engineering proceedings

MDPI

Proceeding Paper

AI Engine for Engineering Talent Development: Automated Competency Graph Modeling and Validation †

Zheng Wen [1] and Wai Yie Leong [2,*]

[1] Faculty of Education and Liberal Arts, INTI International University, Nilai 71800, Negeri Sembilan, Malaysia; i25031583@student.newinti.edu.my
[2] Faculty of Engineering and Quantity Surveying, INTI International University, Nilai 71800, Negeri Sembilan, Malaysia
* Correspondence: waiyie.leong@newinti.edu.my
† Presented at the 9th Eurasian Conference on Educational Innovation 2026 (ECEI 2026), Da Nang City, Vietnam, 30 January–2 February 2026.

Abstract

Universities and enterprises are using AI for engineering education and technical talent cultivation. For better education, we developed the competency engineering engine (CEE), an AI framework that models and validates engineering competencies in integrated circuit electromagnetic susceptibility (IC-EMS). CEE adopts natural language processing, knowledge graphs, and automated routing into engineering workflows to learn technical standards, tokenize texts, and build graphs linking electromagnetic theory to electronic design automation. Students can learn adaptive pathways in foundational, integrated, and advanced research tiers. CEE can enhance reliability, accuracy, and simulation performance in engineering education. Learners improved their problem analysis ability by 19.4%, simulation optimization ability by 21.7%, data-integration accuracy by 20.6%, and system integration level by 17.2% when using CEE. Student engagement increased by 20%, and lab participation by 19% when using electronic design automation (EDA). CEE can be used to effectively adopt AI technology to learn how to automate competency modeling and accelerate skill acquisition. Through an integration with open-source EDA and reinforcement learning agents, CEE can be used for adaptive learning to adapt to rapidly advancing engineering technologies.

Keywords: engineering education; integrated circuit; electromagnetic susceptibility; interdisciplinary research; talent cultivation

1. Introduction

Modern computer chips operate at extremely high speeds (gigahertz) and have microscopic parts packed closely together. This makes them vulnerable to interference. Because of high speeds and tiny sizes, computer chips suffer from electrical noise from nearby devices or environments, which is called electromagnetic susceptibility (EMS). If a chip is susceptible, the hardware fails, which is linked to a security risk. The chip industry is concentrated in Asia, which produces 75% of the world's semiconductor wafers, and China aims to be the top producer by 2030 [1,2]. Because chip production is so huge in this region and it becomes difficult to ensure reliability [3–5], the industry needs trained engineers who know how to design chips that resist electromagnetic interference [5–7].

Despite such demand, engineering education has limitations in EMS instruction. Traditional curriculum models frequently marginalize EMS, treating it as an isolated theoretical

Check for updates

Academic Editors: Teen-Hang Meen, Chun-Yen Chang and Cheng-Fu Yang

Published: 9 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

concept relegated to fragmented elective courses rather than synthesizing it natively within core design and automated software development workflows [8,9]. In addition, instructional programs lack the cross-disciplinary integration required to bridge electromagnetic theory, semiconductor physics, circuit design, and materials science [10]. This results in a theory–practice gap that makes students rarely engage with hands-on EMS simulation, physical testing, or rigorous hardware-in-the-loop design reinforcement [3]. Consequently, entering technical talent remains poorly equipped to resolve multidimensional, real-world electromagnetic interference problems under complex industrial cognitive loads.

To mitigate fragmented knowledge and automate engineering learning, data-driven, scalable engineering infrastructures are required [11]. Traditional manual curriculum engineering cannot adapt to fast-changing enterprise skill requirements or complex DevOps and Agile software engineering deployment life cycles [12]. Therefore, we introduce the AI-driven competency engineering engine (CEE), a microservices-based software framework that automates engineering competency modeling, programmatic user routing, and targeted education. By deploying specialized natural language processing (NLP), CEE transforms unstructured engineering standards and technical corpora into distributed semantic knowledge graphs across four decoupled layers: root theory, mechanism, technology, and frontier integration [11].

