Backpropagation-Free Learning: On the Emergence of Forward-Only Algorithms

Baichuan Huang, *Student Member, IEEE,*, Alexander Ororbia, and Amir Aminifar, *Senior Member, IEEE*

***Abstract*—Backpropagation of errors (BP) plays a vital role in the deep learning domain and exhibits impressive performance in real-world applications. Training state-of-the-art deep neural networks (DNNs) relies almost exclusively on BP, which consumes massive amounts of resources. However, the human brain learns new tasks with remarkable efficiency. This contrast has led to criticism of BP for its biologically implausible nature, underscoring the significant disparity in performance between DNNs and the human brain. The emerging forward-only adaptation, namely backward-pass-free, is proposed to remove the biologically implausible backward pass in deep learning, with the potential to improve memory, computing, and energy efficiency. This survey presents a comprehensive overview and a rethinking of BP-free learning given the emergence of forward-only adaptation. These BP-free algorithms are categorized across different historical stages according to their technical evolution, and are analyzed based on the core principles in each representative work. Moreover, we establish the taxonomy of supervisory signal and biological plausibility, providing a holistic lens into the extent to which various BP-free algorithms perform compared to forward-only adaptation. We further investigate the scalability of BP-free algorithms. Additionally, we discuss the advantages, practical applications, and current limitations of BP-free algorithms, while outlining potential future directions. This survey paper provides a comprehensive overview to foster progress in BP-free learning in the context of emerging forward-only adaptation.**

***Index Terms*—Biologically-plausible learning, Forward-only, Backpropagation, Bio-inspired algorithms, NeuroAI, BP-free, Forward-Forward**

I. Introduction

Synaptic plasticity, an important mechanism involved in brain function, hypothesizes that long-lasting changes/memories are encoded in the synaptic shape of neuronal population activities [1], [2]. This mechanism further constitutes the biological foundation for credit assignment in neuroscience-oriented models, including those based on artificial neural networks (ANNs) [3]. For credit assignment in deep learning, backpropagation of errors (BP) [4] plays a vital role and has been shown to exhibit impressive performance in many real-world applications, e.g., in large language models (LLMs) [5].

To date, modern state-of-the-art deep neural networks (DNNs) consume massive amounts of resources in terms of memory, computation, and energy, posing a threat to the environment [6]. A prime example is GPT-3, an LLM from OpenAI, Inc., which consumes around 1,287 megawatt-hours for training alone, whereas GPT-4 consumes nearly 50 times more energy, around 62,318 megawatt-hours [7]. This problem is particularly relevant today, as most applications focus on DNNs, which are further trained almost exclusively with BP. In contrast, the human brain learns far more efficiently, consuming only around 20 watts [8]–[10]. The presence of such an efficiency gap motivates a deeper investigation into the inherent properties of BP.

BP itself is known to lack biological plausibility [11] due to many factors [12] including: **(a)** weight transport [13]: the error signals move back along the same neural pathways used to forward propagate information; **(b)** non-locality [14]: error derivatives are backpropagated along a global feedback pathway to generate teaching signals; **(c)** update locking [15]: the updates to activity values must wait for all the layers in the forward pass to have been executed (same goes for the updates to weights in the BP pass); **(d)** frozen activities [16]: neural activities are explicitly stored to be used later for synaptic adjustment. *The key to removing the biological implausibilities of BP is to remove the backward pass*, i.e., the process of propagating the error backwards along the same computation path taken in the forward pass, for which there is no such explicit mechanism in the human brain [17]. Neuroscientifically, there is no convincing evidence that the cortex explicitly propagates error derivatives or stores neural activities for use in a subsequent backward pass(es) [18], [19]. Moreover, the backward pass in BP requires significant memory and energy consumption [20], [21] and incurs large computational overheads [22], especially in LLM domain [23]–[28].

As a result, forward-only adaptation [29], [30] has been proposed as an important biologically-plausible and resource-efficient alternative to BP. Forward-only algorithms are a family of learning algorithms that do not recursively propagate error through the reverse of the forward pass [30] nor require any form of recurrent feedback to approximate the effects of a backward pass. These forward-only algorithms not only lead to a path towards neurally-inspired deep learning [31]–[33] with resource-efficiency, but also offer the potential to relieve certain challenges in the subfield of deep learning [34], [35], such as the demand for massive labeled datasets [36]–[38] and DNN vulnerability to perturbations [39], [40]. Accordingly, forward-only adaptation, as an emerging research track within BP-free learning, stands out as a critical and prospective area of study in both deep learning and biological synaptic plasticity. This motivates us to revisit and provide a

This research has been partially supported by the Swedish Wallenberg AI, Autonomous Systems and Software Program (WASP), Swedish Research Council, Swedish Foundation for Strategic Research, ELLIIT Strategic Research Environment, and an unrestricted gift from Google.

Baichuan Huang, and Amir Aminifar are with the Department of Electrical and Information Technology, Lund University, Sweden (email: baichuan.huang@eit.lth.se). Alexander Ororbia is with the Department of Computer Science and Cognitive Science program, Rochester Institute of Technology, United States.

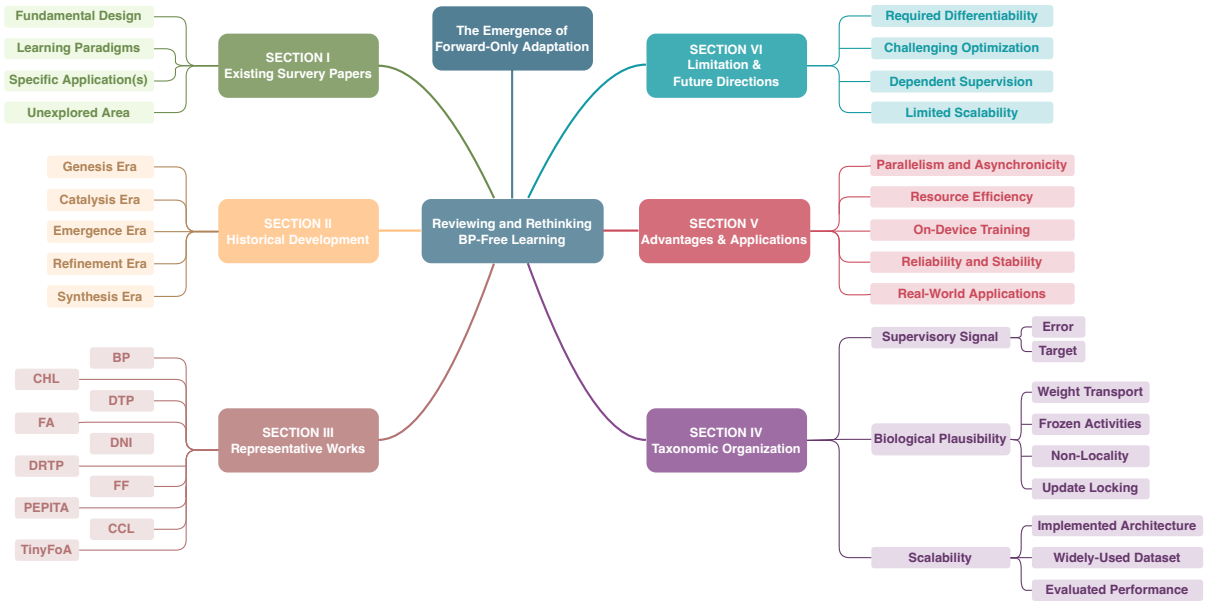


Fig. 1. The structure of our survey: Section I introduces the motivation of the review on the emergence of forward-only adaptation; Section II presents the historical development of BP-free algorithms across different eras; Section III reviews the representative works with a progression from relying on backward pass(es) to more directly leaning on forward pass(es); Section IV establishes the taxonomy of the supervisory signal utilized and the biological plausibility to enable a holistic investigation into the extent to which various BP-free algorithms perform compared to forward-only adaptation, as well as an investigation into the scalability for BP-free algorithms; Section V elaborates on the advantages and practical applications of BP-free algorithms; Section VI discusses the limitations and the potential future of BP-free learning in the context of emerging forward-only adaptation.

comprehensive survey and rethinking BP-free learning given the emergence of forward-only adaptation.

