Proceedings of the DataNinja sAIOnARA 2024 Conference 43-46 DOI: 10.11576/dataninja-1167

Advancements in Neural Network Generations

Sanaullah * SANAULLAH@HSBI.DE
Bielefeld University of Applied Sciences and Arts, Bielefeld, Germany

Shamini Koravuna SKORAVUNA@TECHFAK.UNI-BIELEFELD.DE
Bielefeld University, Germany

Ulrich Rückert RUECKERT@TECHFAK.UNI-BIELEFELD.DE
Bielefeld University, Germany

Thorsten Jungeblut THORSTEN.JUNGEBLUT@HSBI.DE
Bielefeld University of Applied Sciences and Arts, Bielefeld, Germany

1. Introduction

Innovations in Neural Network Generation demonstrate the continual evolution, optimization, and development of artificial neural networks (ANNs) over periods [1]. These improvements include a combination of methodologies, approaches, and technical breakthroughs aimed at increasing the efficiency and abilities of neural network models [1]. Researchers and engineers have repeatedly attempted to push the boundaries of neural network performance, scalability, and applicability across multiple fields. These improvements usually involve changes to network designs, training algorithms, optimization methodologies, and hardware acceleration methods. Moreover, the neural network generations are closely related to key achievements in the machine learning (ML) research domain, such as the development of deep learning (DL) designs like convolutional neural network (CNN) or spiking neural network (SNN) and using both neural generations to introduce natural language processing and advances in computer vision applications [2–4]. Thus, in the field of neural network study, researchers have categorized ANN models into generations based on their computational design and capabilities. Maass' classification approach [5] categorizes ANN evolution into three generations. Therefore, this research study explores the continual evolution and optimization of ANNs, highlighting advancements in methodologies and technical innovation. We discuss the different generations of ANN, based on computational design and capabilities, emphasizing their role in shaping achievements in ML research. The study underscores the significance of these generational milestones in enhancing the adaptability and efficacy of neural network models for computational tasks, such as image classification. Figure 1 demonstrates the visual representation of these generations.

2. First Generation of Neural Network

ML began with the perception neural network, a fundamental component of neural theory. Designed by Frank Rosenblatt in the late 1950s [6], the perceptron represented a unique technique for pattern recognition and classification. It symbolizes the first attempts to recreate the functioning of real neurons called a representation of biological neurons and create a human-like intelligence machine. The architecture was the first attempt to model biological brain network computers and it utilized simple threshold units. At its most basic explanation, the Perceptron is a single-layer neural network designed for binary classification tasks. Its primary element emphasizes its significance as an introduction to more complicated neural network topologies [7, 8]. Although relatively straightforward, first-generation neural networks encountered significant computing and conceptual challenges. The computational capability at the time was not sufficient for training large-scale networks or managing complicated learning algorithms. Therefore, the perceptron's linear decision limitations significantly restrict its ability to address nonlinear issues. Despite its limited extent and functionality, but it was an important step in the development of ANN [9].

* All authors contributed equally

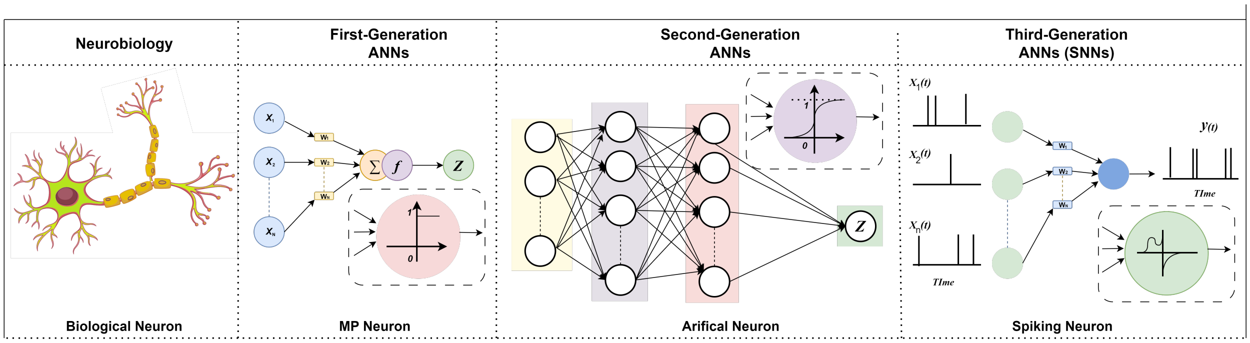

Figure 1: The basic overview of three generations of ANNs.

