Review

Neuromorphic computing at scale

https://doi.org/10.1038/s41586-024-08253-8

Received: 12 June 2023

Accepted: 18 October 2024

Published online: 22 January 2025

Check for updates

Dhireesha Kudithipudi[1]✉, Catherine Schuman[2], Craig M. Vineyard[3], Tej Pandit[1], Cory Merkel[4], Rajkumar Kubendran[5], James B. Aimone[3], Garrick Orchard[6], Christian Mayr[7], Ryad Benosman[6], Joe Hays[8], Cliff Young[9], Chiara Bartolozzi[10], Amitava Majumdar[11], Suma George Cardwell[3], Melika Payvand[12], Sonia Buckley[13], Shruti Kulkarni[14], Hector A. Gonzalez[15], Gert Cauwenberghs[11], Chetan Singh Thakur[16], Anand Subramoney[17] & Steve Furber[18]

Neuromorphic computing is a brain-inspired approach to hardware and algorithm design that efficiently realizes artificial neural networks. Neuromorphic designers apply the principles of biointelligence discovered by neuroscientists to design efficient computational systems, often for applications with size, weight and power constraints. With this research field at a critical juncture, it is crucial to chart the course for the development of future large-scale neuromorphic systems. We describe approaches for creating scalable neuromorphic architectures and identify key features. We discuss potential applications that can benefit from scaling and the main challenges that need to be addressed. Furthermore, we examine a comprehensive ecosystem necessary to sustain growth and the new opportunities that lie ahead when scaling neuromorphic systems. Our work distils ideas from several computing subfields, providing guidance to researchers and practitioners of neuromorphic computing who aim to push the frontier forward.

As neural networks continue to affect a growing range of applications and further advances are sought, human brains remain a vibrant source of inspiration for modelling rich computational prowess. However, the pursuit of brain-inspired machine intelligence will require a change in the way we design and build computational platforms. One of the most promising research efforts in this direction is neuromorphic computing—a brain-inspired approach to hardware and algorithm design that efficiently realizes artificial neural networks[1]. Neuromorphic computing designers apply the principles of biointelligence discovered by neuroscientists to design efficient computational systems, often for applications with size, weight and power constraints.

Extrapolation from the recent rate of progress in prototype neuromorphic systems suggests an enormous potential for future artificial intelligence (AI) applications: the market for neuromorphic computing chips is expected to reach US$556.6 million by 2026 (ref. 2). Some neuromorphic chips are rapidly entering the early-stage commercial market and have demonstrated capabilities to solve computational tasks at varying scales, with extremely low power budget and latency[3,4]. One reason for such an explosion is that these systems are versatile. For example, advances in traditional computing are often focused on a specific class of architecture—exascale for supercomputers or small scale for embedded systems—and the same class is typically not explored to influence both. However, neuromorphic computing has the potential to be disruptive in both classes by using homogeneous computing technology throughout. The question of whether the field is ready to enable substantial computational breakthroughs, such as the 'AlexNet moment' described in Box 1, and how to comprehend the maturity of a given approach is more complex than simply looking at a singular measure of performance. Scale is one of the critical dimensions to track the progress of the field. The field is now at a critical juncture; our intention here is to identify the needs that, if addressed, can usher transformative impacts.

Neuromorphic computing systems offer distinct computational advantages over conventional deep learning accelerators: (1) memory and compute are tightly coupled, avoiding costly data transfer between computing elements and memory devices; (2) sparse distributed information encoding through spikes or events that carry temporal information; (3) dynamic and local learning, which avoids the high power drawn to backpropagate errors across many layers; (4) to reach a stable perception through learning, they make predictions about the sensory signals; and (5) they use dynamics on several timescales for real-time learning and processing. However, these features do not offer a magical solution. We can also achieve these optimizations through a fully top-down engineering approach, but we believe that a quicker route is possible by looking at the solutions that evolution has produced. Neuromorphic computing systems provide a solution to this problem. The advantage is that this field is close to neuroscience and biology, which have found ways to solve these problems through self-organization, dynamic rewiring, 3D growth, modularity, efficient signal encoding, sparsity, event-based computation and so on. The promise is that these biological principles can inspire the design of large-scale systems.

In Fig. 1, we show a historical timeline of the progress of neuromorphic computing systems up to the present. The underlying architectures of these systems represent the critical milestones achieved in

[1]University of Texas at San Antonio, San Antonio, TX, USA. [2]University of Tennessee, Knoxville, TN, USA. [3]Sandia National Laboratories, Albuquerque, NM, USA. [4]Rochester Institute of Technology, Rochester, NY, USA. [5]University of Pittsburgh, Pittsburgh, PA, USA. [6]Intel Labs, Santa Clara, CA, USA. [7]Technische Universität Dresden, Dresden, Germany. [8]U.S. Naval Research Laboratory, Washington, DC, USA. [9]Google DeepMind, Mountain View, CA, USA. [10]Italian Institute of Technology, Genoa, Italy. [11]University of California, San Diego, San Diego, CA, USA. [12]Institute of Neuroinformatics, University of Zürich and ETH Zürich, Zürich, Switzerland. [13]National Institute of Standards and Technology, Boulder, CO, USA. [14]Oak Ridge National Laboratory, Oak Ridge, TN, USA. [15]SpiNNcloud Systems GmbH, Dresden, Germany. [16]Indian Institute of Science, Bengaluru, India. [17]Royal Holloway, University of London, Egham, UK. [18]The University of Manchester, Manchester, UK. ✉e-mail: dhireesha.kudithipudi@utsa.edu

Box 1

AlexNet-like moment for neuromorphic computing

In exploring the potential for large-scale neuromorphic computing, we draw inspiration from the continuing deep learning revolution and its path to viability and impact. Both were based on different levels of neural understanding and both endured long periods of limited success. Similar to convolutional neural networks maturing through digit recognition[140], neuromorphic computing systems stand on the cusp of their own 'AlexNet moment'—a breakthrough realization of their potential[141]. This article outlines the open issues whose resolution will catalyse this breakthrough, enabling neuromorphic computing to achieve impact comparable with deep learning.

Neuromorphic computing progress may hinge on specialized hardware, just as AlexNet was enabled by the performance of general-purpose GPUs. AlexNet triggered a surge in deep learning model scaling, with more powerful machines[142], much larger models[143] and substantially improved chip performance[144]. Neuromorphic computing should similarly identify critical hardware requirements that unlock its potential. Although large-scale deep learning systems use thousands of accelerator chips today, an initial neuromorphic computing breakthrough might emerge from a small hardware configuration—AlexNet used just two GPUs. Such a neuromorphic computing AlexNet moment could then inspire the development of even larger scale systems, mirroring the trajectory of deep learning. A breakthrough with a small-scale neuromorphic computing system would pave the way for large-scale neuromorphic computing deployments.

terms of complexity, versatility and heterogeneity. However, there are important challenges that remain at every level of the stack, which must be addressed to allow practical neuromorphic computer at scale for widespread adoption.

In this article, we discuss the nature, needs, importance and challenges of scalable and practical neuromorphic computing infrastructure[5]. We discuss key features of scale and provide perspectives on how at-scale infrastructure can be made accessible to a broad range of stakeholders. We explore critical aspects at all levels of the neuromorphic computing stack and tool suites by identifying important challenges and opportunities. We anticipate that these perspectives will generate new ideas and collaborations that will accelerate the development of neuromorphic systems at scale.

Progression of neuromorphic computing to scale

We define neuromorphic computing at scale as the capacity of a system (inclusive of algorithms, hardware, architecture and infrastructure) to operate at the size, speed and energy required to address complex, real-world tasks. This can be achieved with several large systems accessed virtually in data centres, large networks of edge devices that exhibit collective distributed intelligence or some combination thereof. Scaling neuromorphic computing requires moving beyond proof of concepts in the lab to at-scale deployment solving real-world tasks. At scale, neuromorphic computing can lead to widespread adoption by millions of users, knowingly or unknowingly. This is a pivotal shift from historical approaches in the neuromorphic community. The recent proliferation of neuromorphic applications relevant to the high-performance computing (HPC) domain offers an initial exemplar. Most scientific HPC systems are general-purpose community resources that can allow scaling of a resource allocation to fit the demands of a particular scientific task. Despite its fundamental differences from conventional von Neumann designs, the general approach to neuromorphic architectures is perhaps ideally suited for HPC-like on-demand resources. This means that the same neuromorphic system could simultaneously be useful for performing large-scale scientific computing simulations and evaluating tiny-scale edge and distributed intelligence configurations.

As we move forward, the ability to scale neuromorphic computing systems will require careful consideration of all aspects of development, deployment and tool suites. Although developments in neuromorphic computing towards systems at a very large scale are accelerating at a rapid pace, only recently have they begun to make a sizable impact. It is challenging to make clear predictions on the extent of future outcomes, other than that we anticipate these to be profound at a level not unlike the revolution in AI.

For decades, the microelectronics industry has measured the progress of innovation by the metric of scale; a measure that encompasses both the density of the underlying hardware (with the well-known trajectory of Moore's law) and the endowed performance metric of supercomputers measuring floating-point operations per second (known as FLOPS). This unified metric offers insight into what performance computing devices may enable, from the smallest microcontrollers that address resource-constrained scenarios to server-class processors that address HPC needs. Hence, it seems natural to consider the scale of neuromorphic computing using a similar and expanded set of metrics. This is probably because of several factors. Not only does it follow the lineage of measuring improved performance by how many computational operations are enabled but there are also intuitive engineering progress measures to relate the number of neurons in a processor or system to the corresponding brain size of various insects or animals[6]. These intuitive constructs align well with measurable progress but underappreciate the unique attributes of neuromorphic computing. Scaling introduces various challenges in areas such as manufacturing, testing and reliability, particularly in terms of performance under uncontrolled conditions, infrastructure and user-friendliness.

At present, research and start-up investments in this field are growing at unprecedented levels, while devices and architectures are maturing[2]. Neuroscientists are progressing in understanding the brain[7], which has inspired neuromorphic engineers in the design of sensing and computing systems that benefit new application domains[8]. This has been demonstrated in several proof-of-concept examples such as scientific computing, artificial vision[9,10], robotics[11], biosignals[12], space computing and computational neuroscience[13,14], as shown in Fig. 2. However, there remains a substantial gap to bridge neuromorphic computing systems at scale. This process needs an understanding of the key features required for at-scale systems and the conditions that would most likely favour its widespread adoption. Notably, we present aspects that play a vital role in the research, engineering design and use of such computing architectures.

Key neuromorphic computing features

Identifying the appropriate features that make neuromorphic systems more efficient and scalable over deep learning accelerators or classical von Neumann processors is still a challenge. Consequently, we identify key features (Fig. 3) that we propose are essential to enable a neuromorphic computing advantage. It should be emphasized that these characteristics do not replace the core features of the neuromorphic computing system[2] but are instead further features required to achieve scale.