To date, there exist several surveys and review papers investigating biologically-plausible BP-free algorithms from different perspectives:

1) *Fundamental design*: the surveys [12], [41] study neurobiological credit assignment, where [12] constructs a taxonomy that mainly considers the source and means of production of the signals that drive synaptic plasticity. Moreover, [42] introduces the vast majority of biologically-plausible models of learning, including synaptic plasticity, neuro-modulation, and meta-plasticity;
2) *Learning paradigms*: the work of [43] provides a comprehensive study of brain-inspired mechanisms for continual learning. [44] introduces biologically-grounded synaptic plasticity models in unsupervised deep learning to spiking seural networks (SNNs), while [45] examines several BP-free algorithms but mainly focuses on the topics of on-device training of neural architecture search (NAS);
3) *Specific application(s)*: the work of [46] broadly reviews brain-inspired studies mainly for the case of remote sensing. [47] reviews brain-inspired deep learning algorithms for learning, perception, and cognition, whereas [48] mainly reviews the brain-inspired paradigm in the context of computer vision.

However, with the emergence of forward-only adaptation as a promising BP-free learning, a systematic categorization of technical evolution and thorough analysis of scalability for BP-free algorithms are more necessary than ever, yet still lacking. In particular, no prior survey paper has offered a comprehensive examination of the relation between the extent to which BP-free algorithms eliminate or approximate the backward pass, and their resulting levels of performance in biological plausibility—an aspect that remains largely unexplored.

In this survey paper, we conduct a comprehensive review and rethink of BP-free learning given the emergence of forward-only adaptation, as presented in Fig. 1. First, we revisit and review BP-free learning schemes from the 1970s through 2025, subsequently categorizing them into different eras of development in terms of technical evolution and historical development, as presented in Section II. Next, we demonstrate the representative works of these BP-free algorithms along these eras in Section III, revealing a progression from relying on backward pass(es) to more directly leaning on forward pass(es). We then, in Section IV-A, establish a taxonomy of the supervisory signal used to drive learning, depending on whether the real label or the activation is exploited to update the weights (or not). To investigate the extent to which various BP-free algorithms perform compared to forward-only adaptation, we further develop our taxonomy of biological plausibilities in terms of weight transport, non-locality, update locking, and frozen activities elaborated in Section IV-B. Subsequently, we extend our taxonomic organization to include the scalability of the implemented (neuronal) architecture, the widely-used dataset that the implemented (neuronal) architecture is applied to, as well as the evaluation of the performance of BP-free algorithms in Section IV-C. Finally, we describe the advantages and applications of BP-free algorithms in Section V, exploring the inherent limitations as well as the potential future of these types of learning algorithms in Section VI. In essence, this survey paper seeks to provide a holistic understanding of biologically-plausible BP-free algorithms to inspire and motivate future developments in bridging the gap between the adaptivity of human brains and

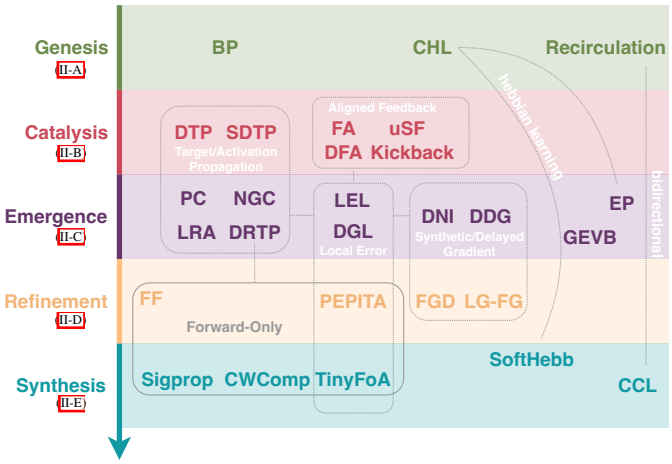


Fig. 2. The technical evolution of BP-free learning across different eras. There are four technical groups (Target/Activation Propagation, Aligned Feedback, Local Error, and Synthetic/Delayed Gradient) and independent technical tracks (Hebbian learning-based and bidirectional-based tracks), as well as the emerging forward-only adaptation.

deep learning, in service of the greater goals of NeuroAI and brain-inspired computing.

II. On the Historical Development of BP-Free Learning

In this section, we review the BP-free algorithms from the 1970s to 2025 and divide these into different eras of development, including the *genesis era*, *catalysis era*, *emergence era*, *refinement era*, and *synthesis era*. Furthermore, for a clear and holistic review of their development, we group these algorithms based on the technical track they have taken, linking them by relevance and similarity, as outlined in Fig. 2. In the technical evolution, we mainly establish four technical groups: **(a)** Target/Activation Propagation – which mainly means that the supervisory signal is not the gradients that are back-propagated as in BP; **(b)** Aligned Feedback – which represents the feedback connections are not the exact symmetric weights happening in forward pass; **(c)** Local Error – which means the supervisory signal is not generated globally; **(d)** Synthetic/Delayed Gradient – which has the potential to train DNNs in parallel and asynchronously. Note that these four groups are not completely independent but are interconnected and interact with each other. Additionally, there are also some relatively independent technical tracks (Hebbian learning-based and bidirectional-based tracks), as well as the emerging forward-only adaptation. All of these developments are treated as a pathway to the BP-free algorithms that are elaborated from Section II-A to Section II-E. Furthermore, the chronological development of BP-free learning since BP's proposal is demonstrated in Fig. 3.

A. Genesis Era

Our first period of history to examine, the genesis era, includes the birth of credit assignment in deep learning and the construction of its foundation. In terms of credit assignment in deep learning, BP, much in the form it is known today, originally emerged from early work done in the 1970s to the 1980s [11], [70], [71]. In particular, the paper [4] marked an important step in the popularization of BP. In BP, there is one forward pass (inference phase) and one backward pass (learning phase). The error information from the forward pass needs to be iteratively propagated back along the pathways taken in the forward pass (i.e., the backward pass) in order to update DNN parameters. Across the decades, BP has catalyzed many successes in deep learning across real-world applications [5]. Although highly effective, BP has long been considered to be biologically-implausible [11] and, in parallel to BP's developmental progress, many biologically-plausible learning schemes have been proposed as alternatives to BP.