3. Second Generation of Neural Network

Second-generation neural networks consist of multilayer perceptrons (MLP), a class of neural networks containing one or more hidden layers [10, 11]. Unlike single-layer or perceptions neural networks, which can only be learned from linearly defined hyper-parameters, MLPs can learn nonlinear mappings from the input and output data by incorporating one or more hidden layers with nonlinear activation functions. This enabled MLPs to reach complex functions and solve a wide range of ML challenges [12, 13]. However, compared to MLP, the most significant achievement in the second generation of ANNs was the development of CNN. CNN appeared as a marked improvement in the neural network history. In the mid-1980s, Kunihiko et al. [14] designed architectures for processing structured grid-like data, such as image-based datasets. They used the ideas of local connectivity and hyper-parameter sharing to effectively process hierarchical representations of graphical data. Additionally, Rumelhart and Williams et al. [15] presented new learning methods, including backpropagation, which transformed computer vision and showed a breakthrough in image recognition and object detection research domain.

4. Third Generation of Neural Network

SNNs are a class of ANN that draws inspiration from the human nervous system, such as the spiking mechanism of neurons in the brain [16, 17]. SNNs neural architecture-based sharing information using discrete spikes rather than continuous-valued signals, as compared to other generations of ANNs process. This spiking neuron function is the fundamental unit of an SNN, stimulating the activity of biological neurons by producing discrete spikes in response to input current. These spikes are frequently described as binary events that emerge at predetermined times and reflect both the timing and stability of neural activity. As a result, the temporal dynamics of the spiking process and propagation are important for information simulation in SNNs for allowing them to encode and interpret temporal patterns in input [18, 19]. Furthermore, SNNs have demonstrated promising performance in different applications, including event-driven processing, pattern recognition, and neuromorphic computing [20–22]. They are especially well-suited to applications that require processing spatiotemporal data, such as sensory processing, robotics, and object identification prediction [23, 24]. Unlike traditional neural network architectures that depend on the rate-based firing of neurons, SNN more closely mimics the behavior of biological neuron manners by communicating between neurons via discrete functions, commonly known as action potentials [25, 26]. Lastly, in terms of parallel processing and implementation on hardware or edge devices, SNNs perform incredibly well, due to their discrete spike trains. Therefore, this feature enables energy-efficient implementations on edge computing and is a particularly useful tool for low-power applications.

Acknowledgments

This research was supported by the research training group "Dataninja" (Trustworthy AI for Seamless Problem Solving: Next Generation Intelligence Joins Robust Data Analysis) funded by the German federal state of North Rhine-Westphalia and the project SAIL. SAIL is funded by the Ministry of Culture and Science of the State of North Rhine-Westphalia under grant no NW21-059B.

References

[1] Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Abubakar Malah Umar, Okafor Uchenwa Linus, Humaira Arshad, Abdullahi Aminu Kazaure, Usman Gana, and Muhammad Ubale Kiru. Comprehensive review of artificial neural network applications to pattern recognition. *IEEE access*, 7:158820–158846, 2019.

[2] Yoav Goldberg. A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57:345–420, 2016.

[3] Sanaullah Sanaullah. A hybrid spiking-convolutional neural network approach for advancing machine learning models. In *Northern Lights Deep Learning Conference*, pages 220–227. PMLR, 2024.

[4] Yoav Goldberg. *Neural network methods for natural language processing*. Springer Nature, 2022.

[5] Wolfgang Maass. Networks of spiking neurons: the third generation of neural network models. *Neural networks*, 10(9):1659–1671, 1997.

[6] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6): 386, 1958.