- Distributed and hierarchical: similar to hierarchical structures observed in certain regions of the brain[15,16], such as the visual cortex, the ability to organize a neural computing system into hierarchies

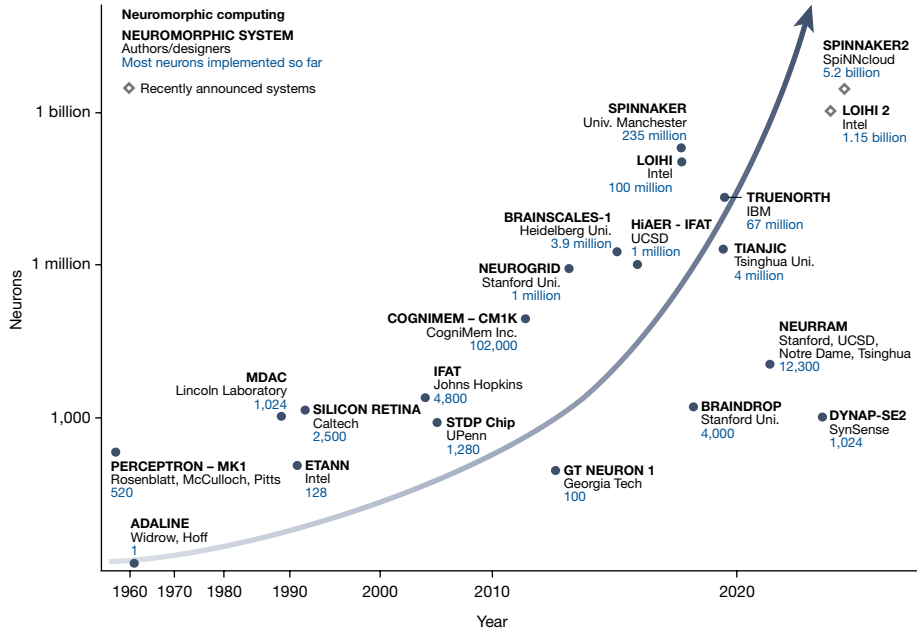

Fig. 1 | Progression of neuromorphic computing systems. We show how the number of neurons (*y* axis), chips and boards have scaled over time (*x* axis). Although this denotes one dimension of scale, there are parallel developments in architectures and communication models to enable this growth. As we move forward, the ability to scale neuromorphic computing systems will require careful consideration of all aspects of development, deployment and tool suites. For brevity, we represent a subset of neuromorphic accelerators[3,37,39–41,49,102,109–121] that is publicly recorded. Several systems such as DeepSouth use commodity chips (for example, field-programmable gate arrays) to achieve scale. Other architectures such as IBM NorthPole[122] explore the acceleration of neural network workloads with neuro-inspired functionality, in which neurons are not the core compute units. Accordingly, we do not include their progress here. The neuromorphic field has a rapidly changing landscape and we anticipate that there will be systems with more than 10 billion neurons available by 2026.

helps to disentangle complex inputs[17,18] and information at different scales[18]. Each hierarchical level can be assigned to deal with a different level of information complexity. This process provides better granular control in large models, as well as increased explainability of the processes at each level. The improved cognition and correlation obtained through data simplification provided by the 'disentangling' can also reduce redundancies that can otherwise occur in natural, linear, non-hierarchical networks[19] without specialized mechanisms[20] or supervision. Hence, we consider the ability of a system to support distributed and hierarchical structures an important aspect for scale.

- Sparsity: the sparsity of activity and connectivity in human brains is a notable factor in their scalability. Studies have shown that human brains, through development, start with relatively sparse synaptic interconnections, experience a period of densification, followed by extensive pruning and then remain at a relatively constant level of sparsity[21]. Sparsity can lead to reduced representational complexity, in which only a subset of the dimensions is used at a time, thereby supporting selectivity and specificity for a model and system. This helps to achieve considerable improvements in computational, storage and energy consumption without loss of accuracy. Sparsification can be either structural (that is, weights, neurons, heads) or ephemeral (that is, activations, gradients, errors)[22]. Although neuromorphic systems inherently have the sparsity advantage owing to event-driven communication[23–26], there are further mechanisms that should be explored in this space to enable large-scale computations and generalization. Sparsity in neuromorphic models can potentially make the slope of model scaling steeper because of the potential to use unconventional spatial layouts of computing devices[27].
- Neuronal scalability: features such as neuronal scalability that support a large number of neurons on single or multichip systems will enable deep spiking or rate-based algorithms that can solve complex real-world problems in a wide range of machine learning (ML) applications[23,28,29]. Such scalability also offers us the ability to simulate full human brain simulations in real time for advances in cognitive applications and neuroscience research developments. The systems should support a dynamic range of neurons based on the problem. In current set-ups, this scale can be achieved when racks of neuromorphic chips are stacked together, as shown in Fig. 1, in which the systems support hundreds of millions of neurons. This greatly expands the application space by targeting approximate solutions for nondeterministic polynomial time (NP)-complete problems, running large-scale neural simulations and complex graph algorithms[30,31].

To encourage greater adoption of neuromorphic systems, it is essential to incorporate features that enable the integration of a wide variety of computing resources and neuromorphic elements[32,33], as well as support the integration of external tools and sensors. This ability to combine distributed resources across different platforms within a single neuromorphic system can be referred to as heterogeneous integration[34,35]. This enables support for several device technologies on a single chip, data fusion from numerous sensors and platforms and deployment of large and complex artificial neural network–spiking neural network hybrid frameworks.

- Asynchronous communication: several of the neuromorphic chips incorporate event-based, asynchronous communication protocols to support the address event representation architectures used for the receiver and transmitter. Addresses are inputs to the chip (address events) and represent the neuron receiving the input event or spike[36]. There are several variants of these protocols that are implemented in chips, from first-generation chips such as silicon retina perceptual systems to cortical cognitive processing systems[3,37,38]. There is notable progress in complex network on-chip architectures[32,39–41], which makes it easier to make use of asynchronous communication for large-scale integration of systems.
- Dynamic reconfigurability: the brain is an inherently dynamic system. Studies show that executive cognition requires highly evolving and dynamically reconfiguring networks of brain regions that interact in complex and transient communication patterns[42]. To this effect, several neuromorphic systems support dynamic reconfigurability[23,43–46]

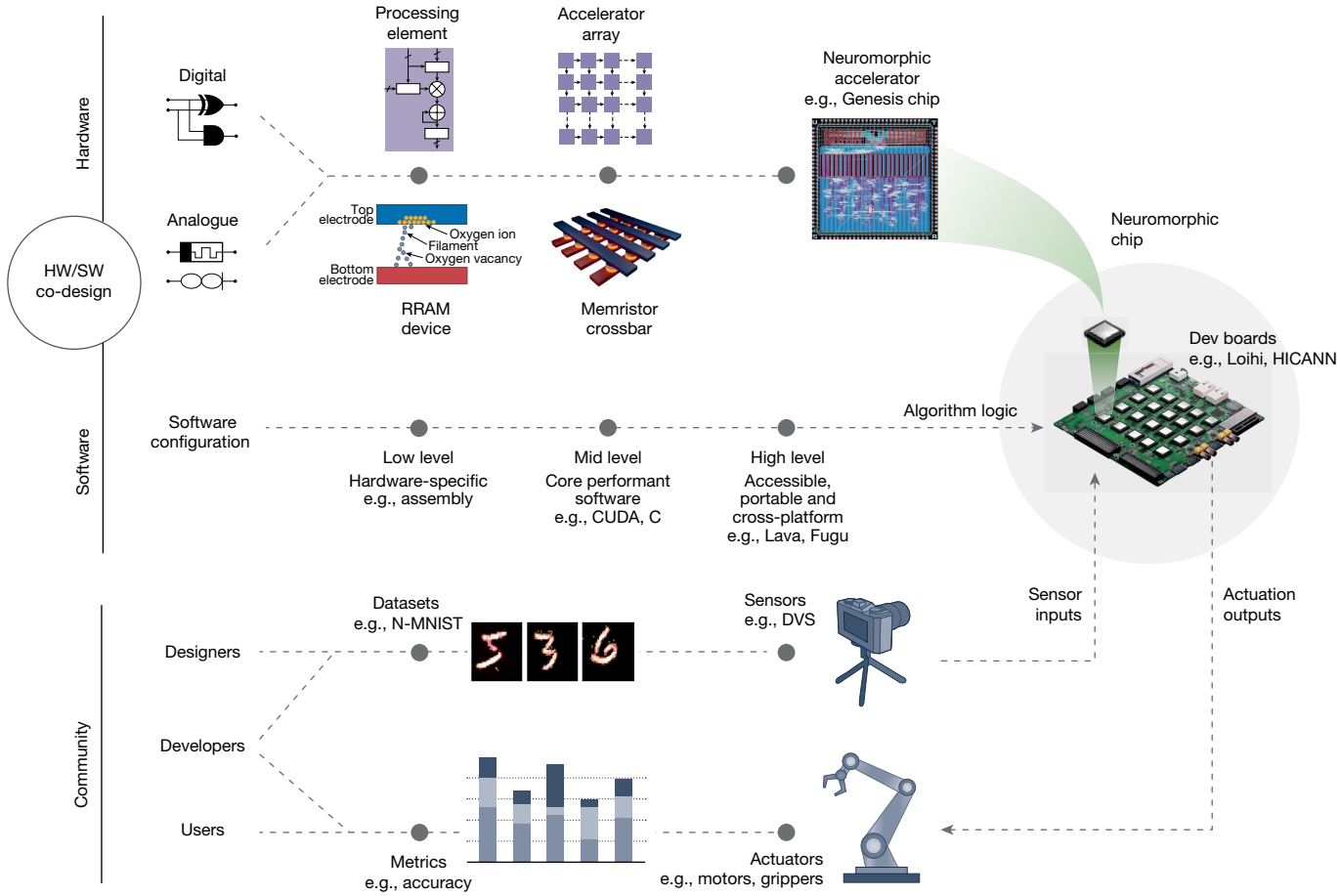

Fig. 2 | Neuromorphic computing ecosystem. The neuromorphic ecosystem, as described here, consists of hierarchies of hardware[123–125] and software[66,67,72,77,126,127] technology, hardware/software co-design approaches, workflows that culminate in flexible neuromorphic ecosystems[4,128,129], prototype boards incorporated into several application domains[9,10,14,128,130,131] through sensors[55] and actuators[130,132], a suite of external tools[133–137] that integrate with neuromorphic frameworks and community feedback that serves as the R&D multiplier. This can be achieved with a few large systems accessed virtually in data centres, large networks of edge devices that exhibit collective distributed intelligence or some combination thereof.

in varying forms. In these systems, it can contribute to the plasticity of primitive building blocks (for example, synapses, neurons, axons, dendrites) or the entire system. For example, in several of the chips, reconfigurability of synaptic connections between neurons and the one-to-many capability allows for linear filtering, including edge and smoothing operators.

