IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 18, 2021, accepted July 26, 2021, date of publication August 2, 2021, date of current version August 11, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3101579

Review of Algorithms for Artificial Intelligence on Low Memory Devices

PETAR RADANLIEV AND DAVID DE ROURE
Department of Engineering Sciences, University of Oxford, Oxford OX1 3QG, U.K.
Corresponding author: Petar Radanliev (petar.radanliev@eng.ox.ac.uk)

This work was supported in part by The Engineering and Physical Sciences Research Council (EPSRC), U.K., under Grant EP/S035362/1, and in part by the Cisco Research Centre under Grant CG1525381.

ABSTRACT The aim of the article is to conceptualise a more compact and efficient version of algorithms for artificial intelligence (AI). The core objective is to construct the design for a self-optimising and self-adapting autonomous artificial intelligence (AutoAI) that can be applied for edge analytics using real-time data. The methodology is based on synthesising existing knowledge on AI (i.e., knowledge modelling, symbolic reasoning, modal logic), with novel concepts from neuromorphic engineering in combination with deep learning algorithms (i.e., reinforcement learning, neural networks, evolutionary algorithms) and data science (i.e., statistics, linear regression, Bayesian methods). Far-reaching implications are expected from the unique integration of approaches in neuromorphic engineering and edge analytics.

INDEX TERMS Artificial intelligence, algorithms, conceptual design, edge analytics, low memory AI, neuromorphic engineering.

I. INTRODUCTION

The 17th-century philosopher René Descartes first compared the human brain with a working machine, arguing that mathematics and mechanics can explain the complexities of the human brain. Inspired by Descartes, in the 20th century Alan Turing developed the idea of a Turing machine and become known as the father of theoretical computer science and artificial intelligence. In fact, one of oldest and the most famous methods for testing consciousness in artificial intelligence is the Turing test. While many new AI applications can pass the Turing test, today we consider the Turing test as a test of behaviour, and not a test of consciousness. The AI systems we are referring to in this study, require a possession of functional intelligence. In other words, they function in the way they are designed and programmed. To explain this differently, the Turing test is measuring intelligence if a machine can think humanly, while in present advancements of AI, we are focused on machines that can act humanly to maximise outcome of different processes.

This article conducts a state-of-the-art review of current AI algorithms and shows that the fundamentals of our current understanding of AI are over 34 years old. The aim of this research is to commence with conceptual developments for rethinking the essence of AI algorithms and focus on developing algorithms that are sparce, compact and efficient and not based on 'back-propagation'. The objective is to conceptualise the development of AI neural networks based on compact representations (and not on dense representations), that can operate with lower memory requirements, compatible with edge devices.

The structure of the article includes a literature review in chapter 2, chapter 3 discusses the differences between artificial and human intelligence, chapter 4 outlines the methodology, chapter 5 presents the hypothesis and engages in a design of a new ordered pipeline approach, and chapter 6 presents the conclusions and limitations of this study.

The associate editor coordinating the review of this manuscript and approving it for publication was Parul Garg.

II. LITERATURE REVIEW

We conducted a Web of Science search on the topic of Artificial Intelligence which resulted with 75.808 research data records (search conducted on the 18th of July 2021). We analysed these results with a Bar graph (Figure 1) of the last 25 years, and the increase in data records is visibly increased since 2016.

The origins of Artificial Intelligence (AI) can be traced back 200 years ago with the discovery of linear regression [1]. Linear regression endures some similarity with the first idea considered as a method that can make machines learn - the

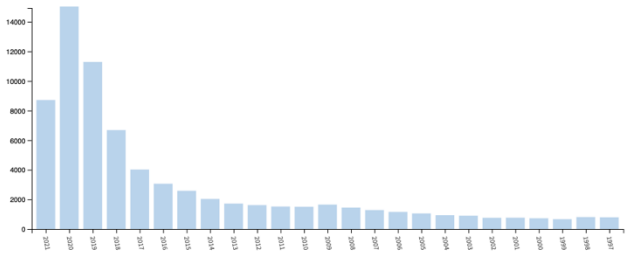

FIGURE 1. Bar graph of research data records on artificial intelligence in the last 25 years.

'Frank Rosenblatt's Perceptron', which resembled a mathematical model of how the neurons in human brains operate [2]. The 'Perceptron' was built upon an earlier design of a 'non-linear function' by 'Mcculoch-Pitts' [3] founded on biological inspiration. The 'Mcculoch-Pitts' work designed a logical calculus of the nervous activity and produced the output of the artificial neuron based on an 'activation function'. Although the 'Mcculoch-Pitts' model didn't include a mechanism for AI learning, the 'Perceptron' presented the first idea on how to make artificial neurons learn. In earlier terminology the 'Perceptron' was commonly referred to as neuron, while in today's terminology it is referred to as a 'unit' and multiple 'units' are structured in a 'layer'. By organising multiple 'units' in a 'layer', the same 'input' can be used to produce multiple 'outputs', as a structure that represents a form of AI neural nets, defined today as 'Artificial Neural Networks' (ANNs). Alternative design of 'neural nets' with artificial neurons defined as 'ADALINE' is to organise 'adaptive linear neurons' incorporated into electrical circuits as chemical 'memistors' i.e., resistors with memory [4]. Although these early developments in the field of AI can be described as different forms of linear regression, they inspired the idea of 'connectionism' i.e., networks of 'units' solving difficult AI problems. The real hopes of that time can be seen in a quote from Dr. Frank Rosenblatt from 1958, claiming that 'the embryo of an electronic computer …will be able to walk, talk, see, write, reproduce itself and be conscious of its existence'.[1] It has been over 60 years since this quote, and the developments of the 'Perceptron' and 'ADALINE', but such AI does not exist yet. This article uses the current knowledge to revisit, review and advance the idea of designing AI based on how neurons in human brains operate.