Within this era, recirculation [49] was proposed, which is a scheme based on the bidirectional passing of error signals to drive parameter updates. The recirculation algorithm is further generalized to learn arbitrary input/output mappings [72]. Contrastive Hebbian learning (CHL) [50], [69], [73], grounded in Hebbian learning [3], was proposed to update DNNs based on the correlation of pre- and post-synaptic activities. Although CHL was originally formulated for the Boltzmann machine [74], we regard its extension to deterministic networks [50] as the representative work. CHL entails a comparison between a "clamped state" and "free state" (each state requiring running the DNN for multiple iterations to convergence) [69]; we remark that a variant of CHL which employed random feedback weights was later studied [75]. In contrast to these schemes, the extreme learning machine (ELM) [76] was also proposed, representing the idea that most of an architecture could be random and fixed with only the output layer learned (via a shallow form of BP). ELM leads to several orders of magnitude speedup in the training process over BP, only effective in single-hidden-layer neural models.

B. Catalysis Era

The catalysis era entails the rapid changes that lead to stimulation (and an increase in momentum) for subsequent developments in biologically-plausible learning procedures. For instance, difference target propagation (DTP) [51] was proposed as means of computing and propagating (backward) targets rather than gradients to drive learning, based on earlier ideas of target propagation (TP) [77] and recirculation [49]. The paper [78] introduces a difference reconstruction loss to improve the training of the feedback parameters in DTP. The fixed-weight DTP [79] keeps the feedback weights constant during training. Moreover, the paper [80] finds that for complex datasets such as CIFAR [81] and ImageNet [82], the variants of DTP perform significantly worse than BP, opening questions about whether new architectures and algorithms are required to scale these biologically-motivated deep learning algorithms. Accordingly, a simplified difference target propagation (SDTP) [80], never transporting gradients by error propagation, is extended to more complicated datasets and network architectures. At the same time, the paper [83] scales DTP to ImageNet [82] by efficiently aligning feedforward and feedback weights, with local difference reconstruction loss.

Other schemes focused on the nature of the propagation pathway (of gradients), such as the kickback procedure [52]

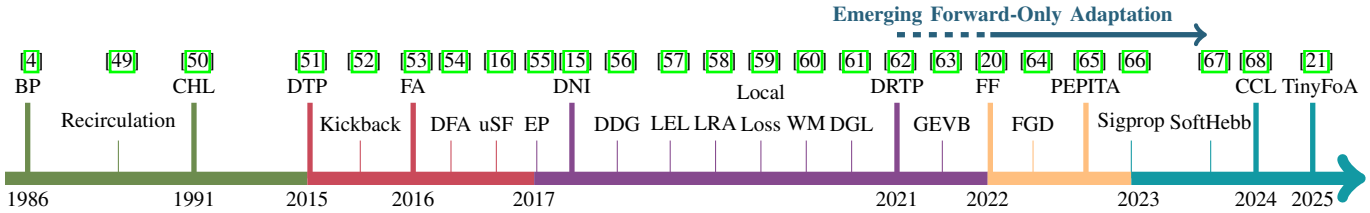


Fig. 3. The chronological development of BP-free learning since backpropagation of errors (BP) [4]. The BP [4], CHL [69], DTP [51], FA [53], DNI [15], DRTP [62], FF [20], PEPITA [65] CCL [68], and TinyFoA [21] are selected as the representative works, revealing a progression from relying on backward pass(es) to more directly leaning on forward pass(es) as the central driver of credit assignment itself.

and feedback alignment (FA) [53]. The kickback procedure [52] proposed a truncation of the feedback in backprop by summing over paths on the way to the output layers, only keeping the error computed at the output layer by decomposing BP. FA [53] centers around the use of random synaptic feedback connections, in place of the symmetrical weights in the backward pass. Although FA resulted in a useful, simple learning process, it was often to perform significantly worse on hard visual recognition problems such as ImageNet [82]. Later, the paper [84] provides a theory of feedback alignment algorithms. Additionally, direct feedback alignment (DFA) [54] built on FA by directly propagating the error at the output layer to hidden layers via fixed random feedback connections, further relaxing some of the restrictions imposed by FA, and the paper [85] scaled DFA to modern deep learning tasks and architectures. Complementary to this development, uniform sign-concordant feedback (uSF) [16] was proposed, which only exploited the sign of weights in its backward pass. uSF indicates that sign-concordance is very important for achieving strong performance and that normalization methods such as batch normalization are necessary for these asymmetric styles of backpropagation-like algorithms to work. Subsequently, the paper [86] extends uSF to large-scale datasets, e.g., ImageNet [87], approaching BP-trained performance, with the aid of ResNet [88] and RetinaNet [89] structures.

C. Emergence Era

The emergence era marks the time when biologically-plausible learning schemes begin to emerge spontaneously, resulting in a surge in complexity and creativity.

Schemes based on the alternating direction method of multipliers (ADMM) and Bregman iteration were proposed, called scalable ADMM for neural networks (ADMM-NN) [90], offering the benefits of parallel and independent weight updates, while not requiring any backpropagation of any gradients. In [91], an activation-propagation approach was designed that also did not rely on the gradient chain rule of backpropagation, offering further support for processes that updated weights across layers in parallel.

In the realm of energy-based learning, and much inspired by CHL, equilibrium propagation (EP) [55] was proposed as a learning framework that compared to two phases of forward-propagation driven computation and was demonstrated to be extensible/applicable to most neural architectures. Useful generalizations of EP have been proposed, further relaxing some of its constraints [92]. In the same vein, predictive coding (PC) [93], in a single phase of energy-based learning, defines a generative model for the expected activation and minimizes the error between the expected and the exact activities. PC approximates BP but in a more biologically-plausible fashion [93], [94] and, more importantly, only depends on local learning (Hebbian-like) rules (with layerwise-parallel weight updates). In essence, schemes that fall under PC are motivated by the idea that the brain centrally focuses on minimizing prediction errors between what is expected to happen and what actually happens [95]. Moreover, PC has been extended to skip connection as well as ResNet and transformer structures [96], showing that PC is a more biologically-plausible equivalent to BP in the way that the updating of parameters on any DNN occurs. Far more biologically-plausible yet effective variants of PC have been proposed, such as neural generative coding (NGC) [97], which have been shown to offer powerful frameworks for learning probabilistic generative models that remain competitive with BP.

Other schemes modify key components of BP so as to sidestep its limitations; for instance, decoupled neural interfaces (DNI) [15] introduces a feedback module that synthesizes gradient in place of the real gradients produced by BP. Notably, DNI can be trained independently and asynchronously. The paper [98] investigates DNI [15] and compares it with FA [53], DFA [54], and kickback [52]. In addition, decoupling the backpropagation algorithm using delayed gradients (DDG) [56] proposed a means of decoupling the BP algorithm using delayed gradients, also resulting in parallel updates to a neural architecture (however, it relies on constraints similar to BP, such as symmetric forward propagation and backward feedback). Other approaches draw inspiration from early deep learning schemes, notably greedy layer-wise training approaches, such as local error learning (LEL) [57], which exploits random auxiliary classifiers to produce errors locally; the training process of LEL can be either independent, layer-by-layer, or simultaneous. Decoupled greedy learning (DGL) [61] was based on a greedy learning objective, yielding parallel updates and parallelism; interestingly a synchronous form of DGL avoids the need for a non-parallel forward pass through replay buffers, which provide better generalization than sequential greedy optimization. Other approaches such as local loss (LL) [59] used two separate single-layer sub-networks, i.e., one that optimized a similarity matching loss and another the cross-entropy loss, to facilitate layer-wise training; however, these schemes often employed BP in some

parts of the system to improve performance. The work of [99] extended the greedy layer-wise learning to deeper networks and large-scale datasets including ImageNet [82].