- Redundancy and correlation: neurons are capable of executing several tasks simultaneously owing to neural redundancy. The correlations between neurons can influence the amount of data encoded in a group of neurons and the methods used to decode the population[47]. Studies have suggested that these redundant activity patterns may be beneficial for neural computation[48], allowing stable computation despite unstable neural dynamics and filtering out unwanted noise. Although there are several examples of this in neuromorphic systems[3,32,39,49], such as skip-zero and zero-weight approaches, there is an opportunity to scale these mechanisms.
- Sensor and compute interfaces: apart from the core computing ecosystem, end-to-end development of applications requires a robust and efficient integration of the external sensors and actuators with the neuromorphic computing tools. Most commercial sensors have non-spiking protocols and require a kernel-level interface for translating the data packets into spike-encoded formats. Having optimized neuromorphic drivers for such interfaces can greatly boost performance in low-latency tasks such as audio and vision processing[50,51]. It is also important to incorporate a common neural information format[52–54] to make such drivers more universal in the neuromorphic ecosystem, rather than hardware specific. Alternatively, the development and adoption of neuromorphic sensors such as event-based cameras[55] can further reduce response time in rapid-motion-capture applications. Similarly, medical devices such as silicon cochlear implants[56] and prosthetics with e-dermal layers[57] can improve sensory fidelity and sensitivity.
- Resource awareness: it is critical that the accelerators are designed with resource awareness for efficiency and versatility. Resource awareness refers to the ability of the system to track its energy, compute and memory size over its lifetime. In some way, the system should be capable of dynamically assigning resources on the basis of changing goals and functionalities. We can draw inspiration from self-aware architectures and continual learning accelerators[58] to support these features.

Each of the key features can be further explored on the basis of its impact on power, performance, ability to scale, adapt and the versatility of integration. Furthermore, we note that some of the features that are innate to neuromorphic systems are equally applicable to the design of conventional systems at scale. For example, extreme parallelism is fundamental to the design of large-scale supercomputers and is not unique to neuromorphic computing. Changing the slope of the scaling curve can allow much larger models than is possible at present. On the other hand, it can enable longer training using larger datasets, even for smaller models. This will make it easier to deploy many smaller models trained on datasets specific to each task, as opposed to the

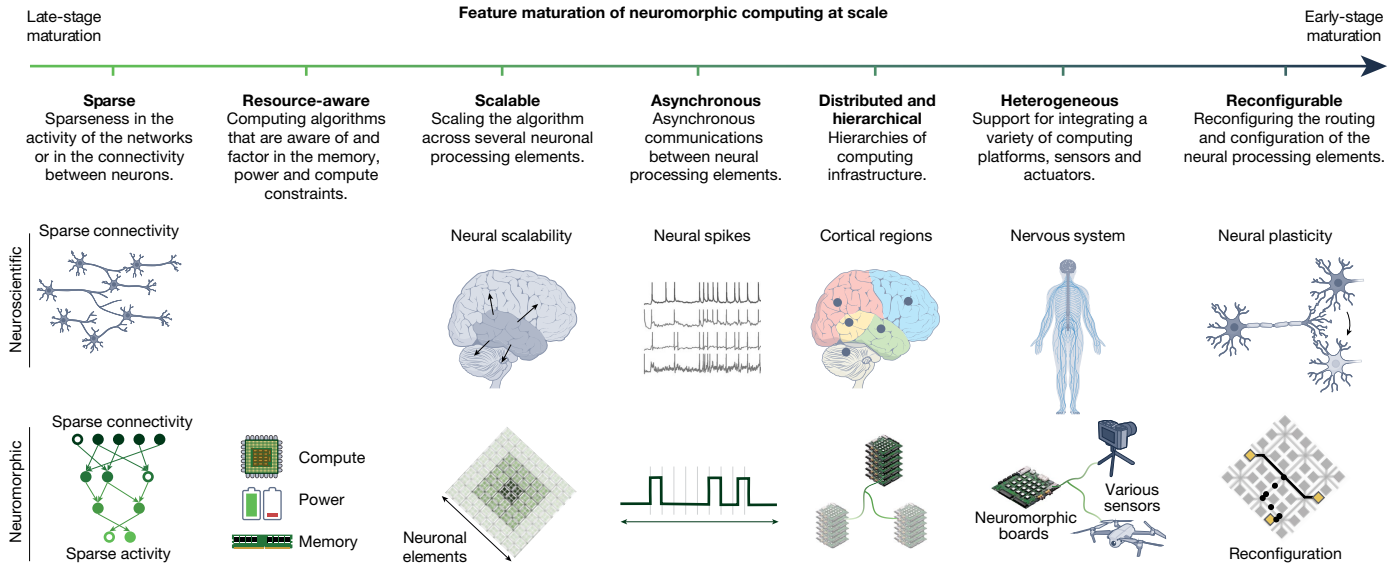

Fig. 3 | Key features of neuromorphic computing systems at scale and their feature maturation timeline. These features augment the core aspects of neuromorphic computing systems. It can be observed that they inherently draw inspiration from the brain and neural processes[138,139] in numerous facets. Some will reach a higher maturation level in the short term, whereas others may require more advanced approaches to reach the same level of maturation.

current trend of training one extremely large model and attempting to use it for many different tasks, which limits the task performance on the long tail of small and niche use cases. Consequently, although some of these design concepts are not unique to neuromorphic computing, their interplay within a unified computing model is where we see the potential.

Challenges and opportunities

As we summarize some of the key challenges and opportunities, it is important to note that our current understanding of the brain is limited. We anticipate that neuromorphic computing may provide an opportunity to improve our understanding.

Despite substantial progress and the potential game-changing effect that neuromorphic systems have in producing sustainable and robust technology for a wide range of applications that benefit society, there are still notable roadblocks and challenges that must be overcome for them to be widely adopted and have a tangible impact. These challenges apply to neuromorphic hardware designers, users and algorithm developers alike. They start with the need for co-design and co-development of hardware and software, as described in Box 2, and extend to technology adopters and founders.

From a developer's perspective, there are challenges on several fronts. First, progress is needed to enable use of the hardware without understanding intricate details of low-level hardware by providing a higher-level 'coding abstraction' (transitioning from assembly to object-oriented code). This can be attributed to the diversity of the neuromorphic hardware, which makes replication of models on different platforms difficult and, thereby, algorithms are hardware dependent. The computational primitives and hardware constraints differ from platform to platform. A set of common standards for hardware and software can minimize the need to alter algorithms when used on different platforms. This, along with a hardware abstraction layer for all neuromorphic platforms, represents a critical step towards a 'compilation scheme' for porting arbitrary spiking neural network models to any hardware architecture. Missing components in the current neuromorphic computing ecosystem are highlighted in Fig. 4. Within the variety of spiking neural network frameworks, interoperability remains limited, greatly differing from the robust connections seen in conventional artificial neural network frameworks. Moreover, missing quality-of-life extensions such as inference optimizers, drag-and-drop editors and cross-platform compatibility aid in ease of operation and enhance newcomer experience. For seasoned developers and large enterprises, the absence of integrated cloud scaling, deployment and life-cycle management resources can impede development at scale. Finally, tools for cross-compilation across different

Box 2

Hardware/software co-design

Intrinsically, neuromorphic computing is an interdisciplinary field that bridges neuroscience, computing, engineering and AI systems. Following the best practices and research approaches of each of these fields leads to varied methods of tackling the system design question. One way to design neuromorphic systems is to draw inspiration from the plasticity and learning of all levels of abstraction in the brain, translate them into models and then collaborate on the hardware and software components of the machine before deploying them. The top-down approach involves researchers designing neuromorphic systems by abstracting hierarchical cortical layers without detailed neuronal or synapse models[15,16] and using these models to guide hardware and algorithm designs. In this case, it is necessary to identify key architectures and connections in the brain that will allow us to create systems at scale. In the bottom-up approach, designers take advantage of the inherent device characteristics to drive new algorithmic advances that will improve architectures and systems[145,146]. Innovations in technologies such as in-memory computing[147–149], emerging device technologies[150,151] and low-precision arithmetic[123,152] are necessary in this context. However, both approaches introduce inefficiencies in the process owing to a lack of awareness of device properties among algorithm designers and architects and vice versa. A potential solution to this problem is to use a hardware/software co-design approach[80,123,153] and to consider scalability throughout the design process, independent of the approach used.

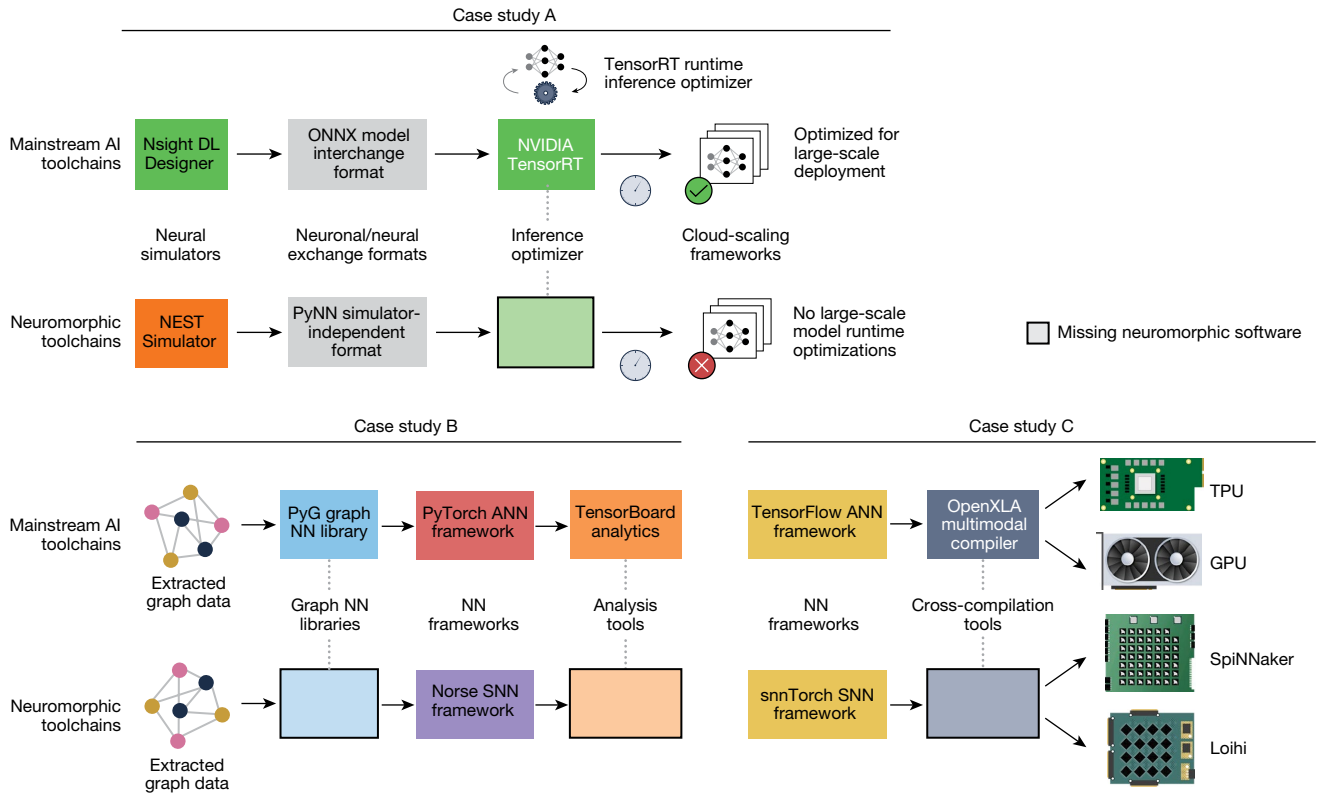

Fig. 4 | Case studies showcasing the gaps in the neuromorphic computing software ecosystem as compared to AI/ML. Mainstream AI/ML (top, per case study) has established pipelines and widespread interplay between frameworks, compilers and hardware. Identifying and filling these missing pieces is essential to make the neuromorphic ecosystem (bottom, per case study) more comprehensive, collaborative, and complete. We provide examples of 3 case studies for comparison. Case study A: Optimizing models for large-scale deployment and inference. Case study B: Development and analysis of graph neural networks. Case study C: Compiling AI models for general-purpose or specific hardware targets. Software that has counterparts in AI/ML and neuromorphic systems is indicated by the labels in the centre. The missing software in the neuromorphic ecosystem is indicated by empty blocks.