Since the invention of the 'Perceptron', AI algorithms have been advanced into completing more complicated learning, with mathematical neurons designed to send an input to arbitrarily many neurons, defined as 'hidden layers'. The 'hidden layers' are used to find 'features' from the data and enable the next layer to be more efficient in processing raw data. The advancements of multiple layers or 'multilayer neural nets' created real challenges for applying the rules of the 'Perceptron'. To resolve the multilayer issue, neural net neurons were seen not as 'Perceptrons', instead calculus (the chain rule) and optimisation (stochastic gradient descent) were used to 'backpropagate' the errors. The first implementation of 'backpropagation' on computers [5] for designing neural nets [6] was also inspired by the human brain. This work didn't attract much attention back then, but every AI algorithm used from 1986 until present-day is based on 'backpropagation' [7].

It has been it proven mathematically that neural nets based on 'backpropagation' with multiple layers can (theoretically) implement any function [8], but would require endless memory and computation power. The very first real-world application of 'backpropagation' on large data [9], resulted with the conclusion that key modifications are needed for neural nets to advance towards deep learning. The present-day neural nets operate with a hidden convolutional layer, where neurons can ignore some features by subsampling from a pooling layer. The convolutional and pooling layers resulted with the emergence of the Convolutional Neural Nets (CNNs) as a distinguished from of artificial neural nets, although the actual term used at the time was 'weight sharing' [10]. The inspiration for this new design emerged again from studies on the human brain, which is clear from earlier models, such as the 'Neocognitron' [11] a self-organising neural network for pattern recognition. The advancements towards unsupervised automation emerged from the 'autoencoder' [12] neural net, which represents unsupervised learning [13] and is used for finding hidden patterns and structures in unlabelled data [14] e.g., clustering or inference of latent variables. Unsupervised applications of neural networks has been applied in real-world practical scenarios for self-organised formation of mapping topologies [15] and self-organising adaptive pattern recognition [16]. Unfortunately, pattern recognition is also one of the best achievements in Machine Learning (ML) until present time. Alternative approach that operates as a neural net and contains 'units' that resemble the 'Perceptron' are the 'Boltzmann Machines' [17], inspired from thermodynamics. In 'Boltzmann Machines' the probability distribution is used to allocate energy to states of particles and learning is conducted by reducing the energy of the system. Such neural net can probabilistically learn the hidden structure in raw data – as a generative model. In 'Boltzmann Machines' there are no layers, everything is connected leading to a 'domain independent learning algorithm that modifies the connection strengths between units'. The approach is based on maximum-likelihood algorithms, using Gibbs Sampling to get units of values. Despite its promising characteristics this algorithm was too slow for practical applications, but it inspired the ideas for similar approaches called a 'belief net' [18], 'wake-sleep' algorithm [19], 'The Helmholtz Machine' [20] and 'The Elman Network' [21]. While none of these approaches was considered practical, they represent a different set of ideas on how unsupervised and supervised learning can be applied to train machines - Figure 2.

[1] https://www.nytimes.com/1958/07/08/archives/new-navy-device-learns-by-doing-psychologist-shows-embryo-of.html

