[1]Mengzi Zhang

[2]Xiao Chen

[3]Yue Jin

[4]Xiaocheng Zhou

[5]Shaowei Zhang*

Talent Cultivation Quality of Software Engineering Majors Based on Deep Learning

Abstract: - Online learning, to cultivate talents it is inevitable to encounter some pictures or videos with poor visual quality. Deep-learning algorithms are both data-hungry and expensive to compute. These algorithms work better after being trained on a broad and extensive collection of samples. The current moment deep learning methods must urgently make use of human intellect to address the issue in a way that reduces the most expensive effort computationally. This paper analyzes the current situation of software engineering talent cultivation quality of software engineering to enhance the quality of the education is improved by Hierarchically Gated Recurrent Neural Network (HGRNN). The aim of the work is to foster the development of world-class software engineering talents. Initially, the input data's are gathered from public dataset train 400 with 400 grey pictures. HGRNN is image de-noising module, as for the smart teaching platform to assist instructors in obtaining teaching photography with high quality and improve teaching quality. The proposed model is implemented in MATLAB/ Simulink platform and the accuracy is compared to various existing approaches such Back Propagation Network (BPN), Artificial Neural Network (ANN) and Decision Tree Algorithm (DTA) our proposed method obtains 98% of accuracy.

Keywords: Software Engineering, Talent Cultivation, Smart Teaching, Teaching Mode, Projects, De-noising, Teaching Quality.

I. INTRODUCTION

Building a theoretical and practical education system that satisfies the demands of the software industry is necessary to cultivate talent in software engineering [1]. It also entails developing a faculty of teachers from a variety of backgrounds and devising a system that integrates learning and production. Formal training, leadership development, succession planning, on-the-job training, and more are all included in mentoring and coaching. To begin with, the majority of these school-based teaching quality assurance systems are macro-level systems, [2, 3] and it is evidently unreasonable to implement the same quality assurance procedures across disciplines [4]. Some universities and colleges have the unilateral belief that the teaching quality guarantee system is merely a means of monitoring the teaching process, which stifles the teachers' and students' enthusiasm and creativity [5] Simultaneously, some universities and colleges only concentrate on building redundant systems while ignoring the management of the process of improving teaching quality, which has the effect of lowering teacher and student participation.

Graduation design, professional and fundamental software engineering courses, and public general courses are typically included in traditional software engineering training [6, 7]. While this type of education can help students gain more specialized knowledge, it does not meet society's need for creative and practical skills [8, 9]. There are currently two main inconsistencies in Chinese higher education. Students' practical software engineering skills are lacking. The current college education model has the first discrepancy between learning in a group and growing personally; all students take the same course and are trained in accordance with the same standards [10]. Similar to assembly line products that are unable to satisfy the demands of society for a diversity of talents, these results in students' incapacity to develop in accordance with their own interests [11, 12].

In this paper, we examine the specific development requirements for software engineering in higher education institutions against the backdrop of new engineering and offer solutions for developing talent in line with those needs [13, 14].

[1]Professor, Department of Computer Engineering, Anhui Wenda University of Information Engineering, Hefei, Anhui, 231201, China
[2]Teacher, Department of Computer Engineering, Anhui Wenda University of Information Engineering, Hefei, Anhui, 231201, China
[3]Professor, Department of Computer Engineering, Anhui Wenda University of Information Engineering, Hefei, Anhui, 231201, China
[4]Teacher, Department of Computer Engineering, Anhui Wenda University of Information Engineering, Hefei, Anhui, 231201, China
[5]Teacher, Department of Computer Engineering, Anhui Wenda University of Information Engineering, Hefei, Anhui, 231201, China*
[5]Email: phdshaoweizhang@gmail.com*
Copyright © JES 2024 on-line: journal.esrgroups.org

Our design includes a module for image de-noising and a Hierarchically Gated Recurrent Neural Network (HGRNN) that can be integrated into an intelligent teaching platform. Notably, this suggested HGRNN can help instructors acquire high-quality teaching photography and enhance the quality of instruction [15, 16].The framework of a top-notch talent-training program, as well as the development of a curriculum group that adheres closely to the core knowledge of the profession [17 -19]. The issue of gradient disappearance and promote the reuse of noise features, thereby enhancing the effects of noise removal.

The main contribution of this paper is,

- By making recommendations for pertinent materials, classes, and learning pathways based on each student's needs and development, HGRNNs can customize learning environments. As a result, the learning environment might become more dynamic and productive.
- HGRNNs can analyze real-time student data and feedback to recommend adjustments to the curriculum, ensuring it remains relevant to industry needs and student interests. This could lead to more efficient and effective learning experiences.
- HGRNNs rely on high-quality educational data. Assist teachers to obtain teaching photography with high quality and improves teaching quality.
- HGRNNs for talent cultivation in software engineering hold promise for improving educational effectiveness and efficiency.

