International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0231

e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1562 - 1566

An AI-Enabled Adaptive Visualization Approach to Overcoming Cognitive Barriers in Technical Terminology Learning

Pooja S Pillai[1], Aswin A Manchakkal[2], Rhishikesh T.S[3], Nithin Saju[4], Devika T.V[5], Vyshnav K.M[6]
[1]Assistant Professor, Department of Linguistic Studies, Yenepoya (Deemed to be University), Bangalore, Karnataka.
[2,3,4,5,6]UG - Bachelor of Computer Application, Yenepoya (Deemed to be University), Bangalore, Karnataka.
Email ID: *Poojaspillai.blr@yenepoya.edu.in[1], 31403@yenepoya.edu.in[2], 31574@yenepoya.edu.in[3], 31424@yenepoya.edu.in[4], 31434@yenepoya.edu.in[5], 30872@yenepoya.edu.in[6]*

Abstract
Acquiring the specialized technical terminology is an important and ongoing requirement of the computer science education. Despite the extensive usage of codified definitional paradigms and traditional static visual instructional aids. The majority of students struggle to achieve the genuine conceptual clarity and epistemic knowledge. This difficulty is the most visible in learner's incapacity to adequately combine terminology with the conceptual and the procedural knowledge. Using the survey-based research approach, this study investigates the complex barriers to the technical terminology proficiency by examining the interconnected factors such as the conceptual integration, the effectiveness of the visualization- based learning strategies, the learner's capacity for a practical application, and the persistent problem of delay in the comprehension error detection. The data collected from the undergraduate computer science students showed that the learners commonly exhibit unfounded metacognitive confidence when dealing with the technical jargon, resulting in false comprehension. This cognitive-epistemological mismatch was evident during evaluative tasks, where the difficulties in applying the theoretical notions reveal major deficits in the basic conceptual frameworks. The findings also showed that the traditional instructional techniques are frequently fail to detect and rectify the growing conceptual mistakes in a timely manner, allowing them to become the cognitively entrenched. This research presents the Conceptual DNA System (CDS), a three-tier educational framework aimed at advancing learning of technical vocabulary and the assimilation of concepts in computer science, thereby addressing existing teaching limitations. It combines context-based visualization, collaborative learning and real-time feedback to support early error detection, deeper conceptual integration and sustained comprehension. The findings show that adaptive visualisation assists learners in organizing and connect technical information, mitigates cognitive overload, and improves lasting retention by transforming static terms into visual forms.
Keywords: *Technical Terminology, Computer Science Education, Visualization-Based Learning, False Comprehension, Conceptual DNA System.*

1. Introduction
While pursuing Computer Science education it mandates that students acquire thorough understanding of specialized technical terminology [1]. Some of the foundational terms of programming like stack, recursion, polymorphism and encapsulation enables students to interpret concepts embedded in any program, engage in technical discussions and theoretical knowledge could be translated to practical problem solving endeavors [2]. However, the textbook visualiztion techniques and rigid teaching approaches combined with the abstract nature of these terms creates substantial cognitive barrier for learners [3]. Conventional teaching techniques frequently fails to align with the existing mental models of students because it mostly relies on static diagrams, definition-centric explanations, and

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0231
e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1562 - 1566

generic examples [4]. As a result, learners will often have superficial understanding without fully grasping the concepts which creates an illusion of competence a form of false comprehension which will be evident during practical applications or assessments then they resort to memorizing definitions and diagrams [5]. This study tackles these challenges through data-based analysis and technological innovation. A survey of 80 Bachelors of Computer Application (BCA) students was conducted to identify where precisely does terminology understanding breaks down at the early stage of academic journey. Based on the findings, we present the Conceptual DNA System (CDS), an educational framework with three layers, which combines AI-powered personalized learning narratives, real-time behavioral confusion detection, and collaborative peer-learning networks.

2. Methodology

2.1.Research Design and Participants

This study was conducted among BCA students which had a quantitative design. It took place in the months of February and March and it received 80 responses. The distribution of fields includes Data Science & AI (44.1%), Other specializations (26.5%), Cybersecurity (23.5%), and General Computer Science (5.9%). In terms of programming language background split: 55.9% had no prior experience (under 1 year), and 5.9% reported intermediate experience (2-3 years).

2.2.Survey Instrument

The survey evaluates commonly confused term pairs, initial confidence levels, frequency of misunderstanding recognition, openness to personalized explanations, contributing difficulty factors measured on a 5-point Likert scale [6], current learning resources and dynamic visualization. The survey questions were modeled section-wise to gather all the crucial information we required. A large number of students form a perception that they can understand a concept by reading or hearing its definition, but that perception breaks down only when they are required to apply it in code. This is corroborated by the findings in the survey which state that 69.1% of students only caught their misunderstanding while writing a program. As Shown in Table 1.

