Auto-Algorithm Selection Technique for Computational Intelligence in Cloud-Based Computing Environment

Dr.M.P.Revathi,
Professor, Department of Computer Science and Engineering, J.J. College of Engineering and Technology, Trichy, Tamilnadu
G.Keerthana,
Assistant Professor, Department of Computer Science and Engineering, J.J. College of Engineering and Technology, Trichy, Tamilnadu
J.S.Jaslin,
Assistant Professor, Department of Computer Science and Engineering, J.J. College of Engineering and Technology, Trichy, Tamilnadu
T.Vency Stephisia,
Assistant Professor, Department of Computer Science and Engineering, J.J. College of Engineering and Technology, Trichy, Tamilnadu

Abstract

This research presents an Auto-Algorithm selection technique for Computational Intelligence in Cloud-Based computing environment. By leveraging the capabilities of a network information processing platform, users can effortlessly and intelligently build Computational Intelligence models tailored to their specific problems without the need for manually configuring the Computational Intelligence environment, selecting algorithms, or adjusting complex Computational Intelligence functions and parameters. The proposed procedure allows users to simply upload sample data through a web interface, freeing Computational Intelligence applications from environmental constraints and taking advantage of the network information processing platform's capabilities. This approach transparently handles the model building process, significantly reducing the barriers to entry for utilizing Computational Intelligence. By addressing the issues of unpredictable model selection, manual parameter adjustment, and the challenges faced by common users, this auto selection procedure empowers the practical application of Computational Intelligence in various domains.

Keywords: Computational Intelligence, Network information processing, Auto-selection, Model building, Parameter adjustment, User-friendly.
DOI Number: 10.48047/nq.2020.18.8.nq20220 **NeuroQuantology 2020;18(8):162-167**

Introduction

Computational Intelligence has gained significant attention in recent years due to its ability to analyze large datasets and make predictions or derive valuable insights. However, the process of building Computational Intelligence models requires expertise in selecting appropriate algorithms, adjusting parameters, and managing the computational resources necessary for training.[1] This complexity poses a challenge for users with limited knowledge of Computational Intelligence techniques and resources.Network information processing platforms offer a promising solution by providing on-demand access to scalable computing resources. Leveraging the capabilities of network information processing, this research aims to develop an auto-selection procedure for Computational

Intelligence models that overcomes the challenges associated with manual model building and resource management.[2]

The proposed procedure allows users to build Computational Intelligence models intelligently and systematically within the network information processing environment. By utilizing a web-based interface, users can upload their sample data and have the system systematically select the most suitable Computational Intelligence algorithm, adjust the necessary parameters, and build the model tailored to their specific problem domain.[3] This approach eliminates the need for users to manually set up the Computational Intelligence environment, select algorithms, or fine-tune complex parameters.The main objective of this research is to reduce the barrier to entry for Computational Intelligence applications by simplifying the model building process in a network information processing environment. By leveraging the advantages of network information processing platforms, such as scalability and ease of access, the proposed auto-selection procedure aims to make Computational Intelligence more accessible to a broader range of users, including those without extensive Computational Intelligence expertise.

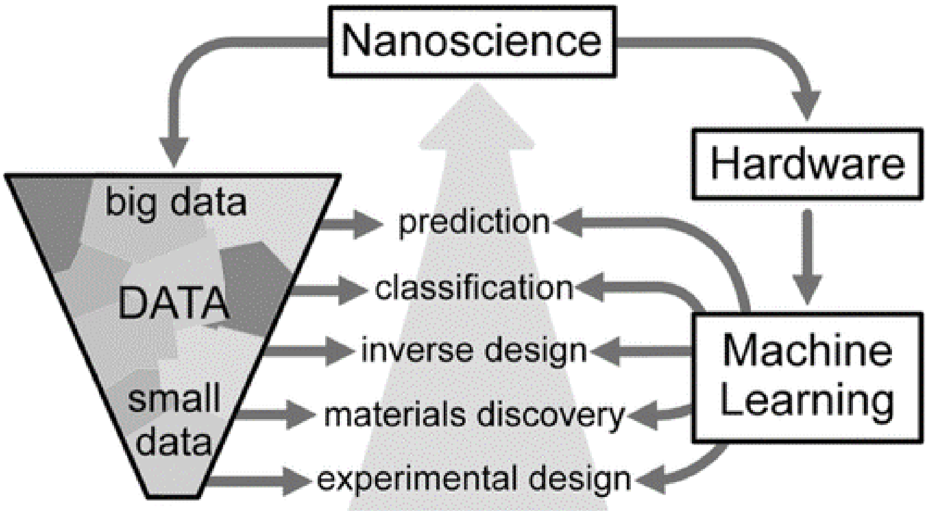

Figure 1. The scope of machine learning methods to progress "safe-by-design" nanotechnology