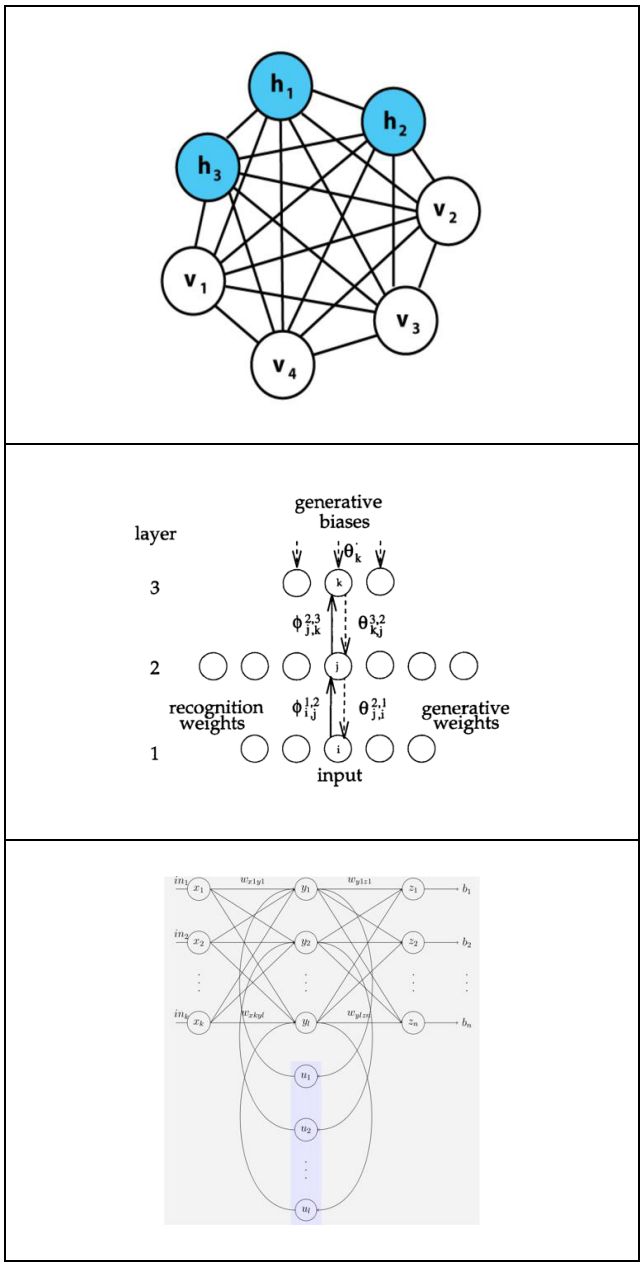

FIGURE 2. 'The Boltzmann Machine' (top), 'The Helmholtz Machine' (middle) and The Elman network (bottom).

'The Elman Network' [21] represents the third branch of algorithms, based on reinforcement learning and called recurrent neural nets (RNNs), to discuss in the current state-of-the-art. This algorithm is slightly challenging to describe, but to summarise in one sentence reinforcement learning is based on an agent deciding on action, based on current state and receiving a reward for making good decisions that maximise the utility. Since its early invention, it has proven a strong performance in controlling dynamical systems [22], especially in robotics [23]. The time-delay neural networks (TDNN) [24] is one reinforcement learning algorithm that operates comparable to normal neural networks and CNNs. The TDNN has been exceeded by recurrent neural nets (RNNs), which is quite different from other neural networks. Other neural networks can be described as 'feedforward', because the output of neurons in one layer is used as input to neurons in the next layer. The RNNs operate differently, by looping the output back into the network, similarly to the 'Boltzmann Machines'. In other words, the output of the last layer is looped as input to the first layer, or the output of a neuron is looped back to the neuron, giving the neural network memory of past inputs. The challenge with this approach is that errors will also be 'propagating' in the loop, but that was resolved with the 'backpropagation through time' i.e., limiting the number of loops/times. This approach to RNNs has proven effective for speech recognition [25], the issue is the emphasis on 'short term dependencies and not long term dependencies' [26], because 'learning long-term dependencies with gradient descent is difficult' [27], and require alternative optimisation methods [28]. The Long Short Term Memory (LSTM) [29] method was proposed as a solution, but the general scientific perception at that time was that neural nets could not be made to work fast and efficiently on computers. This triggered a rebranding of neural nets into 'deep learning' [30] and new solutions emerged based on training a Restricted Boltzmann Machine (RBM) efficiently [31]. These solutions are defined as deep belief networks (DBNs) and remain the state-of-the-art in semi-supervised learning (i.e., combinations of unsupervised and supervised learning). DBNs show that deep machine learning methods are more efficient [32] for difficult problems, than two-layer ANNs or support vector machines. These algorithmic advancements require training data and efforts have been made to create such datasets e.g., Caltech 101,[2] Caltech 256,[3] ImageNet,[4] in combination with high computational power [33]. The combination of the two enables the brute force approach for fast computations with big training sets. However, this is a different approach to what this research is proposing i.e., algorithmic solutions.

This study targets the fundamental problems with 'back-propagation', such as the difficulty of training deep feed-forward neural networks [34]. Deep learning operates and learns best from big datasets, which creates major limitations for resolving many AI problems. By conceptualising the design of novel algorithms that can operate on low-memory, low-computation devices, this article attempts to resolve the major limitation of deep-learning algorithms. Although there are some limited solutions developed in a form of very simple functions e.g., best activation function [35], how these solutions work is still unknown. The methodology in this article builds upon ensemble learning approach called 'Dropout' [36] based on randomly ignoring some neurons in the training datasets, similar to how the random forest method operates.