Other important approaches emerged during this time as well, often offering more flexible mechanisms to do what earlier algorithms such as DTP [51] and recirculation [49] did. Notably, local representation alignment (LRA) [58] focused on the integration of separate random feedback matrices to produce targets, rather than weight updates. Variants such as LRA-fdbk [58] offered tools for resolving certain issues related to BP and were even shown to work for non-differentiable neuronal units. Error-driven local representation Alignment (LRA-E) [100] later introduced biologically-plausible ways of learning the feedback connections themselves, even though targets would still need to be propagated sequentially as in DTP [51]; Rec-LRA [101] broke free of this sequential target propagation and was shown to scale to large-scale architectures and large benchmarks, again including ImageNet [82]. Building on FA [53] and DFA [54], schemes such as weight mirrors (WM) [60] would alternate between an "engaged" mode and a "mirror" mode. The engaged mode operated just like the FA [53], while the mirror mode tried to push the fixed random matrices closer to the forward propagation weights. Later approaches [102] relieved several issues in WM, such as the need for bidirectional connections [60].

Other later approaches offered learning formats that more closely resembled forward-only learning itself, i.e., *they began to rely less on feedback pathways in general and more on what could be computed with forward propagation mechanisms*. Direct random target projection (DRTP) [62] proposed the use of one-hot-encoded labels as the target to drive the error computed for each layer of a DNN; however, it was empirically demonstrated that DRTP resulted in a higher accuracy degradation (than other schemes such as FA [53]). Global error-vector broadcasting (GEVB) [63] entailed a scheme for "spreading" the error vector to all hidden neurons; the learning rule of GEVB could be interpreted as a form of three-factor Hebbian learning. GEVB requires each node to be represented by a vector unit with the same dimensionality as the output class, unfortunately implying a higher computational overhead. In contrast, deep feedback control (DFC) [103] extended the control theory approach to credit assignment, resulting in a learning rule that was fully local in space and time. From a Hebbian learning perspective, biologically-plausible convolutional neural network (BioCNN) [104] added lateral connectivity to a locally-connected network (LC) and updated weights with a cross-correlational scheme.

D. Refinement Era

The refinement era resulted in biologically-plausible forward-only algorithms that more directly lean on forward propagation as the central driver of credit assignment itself. Notably, the forward-forward algorithm (FF) [20] was proposed as a framework that was driven by two forward passes, i.e., one "positive" pass and one "negative pass" (but without the dependency or long iteration time of older schemes such as CHL [69]), in place of the forward and backward passes of BP. Collaborative FF [105] represented a refinement of FF through a "collaboration" between layers of the whole network. Other variants of FF, e.g., the work of [106], exploit the local backpropagation, which converges faster than the original FF. The local backpropagation avoids the gradient calculation for layers such as non-differentiable areas of the network. FF observes the high sparsity of representations [107] and has been extended to convolutional neural network (CNN) [108]. Additionally, the lightweight inference scheme specifically designed for DNNs trained by FF [20], e.g., LightFF [109], is proposed.

Forward gradient descent (FGD) [64] proposes the gradient calculation based on the forward pass without backpropagation, having a faster runtime compared to backpropagation. FGD is based on automatic differentiation [110]. Local Greedy Forward Gradient (LG-FG) [111] scales the forward gradient to large-scale tasks. The paper [112] reduces the variance of FGD by activity perturbation with DFA [54] and momentum. AsyncFGD [113] is proposed as an asynchronous version of FGD [64], for efficient parallel training.

Present-the-error-to-perturb-the-input-to-modulate-activity (PEPITA) [65] is proposed to replace the backward pass with a modulated forward pass. In the modulated forward pass, the input is modulated by the error from the previous forward pass. A PEPITA variant [114] comes out as a memory-efficient version, removing the need to store the errors and activations. Besides, the paper [115] investigates BP and forward-only algorithms, finding that PEPITA [65] and FF [20] are more vulnerable to binary activations.

E. Synthesis Era

The synthesis era demonstrates that multiple inspirations from previous eras are integrated, forming a stable and innovative pattern. For instance, feed-forward with delayed feedback ($\mathbf{F}^3$) [116] uses the delayed error information to approximate gradients, side-stepping the need for shared backward and forward pass parameters. The predictive forward-forward algorithm (PFF) [117] integrates PC [93] with FF [20], simultaneously learning a representation and a generative model, further doing so in a layer-wise parallel manner. Signal propagation (Sigprop) [66] only utilizes forward passes to update the model; the input data and learning signals use the same forward pass, without requiring further structural or computational overheads. Sigprop also does not store any activations or block the next input, which means the updates can be made in parallel. Additionally, layer-wise feedback propagation (LFP) [118] distributes an output-wise reward signal based on the respective contributions, without the need for gradient calculation; LFP [118] is based on layer-wise relevance propagation (LRP) [119], which backpropagates the one-hot relevance value at the output layer in the backward pass. Beyond this, tiny restricted Coulomb energy (TinyRCE) [120], [121] was proposed as a forward-only learning approach based on a hyperspherical classifier.

Multilayer SoftHebb [67], which combines Hebbian learning in a soft winner-take-all-style network, offered a softmax-based plasticity rule [122] that could be exploited to train

DNNs efficiently. The work of [123] based its premise around multi-compartment pyramidal neuron models [124], proposing a correlative information maximization scheme between layer activations. CwComp [125] was proposed as a form of channel-wise competitive learning in the context of CNNs, derived from FF [20], where [68] proposed counter-current learning (CCL) motivated by the counter-current exchange mechanisms observed in biological systems.

Recently, TinyFoA [21] was proposed as a memory-efficient on-device learning algorithm, relying solely on layer-wise forward passes and partially updating each layer to reduce dynamic training memory requirements. In contrast, [126] exploited a basis of periodic vectors, taken from studies that neuronal ensembles in the brain synchronize their activity at particular frequencies, in order to replace the fixed random matrix in Bio-FO [127]. Finally, the cascaded forward (CaFo) algorithm [128] only utilized the forward pass to directly estimate the prediction errors as well as update the parameters, eliminating the necessity for negative sampling as in FF schemes [20].

III. Representative Works

In this section, we review the representation of typical BP-free algorithms in comparison to BP, revealing a progression from relying on backward pass(es) (Fig. 4) to more directly leaning on forward pass(es) (Fig. 5). Based on the technical evolution shown in Fig. 2 and the chronological development shown in Fig. 3, we select particular representative works: 1) BP [4]; 2) CHL [69] from the Hebbian learning-based technique track; 3) DTP [51] and DRTP [62] from the technique group of Target/Activation Propagation; 4) FA [53] from the technique group of Aligned Feedback; 5) DNI [15] from the technique group of Synthetic/Delayed Gradients; 6) FF [20] from the emerging forward-only adaptation; 7) PEPITA [65], and TinyFoA [21] from the technique groups of Local Error and the emerging forward-only adaptation; and, 8) CCL [68] from the bidirectional-based technique track.