Rest of these manuscripts is organized as follows: The literature review is reviewed in Part 2, the methodology is explained in Part 3, the results are proven in Part 4, and the document is concluded in Part 5.

II. RELATED WORK

Several works were suggested in the literature related to deep learning based talent cultivation quality of software engineering majors, a few recent works are reviewed here,

Han et al. [20] have suggested that the study gains knowledge of deep learning, neural networks, creative talent traits, and training models associated considering the seasonal variation Model of Autoregressive Moving Average. Autoregressive seasonal integrated moving average regression back propagation was suggested using these ideas. First, the SARIMA-BP evaluates the model's superiority, versatility, and speed of convergence while artificially setting the weight parameter values. After that, the model underwent pre-processing, and the independence of the particle swarm optimization (PSO) algorithm. Additionally, wavelet de-noising was used to test and optimize the established mode.

Bai and Huang [21] have developed that the ability of ANNs to handle nonlinear information adaptively allows for the assessment of the integration impact between high-quality education and skill-based education. Prior to the merging of the two models, the issues and incentive systems were the subject of an investigation. For the integration impact, an evaluation index system was then developed. Next, an ANN model for the evaluation index compatibility was created in order to forecast the integration effect.

B-Fernández et al. [22] have suggested that using technology to teach and learn improves academic quality and, in turn, advances or advances society as a whole. Consequently, there has been a dearth of study on tools that support kids in drawing conclusions from their own past grades. This essay offers a methodology wherein data collection and pre-processing were done first, and then students with comparable academic performance patterns were grouped in a subsequent stage. The most suitable supervised learning algorithm, the Decision Tree algorithm, was chosen in the following phase based on the patterns that had been found, and the experiment was then run.

Ding [23] have utilized that Domestic engineering colleges have gradually come to honor China's CDIO engineering education model, which is another name for the software engineering education paradigm. This article will go over how colleges and universities teach software engineering using the CDIO engineering education paradigm using the goal of improving the quality of engineering talent in China. Using University A's software engineering major as an example, a questionnaire survey covering course design, instruction, and skill development was given to major students. It was determined that problem-driven learning strategies and project-based curriculum design techniques significantly enhance students' skills.

Zhang et al. [24] smart manufacturing (SM) was lacking in the industrial engineering (IE) majors' existing talent training model, and there was a dearth of hands-on learning opportunities for SM environments. Because of this, it was challenging for students pursuing traditional degrees in IE to adjust to the current industrial intelligence trend and fulfill the demands of enterprise development and the market. Consequently, IE majors now have an urgent problem to solve: how to develop IE talents for SM. This research proposes a novel talent

training approach, named "SM+IE," which aims to produce more superior composite application talents, to accomplish this purpose. This model examines the impact of the SM training mode and was inspired by the Lean Manufacturing course.

Cico et al. [25] have suggested that the in order to keep students abreast of software technologies, procedures, and practices that were popular in industry, software engineering education was essential. Our goal was to determine how much major software engineering trends were addressed in academic settings by software engineering education, based on literature review. Method: In order to look into how important trends in software engineering are taught in project-based courses, we carried out a systematic mapping study. Based on our investigation of Software Engineering Trends, including practices and processes, educational strategies, as well as how Software Engineering Trends have changed throughout time, we categorized 126 papers.

Gurgan and Cagiltay [26] have developed that Software engineering was an essential component of data science and a data-driven field. As a result, in order to meet industry demands and stay up to date with emerging trends, educational programs must now adjust to recent advancements by first figuring out the skills required for big data software engineering. A semi-automated approach was suggested for the semantic examination of the written content of big data software engineering-related internet job postings. This method uses a probabilistic topic-modeling technique called LDA to extract the hidden semantic structures from a given textual corpus.

A. Motivation

The general overview of current studies indicates that, talent cultivation quality of software engineering majors is an important creating a mechanism that combines learning and production, and building a diverse teaching faculty. The talent cultivation results innovate of new ideas and create revolution in the software engineering. While this type of education can help students gain more specialized information, it falls short of what society needs in terms of creativity and practical application. The critical criterion for assessing the efficacy of education and the beginnings and ends of talent development. The Talent Cultivation Reform 805 evaluation of curricular teaching quality uses quantitative metrics to help college professors become more effective teachers. Numerous researchers are addressing this issue using various technologies found in literature, such as DTA, ANN, and BPN. Larger prediction error BPNs are simply disregarded, possibly resulting in the loss of some important data. ANN Is only processing nonlinear information and DTA the results image de-noising accuracy is low. The approaches not focus accuracy and error rates. Very few approach-based works have been provided in the literature to address this issue; these shortcomings and issues are what have inspired this study effort.