The extensive application of high throughput synthesis and characterization techniques in the field of nanomaterials has been relatively limited compared to other domains of materials science. As a result, the adoption of machine learning (ML) approaches for analysing nanomaterials datasets has not gained significant traction. However, it is undeniable that these empowering technologies will witness broader implementation in the coming years. This will be facilitated by harnessing the advancements made in high throughput synthesis and characterization methodologies, as well as ML modelling techniques originally developed for bulk materials (**Figure 1**).

The auto-selection procedure for Computational Intelligence in a network information processing environment addresses the challenges associated with model building selection, parameter adjustment, and the complexities faced by common users when applying Computational Intelligence in real-life scenarios.[5] By providing a transparent and user-friendly approach, this research aims to empower users to leverage Computational Intelligence techniques effectively and efficiently, leveraging the advantages of network information processing platforms.

Related Work

Computational Intelligence is a core research topic in the field of artificial intelligence and has emerged as a significant application alongside expert systems and other AI applications. Its aim is to enable processing machines to simulate or mimic human knowledge acquisition, thereby acquiring knowledge or improving their performance based on new information. The ability of Computational Intelligence to adapt and improve is a crucial feature, as highlighted by H.A. Simon, who views learning as the system's ability to achieve the same or better results in similar tasks. R.S. Michalski, on the other hand, considers learning as the process of constructing or revising experiences. From the perspective of knowledge engineering, those involved in the development of expert systems emphasize the acquisition of knowledge.[4]

These different viewpoints emphasize various aspects, with the first emphasizing external behavioral effects, the second focusing on the internal learning process, and the third highlighting the practicality of knowledge engineering.The research procedureology in Computational Intelligence draws insights from fields such as physiology, psychology, and cognitive science to understand the self-teaching mechanisms of humans. Based on this understanding, computational models or cognitive models of human knowledge processes are developed to formulate various learning theories and procedures. The goal is to establish learning systems that are tailored to specific applications. These research goals influence and reinforce each other, leading to the rapid development of Computational Intelligence as a central topic since the first Computational Intelligence scientific seminar held by KaNeiji-Mei Long University in 1980.[6]

The history of Computational Intelligence's development can be divided into four stages: the fervent period of the mid-1950s to the mid-1960s, the calm period of the mid-1960s to the mid-1970s, the recovery period of the mid-1970s to the mid-1980s, and the current stage since 1986, which represents the latest advancements in Computational Intelligence. At present, Computational Intelligence has become an emerging frontier branch of science, encompassing various learning procedures.[7] Its range of applications continues to expand, and academic activities in the field are highly active.Computational Intelligence has gained widespread usage, with numerous notable algorithms developed. These algorithms can be broadly classified into symbol-based and non-symbol-based learning. The former includes algorithms such as decision trees, genetic algorithms, Bayesian statistics, artificial neural networks, support vector machines, and association rules. This paper proposes a procedure that combines these common algorithms, namely the MBM (Multiple Base Models) procedure, and uses the Expectation-Maximization (EM) algorithm for parameter estimation.

However, when applying Computational Intelligence techniques to specific tasks, three main challenges arise. First, setting up a Computational Intelligence model for a specific task can be time-consuming and energy-intensive, as each task has unique details that make it difficult to directly reference pre-existing system models. Second, even for subtasks, selecting the most suitable Computational Intelligence algorithm and arranging its complex parameters pose challenges that may require heuristic approaches or extensive computation. Third, users who need to address problems quickly may struggle with learning and using specific Computational Intelligence software, as the algorithms are numerous and complex, requiring significant time for autonomous learning. Additionally, different users may have varying degrees of familiarity with specific algorithms, adding to the complexity of selecting the appropriate approach for each task.