Production data transmission from a semester-long comparative tracking deployment validates that the automated AI engine yields rigorous engineering gains. Results show significant score enhancements across all monitored curricula layers and advancements in analytical and simulation engine outputs with large statistical effect sizes. Furthermore, front-end user engagement with electronic design automation (EDA) simulation scripts and laboratory testing workflows was also increased.

In this study, we designed CEE with backend text-mining methodology and validation data for advanced engineering education. Front-end user interface (UI)/user experience (UX) was developed to reduce complexity and lower platform fatigue [13]. CEE is a scalable, efficient learning tool for industrial-grade engineering education and enhancing core structural competencies.

2. Theoretical Foundations

The systematic cultivation of technical expertise in integrated circuit electromagnetic susceptibility (IC-EMS) necessitates an architectural paradigm that bridges established pedagogical frameworks with contemporary, automated software engineering pipelines [7]. To achieve this integration, the AI-driven CEE is developed, comprising four foundational theoretical pillars: the Conceive–Design–Implement–Operate (CDIO) engineering life cycle [1], Outcome-Based Education (OBE) [9], constructivist learning theory [2,4,9], and interdisciplinary education principles [7,8]. These frameworks are operationalized through decoupled microservices that enable active state-tracking and data-driven curriculum routing [11].

- Conceive layer: It automates ingestion and tokenization of unstructured technical standards and documentation, accelerating the identification of electromagnetic risk vectors [3];
- Design layer: It functions as a semantic parser, transforming unorganized technical structures into relational schemas and conceptual entities, thereby informing predictive simulation parameters [11];
- Implementation layer: It compiles extracted dependencies into a graph-relational database, translating academic constructs into functional code repositories [11];

https://doi.org/10.3390/engproc2026141031

- Operate layer: It delivers low-latency application programming interface (API) responses to student environments, enabling hardware-in-the-loop validation and performance profiling [13].

OBE requires instructional design to be governed by measurable competencies [9]. CEE formalizes this principle through a deterministic competency routing service. Rather than relying on static evaluations, OBE requires cloud-based EDA logs, linting diagnostics, and repository commit rates [11]. These multidimensional inputs are assessed against verification matrices to generate automated routing vectors within the talent graph, thereby translating abstract educational objectives into discrete software states and reinforcing industrial resilience [14].

Constructivist principles emphasize experiential iteration and collaborative problem-solving [10]. Within CEE, these principles are embedded in the UI of the interactive student development and operations and live tracking sandboxes [15]. As learners engage with containerized environments, the engine records real-time telemetry and computes variance between simulated configurations and physical instrumentation logs. This feedback loop transforms passive theoretical study into active engineering practice, mitigating the disconnect between classroom learning and industrial application [6].

IC-EMS is inherently interdisciplinary, requiring integration of electromagnetic field theory, semiconductor physics, digital systems design, and materials science [3,7]. To dismantle disciplinary silos, CEE employs an AI knowledge graph engine that structures multi-source domain knowledge into interconnected conceptual networks [11]. As illustrated in Figure 1, the architecture is organized into four decoupled layers.

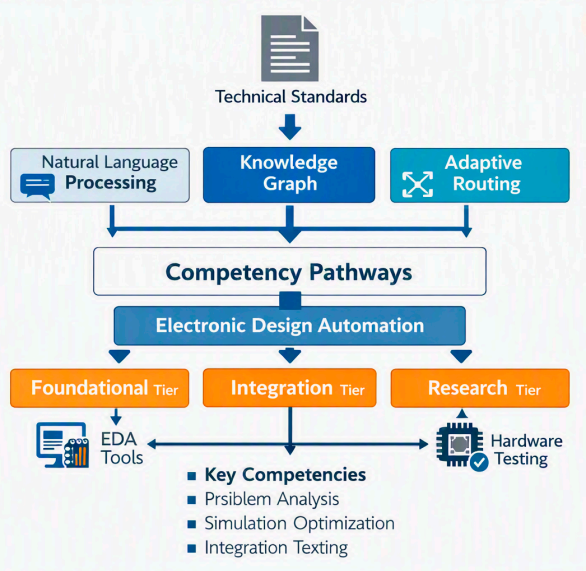

Figure 1. The schematic diagram of CEE developed in this study (created in this study using AI).