III. PROPOSED METHODOLOGY

The study may explore various aspects such as curriculum design, teaching methods, practical training, and evaluation technique. The talent cultivation quality in engineering majors diagram is displayed in Figure 1. These features can include student performance metrics, curriculum outcomes, teaching methods, practical training opportunities, industry collaborations, research projects, and other relevant factors that contribute to talent development. Nowadays, software engineers should be able to use their creative talents to apply software engineering technologies to address challenges in various sectors. To develop a system framework for strong aptitude, practical knowledge-training, and superior talent-training; additionally, to create a curriculum group closely aligned with fundamental professional knowledge. When learning online, certain images or movies with low visual quality will inevitably be seen. Our grey synthetic noisy image de-noiser was trained on 400 grey images from the public dataset Train 400. A channel mixing module called GLU plus a token mixing module called HGRU make up HGRNN. A basic gated linear recurrent linear unit is described as follows: a noisy image is reshaped into down sampled sub-images in the first layer. a reversible down sampling operator and de-noised image has high quality. The quality of the teaching is also improved it is major characteristics in the talent cultivating platform.

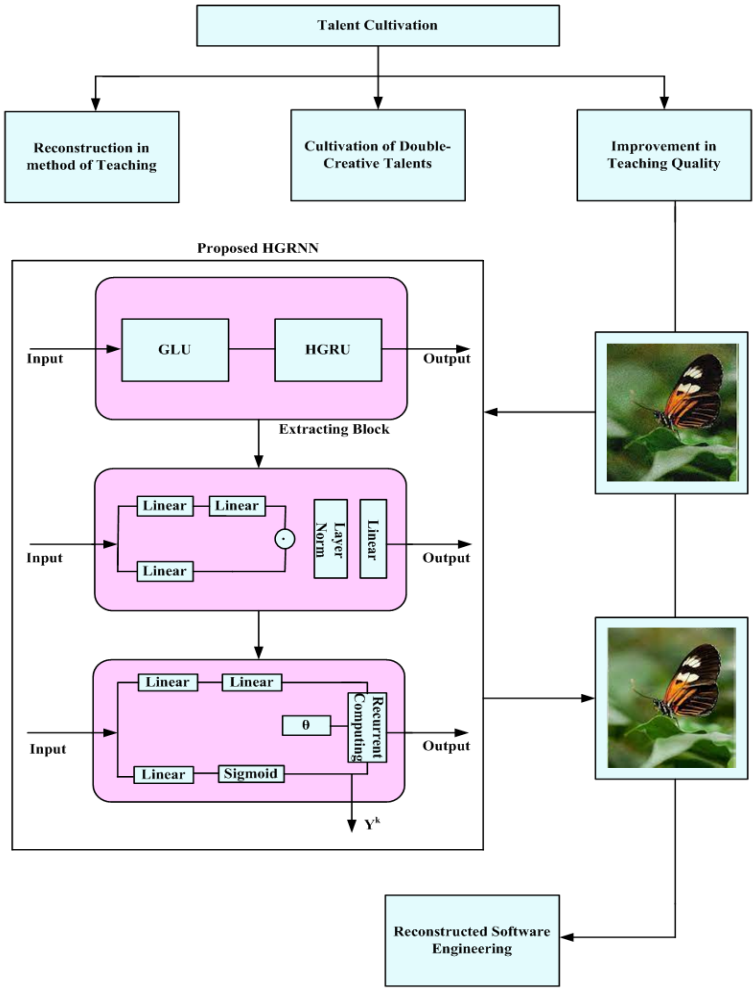

Figure 1: Proposed Methodology Diagram of talent cultivation quality in engineering majors

A. Reconstruction in Method of Teaching

Teachers should adapt traditional teaching modes into advanced ones, such blended learning, flipped classrooms, and practice teaching, to enable students to get new knowledge and further graduate abilities. The project team, guided by this understanding, focuses particularly on creativity (entrepreneurship) and high standards of instruction while creating a cutting-edge test site for software engineering's application-focused talent training approach. Combine and create the two educational concepts, implement innovative personnel training, adjust to comprehensive curriculum content changes, and promote the establishment of practice training bases; with a focus on developing "applicable talent." A curriculum group that is tightly paired with typical professional fundamental knowledge and comprehensive practice design capabilities should be able to be constructed using relevant grasping and construction measures. This will support the development of a strong-ability, high-quality talent-training, and practical knowledge-training system framework.

B. Cultivation of Double-Creative Talents in Software Engineering

Universities ought to support students in their independent study and motivate them to actively participate in relevant innovation-related endeavors. Regularly working on coding challenges, personal projects, or contributing to open-source projects can enhance your problem-solving abilities and coding skills. Engaging in code reviews, pair programming, and participating in tech communities or meet ups can provide valuable feedback and insights from peers. Think about focusing on web development, mobile app development, or machine learning as your area of specialization in software engineering. These activities can help students become more proficient in experimental design and analysis as well as more collaborative and competitive. Additionally, it promotes the development of multifaceted skills with inventive and enterprising capabilities for the nation. These measures are improving the students or professional’s problem solving skills and design skills, creativity skills and coding skills.