[2]http://www.vision.caltech.edu/Image_Datasets/Caltech101/
[3]http://www.vision.caltech.edu/Image_Datasets/Caltech256/
[4]http://image-net.org/

III. ARTIFICIAL INTELLIGENCE VS HUMAN INTELLIGENCE

The motivation for this article was the significant differences in how artificial and human intelligence operate. When we compare the differences between AI algorithms and a human nervous system, we find that even the most advanced deep learning algorithms represent very simple mathematical approaches in comparison to a complex nervous system. In other words, the complexities of a human cognitive process are far greater than the current AI algorithms. The problem with current algorithms is that edge devices have very low memory and AI cannot run on those devices. This article conceptualises the design for faster and more efficient processing that will make running AI on edge devices possible. Current state-of-the-art assumes that for better, faster and more efficient processing, we need a better hardware. The new design can construct algorithmic solution to this, based on the knowledge that human brain operates at 20 Watts, while a single GPU operates at 300 Watts. This means that probably there are more efficient versions of the AI algorithms that we are using. Looking at this problem from a different perspective, the current state-of-the-art assumes that to deploy AI in edge devices, we need more memory in edge devices. While the design proposed in this research is to construct algorithmic solution that are more compact and efficient. The solution is founded on the human brain and the knowledge that current AI neural networks are based on dense representations, such as dense multidimensional metrics called 'tensor'. While human brain is extremely sparce, compact and efficient. Therefore, the solution is to develop AI algorithms that are more compact and efficient, so that can be deployed on edge devices. Through synthesising existing knowledge, this article develops the concept for a new form of algorithm (A) that can build upon other algorithms to reach a state of AI that is sparce, compact and efficient and can be used on edge devices. The aim of developing the new concept algorithm is to construct a new AI that operates closer to a human nervous system, and to discover fundamentally new AI approaches. The objective can be summarised as research efforts to construct a self-evolving, self-procreating, self-optimising and self-adapting autonomous AI (AutoAI), that can operate with real-time data on low-memory edge devices (Table 1).

The conceptual design assumes that an iterative progress will enable individual stages to be used as a stepping-stone for developing a novel AI based on compact representations, similar to the human brain. The iterative methodology is best suited for this concept AI design, because all of the steps will benefit from the different phases in individual cycles. For example, the self-evolving algorithm will enhance our understanding of the requirements for a self-evolving and self-procreating AI. The design combines conventional e.g., 'The Boltzmann Machine', 'The Helmholtz Machine' and The Elman network with novel research approaches e.g., autonomous feature selection and feature extraction, automated hyperparameter optimisation, and automated model selection for pipeline optimisation.

TABLE 1. Summary map of the concept for a new form of algorithm (A).

Concept algorithm (A)	Scientific approaches and novel methodologies in individual development stages of the concept algorithm (A).
A1: Develop a self-evolving AutoAI algorithm.	Challenge: AI neural networks need to be based on compact representations, that are autonomous, sparce, compact and efficient, similar to the human brain.
	Concept: Review the basic fundamentals of AI - e.g., 'The Boltzmann Machine', 'The Helmholtz Machine', The Elman network – and combine with state-of-the-art deep learning algorithms to discover novel methods for training AI algorithms how to decode human cognition.
	Output: A self-evolving AI founded on compact representations, that can operate with lower memory requirements, compatible with edge devices.
A2: Develop a self-procreating AutoAI algorithm.	Challenge: A self-procreating autonomous AI algorithm (AutoAI) that can write its own improved algorithms.
	Concept: Autonomous feature selection and feature extraction, automated hyperparameter optimisation, and automated model selection for pipeline optimisation.
	Output: Self-driving, self-securing, self-repairing and self-procreating AI.
A3: Develop self-optimising and self-adaptive autonomous AI algorithms.	Challenge: Construct training scenarios for autonomous AI.
	Concept: Self-optimising for preventing cyber-attacks and self-adaptive to continue operating even when compromised.
	Output: Construct alternative vaccine delivery systems based on new technologies - for resolving shortages of supplies in critical times.

IV. METHODOLOGY–CONCEPTUAL DESIGN

The primary scientific challenge and contribution of this conceptual design is the advancement of a more compact and efficient AI algorithms based on the human nervous system. Secondary scientific challenges and contributions emerge from the scenarios constructed for teaching, training and improving the algorithm.

This design is investigating the ability to use new and emerging forms of data (NEFD) to make AI decisions on low-memory devices. AI is effectively a machine that can learn from structured, semi-structured or unstructured data to build intelligent systems. AI can be classified based on capabilities (e.g., narrow AI, general AI, and super AI) and functionalities (e.g., reactive machine, limited memory, theory of mind, and self-awareness). The current state-of-the-art in AI capabilities is narrow AI (e.g., Apple Siri, Google Translate, IBM Watson, image recognition software), strong AI has not been achieved and requires a full set of cognitive abilities, while super AI is still hypothetical. The methodology initiates with research on narrow AI and advances into strong AI while touching upon conceptual concepts from super AI. The state-of-the-art in AI based on functionalities is more diverse, a reactive machine cannot learn with practice, but it uses the present moment to

make predictions e.g., IBM Deep Blue. Limited memory AI can learn from past data to make decision, but their memory is short-lived e.g., autonomous vehicles. Theory of mind should be able to understand emotions but only exists only as a concept and in limited lab-based models e.g., Sophia. The last AI based on functionalities is self-aware AI will be smarter than the human mind, but the concept is still far away from becoming a reality. The methodology is based on limited memory AI and engages with theory of mind, while contemplating the functionalities of self-aware AI.