- Root theory layer: It encapsulates mathematical constraints, transmission line algorithms, and microwave network functions essential for EMS evaluation [3];
- Mechanism layer: It computes hierarchical dependencies across chip-level, board-level, and firmware modules [11];
- Technology layer: It interfaces with automated EDA scripts, compliance testing, and code-hardening routines to cultivate practical debugging skills [6];
- Frontier integration layer: It synthesizes real-time telemetry, thermodynamic data, and cross-disciplinary updates, enabling dynamic scalability alongside semiconductor innovations [15].

By modeling these domains as interconnected software classes, CEE computes structural skill adjacencies and semantic similarity scores, thereby optimizing technical talent development under diverse cognitive loads [11,13].

3. Methodology

CEE was designed as a microservice-based AI framework that integrates NLP, a semantic knowledge graph, and automated routing. CEE transforms fragmented instructions in IC-EMS into a structured, scalable pipeline that connects theory with electronic design automation (EDA) and hardware validation. CEE comprises four layers.

The root theory layer is responsible for encoding mathematical problems and algorithms for EMS evaluation. The mechanism layer is responsible for modeling dependencies across chip-level, board-level, and firmware modules. The technology layer is designed to correspond to EDA scripts, compliance testing, and code-hardening routines. The frontier integration layer processes thermodynamic data and updates cross-disciplinary knowledge to include semiconductor innovations. The outcomes of the layers are included in a graph-relational database to compute semantic similarity scores and structural skill adjacencies, enabling adaptive routing of learners through competency pathways (Figure 1).

Data were collected from 20 university instructors, 10 industry technical experts, and 150 students. The respondents were involved in or learned integrated circuit design (37.5%), electronic packaging (35.0%), and microelectronics software tools (27.5%). To identify bottlenecks in tasks and learning, learning difficulties were analyzed. Table 1 shows the most common challenges: fragmented and scattered knowledge was reported by 85% of the respondents, lack of hands-on practice and insufficient simulation skills by 87.5%, and weak engineering thinking by 72.5%. Limited tools and low efficiency were also pointed out by 66.3%. This result showed that traditional instruction made learners underprepared for EMS problems.

Table 1. Analysis of learning difficulties (n = 80).

Learning Difficulty	Proportion (%)
Fragmented and scattered knowledge	68 (85.0%)
Outdated knowledge system and insufficient learning motivation	62 (77.5%)
Lack of hands-on practice and insufficient simulation/verification skills	70 (87.5%)
Weak engineering thinking and limited ability to tackle complex problems	58 (72.5%)
Limited learning tools and low efficiency	53 (66.3%)

Industry experts recommended addressing these problems. In total, 92.5% of the respondents thought that university–industry internship programs are required, while 90% supported the integration of case studies of enterprises into teaching. Involvement of industry experts in curriculum design (85%), dual-instructor mentoring (82.5%), and regular seminars (80%) were also pointed out. As shown in Table 2. These results can be used to design CEE's routing service and collaborative modules.

Table 2. Recommendations for industry–academia integration (n = 40).

Learning Difficulty	Proportion (%)
Integration of real enterprise cases into teaching	36 (90.0%)
Co-establishment of university–industry internship bases	37 (92.5%)
Involvement of industry experts in curriculum design	34 (85.0%)
Dual-instructor project mentoring	33 (82.5%)
Regular seminars	32 (80.0%)

https://doi.org/10.3390/engproc2026141031

CEE can be used in three ways. The foundational function supports early learners to learn basic data and script tasks. The integration function provides cloud-based sandboxes for simulations and hardware comparison. The research function is used to construct open-source frameworks and produce deployment modules. Performance is automatically monitored through continuous integration and delivery loops, combining theoretical knowledge (60%), project reports (20%), and repository metrics (20%).

CEE embeds AI functions for simulation and hardware validation. In AI-enhanced engineering education, NLP-based knowledge graphs and adaptive routing are used for technical skill development [14,15]. Knowledge graphs are used to improve interoperability between design and production systems and validate the engineering relevance of CEE's architecture. AI introduction in engineering education indicates the importance of adaptive routing and automated assessment [14]. The generative AI framework is validated for the scalability of CEE for personalized education.

