The International Journal of Engineering and Science (IJES)
|| Volume || 13 || Issue || 12 || Pages || PP 32-36 || 2024 ||
ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805

Exploring Artificial Intelligence and Machine Learning Programming for Automated Problem-Solving Simulations: A Review

Shakila Begam[1], Mr. Kamlesh Kumar Yadav[2]

Faculty of Computer Science and Information Technology[1,2],
Kalinga University, Raipur, Chhattisgarh, India

ABSTRACT

This study explores the use of an AI-driven machine learning approach implemented in current scenario for solving the real world problem. A machine learning based code needs a similar datasets of the particular problem so that it will work and behave the same as what human think i.e. general intelligence. This paper presents a review of past research on the aforementioned topic and draws conclusions based on the reviews. The major contributors of the different studies shows the multi-layer perceptron neural network (MLPNN), convolutional neural network (CNN), and recurrent neural network (RNN), applied to one-dimensional, two-dimensional, and time-series data for application of understanding by the code. It is concluded that the selection of AI based programming language should be selected first that learn the datasets and respond according to general intelligence and then convert it into artificial intelligence.

Keywords— Machine Learning, Multistory Buildings, Optimization, Data Analysis, Computational Efficiency

Date of Submission: 01-12-2024 Date of acceptance: 10-12-2024

I. INTRODUCTION

The transformation of problem-solving methods across various fields has been driven by the integration of artificial intelligence (AI) and machine learning (ML). Complex tasks are now being automated, with traditional approaches being replaced by efficient, scalable solutions. Through AI-based ML programming, new opportunities for automated simulations have been uncovered, enabling problems to be addressed with remarkable ease and accuracy. This process reduces manual involvement while fostering innovation in fields such as engineering, healthcare, and economics.

The feasibility of utilizing AI in simulation is demonstrated by its ability to process extensive datasets, recognize patterns, and generate optimized solutions. Unlike static programming methods, AI systems are designed to adapt to changing conditions and continuously improve their performance through iterative learning processes. Such adaptability makes these systems highly effective for addressing intricate and dynamic challenges.

The practicality of AI systems has been significantly enhanced by advances in algorithms and computing capabilities. Techniques such as deep learning, reinforcement learning, and natural language processing have been applied to broaden the horizons of simulation and predictive modeling. These developments emphasize the growing relevance of AI as a pivotal focus for future innovation and research.

In this paper, the use of AI-based ML programming for automated problem-solving will be analyzed, with key frameworks, successful applications, and existing challenges being explored. Insights into the strengths and limitations of AI-driven simulation systems will be provided, offering a foundation for the development of intelligent solutions to complex issues.

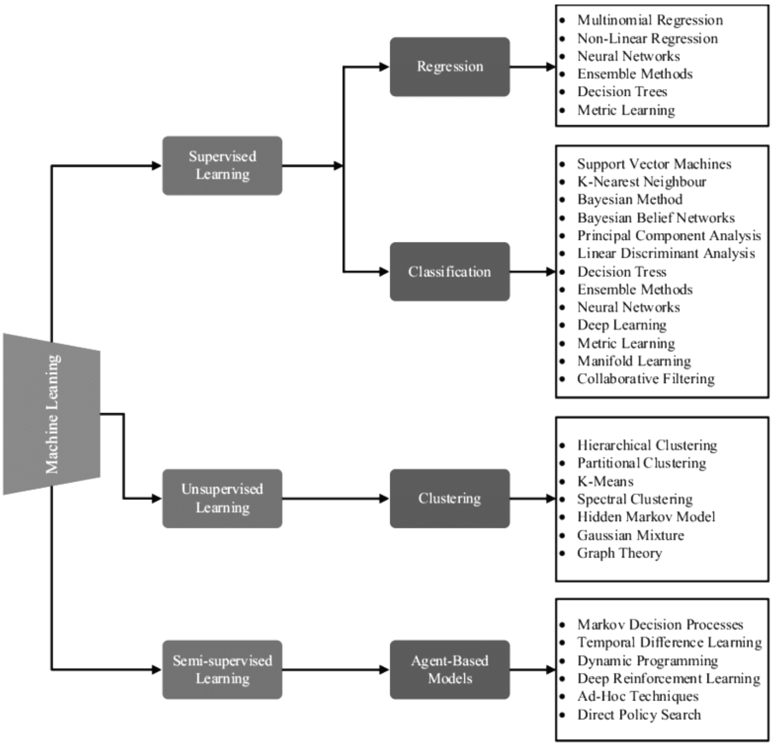

Fig. 1: Machine learning structure

II. LITERATURE REVIEW

The following literature articles have been selected for the current research study based on usage of programming language as AI and simulate that code for machine learning pragramming. The individual summaries are as follows:

Raimondas Sasnauskas et. al. (2016)

In this research, the compiler optimizations can be automatically derived, some of the substantial engineering challenges associated with creating and maintaining a high-quality compiler might be sidestepped has discussed. Souper, a synthesizing superoptimizer, was developed to explore how far these ideas could be extended within the context of LLVM. During this process, it was discovered that Souper's intermediate representation was sufficiently similar to the one used in Microsoft Visual C++, enabling its application to that compiler as well. Optimizations suggested by Souper but implemented manually have been included in shipping or near-shipping versions of both compilers. Additionally, when Souper is utilized as a fully automated optimization pass, a Clang compiler binary approximately 3MB (4.4%) smaller than the one compiled by LLVM is produced.

Noam Shazeer et. al. (2019)

In this research, Multi-head attention layers, as employed in the Transformer neural sequence model, are regarded as a powerful alternative to RNNs for transferring information across and between sequences has discussed. Training these layers is generally considered fast and straightforward due to their parallelizability along the sequence length. However, incremental inference, where such parallelization is not feasible, is often slowed by the memory-bandwidth cost associated with repeatedly loading the large "keys" and "values" tensors. A variant called multi-query attention is proposed, in which the keys and values are shared across all attention "heads." This approach significantly reduces the size of these tensors and, consequently, the memory bandwidth requirements for incremental decoding. Experimental verification has confirmed that the resulting models are capable of much faster decoding while incurring only minor quality degradation compared to the baseline.

Mark Chen et. al. (2021)

In this research, Codex, a GPT language model fine-tuned on publicly available code from GitHub, is introduced, with its Python code-writing capabilities has discussed. A distinct production version of Codex powers GitHub Copilot. On HumanEval, a newly released evaluation set designed to measure functional correctness for program synthesis from docstrings, the model solves 28.8% of the problems, compared to 0% by

GPT-3 and 11.4% by GPT-J. Additionally, repeated sampling from the model is found to be an unexpectedly effective strategy for generating working solutions to complex prompts. Using this approach, 70.2% of the problems are solved with 100 samples per problem. A detailed investigation of the model reveals limitations, such as challenges with docstrings describing lengthy chains of operations and difficulties in binding operations to variables. Finally, the broader implications of deploying advanced code generation technologies are discussed, including considerations related to safety, security, and economics.

Hammond Pearce et. al. (2022)

In this research, code developed by human programmers can often contain cybersecurity vulnerabilities has discussed. Emerging "smart" code completion tools are being evaluated for their potential to repair such bugs. This work investigates the use of large language models (LLMs), such as OpenAI's Codex and AI21's Jurassic J-1, for zero-shot vulnerability repair. Challenges in designing prompts to guide LLMs in generating secure and repaired versions of vulnerable code are explored, given the complexity and variability in phrasing key information both semantically and syntactically in natural language. A large-scale study is conducted using five commercially available, black-box, "off-the-shelf" LLMs, along with an open-source model and a locally trained model. The evaluation covers synthetic, handcrafted, and real-world security bug scenarios. Results show that the approach holds promise, with LLMs collectively repairing 100% of synthetically generated and handcrafted scenarios. However, a qualitative analysis of the models' performance on historical real-world examples reveals significant challenges in generating functionally correct and secure code.

Alexandrina-Elena Andon et. al. (2022)

In this research, Over the past five years, image processing using artificial neural networks (ANN) has garnered significant interest among researchers in the field of Civil Engineering has discussed. As traditional artificial neural networks, composed of several neurons, may struggle to extract features from images due to high computational requirements, the Convolutional Neural Network (CNN) was developed. CNN, a machine learning algorithm, processes images as input and assigns importance to various aspects (such as objects within the image) to enable differentiation. Recorded results in the Civil Engineering domain have demonstrated notable success in employing this approach.

Arshad Shaikh et. al. (2024)

This research aims to ensure that raisins meet high-quality standards, thereby maintaining consumer satisfaction. Sorting plays a crucial role in eliminating defective raisins, ensuring that only the best quality reaches the market. The manual sorting process is time-consuming and prone to inaccuracies, which may result in improper differentiation between good and bad raisins. Automated sorting addresses these issues by ensuring uniformity in size, color, and texture, thus meeting consumer expectations. Proper sorting is also essential for compliance with food safety standards. The project focuses on evaluating and determining the quality of raisins through thorough analysis. The goal is to establish a robust methodology for assessing raisin quality, considering factors such as color, shape, size, texture, and overall appearance. In this project, a Convolutional Neural Network (CNN) model is employed for raisin sorting. Since the CNN model requires a dataset, both good and bad quality raisin images are used to train the learning model.