V. CONCEPTUAL DESIGN OF A SELF-EVOLVING AI

A. HYPOTHESIS:IT IS POSSIBLE TO CONSTRUCT AI ALGORITHMS BASED ON COMPACT REPRESENTATIONS, SIMILAR TO OUR COMPLEX NERVOUS SYSTEM

1) PHASE 1: IoT-BASED AI

Although AI has made some significant advancements, at present it requires strong computing power and a lot of data, because neural networks are 'over-parametrized' creating significant redundancies [37]. There are many proposed methods for reducing the parameters, but the solutions are not necessarily faster in practical applications. The first method applied in the conceptual design is the 'Adaptive Fastfood' for transforming and 'reparametising' convolutional networks, which results with 'deep fried convnets' [38]. This approach enables parameter reduction without affecting the accuracy. Second method to be applied is the 'Baseline Caffemodel' [39] enhanced with use of 'sparsity-inducing regularizers' [40], because of its ease of use i.e., C++ library with Python and MATLAB bindings for training and deploying general purpose deep models. The methods will be tested in combination with manipulating the linear structure in the convolutional filters to derive approximations and reduce the required computation [41]. The conceptual design will build upon previous related research and will place network parameters into buckets and store only the values of the buckets. One such approach to use is 'HashedNets' [42] to reduce model sizes. Another is to compress deep convnets using vector quantization [43], and to replace layers with global average pooling e.g., 'GoogLenet' [44]. However, these models have limitations in transferring learning from one dataset to another e.g., reusing features for solving a new problem remains a challenge. To continue to evolve, AI needs to advance into a transferable low-memory / low-power technology. Constantly building algorithms for specific problems is not cost effects and large storage requirements prevent the deployment of deep neural networks in mobile apps and IoT devices. There is also the problem with large energy consumption, disabling AI deployment in battery constrained devices. The Phase 1 of the conceptual design sees this evolution operating through a pipeline of methods applied in ordered structure, similar to the 'deep compression' method [45]. This method creates a pipeline of three different approaches, starting with 'pruning' reaching same accuracy with 9x-13x reduction, followed by

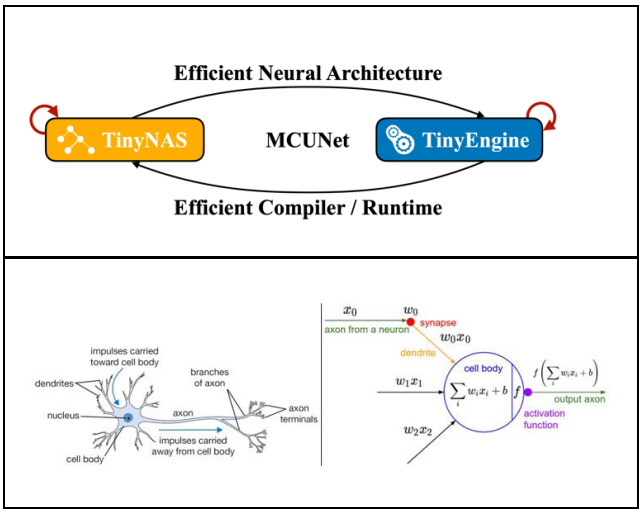

FIGURE 3. IoT-based ML technique with efficient neural architecture and lightweight inference engine (up) and biological neuron explained in a mathematical model (down).

Quantization reaching same accuracy with 27x-31x reduction, and Huffman Encoding reaching same accuracy with 35x-49x reduction. The origins of this method are motivated by how 'the mammalian brain, operates by learning which connections are important' [46].

The ordered pipeline approach will continue with applying an energy efficient inference engine on the compressed deep neural network [47], in combination with a small CNN architecture e.g., SqueezeNet [48] and enable real-time batching to be employed and to improve re-use in layers that are memory limited. To improve the efficiency, the ordered pipeline approach will be applied to reduce the energy needed to fetch their parameters. The ordered pipeline approach adapts and builds upon the recent IoT based learning technique referred to as microcontroller units MCUNet framework (Machine learning on tiny IoT devices based on microcontroller units - MCU), that designs efficient neural architecture (TinyNAS - a two-stage neural architecture search method that first optimises the search space to fit the tiny and diverse resource constraints, and then performs neural architecture search within the optimized space.) and lightweight inference engine (called TinyEngine) [49]. As seen in Figure 3 there is a striking resemblance in the looping approach seen in the 'The Boltzmann Machine', 'The Helmholtz Machine' and 'The Elman network' (Figure 2). The IoT based machine learning technique will be adapted to automatically handle various unpredictable constraints (e.g., different IoT device types, data latency, different energy and memory) while maintaining efficient low energy search. With the new ordered pipeline approach (see Table 2) the IoT based ML technique will be advanced into IoT based AI technique, that adapts to the overall network topology for memory scheduling instead of layer-wise optimisation, to reduce further the memory usage.

2) PHASE 2: NEUROMORPHIC ENGINEERING

Reducing and compacting deep learning algorithm with the new ordered pipeline approach in Table 2 will inevitably

TABLE 2. New ordered pipeline approach.