frameworks and hardware are crucial to minimize unnecessary duplication of existing components and to facilitate transitions between ecosystems. There has been substantial progress in recent years to bridge some of these gaps. AI vision toolkits, simulators/analytical tools and automatic-hyperparameter-search frameworks are integrated into existing toolchains[40,50,52,53,59–72]. Furthermore, the support for dynamic architectures and low-level primitive simulators rival the best of mainstream AI equivalents. Although there is a widely adopted and established exchange format for low-level neuronal primitives, the equivalent high-level neuronal topology exchange format lacks the same level of adoption between frameworks. Mainstream AI/ML toolchains faced similar hurdles and were largely mitigated by the united development and widespread adoption of the ONNX format[73,74]. Similar efforts in the neuromorphic community are required to establish or adopt standards for spiking neural network model description and exchange. Moreover, improved accessibility, comprehensive documentation and robust community support for these tools and platforms will increase adoption among new users. Such progress can foster development of components for an integrated system that are reusable (for example, ROS, Linux) instead of focusing on isolated motifs for hardware-specific tasks.

Scientific research communities prefer tool flexibility, whereas end users and product designers give priority to tool efficiency. Thus, an approach to increase the adoption of neuromorphic systems would be to emulate the principles of early AI toolchains such as Torch and Theano[75,76], in which the initial efforts focused on flexibility over efficiency, bringing a larger scientific community interested in development, which spurred community-driven development of more efficient back-ends (for example, PyTorch[77]).

At the same time, a substantial roadblock for application developers is that these systems require a fundamentally different approach to combining neural computational primitives and write a 'program', which creates an entry barrier for researchers and engineers without previous experience in neuromorphic computing or computational neuroscience. A strong theoretical foundation on how to use neural building blocks for implementing computation needs to be developed, such as ordinary differential equation solvers for modelling dynamics. Some initial work has been done through this mathematical approach, such as the Neural Engineering Framework (NEF)[78] and dynamic neural fields[79].

From a designer's perspective, there is a need to work across the hierarchy of the stack. The progress of the field in device, circuit, architecture, algorithms and applications should integrate seamlessly in a 'neuromorphic system' by the designer. Therefore, we need a high level of synchronization in this stack while the neuromorphic systems are being designed. Notably, to scale these systems for complex tasks, we require the modularity of the components. This leads to the question of whether neurons and synapses are the correct abstraction level for neuromorphic systems, as has been implemented so far in the community. Moreover, neuromorphic algorithms often emphasize biological plausibility, which may not always be suitable to achieve the best task performance while remaining scalable and energy efficient. This is in part because of the differences between the neuromorphic and biological substrates and partly because biology admits many levels of complexity that simplified biological models do not reflect. The challenge is to develop the appropriate levels of algorithmic abstractions that allow us to reason about algorithmic efficiency and scalability in a clear and focused manner while ignoring

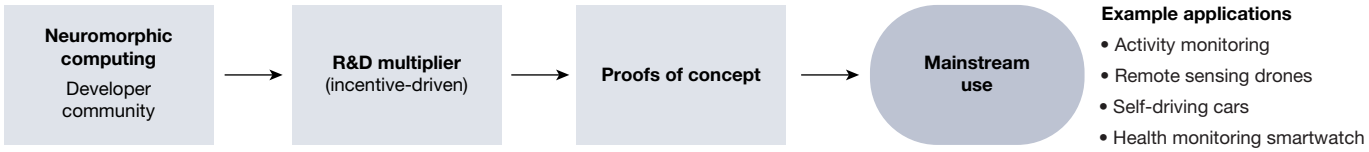

Fig. 5 | Considerations to achieve community readiness. R&D multipliers include developing industry-specific proofs of concept and granting early and easy technology access to R&D groups beyond the traditional neuromorphic ecosystem, which further creates a chain effect. Proof of concept has a greater impact when the selected niche area has the potential to reflect on others as a result, as they get adopted in mainstream applications. By listening to the feedback from early adopters and incentivizing them, it serves as a force multiplier.

the details of biology that may not be relevant. Scaling neuromorphic algorithms also requires a new focus on asynchronous and distributed algorithms, which has historically been the focus of a different community than that working on mainstream deep learning and neuromorphic computing. Finally, identifying key interface requirements such as high-speed-sensor integration and diverse sensor compatibility is an important challenge that must be taken into account during the design phase. Another remaining challenge is to continue to engage neuroscientists for collaboration in multidisciplinary teams in which neuromorphic researchers are actually driving some of the neuroscience experiments.

From a user's perspective, a substantial challenge has been to identify appropriate applications for enabling a neuromorphic advantage. A useful analogy is to look at quantum computing, which has received billions of dollars in financing in part because of the existence of a formal complexity advantage for the factoring of numbers (Shor's algorithm), an important benchmark task with real implications for cryptography. Although certain tasks have been shown practically to have performance or energy advantages on neuromorphic hardware, the lack of this type of formal complexity proof has almost certainly affected investment in neuromorphic hardware. In other words, the theoretical proofs for improved performance of proposed quantum computers have trumped the neuromorphic existence proof (that is, the human brain). Lack of such formal theoretical advantages on important problems has been one challenge in building interest in neuromorphic computing.

Many of the tasks for which neuromorphic computing shows promise, such as lifelong or continual learning[58,80], learning in the presence of sparse data, robustness in the presence of noise and variability and ultralow-power machine learning with real-time sensor data, are difficult to break down into a sequence of benchmark tasks of increasing difficulty in a mathematically provable way. Moreover, to have a closed-loop benchmark, neuromorphic chips need to be interfaced with sensors and actuators. Therefore, there is a need to standardize the communication from the low-level electronic circuit implementation up to the protocol in a way that different chips (sensors, distributed ad hoc processors, multicore million-neuron systems) can seamlessly interface, as well as with non-neuromorphic systems.

Cross-platform neuromorphic software

Mainstream deep learning success can also be attributed to the availability of powerful and relatively easy-to-use open software tools that offer a high level of abstraction to the user and application developer without requiring expertise in theoretical ML. Tools such as TensorFlow, Keras and PyTorch hide complex mathematical processes such as automatic differentiation, allowing the user to focus on high-level abstraction, such as network structure. Neuromorphic computing lacks in this aspect; the few available examples of plug-and-play tools are developed and maintained by a handful of researchers in the field (for example, jAER[81]). Although there is potential for community-driven support and the growth of open-source tools, the appropriate level of abstraction for developing these computational tools is still an open question.

Neural network modelling tools that target neuromorphic hardware are at present less advanced, often operating in the equivalent of a register transfer language rather than at a high-level synthesis. The lack of maturity can also be attributed to divergent efforts of the community, marked algorithm diversity and analogue[82], digital[23] or software[4] neuromorphic platforms having unique implementation requirements. More recent frameworks support a variety of neuron models, learning rules, partially more back-end platforms and mechanisms to import models from well-established deep learning frameworks[59]. The limited community engagement with neuromorphic frameworks is not a result of the lack of tools but rather of the absence of a shared standard, one not dominated by the commercial interests of a particular entity but steered by the community as a whole. Recent initiatives to establish open-source frameworks and common intermediate representation levels contribute to this unifying effort and motivate hardware vendors to follow standards[52].

The community acceptance of a small subset of software frameworks allows hardware companies to focus on producing cutting-edge accelerators without the need to redesign a full software stack for each new chip, which—in turn—contributes to the adoptability of new chips. In the future, we anticipate tools that operate at a higher level of abstraction with common intermediate representation layers, perhaps assembling applications from a library of functional modules composed of spiking neurons. What the functions of such modules would be and how they would be developed remains an open question.

Community readiness and ecosystem

The neuromorphic field has the potential to benefit from the cumulative body of fundamental research and increasing commercial interest fuelled by a growing ecosystem of established and start-up companies. Despite the development of systems that accurately simulate the bottom-up processes of the brain, most users are not willing to switch from their current solutions unless they can clearly see the advantages. Here we outline the traits of a neuromorphic ecosystem and offer a set of considerations to prepare the community for large-scale systems.

- R&D groups: as presented in Fig. 5, granting early and easy technology access to R&D groups beyond the conventional neuromorphic ecosystem creates a chain of multipliers, propagating the advantages (for example, energy consumption, form factor) of this new approach. Early access refers to hardware prototypes made available to a wide range of stakeholders to test the technology. Further, incentivizing the R&D groups with awards, competitions, common benchmarks, onboarding practices for newcomers, continuing support or workshops helps maintain engagement. Community efforts such as the Telluride and CapoCaccia workshops, the NICE and ICONS conferences and the Intel INRC are spearheading efforts in this direction.
- Easy, common and open-source software: as discussed in the 'Cross-platform neuromorphic software' section, easy software access refers to compilers that enable mainstream users to deploy the models on neuromorphic platforms without needing to understand the low-level hardware. The cross-compilers should be developed to facilitate easy integration of new back-ends and incentivize hardware

providers to follow a common standard. A common and accepted framework reduces the individual software efforts of hardware providers and increases community engagement. The use of permissive open-source licences (such as Apache License 2.0) at all levels of the software stack also promotes the collective contribution without imposing restrictions on commercial entities.

