International Journal of Enhanced Research in Management & Computer Applications, ISSN: 2319-7471
Vol. 4 Issue 6, June-2015, pp: (1-6), Impact Factor: 1.296, Available online at: www.erpublications.com

A Comparative study on Computational Intelligence, Granular Computing and Soft Computing technique

Neeti Malik

Abstract: This paper gives an overview and considers impacts of some theories developed around granular computing, theory of computer networks and of evaluative computing on development of computing systems. Possible further development of soft computing and impacts of that development on the development of computing systems in general, is considered. This article has made the summary on the development and the present situation of all these computational techniques.

Keywords: soft computing, granular computing, computational intelligence.

INTRODUCTION

The main objective of this article is to consider the effects of different uncertainty representation paradigms models, such as fuzziness and roughness, and of the neural networks theory and evolutionary computing on computing systems development. The traditional approach to computing sometimes is referred as hard computing. Fuzzy logic with expert systems, neural networks, probabilistic reasoning, belief networks, genetic algorithms, chaos theory and parts of learning theory makes complementary partnership of disciplines and technologies known as soft computing, which gives methods for solving complex problems in designing intelligent systems with the ability to exploit the tolerance for imprecision, uncertainty and partial truth, to achieve tractability, robustness and low solution cost. A combination of hard and soft computing should give combined improvements of computing system's performances.

COMPUTATIONAL INTELLIGENCE

Computational intelligence (CI) is a set of nature-inspired computational methodologies and approaches to address complex real-world problems to which traditional approaches, i.e., first principles modeling or explicit statistical modeling, are ineffective or infeasible. Many such real-life problems are not considered to be well-posed problems mathematically, but nature provides many counterexamples of biological systems exhibiting the required function, practically. For instance, the human body has about 200 joints (degrees of freedom), but humans have little problem in executing a target movement of the hand, specified in just three Cartesian dimensions. Even if the torso were mechanically fixed, there is an excess of 7:3 parameters to be controlled for natural arm movement. Traditional models also often fail to handle uncertainty, noise and the presence of an ever-changing context. Computational Intelligence provides solutions for such and other complicated problems and inverse problems. It primarily includes artificial neural networks, evolutionary computation and fuzzy logic. In addition, CI also embraces biologically inspired algorithms such as swarm intelligence and artificial immune systems, which can be seen as a part of evolutionary computation, and includes broader fields such as image processing, data mining, and natural language processing. Furthermore other formalisms: Dempster–Shafer theory, chaos theory and many-valued logic are used in the construction of computational models.

What Is Computational Intelligence?

Computational intelligence is the study of the design of intelligent agents. An agent is something that acts in an environment—it does something. Agents include worms, dogs, thermostats, airplanes, humans, organizations, and society. An intelligent agent is a system that acts intelligently: What it does is appropriate for its circumstances and its goal, it is flexible to changing environments and changing goals, it learns from experience, and it makes appropriate choices given perceptual limitations and finite computation. The central scientific goal of computational intelligence is to understand the principles that make intelligent behavior possible, in natural or artificial systems. The main hypothesis is that reasoning is computation. The central engineering goal is to specify methods for the design of useful, intelligent artifacts.

Artificial or Computational Intelligence ?

Artificial intelligence (AI) is the established name for the field we have defined as computational intelligence (CI), but the term "artificial intelligence" is a source of much confusion. Is artificial intelligence real intelligence? Perhaps not, just as an artificial pearl is a fake pearl, not a real pearl. "Synthetic intelligence" might be a better name, since, after all, a synthetic pearl may not be a natural pearl but it is a real pearl. However, since we claimed that the central scientific goal is to understand both natural and artificial (or synthetic) systems, we prefer the name "computational intelligence." It also has the advantage of making the computational hypothesis explicit in the name. The confusion about the field's name can, in part, be attributed to a confounding of the field's purpose with its methodology. The purpose is to understand how intelligent behavior is possible.

The methodology is to design, build, and experiment with computational systems that perform tasks commonly viewed as intelligent. Building these artifacts is an essential activity since computational intelligence is, after all, an empirical science; but it shouldn't be confused with the scientific purpose. Another reason for eschewing the adjective "artificial" is that it connotes simulated intelligence. Contrary to another common misunderstanding, the goal is not to simulate intelligence. The goal is to understand real (natural or synthetic) intelligent systems by synthesizing them. A simulation of an earthquake isn't an earthquake; however, we want to actually create intelligence, as you could imagine creating an earthquake. The misunderstanding comes about because most simulations are now carried out on computers. However, you shall see that the digital computer, the archetype of an interpreted automatic, formal, symbol-manipulation system, is a tool unlike any other: It can produce the real thing.