ED	Conceptual design of the new ordered pipeline approach - exploitation **(E)** and dissemination **(D_i)** of algorithms (A), and scientific milestones **(M)**	A	**M**
E_1	Synthesise knowledge from state-of-the-art methods to design parameter reduction without affecting the accuracy.		M_1
E_2	Building upon the parameter reduction method, develop a new algorithm for adjusting the convolutional filters to derive approximations and reduce the required computation.	A_1	M_2
E_3	Design a pipeline method with an ordered structure that can be applied on an energy efficient inference engine.		M_3
D_{i1}	Develop a new IoT based AI technique – based on 'The Boltzmann Machine', 'The Helmholtz Machine' and 'The Elman network'.	A_2	M_4
E_4	Adapt the IoT based AI technique to the overall network topology for memory scheduling.		M_5
E_5	Advance the concept of looping with neuromorphic computing to design AI based on the human brain.		M_6
D_{i2}	Build a neuromorphic AI with the capacity to store and process the information inside individual units, neurons, and their synapses.	A_3	M_7
D_{i3}	Integrate methods related to neural nets and neuromorphic computing to seek a new ground-breaking AI algorithm.	A_4	M_8
D_{i4}	Design a novel self-adaptive and self-modifying event-driven algorithm, using asynchronous spiking neural networks.	A_5	M_9
D_{i5}	Built a novel AI algorithm that can operate on existing neuromorphic chips.	A_6	M_{10}
D_{i6}	Combine unsupervised learning with adaptive pruning to design a novel energy-efficient neuromorphic algorithm.	A_7	M_{11}

result with improved neural nets, but the human brain is probably running on a completely different mechanism (see Figure 3). In Phase 2, the conceptual design will advance the theory of looping with neuromorphic computing [50] to design AI based on the human brain, with the capacity to store and process the information inside individual units, neurons, and their synapses. Current AI requires a lot of training and a lot of data to classify patterns that can fail with a small change in the data, which is very different to how human brains operate. Neural nets are also not very good at generalising what they have learned from one situation to the next and the success generally depends on the success of defining the correct loss function. But neural nets have some advantages over human brains, such as classifying images or predicting trends from noisy data. This article integrates methods related to neural nets and neuromorphic computing; two approaches that have been studied mostly in isolation. The Phase 1 of the conceptual design is building upon work on neural nets. The Phase 2 is using asynchronous spiking neural network (SNN) to design a novel self-adaptive and self-modifying event-driven algorithm. To design a cognitive computer, AI needs algorithms to reproduce the behaviour of a human brain [51] and neuromorphic processors (chips). Built a novel AI algorithm that can operate on existing neuromorphic chips (e.g., TrueNorth, Loihi, SpiNNaker). The starting approach will be to combine unsupervised learning with adaptive pruning to design a novel energy-efficient neuromorphic algorithm, operating as an energy-efficient neuromorphic system [52]. The conceptual design proposes the stages for AI evolving into a new autonomous form of compact and efficient intelligence, capable of self-procreation.

Regarding running time (i.e., training time, execution time) of the ordered pipeline approach, we suggest that for such complex approach, we need to express the 'time performance' of the method in terms of its time complexity regarding of the input (i.e. using big O notation - a mathematical notation that describes the limiting behaviour of a function when the argument tends towards a particular value or infinity). In other words, a raw measurement of training or classification time could be misleading, as it depends heavily on the hardware platform (i.e., CPU, GPU, RAM), software libraries used (i.e., optimised or not) and quality of the profiler, essentially benchmarking the system used to test the algorithm and reflecting a lot about secondary elements, not on the algorithm itself.

Some examples of application prospects for AI algorithms on IoT devices include intelligent robots in Industry 4.0 manufacturing. Manufacturing is one of the main industries that incorporated new technologies like IoT, artificial intelligence, facial recognition, deep learning, robots and many more. Other application examples include intelligent sensors on self-driving cars, retail analytics and intelligent thermostat solutions. Retail analytics is particularly relevant, because high street shops rely on low cost to remain competitive, and low cost intelligent IoT devices can create real impact in this sector. Intelligent sensors on self-driving cars is also a very relevant application example, considering that driving is a risk to life activity. But the most contemporary application example for AI on low memory devices is the healthcare sector during Covid-19 and other Disease X pandemics. Considering that human contact is risk event on its own, the ability to operate AI on mass produced low cost / low memory devices, can reflex with numerous applications in healthcare and medicine, e.g., smart temperature monitoring, vaccine supply chain analytics, or remote patients monitoring.

VI. CONCLUSION

This article undertakes experimental developments in research on how AI algorithms can operate on low memory / low computation IoT devices and how AI can be designed and constructed to procreate and write its own algorithms. The article presents a new ordered pipeline approach, based on integrating a variety of existing methods in an ordered approach, to increase the efficiency of algorithms in low memory / low computation IoT devices. The new ordered pipeline approach builds upon the state-of-the-art literature on AI and IoT devices (i.e., MCUNet, TinyNAS, TinyEngine), existing datasets (e.g., Caltech 101, Caltech 256, ImageNet) and integrates some of the concepts from early literature on AI algorithms (i.e., The Boltzmann

Machine, The Helmholtz Machine and The Elman Network). The conceptual design is multidisciplinary as it integrates knowledge and methods from statistics and mathematical sciences, engineering sciences, computer sciences and healthcare disciplines. Each drawing on their disciplinary knowledge. The conceptual design integrates people from different disciplines, using a real synthesis of algorithmic, mathematical, computational and engineering approaches. For engineering science alone, the evolution from dense multidimensional metrics to sparce, compact and efficient AI algorithms, could produce breakthroughs in numerous practical applications. The proposed iterative approach is focused on the most fundamental understanding of AI and its application in low-memory devices in a variety of domains e.g., healthcare, smart industries, self-driving vehicles.