- Benchmarks: a software stack accepted by the community has a direct implication in the adoption of common measures of progress. The maturity of the field and the inherent heterogeneity of neuromorphic systems impose challenges to identify metrics and benchmarks to compare different algorithms and hardware platforms. In particular, the diversity of neuromorphic implementations and hardware instantiations in turn requires a diversity in metrics and benchmarks. However, formalizing aspects of the technology that are already mature (for example, event-driven interface standards, I/O neurons and so on.) plus a hardware agnostic software stack outline a path to define fair benchmarks. Examples include HPC-inspired system-level benchmarks[83], industry-led benchmarks for spiking systems[84] and tools for benchmarking probabilistic systems[85,86]. Several datasets have been developed specifically for benchmarking spiking systems[87–93]; however, these are still not widely adopted[30]. Similarly, the metrics for measuring hardware performance are also diverse and do not have a consensus from the community. Singular metrics such as energy, power, latency, accuracy, area, robustness and noise tolerance are routinely quoted. The usefulness of these metrics on their own is often uncertain, except when comparing very similar devices or systems. Thus, a combination of metrics is often used, such as the energy delay product, energy per synapse and relative accuracy. However, the development of standards, metrics, and benchmarks remains a crucial and active area of research, with the goal of their future widespread adoption. For example, there is a continuing community-driven effort to improve the situation by defining a set of benchmarks and relevant metrics for comparing both neuromorphic algorithms and hardware systems[94]. There are also tools to evaluate the performance of neuromorphic systems and models for neuroscience and psychological modelling[95,96]. However, defining a single set of benchmarks and metrics that are applicable to different neuromorphic systems with the broad array of features described in the 'Key neuromorphic computing features' section remains a notable challenge.
- Field-crossing technology: neuromorphic systems must contend with further interface losses, which occur during the bidirectional conversion between traditional and neuromorphic APIs. In some cases, these losses can be attributed to the spike-coding schemes used in those interfaces. Addressing these losses is critical to ensure the usability of neuromorphic platforms, as the computational benefit achieved by brain-like engines could be hindered by poor interfacing.
- Proofs of concept: industry-specific proofs of concept are a vehicle to study the benefits offered by neuromorphic systems at scale, such as energy and performance. As is the case with any new technology, selecting the appropriate proof of concept is crucial to maximize the impact of the investment effort. Proof of concept has a greater impact when the selected niche area has the ability to create a ripple effect. For example, designing an energy-efficient proof of concept for an end user of a large-scale conversational engine does not have the same impact as designing it for the technical service provider of large-scale conversational engines. The technical service provider is the one affected by energy consumption and its satisfaction with proof of concept can reach more end users. Furthermore, building prototypes at scale during the proof of concept, rather than incrementally, generates further benefits.
- Listening to the feedback: a recursive feedback loop helps improve research methods, shapes proofs of concept and evaluates the level of technology adoption. Maintaining a user-centric approach forces bottom-up approaches to consider practical applications and encourages hardware/software co-design.

Box 3

Emergent memory technology and its challenges

Many emerging devices such as RRAM[107], spintronic devices[154], ferroelectric transistors[155] and phase-change memory[106] have been recently proposed and used in such neuromorphic systems. Supplementary Table 1 compares a subset of these emerging devices with the standard memory technologies over several relevant metrics. However, the full potential of these devices in large-scale systems is only beginning to be fully exploited[156] owing to several challenges, such as device non-idealities, challenges in integration with CMOS, and leakage. Several emergent devices have addressed device non-idealities by using some form of online training[121,146,157]. This presents an intriguing potential for the incorporation of compensation devices within the hardware itself, which will require the design of new devices and circuits. Some of these non-idealities in the physics of the devices have also been exploited as 'features', for example, the cycle-to-cycle and device-to-device variability can provide a distribution of parameters in Bayesian computation[146,158], and can be used to provide a parameter space for learning in self-organized dynamical networks[145]. Moreover, the internal dynamics of memristive devices that cause volatility have been exploited for motion detection[159] and online learning[160]. However, the scalability of these approaches to large models and practical use cases is an active area of research. As technologies such as RRAM continue to mature, their non-volatility and compute-in-memory capabilities are becoming attractive for application spaces constrained by size, weight and power.

Outlook

Neuroscience exploration

As we look towards brain-scale simulations, making use of neuromorphic computing systems becomes essential. For example, the virtual brain[97] is a brain-sized model incorporating complex biological details, which is difficult to simulate on GPUs at this scale and in real time. This model has notable medical applications, such as personalized Alzheimer's disease detection[98], making fast execution on neuromorphic systems a highly desirable goal for medical diagnostics. As another example, the aforementioned Markram cell study used the Blue Brain IV supercomputer, which—at the time—was the 100th largest supercomputer in the world, to simulate just over 31,000 neurons of the barrel cortex. As the human brain contains roughly 1 million times that many neurons, it is clear that more efficient computing resources will be required once the neuroscience community is prepared for such simulations.

A potential hardware substrate for this is SpiNNaker2 (ref. 99), which was designed in the Human Brain Project with the aim of neuroscience exploration. More importantly, through its flexible numerical accelerators, SpiNNaker2 and Loihi 2 systems can simulate neural behaviour at numerous levels, from very detailed models of synaptic or dendritic computation[23,100] to spiking point neurons and rate neurons[23,99], all the way up to mesoscopic approximations, mean-field models[29] and electroencephalogram behaviour[23,101]. Such systems could also support a multiscale brain model, such as the virtual brain[97], near the size of the human brain and in real time. Several other chips are entering the market that can support neuroscience experiments[23,39,102]. In general, further developments are needed at scale for detailed neuroscience exploration.

Box 4

Short-term questions

- Hardware/software co-design frameworks: can we design user-friendly co-design frameworks with sufficient abstraction for non-experts to develop neuromorphic computing models? Furthermore, can we do this at scale?
- Prototype: what should a prototype test bed look like for at-scale systems? How do we provide the broader community access to such prototypes?
- Integration: how can we improve the ease of integration with conventional computing systems (accelerators)?
- Tools: how do we design open-source software tools that have interoperability with mainstream deep learning frameworks? Can we develop tools that work at higher levels of abstraction with common intermediate representation layers from a library of functional modules?

Box 5

Medium-term questions

- Large-scale test beds: how do we deploy heterogeneous and large-scale demonstrations on common test beds?
- Benchmarking: what should be the common suite of benchmarks for neuromorphic computing at scale? How can these benchmarks be evaluated? What would an open-source protocol for these systems look like?
- Lifelong learning: how can event-driven local learning and plasticity support machines that are capable of lifelong learning? What unified architecture can be used for these systems?
- Dynamic models: how can we incorporate complex dynamics into future neuromorphic systems? We need to explore neuromorphic devices with complex dynamics such as learning synapses, dendritic processing and connectivity.

ML innovations

The field of ML aspires to develop algorithms capable of learning, as opposed to explicitly being programmed to perform a task. These developments have increased the sophistication of neural networks, not only adding to their scale but also exploring the roles of connectivity, information representation, learning and other topics.

Besides algorithmic advances, the computational demands of deep learning algorithms have spurred computational architecture innovations. These advances, however, are not decoupled. Notably, the 'hardware lottery' articulates the impact that some computational ideas win out not because of their intrinsic superiority but because of the enabling hardware[103]. This phenomenology in part hinders neuro-inspired ML innovations without enabling hardware to make the innovative algorithmic approaches highly performant. Consequently, we see effective neuromorphic computing systems at scale as a means to drive neuro-inspired ML innovations[104,105]. The key neuromorphic features identified in Fig. 3 serve as a blueprint for what future ML innovations may look like.

Crucially, the convergence of these advances represents a shift towards understanding algorithms and architectures as interconnected, mirroring the design of the brain, rather than the isolated approach common in von Neumann computation. The development of neuromorphic hardware holds the potential for unparalleled disruption within neural ML. Our brains exemplify the intrinsic nature of learning algorithms, for which parameters adapt to modify functionality. Unlike traditional hardware (for example, data-centre GPUs) and their tailored training patterns, brains learn across several timescales within units that seamlessly blend computation and memory. Neuromorphic computing, by making use of emerging device technologies, offers a distinctive solution to address these fundamental bottlenecks.

Emerging devices and architectures

Efficient memory technology is critical to the scalability of neuromorphic systems. From information storage, consolidation and retrieval, memory serves as the foundation for learning and solving problems based on experience. In particular, emerging memristive devices hold great promise for neuromorphic systems, thanks to their non-volatility, high density and multistate properties described in Box 3. Moreover, an array of such devices implement the multiply and accumulate operation, the core computation of neural networks by virtue of Ohm's law, leading to a compact and low-power implementation[106,107]. For an emerging memory technology to be viable, the following characteristics are desirable: low energy consumption (≈fJ bit^{-1}), low latency, low operating voltages (<1 V), high endurance (>10^{17} cycles), high data retention, scalability (<10 nm) and CMOS integration[108].

Summary

Neuromorphic computing systems have the potential for substantial impact in various domains and the current moment is appropriate for innovation at scale. The field has matured beyond prototype systems developed in academic institutions into a context of production systems with event-driven processing, learning models and design tools coupled with real-world experimentation. Building on these foundations, the industry has further advanced systems with applications in scientific computing, augmented/virtual reality, wearables, smart farming, smart cities and so on. To continue driving progress in the field, large cadres of engineers and scientists from several disciplines and public/private organizations should collaborate on shared goals. This article offers a road map for such possibilities and a summary of key open questions that will inspire future research in the field.

As a first step, we should examine a broad set of capabilities that neuromorphic computing systems can support. We should identify common primitives that can be connected in a heterogeneous fashion and be further integrated to develop multi-use architectures. The human brain is made up of distinct and specialized parts that are connected in a distributed manner. At present, these modules of the brain do not have a one-to-one correspondence with the components of intelligence that are used by the neuromorphic field (with a finer granularity). In this article, we propose a set of features that serve as a guide to identify primitives for the deployment of more capable systems. These systems should support a broad range of learning mechanisms, such as online continual learning, real-time decision-making with event-driven sensor data, sensorimotor fusion algorithms, multimodal learning and predictive modelling, among others. We identify that, rather than concentrating on a single hardware platform or a subset of features, now is the time to investigate diverse capabilities and understand which neuromorphic features are essential for different applications. We do not anticipate that there will be a one-size-fits-all solution for neuromorphic systems at scale but rather a range of neuromorphic hardware solutions with different characteristics based on application needs. Current neuromorphic implementations are on a single chip or board that may not be well suited for large-scale systems. Two threads emanate from this. The first involves the study of robust models and algorithms that are distributed and asynchronous in nature. This requires a more collaborative effort and is a nontrivial challenge. The second thread focuses on scaling up these learning models to larger networks

Box 6

Long-term questions

- Critical applications: how can neuromorphic computing be used for mission-critical or safety-critical applications?

and machines. We can borrow principles from the AI community, in which large models are deployed on cloud or heterogeneous systems. If we can achieve this, there will also be an organic boost in the rate of adoption of neuromorphic systems.