To comprehensively compare the representative works of biologically-plausible procedures, we first define what is meant by a standard forward pass. In the standard forward pass of a DNN, considering a fully connected network with L layers, the input $\boldsymbol{x} \in \mathbb{R}^{D_x \times 1}$ and the one-hot label $\boldsymbol{y} \in \mathbb{R}^{N_c \times 1}$ are leveraged for training the network, where D_x is the size of input $\boldsymbol{x}$ and N_c is the number of classes. The activations of the hidden layer l of the network are denoted as $\boldsymbol{h}_l$, where $\boldsymbol{h}_0 = \boldsymbol{x}$ (the bottom layer is set to be equal to the input data). For the hidden layer l, the activation $\boldsymbol{h}_l$ is computed as follows:

$$\boldsymbol{z}_l = \boldsymbol{W}_l \boldsymbol{h}_{l-1} + \boldsymbol{b}_l, \ \boldsymbol{h}_l = \sigma_l(\boldsymbol{z}_l) \text{ for } L \geq l \geq 1 \quad (1)$$

where, for $\boldsymbol{h}_l \in \mathbb{R}^{D_l \times 1}$ and $\boldsymbol{h}_{l-1} \in \mathbb{R}^{D_{l-1} \times 1}$, D_l is the size of output of layer l and D_{l-1} is the size of input to layer l. $\boldsymbol{W}_l \in \mathbb{R}^{D_l \times D_{l-1}}$ and $\boldsymbol{b}_l \in \mathbb{R}^{D_l \times 1}$ are the DNN weights/biases between hidden layers $l-1$ and l, respectively. Note that, $\boldsymbol{z}_l \in \mathbb{R}^{D_l \times 1}$ contain the "pre-activation" values for layer l, i.e., the weighted sum of inputs $\boldsymbol{h}_{l-1}$. σ_l is the activation function for the hidden layer l; $\sigma_l{}'(z_l)$ is the partial derivative of the activation function σ_l with respect to z_l. Given σ_l is the ReLU activation function for simplification, σ_l' is 1 when the input is positive and 0 otherwise.

A. Backpropagation of Errors (BP)

BP [4] comprises a standard forward pass and a backward pass. In the standard backward pass, i.e., backpropagating the error, given loss, such as cross-entropy, for the last layer $\boldsymbol{h}_L \in \mathbb{R}^{N_c \times 1}$ and the real label $\boldsymbol{y}$, the loss is denoted as $\mathcal{L} = \mathcal{L}_{\text{CE}}(\boldsymbol{h}_L, \boldsymbol{y})$. Based on the chain rule of calculus, the errors for updating weights of the hidden layer l are calculated as follows:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{W}_l} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{h}_L} \cdot \frac{\partial \boldsymbol{h}_L}{\partial \boldsymbol{h}_{L-1}} \cdot \frac{\partial \boldsymbol{h}_{L-1}}{\partial \boldsymbol{h}_{L-2}} \cdot \ldots \cdot \frac{\partial \boldsymbol{h}_{l+1}}{\partial \boldsymbol{h}_l} \cdot \frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{W}_l}. \quad (2)$$

To simplify Eq. 2, we assume that $\sigma_l' = 1$. Then, $\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{h}_{l-1}} = \boldsymbol{W}_l^T$, $\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{W}_l} = \boldsymbol{h}_{l-1}^T$. Consequently, the gradient of $\boldsymbol{W}_l$ is denoted as follows:

$$\nabla \boldsymbol{W}_l = \left[(\boldsymbol{W}_{l+1})^T \ldots (\boldsymbol{W}_{L-1})^T (\boldsymbol{W}_L)^T \delta \boldsymbol{h}_L \right] \otimes (\boldsymbol{h}_{l-1})^T, \quad (3)$$

where $\otimes$ is the Kronecker Product. Next, the weight updating is denoted as:

$$\boldsymbol{W}_l = \boldsymbol{W}_l - \eta \nabla \boldsymbol{W}_l,$$

where η is the learning rate. It also applies to the biases $\boldsymbol{b}_l$.

B. Contrastive Hebbian Learning (CHL)

CHL [69] is an extension of Hebbian learning [3], entailing the construction of an energy-based neural system that has a "free phase" and a "clamped phase". In addition to the standard forward pass, there are feedback connections that are required to be symmetric to the weights in Eq. 1. Taking the layer l as an example, the weights in the feedback connections are $\gamma \boldsymbol{W}_l^T \in \mathbb{R}^{D_{l-1} \times D_l}$, where γ is a positive factor. Therefore, the feedback connection is denoted as follows:

$$\boldsymbol{h}_{l-1} = \sigma_{l-1}(\gamma \boldsymbol{W}_l^T \boldsymbol{h}_l + \boldsymbol{b}_{l-1}).$$

The free phase entails fixing the input layer to a particular data sample $\boldsymbol{x}$ and letting the network converge to a steady state, resulting in the anti-Hebbian gradient of $\boldsymbol{W}_l$:

$$\nabla \boldsymbol{W}_l^{\text{free}} = -\gamma^{l-L} \boldsymbol{h}_l^{\text{free}} \otimes (\boldsymbol{h}_{l-1}^{\text{free}})^T,$$

where γ^{l-L} represents the consecutive product of the positive factor γ from layer l to L. The clamped phase requires that an output layer is clamped at a desired $\boldsymbol{y}$ target value (along with the input clamped to $\boldsymbol{x}$) and then again letting the network converge to a steady state, resulting in the Hebbian gradient of $\boldsymbol{W}_l$ below:

$$\nabla \boldsymbol{W}_l^{\text{clamped}} = \gamma^{l-L} \boldsymbol{h}_l^{\text{clamped}} \otimes (\boldsymbol{h}_{l-1}^{\text{clamped}})^T.$$

After running a clamped and free phase, the gradient of $\boldsymbol{W}_l$ made to the network becomes the following:

$$\nabla \boldsymbol{W}_l = \gamma^{l-L} \left(\boldsymbol{h}_l^{\text{clamped}} \otimes (\boldsymbol{h}_{l-1}^{\text{clamped}})^T - \boldsymbol{h}_l^{\text{free}} \otimes (\boldsymbol{h}_{l-1}^{\text{free}})^T \right).$$

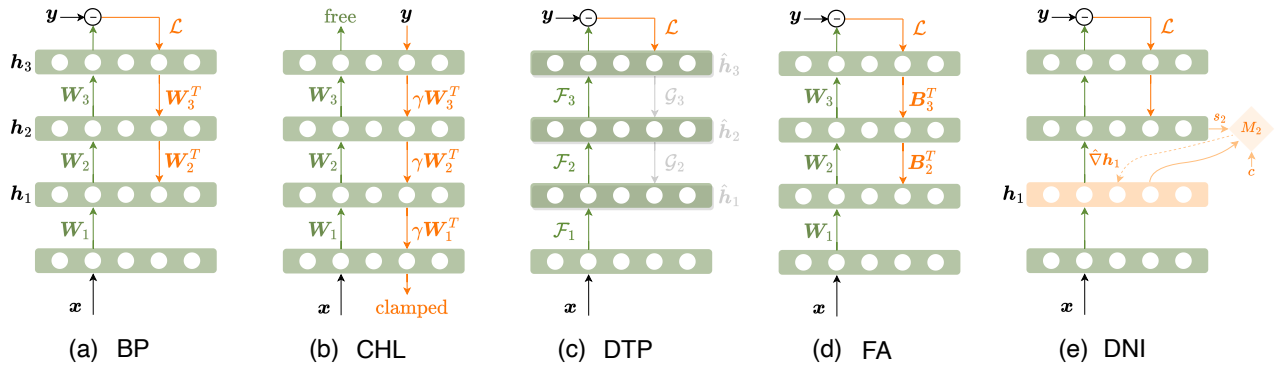


Fig. 4. The representations of (a) BP [4]; (b) CHL [69]; (c) DTP [51]; (d) FA [53]; (e) DNI [15]. The forward passes are illustrated by green arrows, backward passes by orange arrows, and input data/label passes by black arrows. These algorithms, without exception, rely on the backward pass(es).