C. Smart Teaching Platform

Smart teaching is a learner-centered methodology that enhances learning outcomes and student engagement through the use of technology. There will inevitably be some low-quality images or videos in blended learning, which blends online learning with conventional classroom training. In order to tackle this issue, we suggest integrating the image de-noising module HGRNN into the intelligent teaching platform. This would help teachers acquire high-quality teaching photos and enhance the quality of their instruction.

1) Data collection

Our goal was to unbiasedly evaluate the HGRNN's de-noising performance, so we used 400 gray images from the public dataset Train 400 to train a synthetic noisy image de-noisier. Train 400 is precisely 180 by 180 inches in size. Two datasets are used: Set12, which has 12 photos, and Berkeley segmentation dataset (BSD68), which contains 68 images.

2) De-Noising by Hierarchically Gated Recurrent Neural Network (HGRNN)

The purpose of the intelligent teaching platform is to help teachers enhance their instruction and acquire high-quality photography lessons. Compared to regular neural networks, deep neural networks are distinct. We present a fast and flexible de-noising deep learning neural network namely HGRNN [26, 27].

The several stacked layers of a hierarchically gated recurrent neural network are composed of a channel mixing module (GLU) and a token mixing module (HGRU). A noisy image is converted into down sampled sub-images by the first layer, a reversible down sampling operator. This is the definition of a simple gated linear recurrent linear unit. Along with the down sampled sub-images, we also incorporate a noise level map that may be adjusted. HGRU investigation: A basic gated linear recurrent layer is described as follows:

$$F_T = Sigmoid\left(Z_T\ M_F + A_F\right) \in D^{1\times F} \tag{1}$$

$$J_T = Sigmoid\left(Z_T\ M_J + A_J\right) \in D^{1\times F} \tag{2}$$

Here F_T and J_T represent forget and input gates respectively.

Complex valued recurrence: To obtain element-wise linear RNNs with static decay rates and linear recurrence, eigen decompositions are frequently carried out on the recurrent weight matrix. The expressiveness of the model is limited if only real-valued eigen values are permitted, as this narrows the recurrent weight matrix's range and makes it symmetric. On the one hand, we extract deep coarse characteristics using layer. The first layer, where c is the image's channel number, is 64-filtered and uses a size of 3×3×c to convert noisy images into linear features. If the input is a grayscale image, then c = 1; if not, then c = 3. Next, we utilize the 3 × 3 kernel size to fit the HGRN procedure and produce a coarse feature.

We separately parameterize the real and imaginary sections of the input G_T as follows:

$$\mathrm{Re}\ \left(G_T\right) = SiLU\left(Z_T\ M_{GR} + A_{CR}\right) \in D^{1\times F} \tag{3}$$

$$\mathrm{Im}\left(G_T\right) = SiLU\left(Z_T\ M_{Gj} + A_{Cj}\right) \in D^{1\times F} \tag{4}$$

Lower bound on forget gate values: The magnitude argument λ_T is the only factor that influences how much information is remembered. In practical terms, parameterizing lower bounds separately or for every hidden state, with E representing the number of layers. Given a layer index of P, the subsequent computations can be represented as follows:

$$Q = \left(Soft\max\left(\Gamma, \dim = 0\right) \in D^{E\times F}\right) \tag{5}$$

$$\gamma^P = \left[Cumsum\left(Q, \dim = 0\right)\right] p \in D^{1\times F} \tag{6}$$

λ_T is finally parameterized as follows in the P-th layer;

$$\mu_T = Sigmoid\left(Z_T\ M_\mu + A_\mu\right) \in D^{1\times F} \tag{7}$$

The sigmoid activation function's saturated regions will push μ_T away in order to reach the same forget rate value, $\bar{\gamma}$ closed to one.

$$\mu_T = \frac{\bar{\gamma} - \gamma^p}{1 - \gamma^p} < \bar{\gamma} \tag{8}$$

Trying inputting and forgetting gates: Leaky units can be used to reduce the number of parameters, which is typically favorable. These units have shown empirical usefulness and are closely regarding exponential moving averages and the discretization of continuous time systems.

$$E_T = \lambda_T \Theta \exp\left(j\theta\right) \Theta E_{T-1} + \left(1-\lambda_T\right) \Theta G_T \in \subset^{1\times F} \tag{9}$$

Here Θ denotes the element wise product.

Output gates and projection: It has been demonstrated that state space models benefit from having gates included in the repetition layer's output. Before performing the output projection to obtain HGRU, an output gate is applied in the following manner. By extracting diverse aspects of noise and obtain a more comprehensive representation of the objects in the scene.

$$k_T = Sigmoid\left(M_k\ Z_T + A_k\right) \in D^{1\times 2r} \tag{10}$$

Noise reduction effects are improved by HGRNN since it avoids the gradient disappearance issue and makes noise feature reuse easier. On the other hand, compared to the current methods, HGRNN is almost three times faster and requires less memory. Hence, HGRNN considerably increases efficiency while preserving de-noising performance by carrying out de-noising on sub-images.