Table 1 When Students Realize They Misunderstood Terms

Context	Percentage
When writing code	69.1%
During practice problems	58.0%
When explaining to others	49.4%
During exams or quizzes	45.7%

3. Survey Findings

3.1.The Illusion of Understanding

The clear gap between students' perceived confidence and how much they actually understand is a highly revealing result. When students first encounter new terms, only 5.9% expressed high confidence, 35.3% were moderately confident, 26.5% were unsure, and 23.5% said it depends. A striking finding was that 94.1% of students later realized their understanding of a CS term was not as complete as they thought. 50.0% reported that this happened at times, 32.4% indicated that it was rare, and 11.8% indicated it happened frequently.

3.2.Contributing Factors

Students were provided with a set of difficulty factors to rate on a scale from 1 to 5, with 1 being (not a problem) and 5 being (major problem). With a mean score of 2.76, the gap between knowing what a term means and being able to implement it in code was the highest of all. Followed by mixing up similar terms (2.71), textbook examples that were disconnected with reality or boring, static diagrams incapable of demonstrating working of something step by step (2.29), and abstract definitions that were hard to make sense of (2.29) [3].

3.3.Commonly Confused Terms

According to the study there are certain pair of words that creates confusion among students: Class vs Object (17.6%), Compiler vs Interpreter (14.7%), Stack vs Queue (8.8%), Loops vs Recursion (5. 9%).The data indicate that 82.4% of students experience they face this sort of confusion on a regular basis, which proves it is not an isolated problem but something that requires fundamental

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0231
e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1562 - 1566

revamping of the teaching methods.

3.4.Receptiveness to Innovation

Students are open to experiment with different learning methods [8]. 88.2% of students gave positive response to the question of whether linking technical terms to their personal interests would make understanding easier.[9] 50.0% of them responded definitely yes and 38.2% responded probably yes. Similarly, dynamic visualization got a positive response, with 73.5% rating them as very helpful by giving a score of 4 or 5 out of 5, and 44.1% rating all the way to the top score of 5. The predominantly positive response for visualization-based learning confirms that students are not only open to it also find it more effective than traditional methods.

4. The Conceptual DNA System

4.1.System Overview

The Conceptual DNA System (CDS) is a three-layer educational framework designed to address how students struggle with technical terminology. The name borrows from biology DNA carries the blueprint that makes every individual distinct, and CDS works on a similar logic, weaving three interconnected components into a learning experience that adapts to how each student thinks and builds understanding. It is built directly from what the survey data revealed.

4.2.Layer 1: AI-Generated Personalized Narratives

CDS takes a different approach it changes how a concept is explained based on who the student is. Most adaptive systems adjust difficulty while keeping the same explanations for everyone. [10] A student who plays RPGs might learn about Stack through inventory management in games like World of Warcraft or Cyberpunk. A student who cooks might grasp the same concept through stacking plates. It works by anchoring new ideas to what students already know, pulling comparisons from contexts they're comfortable with, easing toward the formal definition rather than leading with it, tying code examples to that same familiar ground, and highlighting the mistakes that tend to trip most students up along the way.

4.3.Layer 2: Real-Time Micro-Behavioral Confusion Detection

It works by anchoring new ideas to what students already know, pulling comparisons from contexts they are comfortable with, easing toward the formal definition rather than leading with it, tying code examples to that same familiar ground, and highlighting the mistakes that tend to trip most students up along the way. [14] The survey should have showed that 69.1% of students only recognize misunderstandings while writing code by which point the damage is already done.

4.4.Layer 3: Peer Distinction Visualization Networks

The third layer targets the term confusion that 82.4% of student's experience regularly, through a community-built knowledge base that maps out how peers worked through those same distinctions. Students can browse comparison maps breaking down the differences between terms, read first-hand accounts of how others finally got a concept to click, spot recurring patterns in common mistakes, place code examples side by side, and use visual templates that lay the contrast out clearly. [11] Take Class Vs Object a student stuck on that distinction might find a peer who cracked it with the cookie cutter analogy, backed by a diagram separating template from instance, and a breakdown of where most people go wrong.

4.5.Theoretical Foundations

CDS rests on three well-established teaching theories. Constructivism shapes the personalized narrative layer new concepts land better when tied to what students already understand rather than introduced cold. The Zone of Proximal Development drives the behavioral detection layer, stepping in with support at exactly the point where a small push can get a student past something they couldn't work through alone. Communities of Practice underpins the peer network, treating learning as something that happens naturally when people share how they worked through the same difficulties. [13] On top of this, CDS draws from how biological memory works using emotional anchoring through relevant narratives, pattern recognition through behavioral analytics, and social reinforcement through shared community knowledge. [7]

5. Discussion

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0231
e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1562 - 1566

5.1.Novelty and Contributions

The adaptive learning platforms already exist, but none combine dynamic narrative personalization, process-level confusion detection, and peer-generated distinction mapping in a single framework. CDS brings three things together that existing systems handle separately, if at all. Most of the adaptive systems adjust to what students learn content difficulty rather than how concepts are explained. And while collaborative tools can support joint concept mapping, CDS enables something different asynchronous sharing of the personal insights of students to build as they work through confusion on their own terms. Predictive analytics in current tools tend to focus on big-picture outcomes like dropout risk, not the smaller moment-to-moment signs of confusion during learning.