Further, the progress of neuromorphic systems is often limited by the myriad of small-scale prototypes that are housed in individual labs and the inability to access these test beds by experts and non-experts outside a select group of specialists. This creates barriers to system growth in terms of portability, standardization and cross-functional knowledge transfer. By promoting open prototype test beds and benchmark repositories, we can enable the development of a new generation of large-scale, adaptive neuromorphic systems. When prototype test beds are integrated with conventional computing systems, it can lead to a surge in R&D and can accelerate application-centric research. Often, these open resources must have measurable performance metrics or open benchmarks[94] to bootstrap a productive growth of scalable systems. We can also take cues from successful industry academic partnerships in deep learning and related fields to further accelerate R&D cycles. This, in turn, can also benefit the deep learning community by offering new pathways to solve problems similar to human-like learning.

In recent years, there has been a upward trend in the development of software and hardware design tools for rapid prototyping or evaluation. However, there is a lack of a comprehensive software tool ecosystem and community forums to support experts and non-experts alike. Although some of these gaps are addressed by industry partners, much of the tool development is still taking place in isolation. Figure 4 highlights the potential for toolchain development that can be integrated with or benefit from the widely adopted deep learning tools. Moreover, there is a need to consider critical applications that can benefit from large-scale event-based systems. These are now being explored in non-cognitive applications, distributed intelligence on the edge and special cases in which neuromorphic systems are integrated with the deep learning systems.

To conclude, we present a list of open questions that can be investigated in the near future (Box 4), over the medium term (Box 5) and into the distant future (Box 6). Now is the ideal time to invest resources towards large-scale neuromorphic computing, as it can lead to notable breakthroughs for both natural and AI systems of the future.

1. Mead, C. Neuromorphic electronic systems. *Proc. IEEE* **78**, 1629–1636 (1990).
   Original article launching the field of neuromorphic electronic systems engineering founded in the physics of computing.
2. Mehonic, A. & Kenyon, A. J. Brain-inspired computing needs a master plan. *Nature* **604**, 255–260 (2022).
   A discussion of the potential of neuromorphic computing to revolutionize information processing, with a focus on bringing together disparate research communities to provide them with the necessary financing and support.
3. Davies, M. et al. Loihi: a neuromorphic manycore processor with on-chip learning. *IEEE Micro* **38**, 82–99 (2018).
   An introduction to Loihi, a neuromorphic chip that models spiking neural networks in silicon and achieves more than three orders of magnitude better energy–delay product over conventional solvers.
4. Furber, S. & Bogdan, P. (eds) *SpiNNaker: A Spiking Neural Network Architecture* (now publishers, 2020).
   A book that explores the development of SpiNNaker-1, a large-scale neuromorphic computing (1 million core) processor platform optimized for simulating spiking neural networks, which will make use of advanced technology features to achieve cutting-edge power consumption and scalability.
5. NSF International Workshop on Large Scale Neuromorphic Computing. https://www.nuailab.com/workshop.html (2022).
6. Jürgensen, A.-M., Khalili, A., Chicca, E., Indiveri, G. & Nawrot, M. P. A neuromorphic model of olfactory processing and sparse coding in the *Drosophila* larva brain. *Neuromorph. Comput. Eng.* **1**, 024008 (2021).
7. Calimera, A., Macii, E. & Poncino, M. The human brain project and neuromorphic computing. *Funct. Neurol.* **28**, 191–196 (2013).
8. Aimone, J. B. & Parekh, O. The brain's unique take on algorithms. *Nat. Commun.* **14**, 4910 (2023).
9. Gallego, G. et al. Event-based vision: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**, 154–180 (2020).
   An overview of the emerging field of event-based vision, exploring the unique properties and applications of event cameras that capture asynchronous brightness changes and discussing algorithms and techniques developed to unlock their potential for robotics and computer vision.
10. Finateu, T. et al. in *Proc. 2020 IEEE International Solid-State Circuits Conference - (ISSCC)* 112–114 (IEEE, 2020).
11. Vitale, A., Renner, A., Nauer, C., Scaramuzza, D. & Sandamirskaya, Y. in *Proc. 2021 IEEE International Conference on Robotics and Automation (ICRA)* 103–109 (IEEE, 2021).
12. Kudithipudi, D., Saleh, Q., Merkel, C., Thesing, J. & Wysocki, B. Design and analysis of a neuromemristive reservoir computing architecture for biosignal processing. *Front. Neurosci.* **9**, 502 (2016).
13. Severa, W., Lehoucq, R., Parekh, O. & Aimone, J. B. in *Proc. 2018 International Joint Conference on Neural Networks (IJCNN)* 1–8 (IEEE, 2018).
14. Bartolozzi, C., Indiveri, G. & Donati, E. Embodied neuromorphic intelligence. *Nat. Commun.* **13**, 1024 (2022).
15. Volzhenin, K., Changeux, J.-P. & Dumas, G. Multilevel development of cognitive abilities in an artificial neural network. *Proc. Natl Acad. Sci.* **119**, e2201304119 (2022).
16. Rubino, A., Livanelioglu, C., Qiao, N., Payvand, M. & Indiveri, G. Ultra-low-power FDSOI neural circuits for extreme-edge neuromorphic intelligence. *IEEE Trans. Circuits Syst. I Regul. Pap.* **68**, 45–56 (2020).
17. Lee, S.-H., Kravitz, D. J. & Baker, C. I. Disentangling visual imagery and perception of real-world objects. *Neuroimage* **59**, 4064–4073 (2012).
18. Greene, M. R. & Hansen, B. C. Disentangling the independent contributions of visual and conceptual features to the spatiotemporal dynamics of scene categorization. *J. Neurosci.* **40**, 5283–5299 (2020).
19. Wu, B., Liu, Z., Yuan, Z., Sun, G. & Wu, C. in *Proc. Artificial Neural Networks and Machine Learning – ICANN 2017* (eds Lintas, A., Rovetta, S., Verschure, P., Villa, A.) 49–55 (Springer, 2017).
20. Xie, G. Redundancy-aware pruning of convolutional neural networks. *Neural Comput.* **32**, 2532–2556 (2020).
21. Herculano-Houzel, S., Mota, B., Wong, P. & Kaas, J. H. Connectivity-driven white matter scaling and folding in primate cerebral cortex. *Proc. Natl Acad. Sci.* **107**, 19008–19013 (2010).
22. Hoefler, T., Alistarh, D., Ben-Nun, T., Dryden, N. & Peste, A. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. *J. Mach. Learn. Res.* **22**, 10882–11005 (2021).
23. Davies, M. et al. Advancing neuromorphic computing with Loihi: a survey of results and outlook. *Proc. IEEE* **109**, 911–934 (2021).
24. Rathi, N., Agrawal, A., Lee, C., Kosta, A. K. & Roy, K. in *Proc. 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)* 902–907 (IEEE, 2021).
   Exploring various spike representations, training mechanisms and event-driven hardware implementations that can make use of the unique features of spiking neural networks for efficient processing.
25. Cai, J. et al. Sparse neuromorphic computing based on spin-torque diodes. *Appl. Phys. Lett.* **114**, 192402 (2019).
26. Hamilton, K. E., Imam, N. & Humble, T. S. Sparse hardware embedding of spiking neuron systems for community detection. *ACM J. Emerg. Technol. Comput. Syst.* **14**, 1–13 (2018).
27. Boahen, K. Dendrocentric learning for synthetic intelligence. *Nature* **612**, 43–50 (2022).
28. Lin, C.-K. et al. Programming spiking neural networks on Intel's Loihi. *Computer* **51**, 52–61 (2018).
29. Yan, Y. et al. Comparing Loihi with a SpiNNaker 2 prototype on low-latency keyword spotting and adaptive robotic control. *Neuromorph. Comput. Eng.* **1**, 014002 (2021).
30. Schuman, C. D. et al. Opportunities for neuromorphic computing algorithms and applications. *Nat. Comput. Sci.* **2**, 10–19 (2022).
   A review of recent advances in neuromorphic computing algorithms and applications, highlighting the potential benefits and future directions of this emerging technology.
31. Aimone, J. B. et al. A review of non-cognitive applications for neuromorphic computing. *Neuromorph. Comput. Eng.* **2**, 032003 (2022).
32. Sawada, J. et al. in *SC '16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* 130–141 (IEEE, 2016).
33. Disney, A. et al. DANNA: a neuromorphic software ecosystem. *Biol. Inspired Cogn. Archit.* **17**, 49–56 (2016).
34. Cardwell, S. G. Achieving extreme heterogeneity: codesign using neuromorphic processors. Technical Report, Sandia National Laboratories (2021).
   A discussion on the need for innovative co-design tools and architectures to integrate neuromorphic computing, inspired by properties of the brain, with conventional computing platforms to enhance high-performance-computing capabilities.
35. Li, S. et al. in *Proc. 2016 IEEE International Symposium on Circuits and Systems (ISCAS)* 125–128 (IEEE, 2016).
36. Thakur, C. S. et al. Large-scale neuromorphic spiking array processors: a quest to mimic the brain. *Front. Neurosci.* **12**, 891 (2018).
37. Mahowald, M. A. & Mead, C. The silicon retina. *Sci. Am.* **264**, 76–83 (1991).
38. Orchard, G. et al. in *Proc. 2021 IEEE Workshop on Signal Processing Systems (SiPS)* 254–259 (IEEE, 2021).