4. Results

To evaluate the performance of the AI-driven CEE, algorithmic routing accuracy and developer onboarding speed were tested. The experimental group used the automated AI engine (n = 54), and the outcome was evaluated against that of the control group (n = 106). The backend methodology optimized operations using knowledge graphs, competency routing vectors, and active repository engagement loops. However, its immediate structural effects on the users' subjective front-end experience required longer execution cycles to smooth out individual variations (Table 3). The multi-tiered execution strategy of CEE enabled substantial, repeatable engineering gains.

- Adaptive routing: Average user performance scores within the foundational ingestion, core integration, and advanced research streams increased by 8–10 points. Independent t-test validations showed that these advancements were statistically significant ($p < 0.001$), with substantial statistical effect sizes ($d > 1.0$). The degree of change reached its peak at the advanced research tier (+16.70%, d = 1.15), confirming that CEE offers exceptional educational value when navigating high-order systemic complexity;
- Competency vector hardening: CEE showed extensive improvement in problem analysis ability (+19.40%, d = 1.44), simulation engine output optimization (+21.70%, d = 1.51), telemetry testing accuracy (+20.60%, d = 1.54), and system trade-off constraint calculations (+17.20%, d = 1.06). Such results validate CEE's capacity to accelerate functional, industrial-grade troubleshooting skills;
- Sandbox engagement metrics: Active developer participation was enhanced significantly. User interaction with automated EDA simulation scripts increased by 20% (χ^2 = 8.82, p = 0.003), and the active repository commit and hands-on loop rate grew by 19% (χ^2 = 5.81, p = 0.016), proving the efficacy of the interactive front-end feedback loop;
- Industrial quantization: CEE also enabled the enhancement of collaborative output, including completed code modules (+13%), documented academic scripts (+19%), and specialized solutions adopted into enterprise sandboxes (+5%).

Objective software competencies and repository outputs increased significantly by using CEE.

- Cognitive fatigue and conflict with UI: User satisfaction showed a non-significant growth of 8% (p = 0.186, φ = 0.10). This disconnect shows that while the backend routing logic optimizes skill acquisition, the raw pipeline complexity introduces cognitive fatigue, confirming the necessity of a dedicated UI/UX layer to abstract away background operations;

- Grade distribution invariance: An 8% increase in excellent performance and a 10% reduction in pass-level states were observed. However, structural alterations across global grade levels remained non-significant during this initial pilot deployment ($\chi^2 = 1.26$, $p = 0.262$ for excellent profiles). This indicates that large-scale application deployments require multi-cohort datasets over longer life cycles to completely map macro-level performance improvement.

Table 3. IC-EMS education system's pilot-teaching outcomes.