GRANULAR COMPUTING

Granular Computing (GrC) is an emerging computing paradigm of information processing. It concerns the processing of complex information entities called information granules, which arise in the process of data abstraction and derivation of knowledge from information or data. Generally speaking, information granules are collections of entities that usually originate at the numeric level and are arranged together due to their similarity, functional or physical adjacency, indistinguishability, coherency, or the like.

At present, granular computing is more a theoretical perspective than a coherent set of methods or principles. As a theoretical perspective, it encourages an approach to data that recognizes and exploits the knowledge present in data at various levels of resolution or scales. In this sense, it encompasses all methods which provide flexibility and adaptability in the resolution at which knowledge or information is extracted and represented. It provides a conceptual framework for studying many issues in data mining. It is demonstrated that one of the fundamental tasks of data mining is searching for the right level of granularity in data and knowledge representation.
The concept of granular computing has been defined and studied by many authors from different points of views, using different notions, based on different conceptual models, and in different contexts. Although a concise and precise definition of granular computing is desirable, any such a definition may unnecessarily limit its scope, generality and potential. For the time being, it is suffice to rely on our intuitive interpretation. Broadly speaking, granular computing may be considered as a label of a new field of multi-disciplinary study, dealing with theories, methodologies, techniques, and tools that make use of granules in the process of problem solving. While concrete granular computing models have been proposed, there is still a lack of a well-accepted framework. It is evident that we must describe and study granular computing from many perspectives, in a wider context, and independent of any particular problem domain. The results from recent studies seem to converge to a view that granular computing provides a common unifying, conceptual framework for modeling human thinking and problem-solving.

Granulation:

Granulation involves the construction of the basic components of granular computing, namely, granules, granulated views, web of granules, and hierarchies. Issues involved are:

– granulation criteria;
– granulation algorithms/methods;
– representation/description of granules and granular structures;
– qualitative/quantitative characterization of granule and granular structures.

Granulation Criteria: A granulation criterion deals with the semantic interpretation of granules and addresses the question of why two objects are put into the same granule. It is domain specific and relies on the available knowledge. In many situations, objects are usually grouped together based on their relationships, such as indistinguishability, similarity, proximity, or functionality. One needs to build models to provide both semantical and operational interpretations of these notions. A model enables us to define formally and precisely various notions involved, and to study systematically the meanings and rationalities of granulation criteria.

Granulation Methods: From the algorithmic aspect, a granulation method addresses the problem of how to put two objects into the same granule. It is necessary to develop algorithms for constructing granules and granulations efficiently based on a granulation criterion. The construction process can be modeled as either top-down or bottom-up. In a top-down process, the universe is decomposed into a family of subsets, each subset can be further decomposed into smaller subsets. In a bottom-up process, a subset of objects can be grouped into a granule, and smaller granules can be further grouped into larger granules. Both processes lead naturally to a hierarchical organization of granules and granulations.

Computing and reasoning with granules

Computing and reasoning with granules depend on the previously discussed notion of granulations. They can be similarly studied from both the semantic and algorithmic perspectives. One needs to design and interpret various methods based on the interpretation of granules and relationships between granules, as well as to define and interpret operations of granular computing. The two level structures, the granule level and the granulation level, provide the inherent relationships that can be explored in problem solving. The granulated view summarizes available information and knowledge about the universe. As a basic task of granular computing, one can examine and explore further relationships between granules at a lower level, and relationships between granulations at a higher level. The relationships include closeness, dependency, and association of granules and granulations. Such relationships may not hold fully and certain measures can be employed to quantify the degree to which the relationships hold. This allows the possibility to extract, analyze and organize information and knowledge through relationships between granules and between granulations.

Different interpretations of granular computing

Granular computing can be conceived as a framework of theories, methodologies, techniques, and tools that make use of information granules in the process of problem solving. In this sense, granular computing is used as an umbrella term to cover topics that have been studied in various fields in isolation. By examining all of these existing studies in light of the unified framework of granular computing and extracting their commonalities, it may be possible to develop a general theory for problem solving.

In a more philosophical sense, granular computing can describe a way of thinking that relies on the human ability to perceive the real world under various levels of granularity (i.e., abstraction) in order to abstract and consider only those things that serve a specific interest and to switch among different granularities. By focusing on different levels of granularity, one can obtain different levels of knowledge, as well as a greater understanding of the inherent knowledge structure. Granular computing is thus essential in human problem solving and hence has a very significant impact on the design and implementation of intelligent systems.