39. Schemmel, J. et al. in *Proc. 2010 IEEE International Symposium on Circuits and Systems (ISCAS)* 1947–1950 (IEEE, 2010).
40. Richter, O. et al. DYNAP-SE2: a scalable multi-core dynamic neuromorphic asynchronous spiking neural network processor. *Neuromorph. Comput. Eng.* **4**, 014003 (2024).
41. Benjamin, B. V. et al. Neurogrid: a mixed-analog-digital multichip system for large-scale neural simulations. *Proc. IEEE* **102**, 699–716 (2014).
42. Braun, U. et al. Dynamic reconfiguration of frontal brain networks during executive cognition in humans. *Proc. Natl Acad. Sci.* **112**, 11678–11683 (2015).
43. Mack, J. et al. RANC: reconfigurable architecture for neuromorphic computing. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **40**, 2265–2278 (2020).
44. Liu, X. et al. in *Proc. 52nd Annual Design Automation Conference* 1–6 (ACM, 2015).
45. Liu, B., Chen, Y., Wysocki, B. & Huang, T. Reconfigurable neuromorphic computing system with memristor-based synapse design. *Neural Process. Lett.* **41**, 159–167 (2015).
46. Pandit, T. & Kudithipudi, D. in *Proc. Neuro-inspired Computational Elements Workshop* 1–9 (ACM, 2020).
47. Averbeck, B. B., Latham, P. E. & Pouget, A. Neural correlations, population coding and computation. *Nat. Rev. Neurosci.* **7**, 358–366 (2006).
48. Hennig, J. A. et al. Constraints on neural redundancy. *Elife* **7**, e36774 (2018).
49. Pei, J. et al. Towards artificial general intelligence with hybrid Tianjic chip architecture. *Nature* **572**, 106–111 (2019).
50. Lenz, G. et al. Tonic: event-based datasets and transformations. Zenodo https://doi.org/10.5281/zenodo.5079802 (2021). Documentation available under https://tonic.readthedocs.io.
51. Rockpool - Rockpool Documentation. https://rockpool.ai/ (2023).
52. Abreu, S. et al. Neuromorphic intermediate representation. Zenodo https://doi.org/10.5281/zenodo.8105042 (2023).
53. Gleeson, P. et al. NeuroML: a language for describing data driven models of neurons and networks with a high degree of biological detail. *PLoS Comput. Biol.* **6**, e1000815 (2010).
NeuroML, an open-source, XML-based language to describe biologically detailed neuron and network models, enabling their use across several simulators and archiving them in a standardized format.
54. Davison, A. P. et al. PyNN: a common interface for neuronal network simulators. *Front. Neuroinform.* **2**, 11 (2009).
PyNN, an open-source interface that allows users to write a simulation script once and run it without modification on several supported neural network simulators, promoting code sharing, productivity and reliability in computational neuroscience.
55. Baby, S. A., Vinod, B., Chinni, C. & Mitra, K. in *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)* 316–321 (IEEE, 2017).
56. Chan, V., Liu, S.-C. & van Schaik, A. AER EAR: a matched silicon cochlea pair with address event representation interface. *IEEE Tran. Circuits Syst. I Regul. Pap.* **54**, 48–59 (2007).
57. Osborn, L. E. et al. Prosthesis with neuromorphic multilayered e-dermis perceives touch and pain. *Sci. Robot.* **3**, eaat3818 (2018).
58. Kudithipudi, D. et al. Design principles for lifelong learning AI accelerators. *Nat. Electron.* **6**, 807–822 (2023).
An exploration of the design of artificial intelligence accelerators for lifelong learning, which enables neuromorphic systems to learn throughout their lifetime, highlighting key capabilities and metrics to evaluate such accelerators, as well as considering future designs and emerging technologies.
59. Manna, D. L., Vicente-Sola, A., Kirkland, P., Bihl, T. J. & Di Caterina, G. in *Proc. Engineering Applications of Neural Networks. EANN 2023. Communications in Computer and Information Science* (eds Iliadis, L., Maglogiannis, I., Alonso, S., Jayne, C. & Pimenidis, E.) 227–238 (Springer, 2023).
60. Pehle, C.-G. & Pedersen, J. E. Norse - a deep learning library for spiking neural networks. Zenodo https://doi.org/10.5281/zenodo.4422024 (2021).
61. Severa, W., Vineyard, C. M., Dellana, R., Verzi, S. J. & Aimone, J. B. Training deep neural networks for binary communication with the whetstone method. *Nat. Mach. Intell.* **1**, 86–94 (2019).
62. Rhodes, O. et al. sPyNNaker: a software package for running PyNN simulations on SpiNNaker. *Front. Neurosci.* **12**, 816 (2018).
63. Eshraghian, J. K. et al. Training spiking neural networks using lessons from deep learning. *Proc. IEEE* **111**, 1016–1054 (2023).
A tutorial and perspective on applying lessons from decades of deep learning and neuroscience research to biologically plausible spiking neural networks, exploring topics such as gradient-based learning, temporal backpropagation and online learning.
64. Liu, Y., Yanguas-Gil, A., Madireddy, S. & Li, Y. in *Proc. 2023 Design, Automation & Test in Europe Conference & Exhibition (DATE)* 1–6 (IEEE, 2023).
65. Sheik, S., Lenz, G., Bauer, F. & Kuepelioglu, N. SINABS: a simple Pytorch based SNN library specialised for Speck. GitHub https://github.com/synsense/sinabs (2024).
66. Bekolay, T. et al. Nengo: a Python tool for building large-scale functional brain models. *Front. Neuroinform.* **7**, 48 (2014).
67. Aimone, J. B., Severa, W. & Vineyard, C. M. in *Proc. International Conference on Neuromorphic Systems*, 1–8 (ACM, 2019).
68. Vitay, J., Dinkelbach, H. Ü. & Hamker, F. H. ANNarchy: a code generation approach to neural simulations on parallel hardware. *Front. Neuroinform.* **9**, 19 (2015).
69. Magma — Lava documentation. https://lava-nc.org/lava/lava.magma.html (2021).
70. Yavuz, E., Turner, J. & Nowotny, T. GeNN: a code generation framework for accelerated brain simulations. *Sci. Rep.* **6**, 18854 (2016).
71. The NEURON simulator — NEURON documentation. https://nrn.readthedocs.io/en/8.2.3/ (2022).
72. Rothganger, F., Warrender, C. E., Trumbo, D. & Aimone, J. B. N2A: a computational tool for modeling from neurons to algorithms. *Front. Neural Circuits* **8**, 1 (2014).
73. ONNX: Open Neural Network Exchange. https://onnx.ai/ (2019).
74. Jajal, P. et al. Interoperability in Deep Learning: A User Survey and Failure Analysis of ONNX Model Converters. In *Proc. of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA)* (ACM, 2024).
75. Bergstra, J. et al. in *Proc. 9th Python in Science Conference* (eds van der Walt, S. & Millman, J.) 18–24 (2010).
76. Collobert, R., Bengio, S. & Mariéthoz, J. Torch: a modular machine learning software library. Technical Report (IDIAP, 2002).
77. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **32**, 8024–8035 (2019).
78. Stewart, T. C. A technical overview of the Neural Engineering Framework. *Univ. Waterloo* **110** (2012).
79. Sandamirskaya, Y. Dynamic neural fields as a step toward cognitive neuromorphic architectures. *Front. Neurosci.* **7**, 276 (2014).
80. Soures, N., Helfer, P., Daram, A., Pandit, T. & Kudithipudi, D. in *Proc. ICML 2021 Workshop on Theory and Foundation of Continual Learning* (2021).
81. Delbruck, T. jAER open source project. https://jaerproject.org (2007).
82. Schmitt, S. et al. in *Proc. 2017 International Joint Conference on Neural Networks (IJCNN)* 2227–2234 (IEEE, 2017).
A demonstration of how training on an analogue neuromorphic device (the BrainScaleS wafer-scale system) can correct for anomalies induced by the hardware and achieve high accuracy in emulating deep spiking neural networks.
83. Vineyard, C. et al. in *Proc. Annual Neuro-Inspired Computational Elements Conference* 40–49 (ACM, 2022).
84. Davies, M. Benchmarks for progress in neuromorphic computing. *Nat. Mach. Intell.* **1**, 386–388 (2019).
85. Theilman, B. H. et al. in *Proc. 2023 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* 779–787 (2023).
86. Cardwell, S. G. et al. in *Proc. 2022 IEEE International Conference on Rebooting Computing (ICRC)* 57–65 (IEEE, 2022).
87. Orchard, G., Jayawant, A., Cohen, G. K. & Thakor, N. Converting static image datasets to spiking neuromorphic datasets using saccades. *Front. Neurosci.* **9**, 437 (2015).
88. Amir, A. et al. in *Proc. 2017 IEEE Conference on Computer Vision and Pattern Recognition* 7243–7252 (IEEE, 2017).
89. Cramer, B., Stradmann, Y., Schemmel, J. & Zenke, F. The Heidelberg spiking data sets for the systematic evaluation of spiking neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **33**, 2744–2757 (2020).
90. See, H. H. et al. ST-MNIST – the spiking tactile MNIST neuromorphic dataset. Preprint at https://arxiv.org/abs/2005.04319 (2020).
91. Zhu, A. Z. et al. The multivehicle stereo event camera dataset: an event camera dataset for 3D perception. *IEEE Robot. Autom. Lett.* **3**, 2032–2039 (2018).
92. Ceolini, E. et al. Hand-gesture recognition based on EMG and event-based camera sensor fusion: a benchmark in neuromorphic computing. *Front. Neurosci.* **14**, 637 (2020).
93. Perot, E., de Tournemire, P., Nitti, D., Masci, J. & Sironi, A. Learning to detect objects with a 1 megapixel event camera. *Adv. Neural Inf. Process. Syst.* **33**, 16639–16652 (2020).
94. Yik, J. et al. NeuroBench: advancing neuromorphic computing through collaborative, fair and representative benchmarking. Preprint at https://arxiv.org/abs/2304.04640 (2024).
A collaborative framework, NeuroBench, from more than 100 co-authors across academic institutions and industry, aims to standardize the evaluation of neuromorphic computing algorithms and systems through a set of inclusive benchmarking tools and guidelines.
95. Schrimpf, M. et al. Brain-Score: which artificial neural network for object recognition is most brain-like? Preprint at https://www.biorxiv.org/content/10.1101/407007v2 (2020).
96. Schrimpf, M. et al. Integrative benchmarking to advance neurally mechanistic models of human intelligence. *Neuron* **108**, 413–423 (2020).
97. Ritter, P., Schirner, M., McIntosh, A. R. & Jirsa, V. K. The virtual brain integrates computational modeling and multimodal neuroimaging. *Brain Connect.* **3**, 121–145 (2013).
98. Zimmermann, J. et al. Differentiation of Alzheimer's disease based on local and global parameters in personalized Virtual Brain models. *NeuroImage Clin.* **19**, 240–251 (2018).
99. Höppner, S. et al. The SpiNNaker 2 processing element architecture for hybrid digital neuromorphic computing. Preprint at https://arxiv.org/abs/2103.08392 (2022).
100. Yan, Y. et al. Efficient reward-based structural plasticity on a SpiNNaker 2 prototype. *IEEE Trans. Biomed. Circuits Syst.* **13**, 579–591 (2019).
101. Gonzalez, H. A. et al. Hardware acceleration of EEG-based emotion classification systems: a comprehensive survey. *IEEE Trans. Biomed. Circuits Syst.* **15**, 412–442 (2021).
102. Barnell, M., Raymond, C., Brown, D., Wilson, M. & Cote, E. in *Proc. 2019 IEEE High Performance Extreme Computing Conference (HPEC)* 1–5 (IEEE, 2019).
103. Hooker, S. The hardware lottery. *Commun. ACM* **64**, 58–65 (2021).
104. Subramoney, A., Nazeer, K. K., Schöne, M., Mayr, C. & Kappel, D. Efficient recurrent architectures through activity sparsity and sparse back-propagation through time. In *The Eleventh International Conference on Learning Representations (ICLR)* (2023).
Spiking event-based architectures going beyond biologically plausible dynamics, achieving state of the art results in language modelling and gesture recognition.
105. Gonzalez, H. A. et al. SpiNNaker2: a large-scale neuromorphic system for event-based and asynchronous machine learning. *Machine Learning with New Compute Paradigms Workshop at NeurIPS (MLNPCP)* (2023).
106. Sebastian, A., Le Gallo, M., Khaddam-Aljameh, R. & Eleftheriou, E. Memory devices and applications for in-memory computing. *Nat. Nanotechnol.* **15**, 529–544 (2020).
107. Song, M.-K. et al. Recent advances and future prospects for memristive materials, devices, and systems. *ACS Nano* **17**, 11994–12039 (2023).
A comprehensive overview of recent advances and future directions in memristive technology, exploring its potential applications in artificial intelligence, in-sensor computing and probabilistic computing.
108. Zahoor, F., Azni Zulkifli, T. Z. & Khanday, F. A. Resistive random access memory (RRAM): an overview of materials, switching mechanism, performance, multilevel cell (MLC) storage, modeling, and applications. *Nanoscale Res. Lett.* **15**, 90 (2020).
109. Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **65**, 386 (1958).
110. Widrow, B. *Adaptive "Adaline" Neuron Using Chemical "Memistors"* (Stanford Univ., 1960).