D. Reconstructed Software Engineering Majors

Colleges and universities should update and enhance their talent training programs in light of the impact of new technologies and the demands of business and society. Universities should specifically invite software engineers and industry experts to give lectures and teach students about cutting-edge technology and relevant industry knowledge in software engineering. They can also enhance the curriculum system and introduce students to popular software engineering technology including cloud computing, blockchain, big data, and artificial intelligence. A number of curriculum groups are developed in the software engineering profession to aid in the development of the application capabilities and professional knowledge structure. Utilizing the complete design and hands-on ability-building curriculum, which also constructs the construction plan for the experimental practice circumstances and puts the related research concepts into practice for the teaching staff's construction, the application-oriented personnel training process can be made more creative and useful. The research focuses on how to tightly integrate "theory, knowledge, and ability," including how information is transformed into ability and the ways in which interaction between ability-based training programs and theoretical courses promotes the development of practical abilities.

IV. RESULT AND DISCUSSION

In this part, the experimental findings of the talent cultivation quality of software engineering majors using the HGRNN technique based on deep learning is discussed. MATLAB is used to conduct the simulations. MATLAB is used to simulate the proposed method under various performance criteria. Results of HGRNN examined using BPN, ANN, and DTA, three popular techniques.

A. Metric Performance Measures

The efficiency is determined to assess the model's performance. Metrics are employed in measuring. Any model will assess the output of the pattern in relation to these metrics. True Positive (TP): It's the proportion of true positives that are expected to be positive.

True Negative (TN): It is the proportion of real negatives expected to be negative.

- False Positive (FP): It is the proportion of fake positives expected to be positive.
- False Negative (FN): It is proportion of fake negative expected to negative.

1) Accuracy

It is total accurate degree or grouping accuracy, represented in equation (11),

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{11}$$

2) Precision Rate

The precision rate is represented in equation (12),

$$\Pr ecisionrate = \frac{pt}{\left(tp+\ fp\right)} \tag{12}$$

3) Error Rate

Error rate is represented as in equation (13),

$$error\ rate = 100 - accuracy \tag{13}$$

In the software engineering the 5000 people talent demand in 2017 and the talent demand is increased to 6400 people needs talent demands to enroll the software companies in 2020 and the talented people demand is increased to 7000 in 2023 it is displayed in Figure 2. In the software engineering the 5000 talented people enroll in 2017 and the talent people enroll is increased to 6000 in 2019 and talent people to enroll the software companies in 2020 are 8000. In 2021 the 9500 people enroll in the reputed software companies. In 2023 the number of talented people enrollment is 10000 it is displayed in Figure 3. Talent cultivation in software engineering majors using HGRNN technique PSNR is 42dB. The degree of noise in the image is 10. There is a 20dB picture noise level and a 30dB PSNR drop. The PSNR drops to 21dB once more, and the level of picture noise is 30. 18dB is the PSNR. The PSNR is decreased to 14dB the image noise level is 40 and the PSNR is decreased to 12dB the image noise level is 50. Hence the HGRNN technique used to reduce the PSNR is lower it is shown in Figure 4.