Dominik Fuchsgruber et. al. (2024)

In this research, Many applications in traffic, civil engineering, and electrical engineering depend on edge-level signals, which can be either inherently directed, like water flow in a pipe network, or undirected, like the diameter of a pipe has discussed. Traditional topological methods model directed edge signals by assigning an orientation to each edge but cannot handle undirected signals or distinguish between directed and undirected edges. This work addresses these challenges by (i) revising the concept of orientation equivariance for edge direction-aware topological models, (ii) introducing orientation invariance for undirected signals, and (iii) developing EIGN, an architecture with novel direction-aware graph shift operators. EIGN is the first general-purpose topological GNN capable of modeling both directed and undirected signals while distinguishing between them. A comprehensive evaluation shows that EIGN outperforms previous methods, improving RMSE by up to 43.5% in flow simulation tasks.

Asraar Anjum et. al. (2024)

In this research, the transformative potential of optimization methodologies in civil engineering is emphasized in this review. Advanced computational techniques such as DOE, fuzzy logic, and ANN are being integrated to provide adaptable, efficient, and cost-effective solutions for structural design, material optimization, and construction methods. These techniques have been proven effective in handling uncertainties, leading to more resilient and sustainable civil structures. Innovation, reliability, and advancement in civil engineering are fostered through these methods. However, practical challenges, such as the integration of AI methods, computational resource demands, and concerns over data privacy, remain. The gap between theoretical advancements and real-world applications must be bridged through interdisciplinary collaboration. Efficient computational algorithms are required to reduce the computational load, making these tools more accessible to smaller engineering firms.

Xiaoming Li et. al. (2007)

In this research, the growing complexity of modern processors has made the development of highly efficient code increasingly difficulty has discussed. Manually developing efficient code is typically expensive but necessary due to the limitations of today's compilers. A promising automatic code generation strategy, implemented by library generators such as ATLAS, FFTW, and SPIRAL, relies on empirical search to identify code characteristics that deliver optimal performance for each target machine. This approach has primarily been applied to scientific codes optimized for machine-dependent characteristics. In this paper, the generation of sorting routines, whose performance also depends on input data characteristics, is studied. Two approaches to generate efficient sorting routines are presented. First, the best "pure" sorting algorithm is selected based on input data characteristics, with machine learning algorithms used to compute a function that selects the best algorithm at runtime. The second approach generalizes the first by building hybrid sorting algorithms using genetic algorithms and a classifier system, which adapt to input data. Results show that algorithms generated with this approach significantly outperform conventional sorting implementations, including IBM ESSL, INTEL MKL, and C++ STL, with the best algorithm being 26% and 62% faster than IBM ESSL on IBM Power 3 and Power 4, respectively.

Xu Yang et. al. (2021)

In this research, artificial neural networks (ANN) have been explored and applied to address complex problems in pavement engineering, given the significant advancements in soft computing and data science. A state-of-the-art review was conducted to survey the recent progress of ANN applications at different stages of pavement engineering, including design, construction, inspection and monitoring, and maintenance. The study focused on papers published over the last three decades, particularly those since 2013. Through literature retrieval, a total of 683 papers were identified, of which 143 were selected for an in-depth review. The ANN architectures used in these studies primarily included multi-layer perceptron neural network (MLPNN), convolutional neural network (CNN), and recurrent neural network (RNN), applied to one-dimensional, two-dimensional, and time-series data. CNN-based pavement health inspection and monitoring attracted the most research interest due to its potential to replace human labor. While ANN has proven to be effective in pavement material design, cost analysis, defect detection, and maintenance planning, it faces significant challenges in data collection, parameter optimization, model transferability, and low-cost data annotation. More attention is needed to integrate multidisciplinary techniques into pavement engineering to address existing challenges and expand future opportunities.

Daniel J. Mankowitz et. al. (2023)

In this research, fundamental algorithms such as sorting and hashing, which are used trillions of times daily, are shown to benefit from the advancements in artificial intelligence. As the demand for computation grows, the need for highly performant algorithms has become crucial. Despite remarkable progress in the past, further improvements in the efficiency of these routines have proven challenging for both human scientists and computational approaches. The task of finding better sorting routines was formulated as a single-player game, and a new deep reinforcement learning agent, AlphaDev, was trained to play this game. AlphaDev discovered small sorting algorithms from scratch that outperformed previously known human benchmarks. These algorithms have been integrated into the LLVM standard C++ sort library, replacing a component with an algorithm automatically discovered using reinforcement learning. Results from additional domains are also presented, highlighting the generality of this approach.

