirical study', *IEEE Access*, Vol. 9, No. 1, pp.71942–71953.

Dang, X., Gong, D., Yao, X., Xu, F. and Yang, K.L. (2021) 'Enhancement of mutation testing via fuzzy clustering and multi-population genetic algorithm', *IEEE Transactions on Software Engineering*, Vol. 48, No. 6, pp.2141–2156.

Fu, Y., Wen, Z. and Wang, Y. (2022) 'A comparison of reliability estimation based on confirmatory factor analysis and exploratory structural equation models', *Educational and Psychological Measurement*, Vol. 82, No. 2, pp.205–224.

Hu, N.L.C. (2020) 'Quality evaluation of practical training of innovative and entrepreneurial talents in universities based on statistical learning theory after COVID-19 epidemic', *Journal of Intelligent and Fuzzy Systems: Applications in Engineering and Technology*, Vol. 39, No. 6Pta2, pp.9045–9051.

Hu, X.G. and Xiong, X. (2021) 'Construction of a personalized talent training evaluation model based on cloud database', *Journal of Wenzhou Polytechnic*, Vol. 21, No. 1, pp.52–56, 61.

Hua, J. and Tao, S. (2022) 'Evaluation algorithm of skilled talents' quality based on deep belief network model', *Journal of Interconnection Networks*, Vol. 22, Supp.02, pp.2143035–2143046.

Jian, D. (2018) 'Study on talent training mode of software engineering based on CDIO background', *International Journal of New Developments in Engineering and Society*, Vol. 2, No. 2, pp.55–57.

Jiang, C. (2023) 'Research on talent cultivation of software engineering under the background of integration of industry and education', *SHS Web of Conferences*, Vol. 166, p.01031, https://doi.org/10.1051/shsconf/20231660103

Li, X., Zhu, J., Shi, H., Chen, M. and Du, Q. (2021) 'Surface defect detection of seals based on K-means clustering algorithm and particle swarm optimization', *Scientific Programming*, Vol. 21, Pt. 9, pp.3965247.1–3965247.12.

Prakash, L., Suryanarayana, G. and Ansari, M.D. (2022) 'Instantaneous approach for evaluating the initial centers in the agricultural databases using K-means clustering algorithm', *Journal of Mobile Multimedia*, Vol. 18, No. 1, pp.43–59.

Shrifan, N., Akbar, M.F. and Isa, N. (2021) 'An adaptive outlier removal aided K-means clustering algorithm', *Journal of King Saud University – Computer and Information Sciences*, Vol. 34, No. 4, pp.6365–6376.

Wang, C.H., Liu, X.L., Han, Y.J., Liu, D. and Zhang, M. (2022) 'An improved genetic algorithm based annulus□sector clustering routing protocol for wireless sensor networks', *Wireless Personal Communications: An Internaional Journal*, Vol. 123, No. 4, pp.3623–3644.

Wang, X., Shen, A., Hou, X., He, K. and Chen, S.C. (2022) 'Research on cluster system distribution of traditional fort-type settlements in Shaanxi based on K-means clustering algorithm', *PLoS ONE*, Vol. 17, No. 3, pp.e0264238–e0264259.

Wu, D. and Luo, X. (2022) 'Robust latent factor analysis for precise representation of high-dimensional and sparse data', *IEEE/CAA Journal of Automatica Sinica*, Vol. 8, No. 4, pp.796–805.

Xie, Y. and Song, Y.Y. (2021) 'Research on the construction of the evaluation index system of the quality of physical education professional training in colleges and universities', *Journal of Huaibei Normal University(Natural Sciences)*, Vol. 35, No. 1, pp.89–96.

Zeng, W.Y. (2022) 'Research on modern apprenticeship talent quality evaluation system based on multi-index fuzzy judgment method', *Education*, Vol. 13, No. 7, pp.217–219.