111. Raffel, J. I., Mann, J. R., Berger, R., Soares, A. M. & Gilbert, S. A generic architecture for wafer-scale neuromorphic systems. *Lincoln Lab. J.* **2**, 63–76 (1989).
112. Brink, S. et al. A learning-enabled neuron array IC based upon transistor channel models of biological phenomena. *IEEE Trans. Biomed. Circuits. Syst.* **7**, 71–81 (2012).
113. Holler, Tam, Castro & Benson. in *Proc. International 1989 Joint Conference on Neural Networks* 191–196 (IEEE, 1989).
114. Vogelstein, R. J., Mallik, U. & Cauwenberghs, G. in *2004 IEEE International Symposium on Circuits and Systems (IEEE Cat. No. 04CH37512)* V–V (IEEE, 2004).
115. Arthur, J. V. & Boahen, K. Learning in silicon: timing is everything. *Adv. Neural Inf. Process. Syst.* **18** (2005).
116. Wysocki, B., McDonald, N. & Thiem, C. Hardware-based artificial neural networks for size, weight, and power constrained platforms. *Proc. SPIE* **9119**, 911909 (2014).
117. Akopyan, F. et al. TrueNorth: design and tool flow of a 65 mW 1 million neuron programmable neurosynaptic chip. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **34**, 1537–1557 (2015).
118. Müller, E. et al. The operating system of the neuromorphic BrainScaleS-1 system. *Neurocomputing* **501**, 790–810 (2022).
119. Neckar, A. et al. Braindrop: a mixed-signal neuromorphic architecture with a dynamical systems-based programming model. *Proc. IEEE* **107**, 144–164 (2018).
120. Painkras, E. et al. SpiNNaker: a 1-W 18-core system-on-chip for massively-parallel neural network simulation. *IEEE J. Solid-State Circuits* **48**, 1943–1953 (2013).
121. Wan, W. et al. A compute-in-memory chip based on resistive random-access memory. *Nature* **608**, 504–512 (2022).
Co-optimization for combined energy efficiency, functional versatility, and accuracy performance in a fully integrated CMOS-RRAM compute-in-memory microchip for AI on the edge.
122. Modha, D. S. et al. Neural inference at the frontier of energy, space, and time. *Science* **382**, 329–335 (2023).
123. Karia, V., Zohora, F. T., Soures, N. & Kudithipudi, D. in *Proc. 2022 IEEE International Symposium on Circuits and Systems (ISCAS)* 1372–1376 (IEEE, 2022).
124. Wong, H.-S. P. & Salahuddin, S. Memory leads the way to better computing. *Nat. Nanotechnol.* **10**, 191–194 (2015).
125. Fatemi, H., Karia, V., Pandit, T. & Kudithipudi, D. in *Proc. Research Symposium on Tiny Machine Learning* 1–8 (2021).
126. Intel. Lava Software Framework. https://lava-nc.org/ (2021).
127. Abadi, M. et al. TensorFlow: large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/ (2015).
128. Davies, M. Benchmarks for progress in neuromorphic computing. *Nat. Mach. Intell.* **1**, 386–388 (2019).
A perspective on how the neuromorphic computing field needs to shift its focus from exploring complex brain-inspired concepts to establishing quantifiable gains, standardized benchmarks and feasible application challenges to advance into mainstream computing.
129. Schemmel, J., Grübl, A., Millner, S. & Friedmann, S. Specification of the HICANN microchip. FACETS project internal documentation (2010).
130. Patel, K., Jaworski, P., Hays, J., Eliasmith, C. & DeWolf, T. Adaptive spiking control of a 7 DOF arm. Naval Application in Machine Learning (NAML) Workshop (2022).
131. Iyer, L. R., Chua, Y. & Li, H. Is neuromorphic MNIST neuromorphic? Analyzing the discriminative power of neuromorphic datasets in the time domain. *Front. Neurosci.* **15**, 608567 (2021).
132. D'Angelo, G., Perrett, A., Iacono, M., Furber, S. & Bartolozzi, C. Event driven bio-inspired attentive system for the iCub humanoid robot on SpiNNaker. *Neuromorph. Comput. Eng.* **2**, 024008 (2022).
133. Quigley, M. et al. in *Proc. ICRA Workshop on Open Source Software* 5 (2009).
134. Peng, X., Huang, S., Jiang, H., Lu, A. & Yu, S. DNN+ NeuroSim V2. 0: an end-to-end benchmarking framework for compute-in-memory accelerators for on-chip training. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **40**, 2306–2319 (2020).
135. Goodman, D. F. M. & Brette, R. The Brian simulator. *Front. Neurosci.* **3**, 192–197 (2009).
136. Jordan, J. et al. NEST 2.18. 0. Technical Report, Jülich Supercomputing Center (2019).
137. Gleeson, P. et al. Open Source Brain: a collaborative resource for visualizing, analyzing, simulating, and developing standardized models of neurons and circuits. *Neuron* **103**, 395–411 (2019).
The Open Source Brain platform, developed to share, view, analyse and simulate standardized neural circuit models from different brain regions and species, aiming to increase accessibility, transparency and reproducibility for the wider neuroscience community.
138. Feinberg, I. & Campbell, I. G. Sleep EEG changes during adolescence: an index of a fundamental brain reorganization. *Brain Cogn.* **72**, 56–65 (2010).
139. Rossant, C. et al. Fitting neuron models to spike trains. *Front. Neurosci.* **5**, 9 (2011).
140. LeCun, Y. et al. Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1**, 541–551 (1989).
141. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **25** (2012).
A turning point in artificial intelligence research. The introduction of AlexNet was important because it introduced a deep convolutional neural network trained on a massive ImageNet dataset using GPUs, making use of transfer learning and achieving human-level recognition rates with very low error rates.
142. Jouppi, N. P. et al. Tpu v4: in *Proc. 50th Annual International Symposium on Computer Architecture* 1–14 (ACM, 2023).
143. Brown, T. et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **33**, 1877–1901 (2020).
144. Choquette, J. NVIDIA Hopper H100 GPU: scaling performance. *IEEE Micro* **43**, 9–17 (2023).
145. Payvand, M. et al. Self-organization of an inhomogeneous memristive hardware for sequence learning. *Nat. Commun.* **13**, 5793 (2022).
146. Dalgaty, T. et al. In situ learning using intrinsic memristor variability via Markov chain Monte Carlo sampling. *Nat. Electron.* **4**, 151–161 (2021).
147. Zyarah, A. M. & Kudithipudi, D. Neuromemristive architecture of HTM with on-device learning and neurogenesis. *ACM J. Emerg. Technol. Comput. Syst.* **15**, 1–24 (2019).
148. Zohora, F. T., Zyarah, A. M., Soures, N. & Kudithipudi, D. in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)* 1–5 (IEEE, 2020).
149. Li, H. et al. in *Proc. 2016 IEEE Symposium on VLSI Technology* 1–2 (IEEE, 2016).
150. Lee, S., Sohn, J., Jiang, Z., Chen, H.-Y. & Philip Wong, H.-S. Metal oxide-resistive memory using graphene-edge electrodes. *Nat. Commun.* **6**, 8407 (2015).
151. Bai, Y. et al. Study of multi-level characteristics for 3D vertical resistive switching memory. *Sci. Rep.* **4**, 1–7 (2014).
152. Langroudi, H. F. et al. in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition* 3100–3109 (2021).
153. Zohora, F. T., Karia, V., Daram, A. R., Zyarah, A. M. & Kudithipudi, D. in *Proc. 2021 IEEE International Symposium on Circuits and Systems (ISCAS)* 1–5 (IEEE, 2021).
154. Hirohata, A. & Takanashi, K. Future perspectives for spintronic devices. *J. Phys. D Appl. Phys.* **47**, 193001 (2014).
155. Mulaosmanovic, H. et al. Ferroelectric field-effect transistors based on HfO_2: a review. *Nanotechnology* **32**, 502002 (2021).
156. Le Gallo, M. et al. A 64-core mixed-signal in-memory compute chip based on phase-change memory for deep neural network inference. *Nat. Electron.* **6**, 680–693 (2023).
157. Buckley, S. M., Tait, A. N., McCaughan, A. N. & Shastri, B. J. Photonic online learning: a perspective. *Nanophotonics* **12**, 833–845 (2023).
158. Harabi, K.-E. et al. A memristor-based Bayesian machine. *Nat. Electron.* **6**, 52–63 (2023).
159. Wang, W. et al. Neuromorphic motion detection and orientation selectivity by volatile resistive switching memories. *Adv. Intell. Syst.* **3**, 2000224 (2021).
160. Demirağ, Y. et al. in *Proc. 2021 IEEE International Symposium on Circuits and Systems (ISCAS)* 1–5 (IEEE, 2021).

Acknowledgements We thank A. Kanaev of the National Science Foundation, who has supported the large-scale neuromorphic computing workshop under NSF project #2231027. Other grants supporting the effort are NSF grant #2317706, #2332744 and DOE ASCR. We appreciate the valuable guidance on scalability from M. Davies (Intel). We also thank members of the Neuromorphic AI Lab—P. Helfer, A. Daram and V. Karia—for helpful suggestions and feedback. L. Aimone provided support in editing. We acknowledge members of the neuromorphic computing community who contributed to decades of research progress in the field. Certain commercial products, suppliers and software are identified in this paper to foster understanding. This identification does not imply recommendation or endorsement by the authors or their institutions, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

Author contributions D.K. conceptualized the article. D.K., C.S., C.M.V., T.P., C.Me., C.B., R.B., J.B.A., G.O., C.Ma., J.H., C.Y., A.M., S.G.C., M.P., S.B., H.A.G., G.C., C.S.T., A.S., S.F. and S.K. had several rounds of discussions in conceptualization for all sections of the paper and have contributed to the draft manuscript preparation and the main manuscript text. T.P. designed the concept and prepared all of the figures, with feedback from the authors, primarily D.K., C.M.V., C.S., G.C., J.B.A., M.P., C.Ma., H.A.G. and S.G.C. D.K., C.S., C.M.V., T.P., C.Me., R.B., J.B.A., C.Ma., R.B., C.B., C.Y., M.P., H.A.G., G.C., C.S.T., A.S. and S.F. revised the paper critically for important intellectual content. All authors commented on the manuscript and reviewed the final version of the manuscript.

Competing interests The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41586-024-08253-8.

Correspondence and requests for materials should be addressed to Dhireesha Kudithipudi.

Peer review information *Nature* thanks Simon McIntosh-Smith and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.