5.2.Implementation Considerations

The narratives that AI generates need to remain pedagogically sound even as they become personal. Familiar analogies have to transfer cleanly to a generic technical discourse. Monitoring behaviors triggers privacy concerns that need to be dealt with early on. [12] Peer-generated content needs to be vetted, And the system needs to be able to distinguish between productive struggle and confusion that needs actual intervention. And all across it, CDS should support human instruction rather than try to replace it. Deploying CDS introduces questions worth thinking through.

5.3.Future Research

The next step is to determine whether the CDS are actually doing what we want them do. So, the controlled studies have to ask whether truly good personalized narratives are actually improving retention and transfer, whether the behavior signals actually help us to detect flag confusion before it becomes and whether the peer visualization's actually help the students to un-flag terms that are prone to confusion. And the research has to look into how the three layers are interacting, when the scaffolding is effective, when it isn't, whether there is a threshold for the analogy-based approach and whether the terminology gains actually do translate into programming gains in general.

Conclusion

This study investigates the experience of 80 BCA students with problems associated with technical terminology and the result is a remarkably coherent picture. 94.1% of students came to realize too late that they didn't understand a term as well as they thought they did. 69.1% of students came to realize that at the time of writing the code. The main issues were the distance between the definition and application of the term (mean 2.76/5), confusion between terms that are similar (2.71/5), and examples that had nothing to do with the real world of practice (2.50/5). It provides the combination of the AI-generated narratives customized to each student interest, something that 88.2% of students indicated would complement the real-time behavioral detection that prevents confusion before it occurs and a peer network paired with the 82.4% of students that regularly confuse terms that are similar. The Conceptual DNA System was designed with these problems in mind. CDS is based on Constructivism, Zone of Proximal Development, and Communities of Practice; it doesn't simply customize what students are learning but how the concepts are presented to them in the first place. When learning is relevant, confusion is removed early, and the students participate in structuring the idea, focuses the probability of the peers leaving with an understanding that is not just enough to succeed in an examination but enough to matter are far higher. When we all teach the same way, with the same definitions and diagrams, it serves a few people and doesn't add to the knowledge of the larger crowd.

References

[1]. Luxton-Reilly, A., et al. (2018). Introductory programming: A systematic literature review. *Proceedings of the ACM Conference on Innovation and Technology in Computer Science Education*, pp. 55–106.

[2]. Qian, Y., & Lehman, J. (2017). Students' misconceptions and other difficulties in introductory programming: A literature review. *ACM Transactions on Computing Education*, 18(1), 1–24.

[3]. Guzdial, M. (2015). *Learner-Centered Design of Computing Education: Research on*

e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1562 - 1566

Computing for Everyone. Morgan & Claypool Publishers.

[4]. Sorva, J. (2012). Visual program simulation in introductory programming education. *Doctoral Dissertation, Aalto University*.

[5]. Kruger, J., & Dunning, D. (1999). Unskilled and unaware of it: How difficulties in recognizing one's own incompetence lead to inflated self-assessments. *Journal of Personality and Social Psychology*, 77(6), 1121–1134.

[6]. Likert, R. (1932). A technique for the measurement of attitudes. *Archives of Psychology*, 22(140), 1–55.

[7]. Dunning, D., Johnson, K., Ehrlinger, J., & Kruger, J. (2003). Why people fail to recognize their own incompetence. *Current Directions in Psychological Science*, 12(3), 83–87.

[8]. Walkington, C. A. (2013). Using adaptive learning technologies to personalize instruction to student interests. *Journal of Educational Psychology*, 105(4), 932–945.

[9]. VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist*, 46(4), 197–221.

[10]. Walkington, C., & Bernacki, M. L. (2018). Personalization of instruction: Design dimensions and implications for cognition. *Journal of Experimental Education*, 87(1), 50–68.

[11]. D'Mello, S., & Graesser, A. (2012). Dynamics of affective states during complex learning. Learning and Instruction, 22(2), 145–157.

[12]. Baker, R. S. J. d., & Inventado, P. S. (2014). Educational data mining and learning analytics. In Learning and Knowledge Analytics in Educational Technology. Springer.

[13]. Lave, J., & Wenger, E. (1991). Situated Learning: Legitimate Peripheral Participation. Cambridge University Press.

[14]. Piaget, J. (1972). The Psychology of the Child. Basic Books.

[15]. Vygotsky, L. S. (1978). Mind in Society: The Development of Higher Psychological Processes. Harvard University Press.