C. Difference Target Propagation (DTP)

DTP [51] aims to compute targets rather than gradients, where the target, in this context, means targeted activation values, as opposed to targeted error. To concisely represent the standard forward pass for DTP, we define a new function composition $\mathcal{F}$, where $\mathcal{F}_l : \boldsymbol{h}_{l-1} \rightarrow \boldsymbol{h}_l$. Given this function composition, Eq. 1 is simplified as $\boldsymbol{h}_l = \mathcal{F}_l(\boldsymbol{h}_{l-1})$. For each $\mathcal{F}_l$, there is an approximate inverse $\mathcal{G}_l$ that results in:

$$\mathcal{G}_l(\mathcal{F}_l(\boldsymbol{h}_{l-1})) \approx \boldsymbol{h}_{l-1}.$$

For the last layer L, the target $\hat{\boldsymbol{h}}_L$ is directly driven from the gradient of the global loss, i.e., $\hat{\boldsymbol{h}}_L = \boldsymbol{h}_L - \hat{\eta}\frac{\partial \mathcal{L}}{\partial \boldsymbol{h}_L}$ where $\hat{\eta}$ is a small step size. The target $\hat{\boldsymbol{h}}_{l-1}$ for hidden layer l is:

$$\hat{\boldsymbol{h}}_{l-1} = \boldsymbol{h}_{l-1} + \mathcal{G}_l(\hat{\boldsymbol{h}}_l) - \mathcal{G}_l(\boldsymbol{h}_l).$$

Then, for updating the weights $\boldsymbol{W}_l$ in the standard forward pass, DTP exploits the per-layer loss $\mathcal{L}_l^{\text{DTP}}$ for layer l. The per-layer loss $\mathcal{L}_l^{\text{DTP}}$ is denoted as follows:

$$\mathcal{L}_l^{\text{DTP}} = \left\| \mathcal{F}_l(\boldsymbol{h}_{l-1}) - \hat{\boldsymbol{h}}_l \right\|_2^2.$$

Therefore, the gradient of $\boldsymbol{W}_l$ is denoted as follows:

$$\nabla \boldsymbol{W}_l = \frac{\partial \mathcal{L}_l^{\text{DTP}}}{\partial \boldsymbol{h}_l} \cdot \frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{h}_{l-1}}.$$

D. Feedback Alignment (FA)

FA [53], in its simplest, original form, demonstrates the precise and symmetric backward connectivity in BP is not required for effective error propagation. FA replaces the symmetry weights in Eq. 3, i.e., $\boldsymbol{W}_l^T$, with a matrix of fixed random weights, $\boldsymbol{B}_l^T \in \mathbb{R}^{D_{l-1} \times D_l}$. For the chain rule in Eq. 2, FA obtains the asymmetric projection of the error, which is represented as follows:

$$\frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{h}_{l-1}} = \boldsymbol{B}_l^T.$$

Then, the gradient of $\boldsymbol{W}_l$ is denoted as follows:

$$\nabla \boldsymbol{W}_l = \left[(\boldsymbol{B}_{l+1})^T ... (\boldsymbol{B}_{L-1})^T (\boldsymbol{B}_L)^T \delta \boldsymbol{h}_L\right] \otimes (\boldsymbol{h}_{l-1})^T.$$

E. Decoupled Neural Interfaces (DNI)

DNI [15] synthesizes gradients locally in place of true gradients that are backpropagated in the backward pass of BP, hence decoupling the DNN and updating the layer parameters independently and asynchronously. For the layer $l+1$, a synthetic gradient model M_{l+1} is constructed based on the message $\boldsymbol{h}_l$ from layer l, the current state s_{l+1} of layer $l+1$, and potentially any other information c (for example, real label or context information). The predicted gradient $\hat{\nabla}\boldsymbol{h}_l$ for layer l is then denoted as follows:

$$\hat{\nabla}\boldsymbol{h}_l = M_{l+1}(\boldsymbol{h}_l, s_{l+1}, c),$$

where the parameters of the synthetic gradient model M_{l+1} are trained by minimizing the difference between the true gradients $\nabla \boldsymbol{h}_l$ and the predicted gradients $\hat{\nabla}\boldsymbol{h}_l$, i.e., $\|\hat{\nabla}\boldsymbol{h}_l - \nabla \boldsymbol{h}_l\|_2^2$. Then, the gradient of $\boldsymbol{W}_l$ is denoted as follows:

$$\nabla \boldsymbol{W}_l = \hat{\nabla}\boldsymbol{h}_l \otimes (\boldsymbol{h}_{l-1})^T.$$

F. Difference Random Target Projection (DRTP)

DRTP [62] demonstrated that the one-hot labels normally available in supervised classification scenarios can serve as a surrogate for the error sign, instead of the targets involved in the reverse feedback pathways of DTP [51] or LRA [100]. DRTP exploits fixed random weights $\boldsymbol{B}_l^T \in \mathbb{R}^{D_l \times N_c}$ for each layer and, notably, represents one of the earlier versions of what could be considered to be (layerwise) forward-only training. The gradient of $\boldsymbol{W}_l$ is denoted as in the following manner:

$$\nabla \boldsymbol{W}_l = (\boldsymbol{B}_l^T \cdot \boldsymbol{y}) \otimes (\boldsymbol{h}_{l-1})^T.$$

G. Forward-Forward (FF) Learning

FF [20] replaces the forward and backward pass in BP with instead two forward passes; namely, the positive (denoted by superscript p) forward pass and the negative (denoted by superscript n) forward pass. In its supervised format, the positive forward pass consists of transmitting information related to data samples that contain the correct labels and the negative forward pass consists of the samples augmented with incorrect labels [20]. For hidden layer l, the gradient of the