[7] Silvia Curteanu and Hugh Cartwright. Neural networks applied in chemistry. i. determination of the optimal topology of multilayer perceptron neural networks. *Journal of Chemometrics*, 25 (10):527–549, 2011.

[8] Richard P Lippmann. Pattern classification using neural networks. *IEEE communications magazine*, 27(11):47–50, 1989.

[9] Imad A Basheer and Maha Hajmeer. Artificial neural networks: fundamentals, computing, design, and application. *Journal of microbiological methods*, 43(1):3–31, 2000.

[10] Simone Marinai, Marco Gori, and Giovanni Soda. Artificial neural networks for document analysis and recognition. *IEEE Transactions on pattern analysis and machine intelligence*, 27(1): 23–35, 2005.

[11] Sanaullah, Hasan Baig, Jan Madsen, and Jeong-A Lee. A parallel approach to perform threshold value and propagation delay analyses of genetic logic circuit models. *ACS Synthetic Biology*, 9 (12):3422–3428, 2020.

[12] Hind Taud and Jean-Franccois Mas. Multilayer perceptron (mlp). *Geomatic approaches for modeling land change scenarios*, pages 451–455, 2018.

[13] Sanaullah, Shamini Koravuna, Ulrich Rückert, and Thorsten Jungeblut. Streamlined training of gcn for node classification with automatic loss function and optimizer selection. In *International Conference on Engineering Applications of Neural Networks*, pages 191–202. Springer, 2023.

[14] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.

[15] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[16] Samanwoy Ghosh-Dastidar and Hojjat Adeli. Spiking neural networks. *International journal of neural systems*, 19(04):295–308, 2009.

[17] Sanaullah, Shamini Koravuna, Ulrich Rückert, and Thorsten Jungeblut. Exploring spiking neural networks: a comprehensive analysis of mathematical models and applications. *Frontiers in Computational Neuroscience*, 17:1215824, 2023.

[18] Qiang Yu, Huajin Tang, Kay Chen Tan, and Haoyong Yu. A brain-inspired spiking neural network model with temporal encoding and learning. *Neurocomputing*, 138:3–13, 2014.

[19] Sana Ullah, Amanullah Amanullah, Kaushik Roy, Jeong-A Lee, Son Chul-Jun, and Thorsten Jungeblut. A hybrid spiking-convolutional neural network approach for advancing high-quality

image inpainting. In *International Conference on Computer Vision (ICCV) 2023*, 2023.

[20] Sanaullah, Shamini Koravuna, Ulrich Rückert, and Thorsten Jungeblut. Snns model analyzing and visualizing experimentation using ravsim. In *International conference on engineering applications of neural networks*, pages 40–51. Springer, 2022.

[21] Mike Davies, Andreas Wild, Garrick Orchard, Yulia Sandamirskaya, Gabriel A Fonseca Guerra, Prasad Joshi, Philipp Plank, and Sumedh R Risbud. Advancing neuromorphic computing with loihi: A survey of results and outlook. *Proceedings of the IEEE*, 109(5):911–934, 2021.

[22] Shamini Koravuna, Ulrich Rückert, Thorsten Jungeblut, et al. Evaluation of spiking neural nets-based image classification using the runtime simulator ravsim. *International Journal of Neural Systems*, pages 2350044–2350044, 2023.

[23] Sana Ullah, Shamini Koravuna, Ulrich Rückert, and Thorsten Jungeblut. A novel spike vision approach for robust multi-object detection using snns. 2023.

[24] Seamus Cawley, Fearghal Morgan, Brian McGinley, Sandeep Pande, Liam McDaid, Snaider Carrillo, and Jim Harkin. Hardware spiking neural network prototyping and application. *Genetic Programming and Evolvable Machines*, 12:257–280, 2011.

[25] Kashu Yamazaki, Viet-Khoa Vo-Ho, Darshan Bulsara, and Ngan Le. Spiking neural networks and their applications: A review. *Brain Sciences*, 12(7):863, 2022.

[26] Sana Ullah, Shamini Koravuna, Ulrich Rückert, and Thorsten Jungeblut. Design-space exploration of snn models using application-specific multi-core architectures. 2023.