SOFT COMPUTING

In computer science, soft computing is the use of inexact solutions to computationally hard tasks such as the solution of NP-complete problems, for which there is no known algorithm that can compute an exact solution in polynomial time. Soft computing differs from conventional (hard) computing in that, unlike hard computing, it is tolerant of imprecision, uncertainty, partial truth, and approximation. In effect, the role model for soft computing is the human mind.

Soft computing (SC) solutions are unpredictable, uncertain and between 0 and 1. Soft Computing became a formal area of study in Computer Science in the early 1990s. Earlier computational approaches could model and precisely analyze only relatively simple systems. More complex systems arising in biology, medicine, the humanities, management sciences, and similar fields often remained intractable to conventional mathematical and analytical methods. However, it should be pointed out that simplicity and complexity of systems are relative, and many conventional mathematical models have been both challenging and very productive. Soft computing deals with imprecision, uncertainty, partial truth, and approximation to achieve practicability, robustness and low solution cost. As such it forms the basis of a considerable amount of machine learning techniques. Recent trends tend to involve evolutionary and swarm intelligence based algorithms and bio-inspired computation.

There are main difference between soft computing and possibility. Possibility is used when we don't have enough information to solve a problem but soft computing is used when we don't have enough information about the problem itself. These kinds of problems originate in the human mind with all its doubts, subjectivity and emotions; an example can be determining a suitable temperature for a room to make people feel comfortable.

Components of Soft Computing

Components of soft computing include:

- Firefly algorithm
- Cuckoo search
- Evolutionary algorithms
- Genetic algorithms
- Differential evolution
- Ideas about probability including:
- Bayesian network
- Chaos theory
- Neural networks (NN)
- Perceptron
- Support Vector Machines (SVM)
- Fuzzy logic (FL)
- Evolutionary computation (EC), including:
- Metaheuristic and Swarm Intelligence
- Ant colony optimization
- Particle swarm optimization

Generally, soft computing techniques resemble biological processes more closely than traditional techniques, which are largely based on formal logical systems, such as sentential logic and predicate logic, or rely heavily on computer-aided numerical analysis (as in finite element analysis). Soft computing techniques are intended to complement each other.

Unlike hard computing schemes, which strive for exactness and full truth, soft computing techniques exploit the given tolerance of imprecision, partial truth, and uncertainty for a particular problem. Another common contrast comes from the observation that inductive reasoning plays a larger role in soft computing than in hard computing.

Soft Computing is the fusion of methodologies that were designed to model and enable solutions to real world problems, which are not modeled or too difficult to model, mathematically. Soft computing is a consortium of methodologies that works synergistically and provides, in one form or another, flexible information processing capability for handling real-life ambiguous situations. Its aim is to exploit the tolerance for imprecision, uncertainty, approximate reasoning and partial truth in order to achieve tractability, robustness and low-cost solutions. The guiding principle is to devise methods of computation that lead to an acceptable solution at low cost, by seeking for an approximate solution to an imprecisely or precisely formulated problem. Soft computing differs from conventional (hard) computing. Unlike hard computing, it is tolerant of imprecision, uncertainty, partial truth and approximation. In effect, the role model for soft computing is the human mind. Soft Computing is basically optimization technique to find solution of problems which are very hard to answer.

COMPARATIVE APPROACH OF ALL DISCIPLINES

Computational Intelligence (CI) is a very young discipline. Other disciplines as diverse as granular computing philosophy, neurobiology, evolutionary biology, psychology, economics, political science, sociology, anthropology, control engineering, and many more have been studying intelligence much longer.

Success in building an intelligent agent naturally depends on the problem that one selects to investigate. Some problems are very well-suited to the use of computers, such as sorting a list of numbers. Others seem not to be, such as changing a baby's diaper or devising a good political strategy. We have chosen some problems that are representative of a range of applications of current CI techniques. We seek to demonstrate, by case study, CI's methodology with the goal that the methodology is transferable to various problems in which you may be interested. We establish a framework that places you, the reader, in a position to evaluate the current CI literature and anticipate the future; and, most importantly, we develop the concepts and tools necessary to allow you to build, test, and modify intelligent agents. Finally we must acknowledge there is still a huge gulf between the dream of computational intelligence and the current technology used in the practice of building what we now call intelligent agents. We believe we have many of the tools necessary to build intelligent agents, but we are certain we don't have all of them. We could, of course, be on the wrong track; it is this fallibility that makes CI science and makes the challenge of CI exciting.