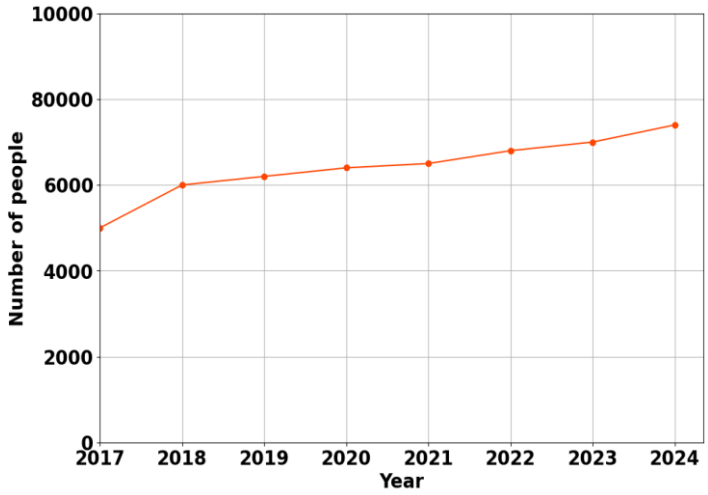

Figure 2: Analysis of talent cultivation demand in software engineering

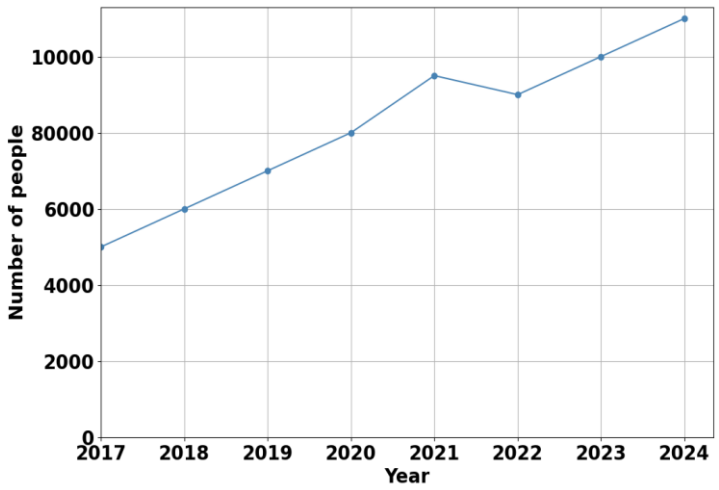

Figure 3: Evaluation of talented people enrollment in software engineering companies