III. CONCLUSION AND OUTLINE OF PROPOSED WORK

It appears that a research gap exists in the literature concerning the various work previously conducted on the current topic. Further research is needed to solve real world problems. Based on the literature review, we have reached a conclusion that highlights the key findings of the research and lists the necessary outcomes:

- Selection of programming language that will be feasible for artificial intelligence with machine learning. (Example-Python, VBA, etc…)
- Selection of environment that runs the programming language code with simulation. (Example-Python launcher, Google Colaboratory etc..)

- Creation of programming code, learn the architecture and hierarchy, implement the learning based on human behavior and then with machine code.
- Optimization of code and debugging wherever possible so that code runs smoothly.

It is found that the motivation of the study is an important part in the design aspect of Machine learning and AI based program.

The primary objective of this study is to assess the feasibility of using artificial intelligence to explore the usage of AI based machine learning programming for automatic simulation of problem solving of a particularproblem and get a solutionwith ease, which will be a focal point for the upcoming proposed work.

ACKNOWLEDGEMENTS

I, Shakila Begam, M. Tech. Student, would like to thank ***Mr. Kamlesh Kumar Yadav,***AssistantProfessor, Faculty of CS and IT, Kalinga University, Naya Raipur, (C.G.), India for his valuable guidance from the commencement of the work up to the completion of the work along with his encouraging thoughts.

REFERENCES

[1] Sasnauskas, R., SES Engineering, Chen, Y., Nvidia, Inc., Collingbourne, P., Google, Inc., Ketema, J., Embedded Systems Innovation by TNO, Lup, G., Microsoft, Inc., Taneja, J., University of Utah, Regehr, J., & University of Utah. (n.d.). A Synthesizing Superoptimizer. Conference Paper.

[2] Shazeer, N. & Google. (2019). Fast Transformer Decoding: One Write-Head is All You Need. Conference Paper.

[3] Evaluating Large Language Models Trained on Code. (2021). In arXiv [Journal-article]. https://arxiv.org/pdf/2107.03374

[4] Pearce, H., Tan, B., Ahmad, B., Karri, R., New York University, & University of Calgary. (2022). Examining Zero-Shot Vulnerability Repair with Large Language Models [Journal-article]. arXiv. https://arxiv.org/abs/2112.02125v3

[5] Andon, A., & Covatariu, G. (2022). A Study on Image Processing Using Artificial Neural Networks in Civil Engineering. Buletinul Institutului Politehnic "Gheorghe Asachi" Din Iaşi. Secţia Construcţii. Arhitectură, 67(3), 85–94. https://doi.org/10.2478/bipca-2021-0027

[6] Shaikh, A., Sawant, S., Patil, A., Bhandare, A., & International Research Journal of Modernization in Engineering, Technology and Science. (2024). RAISIN SORTING USING MACHINE LEARNING ALGORITHM. International Research Journal of Modernization in Engineering Technology and Science, 06. https://www.irjmets.com

[7] Fuchsgruber, D., Poˇstuvan, T., G ¨Unnemann, S., Geisler, S., Department of Computer Science & Munich Data Science Institute, TU Munich, & EPFL. (2024). GRAPH NEURAL NETWORKS FOR EDGE SIGNALS: ORIENTATION EQUIVARIANCE AND INVARIANCE.

[8] Anjum, A., Hrairi, M., Shaikh, A. A., Yatim, N., & Ali, M. (2024). Integrating AI and statistical methods for enhancing civil structural practices: current trends, practical issues, and future direction. Frattura Ed Integrità Strutturale, 19(71), 164–181. https://doi.org/10.3221/igf-esis.71.12

[9] Li, X., Garzaran, M. J., & Padua, D. (2007). Optimizing Sorting with Machine Learning Algorithms. Conference Paper, 1 2, 1–6. https://doi.org/10.1109/ipdps.2007.370499

[10] Research and applications of artificial neural network in pavement engineering: A state-of-the-art review. (2021). In Journal of Traffic and Transportation Engineering (English Edition). http://doi.org/10.1016/j.jtte.2021.03.005

[11] Faster sorting algorithms discovered using deep reinforcement learning. (2023). Nature, 618, 257–258.