The emergence of network information processing technology provides a potential solution to these challenges, making Computational Intelligence more accessible and enabling faster and better value creation. Network information processing, based on distributed systems and grid computing, offers a shared architecture and a service model that provides on-demand access to computational resources and data storage. This model allows users to easily access services through networks, with the ability to scale resources as needed. Network information processing ensures reliable and secure data storage, alleviating user concerns about data loss and security breaches.Moreover, network information processing minimizes the equipment requirements on the user side, as the computational resources are consolidated into large server clusters, including computing servers, storage servers, and broadband resources. By aggregating these resources into various clouds and managing them systematically through software, network information processing eliminates the need for users to handle intricate details, allowing them to focus on their core business, promoting innovation, and reducing costs. In the context of Computational Intelligence, network information processing can provide users with pre-established Computational Intelligence models and related modules, offering a convenient solution that enables users to quickly leverage the achievements of Computational Intelligence techniques.

Research Objective

The main objective of this research is to develop an Auto-Algorithm selection technique for Computational Intelligence in Cloud-Based computing environment. The research aims to overcome the challenges associated with model building selection, manual parameter adjustment, and the complexities faced by common users when applying Computational Intelligence in real-life scenarios. By leveraging the capabilities of network information processing platforms, the research seeks to enable users to effortlessly build Computational Intelligence models that address their specific problems without requiring deep expertise in Computational Intelligence algorithms and configuration. The objective is to provide a user-friendly, transparent approach to Computational Intelligence model building, reducing the barriers to entry and facilitating wider adoption of Computational Intelligence techniques across diverse domains.

Auto-Selection Procedure for Computational Intelligence in Network information processing Environment

The Auto-Algorithm selection technique for Computational Intelligence in Cloud-Based computing environment has the following steps:

Step 1) The individual engages with a web interface offered by the cloud management module to input a general explanation of the issue they wish to address. They choose the issue classification from choices such as intelligent systems, cognitive modelling, decision-making, knowledge discovery, network information retrieval, pattern analysis, anomaly detection, comprehension of human language, automated machinery, interactive entertainment, or alternative categorizations.

Step 2) The primarymodelling cloud is activated based on the user's selected problem category. The user fills in more detailed information, including uploading sample data, selecting expression procedures, determining result interpretation procedures, specifying usability range, and defining expectations.

Step 3) The procedure starts searching for suitable Computational Intelligence algorithms based on the information provided by the user. It compares the user's input with historical examples and determines which Computational Intelligence algorithm should be used. This cloud module continuously adjusts its approach based on the results obtained at each stage.

Step 4) The user's information from Step 2 is inputted into the Computational Intelligence input/output module. The data is standardized, processed to handle missing values and noise, scrubbed, integrated, transformed, and reduced to obtain intermediate results suitable for general algorithms.

Step 5) The valuation functions cloud is activated to set up valuation functions based on the user's input from Step 2. These functions evaluate the quality of the Computational Intelligence solution and predict the performance of specific algorithms.

Step 6) The procedure utilizes the EM algorithm and support cloud to estimate the maximum likelihood in the solution space. It calculates the approximate location of the optimal or better solution, thereby increasing search efficiency.

Step 7) Using the results from the previous steps, the procedure proceeds to the training process of Computational Intelligence. It systematically decides which specific Computational Intelligence cloud modules to call, such as decision tree algorithms, genetic algorithms, Bayesian statistics algorithms, artificial neural network algorithms, support vector machine algorithms, association rule algorithms, or any user-defined Computational Intelligence algorithm.

Step 8) Based on the calculations from Step 7, one or several algorithm clouds are selected and activated. The results and intermediate solutions are communicated back to the user through the web interface, including the calculation steps, obtained intermediate results, and changes in the current optimal solution.

Step 9) The process iterates through Steps 6, 7, and 8, while also evaluating the end condition. If the end condition is met or the computing time is reached without finding a better solution, the iteration stops. The performance prediction algorithm formulated in Step 5 is used to judge the quality of the solution. This step requires significant computational resources, which can be efficiently utilized through network information processing, maximizing the chances of finding an outstanding solution.

