

Convergence of Disciplines in Computer Science Programs*

Conference Tutorial

Mohammad Amin, Bhaskar Sinha, Pradip Peter Dey
School of Technology and Engineering
National University
9388 Lightwave Ave., San Diego, CA 92123
`{ mamin, bsinha, pdey}@nu.edu`

Convergence of disciplines refers to the merging of different fields of study to create capabilities that the market requires. The need for traditional specialties within Computer Science (CS) programs is rapidly being augmented by the demand for integrated knowledge and understanding. Academic institutions are experiencing a shift in the CS education space, due to the rapid integration of interdisciplinary expertise requirements and new problem-solving approaches in a progressively complex digital world. As technology increasingly controls human daily workloads, traditional boundaries between academic fields are disappearing, replaced by inclusive curricula, modern research, and new career paths.

One of the visible examples of this convergence is the smartphone, which includes technical understandings of phones, camera, music, mobile, networking, user interface, GPS, etc. Similarly, the rise of Natural Language Processing (NLP) shows the merging of linguistics, psychology, and CS. Modeling techniques in statistics and computational fields are now essential for developing AI-powered tools like language translation applications. The evolution of NLP demonstrates how an interdisciplinary foundation enhances the ability of computational systems to process human language [1]. Economics and social sciences are also invariably weighing in on the development of CS curricula. With the expansion of data analytics and artificial intelligence, understanding human behavior, motivations, and ethics has become a central issue. Algorithmic fairness, privacy, and bias moderation are now regular discussion points in

*Copyright is held by the author/owner.

data ethics courses [2]. Human-Computer Interface (HCI) is another area of development surrounding convergence. Traditionally driven primarily by user interface (UI) design, HCI now collaborates with cognitive science, computing, data analytics, and graphics to develop adaptive and intelligent systems that understand user perspectives and thought processes. HCI research is increasingly incorporating NLP, computer vision, and ethical design, linking it closely with AI and socio-technical systems [3]. Recently, cybersecurity has also emerged as a multidisciplinary example, uniting theoretical CS, cryptography, and human factors. Academic security programs now include secure coding, behavioral analysis, and policy, all supported by core CS skills. With cyber threats becoming more refined, academia emphasizes collaboration between theoretical and applied CS researchers to devise robust security solutions [4]. Theoretical CS, traditionally perceived as abstract and special, is discovering innovative relevance through its connections within itself. These intersections reflect the evolving nature of theory's role in enabling real-world applications, prompting departments to offer courses that contextualize mathematical rigor with practical relevance [5]. These shifts require academic restructuring, both at the curriculum content and administration levels. Some leading universities have progressed in this direction by offering interdisciplinary specializations and joint programs, such as human-computer interaction, programming with graphic and GUI designs, physics with quantum AI and algorithm design, big data and vector databases, among others, to equip graduates with both domain-specific expertise and computational proficiency. These combinations enhance students' capabilities to confront real-world challenges that require more than just technical prowess.

The convergence of traditional CS specialties in education not only augments technological advancements but also trains learners for future leadership roles in a space where modernization and advances are driven by multi-disciplinary expertise. This merging in academia is not only reshaping how knowledge is prepared and taught but also aligning education with real-world innovations. As computing becomes the basis of research and innovation across all fields, accepting and practicing the interdisciplinary teaching-learning paradigm is no longer an option but is rapidly becoming essential.

References

- [1]. D. Jurafsky and J. H. Martin. Speech and Language Processing (3rd ed.) 2020. Draft online:<https://web.stanford.edu/jurafsky/slp3/>
- [2]. C. O'Neil. Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy. Crown Publishing. 2015.

- [3]. J. Lazar, J. H. Feng and H. Hochheiser. Research Methods in Human-Computer Interaction (2nd ed.). Morgan Kaufmann. 2017.
- [4]. M. Bishop. Computer Security: Art and Science (2nd ed.). Addison-Wesley. 2018.
- [5]. S. Aaronson. Quantum Computing Since Democritus. Cambridge University Press. 2013.

Tutorial Description This tutorial offers an interactive and effective learning session where panelists will raise critical questions, concerns, and discuss potential opportunities. They will also provide valuable suggestions on how the convergence of disciplines can enrich the CS curriculum. The attendees can interact, ask questions, express their ideas, and provide constructive feedback. Examples of some interdisciplinary integrations are introduced. Discussions will include how these can be seamlessly included into the CS curriculum, where students can get opportunities to learn about these emerging technologies and equip themselves to face future challenges and contribute appropriately. The panelists will also address the ethical issues involved and how to maintain academic standards and integrity.

Expected Outcomes Session attendees will gain a useful initial understanding of the driving technologies and their value-adds. They will participate in open discussions, constructive criticisms, and suggestions.

Target Audience Interested faculty who teach or are planning to teach CS courses and related topics.

Prerequisites All educators interested in teaching and learning in CS are welcome to this tutorial session, where intuitive explanations with examples are presented and discussed.