In addition, far-reaching implications are expected from the development of new algorithms, and the new mathematical tools, creating implications in many situations and fields of research.

A. LIMITATIONS AND CHALLENGES

The expected challenges in applying the conceptual design in a practical scenario include addressing multiple objectives at speed. The new concept of autonomous AI depends on data training preparation for multiple AI challenges (self-evolving, self-procreating, self-optimising and self-adaptive) and might be difficult to obtain data at this scale and speed. There could also be an incompatibility of the new and emerging forms of data and the autonomous AI training requirements. Alternatives to mitigate this risk include using reinforcement learning to develop AutoAI that has the capacity to understand or learn any intellectual tasks. If AI algorithms are not trained to take risks and learning from its own experience, then the algorithms are missing the training of experimenting in uncertain environment. To address this challenge, we need to enable AI to learn by itself by exploration and exploitation.

ACKNOWLEDGMENT

The authors would like to thank the Fulbright Visiting Scholar Project.

REFERENCES

[1] X. Su and X. Yan, *Linear Regression Analysis: Theory and Computing*. Singapore: World Scientific, 2009.

[2] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychol. Rev.*, vol. 65, no. 6, pp. 386–408, Nov. 1958.

[3] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bull. Math. Biophys.*, vol. 5, pp. 115–133, 1943. [Online]. Available: https://link.springer.com/article/10.1007/BF02478259#citeas, doi: 10.1007/BF02478259.

[4] B. Widrow, "Adaptive 'adaline' neuron using chemical 'memistors,'" Tech. Rep., 1960.

[5] S. Linnainmaa, "Taylor expansion of the accumulated rounding error," *BIT*, vol. 16, pp. 146–160, 1976. Accessed: Aug. 2, 2021, doi: 10.1007/BF01931367.

[6] P. Werbos, "Beyond regression: New tools for prediction and analysis in the behavioral sciences," Ph.D. dissertation, Harvard Univ., Cambridge, MA, USA, 1974. Accessed: Aug. 2, 2021. [Online]. Available: https://dokumen.pub/qdownload/beyond-regression-newtools-for-prediction-and-analysis-in-the-behavioral-sciences.html

[7] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[8] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989.

[9] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[10] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ. San Diego, La Jolla Inst. Cogn. Sci., Cambridge, MA, USA, Tech. Rep., 1985, ch. 8. Accessed: Aug. 2, 2021. [Online]. Available: https://web.stanford.edu/class/psych209a/ReadingsByDate/02_06/PDPVolIChapter8

[11] K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in *Competition and Cooperation in Neural Nets*. Great Britain, U.K.: Springer, 1982, pp. 267–285. Accessed: Aug. 2, 2021. [Online]. Available: http://www.cs.cmu.edu/~bhiksha/courses/deeplearning/Fall.2016/pdfs/Fukushima_Miyake.pdf

[12] H. Bourlard and Y. Kamp, "Auto-association by multilayer perceptrons and singular value decomposition," *Biol. Cybern.*, vol. 59, nos. 4–5, pp. 291–294, 1988.

[13] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Netw.*, vol. 2, no. 1, pp. 53–58, 1989.

[14] G. E. Hinton and R. S. Zemel, "Autoencoders, minimum description length, and Helmholtz free energy," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 6, 1994, pp. 3–10.

[15] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biol. Cybern.*, vol. 43, no. 1, pp. 59–69, 1982.

[16] G. A. Carpenter and S. Grossberg, "The ART of adaptive pattern recognition by a self-organizing neural network," *Computer*, vol. 21, no. 3, pp. 77–88, 1988.

[17] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A learning algorithm for Boltzmann machines," *Cogn. Sci.*, vol. 9, no. 1, pp. 147–169, 1985.

[18] R. M. Neal, "Connectionist learning of belief networks," *Artif. Intell.*, vol. 56, no. 1, pp. 71–113, 1992.

[19] G. Hinton, P. Dayan, B. Frey, and R. Neal, "The 'wake-sleep' algorithm for unsupervised neural networks," *Science*, vol. 268, no. 5214, pp. 1158–1161, May 1995.

[20] P. Dayan, G. E. Hinton, R. M. Neal, and R. S. Zemel, "The Helmholtz machine," *Neural Comput.*, vol. 7, no. 5, pp. 889–904, 1995.

[21] C.-Y. Liou, J.-C. Huang, and W.-C. Yang, "Modeling word perception using the Elman network," *Neurocomputing*, vol. 71, nos. 16–18, pp. 3150–3157, 2008.