Evaluation Dimension	Specific Indicators	Traditional Teaching Model (N = 106)	IC-EMS Education Model (N = 54)	Magnitude of Change/ Outcomes	Statistic	*p*-Value	Effect Size
IC-EMS Knowledge Graph	Student satisfaction	81%	89%	8%	$\chi^2 = 1.58$	$p = 0.186$	$\varphi = 0.10$
Multi-tiered curricula system	Average score of awareness and the foundation layer	70.0 ± 8.5	78.0 ± 8.2	11.40%	$t = -5.70$	$p < 0.001$	$d = 0.95$
	Average score of the core and integration layer	65.0 ± 9.0	75.0 ± 8.8	15.40%	$t = -6.70$	$p < 0.001$	$d = 1.12$
	Average score of the research and innovation layer	62.0 ± 8.8	72.0 ± 8.5	16.70%	$t = -6.87$	$p < 0.001$	$d = 1.15$
Competency development framework	Problem analysis ability	62.0 ± 8.5	74.0 ± 8.0	19.40%	$t = -8.61$	$p < 0.001$	$d = 1.44$
	Simulation and prediction ability	60.0 ± 8.8	73.0 ± 8.2	21.70%	$t = -9.04$	$p < 0.001$	$d = 1.51$
	Experimental measurement ability	63.0 ± 8.7	76.0 ± 7.9	20.60%	$t = -9.21$	$p < 0.001$	$d = 1.54$
	System integration ability	58.0 ± 9.5	68.0 ± 9.2	17.20%	$t = -6.37$	$p < 0.001$	$d = 1.06$
Strengthening practical teaching	EDA simulation engagement	70%	90%	+20%	$\chi^2 = 8.82$	$p = 0.003$	$\varphi = 0.24$
	Hands-on laboratory engagement	65%	84%	+19%	$\chi^2 = 5.81$	$p = 0.016$	$\varphi = 0.19$
Industry–university–research collaborative education	Total number of projects completed	32	45	+13 items	–	–	–
	Academic output	5	24	+19 items	–	–	–
	Industry-adopted solutions	1	6	+5 items	–	–	–
Diversified evaluation system	Proportion of excellent students (≥90 points)	15%	23%	Increase by 8%	$\chi^2 = 1.26$	$p = 0.262$	$\varphi = 0.09$
	Proportion of good students (70–89 points)	52%	57%	Increase by 5%	$\chi^2 = 0.44$	$p = 0.508$	$\varphi = 0.05$

https://doi.org/10.3390/engproc2026141031

Table 3. *Cont.*

Evaluation Dimension	Specific Indicators	Traditional Teaching Model (N = 106)	IC-EMS Education Model (N = 54)	Magnitude of Change/ Outcomes	Statistic	*p*-Value	Effect Size
Diversified evaluation system	Proportion of pass-level students (60–69 points)	28%	18%	Decrease 10%	$\chi^2 = 1.83$	$p = 0.177$	$\varphi = 0.11$
	Proportion of failing students (<60 points)	5%	2%	Decrease by 3%	$\chi^2 = 0.81$	$p = 0.367$	$\varphi = 0.07$

5. Conclusions

The IC-EMS learning framework integrates CDIO, OBE, and interdisciplinary learning. CEE was developed in this study to apply theory to engineering practice. CEE improved student performance, strengthened professional skills, and increased engagement. CDIO-based training enabled students to simulate and conduct experiments in a continuous learning cycle and use OBE to meet industry requirements to enhance employability.

The AI-driven CEE fills the gap between theories and software development and deployment by automating learning in engineering. By combining CDIO, OBE, and interdisciplinary education, CEE is used to enhance measurable, industry-aligned competencies. It replaces traditional instruction using NLP-based text mining, graph-relational node mapping, and deterministic user routing. The microservices framework of CEE increases coding frequency, improves simulation accuracy, and strengthens engineering capabilities.

Although skill acquisition was improved, challenges remain. The pilot testing period was not long enough to explore the effect of the integration of diverse engineering subjects, and knowledge maps did not ensure long-term capability retention. This necessitates extended evaluations for different groups. Feedback from cross-disciplinary experts must be integrated into open-source EDA tools, and deep reinforcement learning agents need to be adopted to optimize adaptive learning. APIs must be developed for easy access.

Despite the limitations, CEE shows how AI is embedded into engineering education to automate the learning of IC-EMS. CEE transforms fragmented instruction through EDA simulation and hardware validation. Pilot deployment results showed an increase in analytical ability (+19.4%), simulation accuracy (+21.7%), telemetry testing (+20.6%), and system integration (+17.2%). Engagement with EDA increased by 20%, and laboratory participation by 19%. The adoption of five sandbox solutions indicated the validation of CEE's relevance in the industry.

CEE's role as an educational tool and engineering tool aligns with industry needs and technical performance enhancement. To enhance user satisfaction, UI/UX design must be improved to reduce cognitive fatigue. CEE can be improved by adopting open-source EDA platforms, reinforcement learning for adaptive routing, and evaluation with engineering learning platforms.