Granular computing adopts a structured combination of algorithmic and non-algorithmic information processing that mimics human, intelligent synthesis of knowledge from information. By integrating various different agents in which each pursues its own agenda, exploits its environment, develops its own problem solving strategy and establishes required communication strategies, one may form a more effective human-centered information system. In fact, each agent may encounter a diversity of problem-solving approaches and realize their processing at the level of information granules that is the most suitable from their local points of view. To this level, the hybrid model raises a fundamental issue of forming effective interaction linkages between the agents so that they fully broadcast their findings and benefit from interacting with others.

Soft computing is based on natural as well as artificial ideas. It is referred as a computational intelligence. It differs from conventional computing that is hard computing. It is tolerance of imprecision, uncertainty, partial truth to achieve tractability, approximation, robustness, low solution cost, and better rapport with reality. In fact the role model for soft computing is human mind. It refers to a collection of computational techniques in computer science, artificial intelligence, machine learning applied in engineering areas such as Aircraft, spacecraft, cooling and heating, communication network, mobile robot, inverters and converters, electric power system, power electronics and motion control etc. Traditionally soft computing has been comprised by four technical disciplines. The first two, probabilistic reasoning (PR), and fuzzy logic (FL) reasoning systems, are based on knowledge-driven reasoning. The other two technical disciplines, Neuro Computing (NC) and Evolutionary Computing (EC), are data – driven search and optimization approaches.

CONCLUSION

The characteristic of "intelligence" is usually attributed to humans. More recently, many products and items also claim to be "intelligent". Intelligence is directly linked to the reasoning and decision making. Granular Computing forms a unified conceptual and computing platform. Yet, it directly benefits from the already existing and well-established concepts of information granules formed in the setting of set theory, fuzzy sets, rough sets and others. The successful applications of soft computing and the rapid growth suggest that the impact of soft computing will be felt increasingly in coming years. It encourages the integration of soft computing techniques and tools into both every day and advanced applications. Granular computing becomes a layer of computational intelligence, a level of abstraction above soft computing.

REFERENCES

[1]. Fogel, L.J.; A. J. Owens; M. J. Walsh (1966). Artificial Intelligence through Simulated Evolution. New York: John Wiley.
[2]. B. Dengiz, F. Altiparmak, and A. E. Smith (1997), "Local search genetic algorithm for optimal design of reliable networks," IEEE Trans. Evol. Comput., vol. 1, pp. 179–188.
[3]. D. K. Chaturvedi,Soft Computing: Techniques and Its Applications in Electrical Engineering, Springer, (2008).
[4]. Rencher, Alvin C. (2002), *Methods of Multivariate Analysis*, New York: Wiley.
[5]. Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2001), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, New York: Springer, ISBN 978-0-387-84857-0.
[6]. Liao Ying, Zhang Shaoyong, Yi Dawei, Yi Jiawa, Faults diagnosis of diesels based on rough set genetic neural network, Control Eng China, 2009, 16(6): 709-712.

[7]. Pedrycz W., Vukovich G., Granular neural networks, Neurocomputing, 2001, 36(1-4): 205-224.
[8]. D. Kim (2000), “An implementation of fuzzy logic controller on the reconfigurable FPGA system,” IEEE Trans. Ind. Electron., 47, pp703–715.
[9]. P. Baranyi, I. Nagy, P. Korondi, and H. Hashimoto (2000), “General guiding model for mobile robots and its complexity reduced neuro-fuzzy approximation,” in Proc. IEEE Int. Conf. Fuzzy Systems, San Antonio,TX, pp 1029–1032.
[10]. T. Nitta (1993), “Applications of neural networks to home appliances,” in Proc. IEEE Int. Joint Conf. Neural Networks, Nagoya, Japan, 1993, pp. 1056–1060.
[11]. W. Pedrycz, F. Gomide, Fuzzy Systems Engineering: Toward Human-Centric Computing, John Wiley, Hoboken, NJ, 2007 .
[12]. W. Pedrycz and A. Bargiela, Granular clustering: a granular signature of data, IEEE Trans. Systems, Man and Cybernetics, Vol. 32, No. 2, pp. 212-224, 2002.
[13]. Pedrycz, W.: From Granular Computing to Computing Intelligence and Human-centric Systems, IEEE conneCtIonS, Vol. 3, No. 2, May 2005, pp.6-11.
[14]. Polkowski, L., Skowron, A.: Towards adaptive calculus of granules. Proceedings of 1998 IEEE International Conference on Fuzzy Systems (1998) 111-116.
[15]. Inuiguchi, M., Hirano, S., Tsumoto, S. (Eds.): Rough Set Theory and Granular Computing. Springer, Berlin (2003).