

Chapter 1

Convergence of Software Science and Computational Intelligence: A New Transdisciplinary Research Field

Yingxu Wang
University of Calgary, Canada

ABSTRACT

Software Science is a discipline that studies the theoretical framework of software as instructive and behavioral information, which can be embodied and executed by generic computers in order to create expected system behaviors and machine intelligence. Intelligence science is a discipline that studies the mechanisms and theories of abstract intelligence and its paradigms such as natural, artificial, machineable, and computational intelligence. The convergence of software and intelligent sciences forms the transdisciplinary field of computational intelligence, which provides a coherent set of fundamental theories, contemporary denotational mathematics, and engineering applications. This editorial addresses the objectives of the International Journal of Software Science and Computational Intelligence (IJSSCI), and explores the domain of the emerging discipline. The historical evolvement of software and intelligence sciences and their theoretical foundations are elucidated. The coverage of this inaugural issue and recent advances in software and intelligence sciences are reviewed. This editorial demonstrates that the investigation into software and intelligence sciences will result in fundamental findings toward the development of future generation computing theories, methodologies, and technologies, as well as novel mathematical structures.

DOI: 10.4018/978-1-4666-0261-8.ch001

INTRODUCTION

The latest developments in computer science, theoretical software engineering, cognitive science, cognitive informatics, and intelligence science, and the crystallization of accumulated knowledge by the fertilization of these areas, have led to the emergence of a transdisciplinary and convergence field known as software and intelligence sciences. The coverage of the International Journal of Software Science and Computational Intelligence (IJSSCI) includes theories, methodologies, technologies of software science and engineering, denotational mathematics, and their applications in engineering and industries.

Software Science is a discipline that studies the theoretical framework of software as instructive and behavioral information, which can be embodied and executed by generic computers in order to create expected system behaviors and machine intelligence (Wang, 2007a). Intelligence science is a discipline that studies the mechanisms and theories of abstract intelligence and its paradigms such as natural, artificial, machinable, and computational intelligence (Wilson and Frank, 1999; Wang, 2008a, 2009a). The convergence of software science and intelligent science forms the transdisciplinary field of computational intelligence, which provides a coherent set of fundamental theories, contemporary denotational mathematics, and engineering applications.

This editorial addresses the objectives of the International Journal of Software Science and Computational Intelligence (IJSSCI), and explores the domain of the emerging discipline. The historical evolvement of software and intelligence sciences and their theoretical foundations are elucidated. The coverage of this inaugural issue and recent advances in software and intelligence sciences are highlighted. This editorial demonstrates that the investigation into software and intelligence sciences will result in fundamental findings toward

the development of future generation computing theories, methodologies, and technologies, as well as novel mathematical structures.

THE EMERGENCE OF SOFTWARE SCIENCE

Software as instructive behavioral information has been recognized as an entire range of widely and frequently used objects and phenomena in human knowledge. Software science is a theoretical inquiry of software and the laws constrain it on the basis of empirical studies on engineering methodologies and techniques for software development and software engineering organization. In the history of science and engineering, a matured discipline always gave birth to new disciplines. For instance, theoretical physics was emerged from general and applied physics, and theoretical computing was emerged from computer engineering. So will software science emerge and grow in the field of software, computer, information, knowledge, and system engineering (Wang, 2007a).

This section provides perspectives on the emerging discipline of software science along with the maturity of software engineering theories and methodologies in fundamental research. The architecture and roadmap of software science are presented. The theoretical framework, mathematical foundations, and basic methodologies of software science will be briefly introduced.

What is Software Science?

Definition 1. *Software Science is a discipline of enquiries that studies the theoretical framework of software as instructive and behavioral information, which can be embodied and executed by generic computers in order to create expected system behaviors and machine intelligence.*

11 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/convergence-software-science-computational-intelligence/65119

Related Content

On Cognitive Models of Causal Inferences and Causation Networks

Yingxu Wang (2011). *International Journal of Software Science and Computational Intelligence* (pp. 50-60).

www.irma-international.org/article/cognitive-models-causal-inferences-causation/53162

Using Data Masking for Balancing Security and Performance in Data Warehousing

Ricardo Jorge Santos, Jorge Bernardino and Marco Vieira (2013). *Handbook of Research on Computational Intelligence for Engineering, Science, and Business* (pp. 384-409).

www.irma-international.org/chapter/using-data-masking-balancing-security/72501

Empirical Studies on the Functional Complexity of Software in Large-Scale Software Systems

Yingxu Wang and Vincent Chiew (2013). *Advances in Abstract Intelligence and Soft Computing* (pp. 174-192).

www.irma-international.org/chapter/empirical-studies-functional-complexity-software/72781

The Formal Design Models of a Set of Abstract Data Types (ADTs)

Yingxu Wang, Xinming Tan, Cyprian F. Ngolah and Philip Sheu (2010). *International Journal of Software Science and Computational Intelligence* (pp. 72-100).

www.irma-international.org/article/formal-design-models-set-abstract/49133

Automatic Text Classification from Labeled and Unlabeled Data

Eric P. Jiang (2012). *Intelligent Data Analysis for Real-Life Applications: Theory and Practice* (pp. 249-264).

www.irma-international.org/chapter/automatic-text-classification-labeled-unlabeled/67452