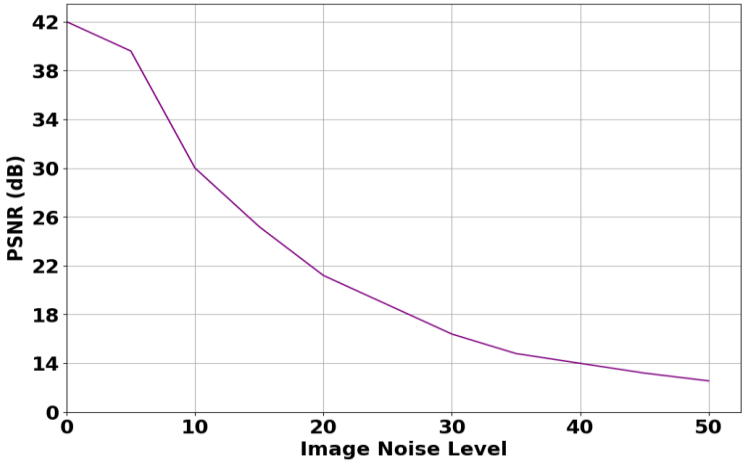

Figure 4: Noise removal effect of using HGRNN technique

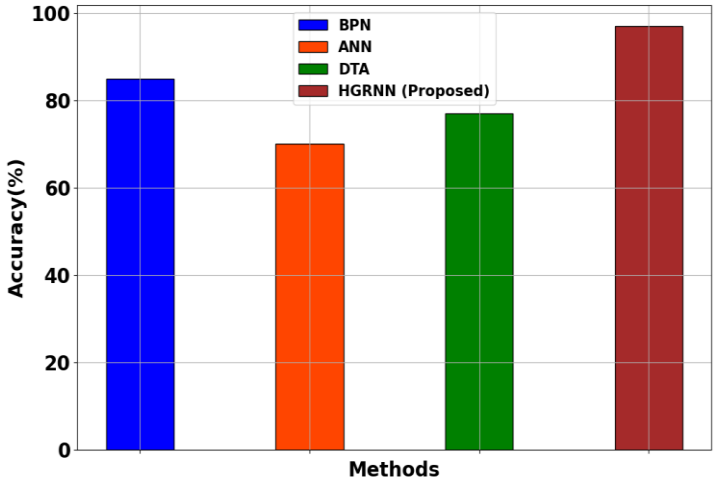

Figure 5: Comparison for accuracy of proposed and existing approaches

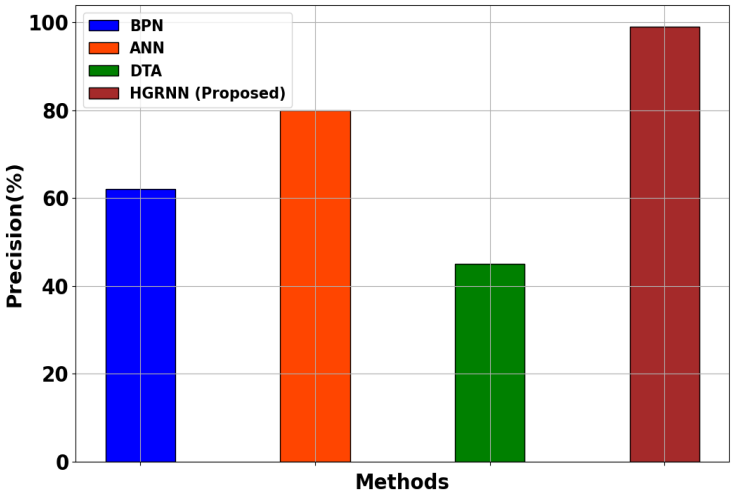

Figure 6: Comparison for precision of proposed and existing approaches

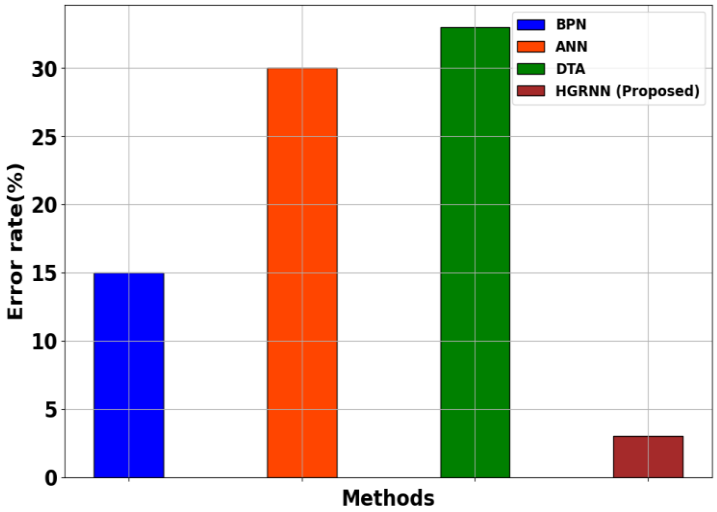

Figure 7: Comparison for error rate of proposed and existing approaches

Table 1: Comparison of PSNR of proposed and Existing Approaches

Models	BPN	ANN	DTA	HGRNN
σ=15	32.33	42.11	20.89	13.11
σ=25	29.67	27.66	18.12	12.34

The accuracy of the BPN approach is 82% and the accuracy of the ANN approach is 70% and the accuracy of DTA approach is 79% and the accuracy of the proposed HGRNN approach gets 98%. Hence the accuracy of the proposed method is higher than the existing method it is shown in Figure 5. The precision of the BPN approach is 61% and the precision of the ANN approach is 80% and the precision of DTA approach is 79% and the precision of the proposed HGRNN approach gets 99%. Hence the precision of the proposed method is higher than the existing approaches it is shown in Figure 6. The error rate of the BPN approach is 15% and the error rate of the ANN approach is 30% and the error rate of DTA approach is 36% and the error rate of the proposed HGRNN approach gets 3%. Hence the error rate of the proposed method is lower than the existing method it is shown in Figure 7. Comparison of PSNR ratio of BPN is the image noise level is 15 the PSNR ratio is 32.33 and the ANN approach is 42.11 the DTA is 20.89 and the proposed method noise level is 13.11. Image noise level is 25 the PSNR of BPN is 29.67Db and the PSNR of ANN is 27.66 and the PSNR of DTA is 18.12Db and the PSNR of the proposed method is 12.34. Hence the PSNR of the proposed approach is gets lower than the existing approaches it is shown in Table 1.

V. CONCLUSION

This paper proposes HGRNN the image de-noising module, as for the smart teaching platform to cultivate the talents in software engineering majors. The talents cultivated in software needs to enhance teaching methods, practical training opportunities, industry collaborations, research projects, and other relevant factors that contribute to talent development. The smart teaching platform the quality of the video or photo of the visual effects is enhances by de-noising images, HGRNN significantly improves efficiency while maintaining de-noising performance. The highest enroll in the software majors is 10000.The accuracy of the proposed HGRNN approach gets 98% and precision is 99%. The error rate of the proposed HGRNN approach gets 3%.Comparison of PSNR ratio of BPN is the image noise level is 15 the PSNR ratio is 32.33 and the ANN approach is 42.11 the DTA is 20.89 and the proposed method noise level is 13.11.Hence the proposed method obtains high accuracy, precision and low error rate and low PSNR ratio.

Acknowledgments

This work was supported by Anhui Provincial Academic program for top professional talents (gxbjZD2022090), Anhui Provincial Scientific research projects (KJ2021A1194,2022AH052847), Anhui Provincial Quality Engineering Teaching and research projects (2021jyxm1287, 2020jyxm0759), Anhui Wenda Information Engineering College scientific research project (XZR2021A18).

REFERENCES

[1] Bharati, R. C., & Singh, A. K. (2019). Predicting rice production using autoregressive integrated moving average model. *Journal of AgriSearch*, *6*(4), 205-210.

[2] Bianco, S., Cadene, R., Celona, L., & Napoletano, P. (2018). Benchmark analysis of representative deep neural network architectures. *IEEE access*, *6*, 64270-64277.

[3] Chen, Y., Xie, Y., Song, L., Chen, F., & Tang, T. (2020). A survey of accelerator architectures for deep neural networks. *Engineering*, *6*(3), 264-274.

[4] Chou, F. I., Tsai, Y. K., Chen, Y. M., Tsai, J. T., & Kuo, C. C. (2019). Optimizing parameters of multi-layer convolutional neural network by modeling and optimization method. *IEEE Access*, *7*, 68316-68330.