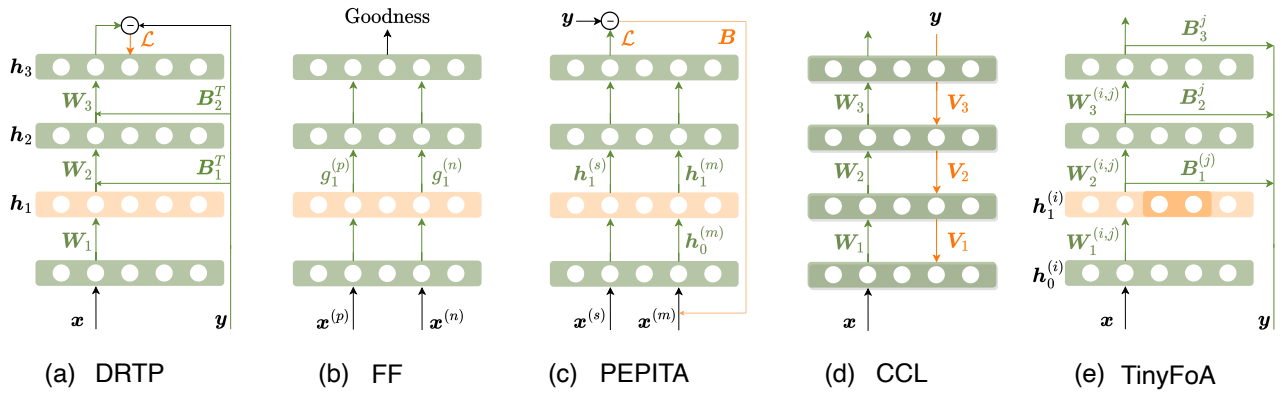


Fig. 5. The representations of (a) DRTP [62]; (b) FF [20]; (c) PEPITA [65]; (d) CCL [68]; (e) TinyFoA [21]. These algorithms (except for CCL) offer learning formats that more closely resemble forward-only learning itself, i.e., they began to rely less on feedback pathways in general and more on what could be computed with forward propagation mechanisms, to more directly lean on forward propagation as the central driver of credit assignment itself.

weights $\boldsymbol{W}_l$ (given the activities computed for layer l, both for positive (p) and negative (n) phases; shown in the first two lines of the equation below) is then denoted as:

$$\begin{aligned}
\boldsymbol{h}_l^{(p)} &= \sigma_l(\boldsymbol{W}_l \boldsymbol{h}_{l-1}^{(p)} + \boldsymbol{b}_l), \\
\boldsymbol{h}_l^{(n)} &= \sigma_l(\boldsymbol{W}_l \boldsymbol{h}_{l-1}^{(n)} + \boldsymbol{b}_l), \\
g_l^{(p)} &= (\boldsymbol{h}_l^{(p)})^T \boldsymbol{h}_l^{(p)}, \\
g_l^{(n)} &= (\boldsymbol{h}_l^{(n)})^T \boldsymbol{h}_l^{(n)}, \\
\nabla \boldsymbol{W}_l &= \boldsymbol{h}_l^{(p)} \cdot \delta g_l^{(p)} \otimes (\boldsymbol{h}_{l-1}^{(p)})^T \\
&\quad + \boldsymbol{h}_l^{(n)} \cdot \delta g_l^{(n)} \otimes (\boldsymbol{h}_{l-1}^{(n)})^T,
\end{aligned} \tag{4}$$

where g_l is the "goodness" measurement [20], roughly reporting the probability that a layer's representation of the input is one within the domain of valid data (as opposed to invalid/negative samples). The gradient of the goodness g_l for hidden layer l is $\delta g_l^{(p)} = \text{sigmoid}(g_l^{(p)} - \theta) - 1$ for samples and layerwise activities of the positive forward pass (where θ is a goodness threshold), and $\delta g_l^{(n)} = \text{sigmoid}(g_l^{(n)} - \theta) - 0$ for samples and activities in the negative forward pass.

H. Present-the-Error-to-Perturb-the-Input-To-Modulate-Activity (PEPITA)

A more modern algorithm known as PEPITA [65], similar in some ways to FF [20], also seeks to replace the backward pass with a second forward pass; however, the second forward pass is specifically modulated by the error of the network. Using a standard forward pass (denoted via superscript s) and a modulated forward pass (denoted via superscript m), for hidden layer l, PEPITA's gradient of the weights $\boldsymbol{W}_l$ is denoted as:

$$\begin{aligned}
\boldsymbol{h}_l^{(s)} &= \sigma_l(\boldsymbol{W}_l \boldsymbol{h}_{l-1}^{(s)} + \boldsymbol{b}_l), \\
\boldsymbol{h}_l^{(m)} &= \sigma_l(\boldsymbol{W}_l \boldsymbol{h}_{l-1}^{(m)} + \boldsymbol{b}_l), \\
\nabla \boldsymbol{W}_l &= (\boldsymbol{h}_l^{(s)} - \boldsymbol{h}_l^{(m)}) \otimes (\boldsymbol{h}_{l-1}^{(m)})^T,
\end{aligned} \tag{5}$$

where $\boldsymbol{h}_0^{(s)} = x^{(s)}$ and $\boldsymbol{h}_0^{(m)} = x^{(m)}$, where $\boldsymbol{x}^{(m)} = \boldsymbol{x}^{(s)} + \boldsymbol{B}(\boldsymbol{h}_L^{(s)} - \boldsymbol{y})$ and $\boldsymbol{B}$ is a fixed random feedback matrix.

I. Counter-Current Learning (CCL)

CCL [68] contains the bidirectional passes, i.e., the separated feedforward pass and feedback pass. The feedforward pass is the standard forward pass, and the feedback pass is denoted as follows:

$$\hat{\boldsymbol{h}}_{l-1} = \sigma_{l-1}(\boldsymbol{V}_l \hat{\boldsymbol{h}}_l + \boldsymbol{b}_{l-1}),$$

where $\hat{\boldsymbol{h}}_l \in \mathbb{R}^{D_l \times 1}$ and $\hat{\boldsymbol{h}}_{l-1} \in \mathbb{R}^{D_{l-1} \times 1}$, and $\hat{\boldsymbol{h}}_L$ is $\boldsymbol{y}$. The $\boldsymbol{V}_l \in \mathbb{R}^{D_{l-1} \times D_l}$ and $\boldsymbol{b}_{l-1} \in \mathbb{R}^{D_{l-1} \times 1}$ are the weights and bias in the feedback pass, respectively.

Then, CCL aims to minimize the difference between activations in the bidirectional passes for all layers except for the output layer; this process is denoted as follows:

$$\mathcal{L}^{\text{CCL}} = \min_{\{\boldsymbol{W}_{l|l\neq L}, \boldsymbol{V}_l\}} \sum_{l=1}^{L} \left\| \text{norm}(\boldsymbol{h}_l) \text{norm}(\hat{\boldsymbol{h}}_l)^T - \boldsymbol{I} \right\|,$$

where $\boldsymbol{I}$ is the identity matrix and norm is the normalization. For updating $\boldsymbol{W}_L$, $\mathcal{L}_{\text{CE}}(\boldsymbol{h}_L, \boldsymbol{y})$ is exploited. Consequently, the gradient of $\boldsymbol{W}_{l|l\neq L}$ is denoted via the following application of the chain rule of calculus:

$$\nabla \boldsymbol{W}_{l|l\neq L} = \frac{\partial \mathcal{L}^{\text{CCL}}}{\partial \boldsymbol{h}_l} \cdot \frac{\partial \boldsymbol{h}_l}{\partial \boldsymbol{h}_{l-1}}.$$