[22] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Netw.*, vol. 1, no. 1, pp. 4–27, Mar. 1990.

[23] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," Dept. Artif. Intell. Psychol., Carnegie Mellon Univ. Pittsburgh, PA, USA, 1989. Accessed: Aug. 2, 2021. [Online]. Available: https://papers.nips.cc/paper/1988/file/812b4ba287f5ee0bc9d43bbf5bbe87fb-Paper.pdf

[24] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 3, pp. 328–339, Mar. 1989.

[25] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, Feb. 2003.

[26] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," *The Handbook of Brain Theory and Neural Networks*, vol. 3361, no. 10. ACM Digital Library, 1995, p. 1995. Accessed: Aug. 2, 2021. [Online]. Available: https://dl.acm.org/doi/10.5555/303568.303704

[27] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[28] Y. Bengio, "A connectionist approach to speech recognition," in *Advances in Pattern Recognition Systems Using Neural Network Technologies*. Singapore: World Scientific, 1993, pp. 3–23.
[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[30] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.
[31] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, 2002.
[32] A.-R. Mohamed, T. N. Sainath, G. Dahl, B. Ramabhadran, G. E. Hinton, and M. A. Picheny, "Deep belief networks using discriminative features for phone recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 5060–5063.
[33] R. Raina, A. Madhavan, and A. Y. Ng, "Large-scale deep unsupervised learning using graphics processors," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 873–880.
[34] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.
[35] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.
[36] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012, *arXiv:1207.0580*. [Online]. Available: http://arxiv.org/abs/1207.0580
[37] M. Denil, B. Shakibi, L. Dinh, M. Ranzato, and N. de Freitas, "Predicting parameters in deep learning," 2013, *arXiv:1306.0543*. [Online]. Available: http://arxiv.org/abs/1306.0543
[38] Z. Yang, M. Moczulski, M. Denil, N. De Freitas, L. Song, and Z. Wang, "Deep fried convnets," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1476–1483.
[39] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.
[40] M. D. Collins and P. Kohli, "Memory bounded deep convolutional networks," 2014, *arXiv:1412.1442*. [Online]. Available: http://arxiv.org/abs/1412.1442
[41] E. Denton, W. Zaremba, J. Bruna, Y. LeCun, and R. Fergus, "Exploiting linear structure within convolutional networks for efficient evaluation," 2014, *arXiv:1404.0736*. [Online]. Available: http://arxiv.org/abs/1404.0736
[42] W. Chen, J. Wilson, S. Tyree, K. Weinberger, and Y. Chen, "Compressing neural networks with the hashing trick," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2285–2294.
[43] Y. Gong, L. Liu, M. Yang, and L. Bourdev, "Compressing deep convolutional networks using vector quantization," 2014, *arXiv:1412.6115*. [Online]. Available: http://arxiv.org/abs/1412.6115
[44] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
[45] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," 2016, *arXiv:1510.00149*. [Online]. Available: https://arxiv.org/abs/1510.00149
[46] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural networks," 2015, *arXiv:1506.02626*. [Online]. Available: http://arxiv.org/abs/1506.02626
[47] S. Han, X. Liu, H. Mao, J. Pu, A. Pedram, M. A. Horowitz, and W. J. Dally, "EIE: Efficient inference engine on compressed deep neural network," *ACM SIGARCH Comput. Archit. News*, vol. 44, no. 3, pp. 243–254, 2016.
[48] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size," in *Proc. ICLR*, 2017, pp. 1–13.
[49] J. Lin, W.-M. Chen, Y. Lin, J. Cohn, C. Gan, and S. Han, "MCUNet: Tiny deep learning on IoT devices," Jul. 2020, *arXiv:2007.10319*. [Online]. Available: http://arxiv.org/abs/2007.10319
[50] D. Monroe, "Neuromorphic computing gets ready for the (really) big time," *Commun. ACM*, vol. 57, no. 6, pp. 13–15, Jun. 2014.
[51] C. Witchalls, "A computer that thinks," *New Sci.*, vol. 224, no. 2994, pp. 28–29, Nov. 2014.
[52] W. Guo, M. E. Fouda, H. E. Yantir, A. M. Eltawil, and K. N. Salama, "Unsupervised adaptive weight pruning for energy-efficient neuromorphic systems," *Frontiers Neurosci.*, vol. 14, p. 1189, Nov. 2020.

PETAR RADANLIEV received the Ph.D. degree from the University of Wales, in 2014, and continued his postdoctoral research at Imperial College London, Massachusetts Institute of Technology, and the University of Oxford. He is currently a Postdoctoral Research Associate at the University of Oxford. His current research interests include cyber risk assessment and governance, artificial intelligence, the Internet of Things, cloudification, and the economic impact of cyber risk.

DAVID DE ROURE received the Ph.D. degree from the University of Southampton, in 1990. He went on to hold the post of a Professor of computer science, later directing the U.K. Digital Social Research Programme. He is currently a Professor of e-research at the University of Oxford. His current research interests include social machines, the Internet of Things, and cybersecurity. He is a fellow of the British Computer Society and the Institute of Mathematics and its Applications.