Step 10) When the end condition is satisfied, and if there is no better solution or further iteration is not possible, the calculated results are converted into easily understandable information using the Computational Intelligence input/output module. The results are then returned to the client through the web interface, and detailed data can be downloaded and preserved for future reuse, avoiding duplication of efforts.

Conclusion

In conclusion, this research proposes an Auto-Algorithm selection technique for Computational Intelligence in Cloud-Based computing environment. By utilizing the capabilities of network information processing platforms, the procedure allows users to build Computational Intelligence models without the need for manual configuration, algorithm selection, or complex parameter adjustment. This approach addresses the limitations and challenges faced by users when applying Computational Intelligence in real-life situations. The procedure transparently handles the model building process, making it more accessible and user-friendly, ultimately reducing the threshold for utilizing Computational Intelligence techniques. By empowering users to leverage Computational Intelligence in a network information processing environment, this research contributes to overcoming the unpredictability of model selection, manual parameter adjustment, and other difficulties encountered in practical Computational Intelligence applications.

Reference

1. Ampatzidis, Y., Partel, V., & Costa, L. (2020). Agroview: Cloud-based application to process, analyze and visualize UAV-collected data for precision agriculture applications utilizing artificial intelligence. Computers and Electronics in Agriculture, 174, 105457. https://doi.org/10.1016/j.compag.2020.105457
2. W. Tong, A. Hussain, W. X. Bo and S. Maharjan, "Artificial Intelligence for Vehicle-to-Everything: A Survey," in IEEE Access, vol. 7, pp. 10823-10843, 2019, doi: 10.1109/ACCESS.2019.2891073.
3. S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar and A. Y. Zomaya, "Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence," in IEEE Internet of Things Journal, vol. 7, no. 8, pp. 7457-7469, Aug. 2020, doi: 10.1109/JIOT.2020.2984887.
4. Yang Lu (2019) Artificial intelligence: a survey on evolution, models, applications and future trends, Journal of Management Analytics, 6:1, 1-29, DOI: 10.1080/23270012.2019.1570365
5. Pan, Y., & Zhang, L. (2021). Roles of artificial intelligence in construction engineering and management: A

critical review and future trends. Automation in Construction, 122, 103517. https://doi.org/10.1016/j.autcon.2020.103517
6. Cui, M., & Zhang, D. Y. (2021). Artificial intelligence and computational pathology. Laboratory Investigation, 101(4), 412-422. https://doi.org/10.1038/s41374-020-00514-0
7. Y. Ma, Z. Wang, H. Yang and L. Yang, "Artificial intelligence applications in the development of autonomous vehicles: a survey," in IEEE/CAA Journal of AutomaticaSinica, vol. 7, no. 2, pp. 315-329, March 2020, doi: 10.1109/JAS.2020.1003021.
8. X. Wang, X. Li and V. C. M. Leung, "Artificial Intelligence-Based Techniques for Emerging Heterogeneous Network: State of the Arts, Opportunities, and Challenges," in IEEE Access, vol. 3, pp. 1379-1391, 2015, doi: 10.1109/ACCESS.2015.2467174.
9. Andronie, M., Lăzăroiu, G., Iatagan, M., Uță, C., Ștefănescu, R., &Cocoșatu, M. (2020). Artificial Intelligence-Based Decision-Making Algorithms, Internet of Things Sensing Networks, and Deep Learning-Assisted Smart Process Management in Cyber-Physical Production Systems. Electronics, 10(20), 2497. https://doi.org/10.3390/electronics10202497
10. H. Fatemidokht, M. K. Rafsanjani, B. B. Gupta and C. -H. Hsu, "Efficient and Secure Routing Protocol Based on Artificial Intelligence Algorithms With UAV-Assisted for Vehicular Ad Hoc Networks in Intelligent Transportation Systems," in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 7, pp. 4757-4769, July 2021, doi: 10.1109/TITS.2020.3041746.
11. Bischl, B., Kerschke, P., Kotthoff, L., Lindauer, M., Malitsky, Y., Fréchette, A., Hoos, H., Hutter, F., Leyton-Brown, K., Tierney, K., &Vanschoren, J. (2016). ASlib: A benchmark library for algorithm selection. Artificial Intelligence, 237, 41-58. https://doi.org/10.1016/j.artint.2016.04.003