[5] Chu, Y., Fei, J., & Hou, S. (2019). Adaptive global sliding-mode control for dynamic systems using double hidden layer recurrent neural network structure. *IEEE transactions on neural networks and learning systems*, *31*(4), 1297-1309.

[6] Farsi, M., Hosahalli, D., Manjunatha, B. R., Gad, I., Atlam, E. S., Ahmed, A., & Ghoneim, O. A. (2021). Parallel genetic algorithms for optimizing the SARIMA model for better forecasting of the NCDC weather data. *Alexandria Engineering Journal*, *60*(1), 1299-1316.

[7] Fetvadjiev, V. H., & He, J. (2019). The longitudinal links of personality traits, values, and well-being and self-esteem: A five-wave study of a nationally representative sample. *Journal of Personality and Social Psychology*, *117*(2), 448.

[8] Gonzalez, M. F., Capman, J. F., Oswald, F. L., Theys, E. R., & Tomczak, D. L. (2019). "Where's the IO?" Artificial intelligence and machine learning in talent management systems. *Personnel Assessment and Decisions*, *5*(3), 5.

[9] Guo, L., Wu, J., & Li, J. (2019). Complexity at Mesoscales: a common challenge in developing artificial intelligence. *Engineering*, *5*(5), 924-929.

[10] Hasson, U., Nastase, S. A., & Goldstein, A. (2020). Direct fit to nature: an evolutionary perspective on biological and artificial neural networks. *Neuron*, *105*(3), 416-434.

[11] Jiang, W., Huang, C., & Deng, X. (2019). A new probability transformation method based on a correlation coefficient of belief functions. *International Journal of Intelligent Systems*, *34*(6), 1337-1347.

[12] Qi, Y., & Wang, J. (2020). A talent cultivation model for improving the innovation ability of college students in scientific research. *International Journal of Emerging Technologies in Learning (iJET)*, *15*(18), 151-164.

[13] Yang, Y., Zhang, H., Chai, H., & Xu, W. (2023). Design and application of intelligent teaching space for blended teaching. *Interactive Learning Environments*, *31*(10), 6147-6164.

[14] Chen, J., Wang, X., Guo, Z., Zhang, X., & Sun, J. (2021). Dynamic region-aware convolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8064-8073).

[15] Zhang, K., Zuo, W., & Zhang, L. (2018). FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Transactions on Image Processing*, *27*(9), 4608-4622.

[16] Khedmati, M., Seifi, F., & Azizi, M. J. (2020). Time series forecasting of bitcoin price based on autoregressive integrated

[17] Lai, Y., & Dzombak, D. A. (2020). Use of the autoregressive integrated moving average (ARIMA) model to forecast near-term regional temperature and precipitation. *Weather and Forecasting*, *35*(3), 959-976.

[18] Liu, Q., Turner, D., & Jing, X. (2019). The "double first-class initiative" in China: Background, implementation, and potential problems. *Beijing International Review of Education*, *1*(1), 92-108.

[19] Hou, C., Wu, J., Cao, B., & Fan, J. (2021). A deep-learning prediction model for imbalanced time series data forecasting. *Big Data Mining and Analytics*, *4*(4), 266-278.

[20] Han, F., Zhang, C., Zhu, D., & Zhang, F. (2022). Talent cultivation of new ventures by seasonal autoregressive integrated moving average back propagation under deep learning. *Frontiers in psychology*, *13*, 785301.

[21] Bai, C., & Huang, W. (2021). Application of an artificial neural network to evaluate the integration effect of quality education and skill education. *International Journal of Emerging Technologies in Learning (iJET)*, *16*(7), 218-232.

[22] Buenaño-Fernández, D., Gil, D., & Luján-Mora, S. (2019). Application of machine learning in predicting performance for computer engineering students: A case study. *Sustainability*, *11*(10), 2833.

[23] Ding, T. (2020). Construction and exploration of university software engineering teaching system based on cdio educational concept. *Frontiers in Educational Research*, *3*(9), 44-48.

[24] Zhang, X., Li, C., & Jiang, Z. (2023). Research on Talent Cultivating Pattern of Industrial Engineering Considering Smart Manufacturing. *Sustainability*, *15*(14), 11213.

[25] Cico, O., Jaccheri, L., Nguyen-Duc, A., & Zhang, H. (2021). Exploring the intersection between software industry and Software Engineering education-A systematic mapping of Software Engineering Trends. *Journal of Systems and Software*, *172*, 110736.

[26] Gurcan, F., & Cagiltay, N. E. (2019). Big data software engineering: Analysis of knowledge domains and skill sets using LDA-based topic modeling. *IEEE access*, *7*, 82541-82552.

[27] Qin, Z., Yang, S., & Zhong, Y. (2024). Hierarchically gated recurrent neural network for sequence modeling. *Advances in Neural Information Processing Systems*, *36*.