Author Contributions: Conceptualization, Z.W. and W.Y.L.; methodology, Z.W.; software, Z.W.; validation, Z.W. and W.Y.L.; formal analysis, Z.W.; investigation, Z.W.; resources, Z.W.; data curation, W.Y.L.; writing—original draft preparation, Z.W.; writing—review and editing, Z.W. and W.Y.L.; visualization, Z.W.; supervision, W.Y.L.; project administration, Z.W.; funding acquisition, W.Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: During the preparation of this manuscript, the authors used ChatGPT (OpenAI, GPT-5.6 Sol) for assistance with manuscript structuring, technical language refinement, and the generation and refinement of illustrative technical Figure 1. The authors reviewed, edited, and verified all AI-assisted outputs and take full responsibility for the scientific accuracy, integrity, and content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Al-Juboori, H.; Noonan, G. Concepts of Engineering Education Innovation and Design Thinking: Implementing the CDIO–Approach Themes. In *Proceedings of the 2024 IEEE Global Engineering Education Conference (EDUCON)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–5. [CrossRef]
2. Aeni, N.; Budiamin, A.; Muhtar, F. Implementation of the Learning Theory of the Constructivism Perspective Jean Piaget (1896–1980) in PAI Learning at SDI Bilal Bin Robah Batulayar Village, Batulayar District. *Edumaspul J. Pendidik.* **2023**, *7*, 5219–5229. [CrossRef]
3. Boyer, A.; Caignet, F.; Laidet, M. An efficient characterization method to predict the susceptibility of integrated circuits to multitone disturbance. *IEEE Trans. Electromagn. Compat.* **2024**, *66*, 1793–1803. [CrossRef]
4. Erawati, N.K.; Adnyana, P.B. Implementation of Jean Piaget's Theory of Constructivism in Learning: A Literature Review. *Indones. J. Educ. Dev. (IJED)* **2024**, *5*, 394–401. [CrossRef]
5. Kim, Y.S.; Rho, S.H. The US–China Chip War, Economy–Security Nexus, and Asia. *J. Chin. Political Sci.* **2024**, *29*, 433–460. [CrossRef]
6. Shi, J. Research on teaching reform path of "Digital Integrated Circuit Design" course under OBE concept. *Educ. Reform Dev.* **2025**, *8*, 40–46. [CrossRef]
7. Kans, M. Applying an innovative educational program for the education of today's engineers. *J. Phys. Conf. Ser.* **2012**, *364*, 012113. [CrossRef]
8. Leung, J.K.L.; Ng, D.T.K.; Tsui, C.Y. A primer for design and systems thinkers: A first-year engineering course for mindset development. *Educ. Sci.* **2024**, *14*, 86. [CrossRef]
9. Baek, S.G.; Ham, E.H. An evaluation study on the educational value of teaching practicum in secondary schools. *Asia Pac. Educ. Rev.* **2009**, *10*, 271–280. [CrossRef]
10. Nurhasnah, N.; Sepriyanti, N.; Kustati, M. Learning theories according to constructivism theory. *J. Int. Inspir. Educ. Technol.* **2024**, *3*, 19–30. [CrossRef]
11. Wan, Y.; Liu, Y.; Chen, Z.; Chen, C.; Li, X.; Hu, F.; Packianather, M. Making knowledge graphs work for smart manufacturing: Research topics, applications and prospects. *J. Manuf. Syst.* **2024**, *76*, 103–132. [CrossRef]
12. Liu, C. A comprehensive review of applications of AI technologies in higher engineering education. *Discov. Educ.* **2025**, *4*, 528. [CrossRef]
13. Lin, K.Y.; Li, M.H.; Lo, F.; Huang, H.C.; Matsuno, K.; Watanabe, R. Adaptive learning with human factors and artificial intelligence: Associations with training effectiveness in programming education. *Int. J. Ind. Ergon.* **2025**, *110*, 103834. [CrossRef]
14. Qian, M.; Ji, H.; Li, L. Empirical validation of a generative AI framework for personalized education assessment. *Sci. Rep.* **2026**, *16*, 11538. [CrossRef] [PubMed]
15. Ruberti, A.; Polenghi, A.; Macchi, M. Maintenance plan adaptation based on health ratings of servitised machines through a fleet-wide machine clustering method. *J. Manuf. Syst.* **2024**, *77*, 368–383. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.