J. Memory-Efficient Forward-Only Algorithm (TinyFoA)

TinyFoA [21], like FF and PEPITA, is based on only the use of forward passes, sidestepping the need for BP, and furthermore targeting some biological implausibility issues related to learning and DNN architecture design. For each layer l, an auxiliary classifier with a fixed random matrix $\boldsymbol{B}_l \in \mathbb{R}^{N_c \times D_l}$ is employed. Next, the layerwise loss, $\mathcal{L}_{\text{CE}}(\boldsymbol{B}_l \boldsymbol{h}_l, \boldsymbol{y})$, generates the supervisory signal required for each layer. To calculate gradients, the input activation $\boldsymbol{h}_l$ and the output activation $\boldsymbol{h}_{l-1}$ are divided into N_s slices. The sliced input activations are denoted as $\boldsymbol{h}_{l-1}^{(i)} \in \mathbb{R}^{\frac{D_{l-1}}{N_s} \times 1}$ and the sliced output activations are denoted as $\boldsymbol{h}_l^{(j)} \in \mathbb{R}^{\frac{D_l}{N_s} \times 1}$, where $i = 1, 2, ..., N_s$ and $j = 1, 2, ..., N_s$. As a result, the weights

are divided into N_s^2 slices, namely, $\boldsymbol{W}_l^{(i,j)} \in \mathbb{R}^{\frac{D_l \times D_{l-1}}{N_s^2}}$. The gradient of weights $\nabla \boldsymbol{W}_l^{(i,j)}$ is then finally denoted as follows:

$$\nabla \boldsymbol{W}_l^{(i,j)} = (\boldsymbol{B}_l^{(j)})^T \frac{\partial \mathcal{L}_{\text{CE}}(\boldsymbol{B}_l \boldsymbol{h}_l, \boldsymbol{y})}{\partial (\boldsymbol{B}_l \boldsymbol{h}_l)} \otimes (\boldsymbol{h}_{l-1}^{(i)})^T,$$

where variants of TinyFoA further utilize binarized variations in the forward pass, i.e., $\boldsymbol{W}_l^b = \frac{1}{n} \sum_{i=1}^{n} |\boldsymbol{W}_l^i| \cdot \text{Sign}(\boldsymbol{W}_l)$ [115] or, alternatively, introduce the sparsity mask $\boldsymbol{S}_l \in \mathbb{R}^{D_l \times D_{l-1}}$ into the forward pass, i.e., $\boldsymbol{h}_l = \sigma_l(\boldsymbol{S}_l \odot \boldsymbol{W}_l \boldsymbol{h}_{l-1} + \boldsymbol{b}_l)$ [127], where $\odot$ is the Hadamard Product.

IV. Taxonomic Organization

In this section, we group the aforementioned BP-free algorithms across three dimensions: supervisory signal, biological plausibility, and scalability. This particular organization will allow us to investigate the extent to which these BP-free algorithms perform compared to forward-only adaptation. First, in Section IV-A, we will divide the supervisory signal into error and target (activation), which helps us to understand the type of signals that drive learning in synaptic plasticity (similar in spirit to what was done in [12]). Second, in Section IV-B, in terms of biological plausibility, we will consider four key issues inherent to BP including weight transport [13], frozen activities [16], non-locality [14], and update locking [15]. Third, in Section IV-C, in terms of scalability, we will consider – for these biologically plausible BP-free algorithms – the implemented architecture used, the widely-used dataset, and how performance is evaluated in terms of accuracy.

A. Supervisory Signal

Here, we establish a simple taxonomy for the supervisory signal, including error and target (activation) values that drive supervision and learning. In principle, error, in the context of this survey, means that the label is explicitly exploited to update the weights, i.e., estimating the error between the output of classifier and label and calculating the gradient or the feedback. Target means that the activation in the hidden layers is exploited directly to update the weights, with the implicit usage or even no usage of labels (see [12] for a more in-depth treatment of the full set of learning signals that can drive biophysical credit assignment.). Based on the type of signals that drive learning and plasticity, we group the aforementioned BP-free algorithms into two classes, namely under error or target signals. Concretely, for error signals, we examine the following: BP [4], uSF [16], Kickback [52], FA [53], DFA [54], WM [60], GEVB [63], LEL [57], DDG [56], DGL [61], $\mathbf{F}^3$ [116], LFP [118], Bio-FO [127], TinyFoA [21], and LL [59][1]. In contrast, for the target signals, we examine: DNI [15], recirculation [49], DTP [51], DRTP [62], LRA [58], LRA-E [100], FF [20], PEPITA [65], PFF [117], PC [94], CHL [69], [75], EP [55], ADMM-NN [90], Sigprop [66], SoftHebb [67], FGD [64], and CCL [68].

B. Biological Plausibility

To investigate the bio-plausible learning in terms of memory, computation, and energy (usage), we further organize the biological implausibilities/issues inherent to BP [4], including: weight transport [13], frozen activity values [16], non-locality [14], and update locking [15]; see Table I. If we consider, for simplicity, a 3-hidden layer fully-connected network (FC) model, the forward pass in Eq. 1 and backward pass in Eq. 2 and 3 are recalled below:

$$\boldsymbol{h}_l = \sigma_l(\boldsymbol{W}_l \boldsymbol{h}_{l-1} + \boldsymbol{b}_l), \ \delta = \frac{\partial \mathcal{L}(\boldsymbol{h}_L, \boldsymbol{y})}{\partial \boldsymbol{h}_L},$$
$$\nabla \boldsymbol{W}_l = \left[\boldsymbol{W}_{l+1}^T ... \boldsymbol{W}_{L-1}^T \boldsymbol{W}_L^T \delta\right] \otimes (\boldsymbol{h}_{l-1})^T.$$

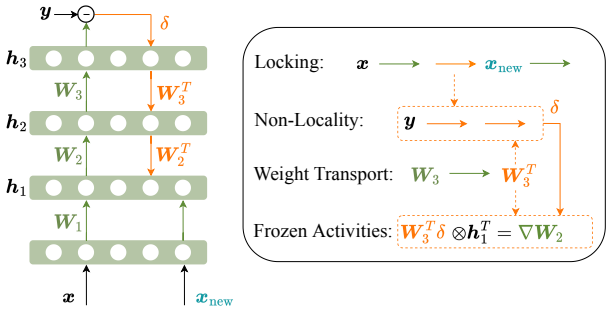


Fig. 6. The overview of biological implausibilities for BP [4]. $\boldsymbol{\delta}$ denotes the supervisory signal; $\boldsymbol{x}_{\text{new}}$ is the new input training sample.

As shown in Fig. 6, the inherent biological implausibilities are primarily reflected in the fact that: **1)** weight transport demonstrates that the symmetric weight value of $\boldsymbol{W}_3$, namely, $\boldsymbol{W}_3^T$ is utilized to backpropagate the supervisory signal (i.e., error in this case); **2)** frozen activities imply that the activations $\boldsymbol{h}_1$ are necessary for generating the gradient information $\nabla \boldsymbol{W}_2$; **3)** non-locality shows that the gradient information, for instance $\nabla \boldsymbol{W}_2$, is not generated locally; and, **4)** locking implies that the new input $\boldsymbol{x}_{\text{new}}$ is locked/unavailable until the previous input $\boldsymbol{x}$ finishes its forward pass and backward pass. Based on this set of biological implausibilities, we next investigate the extent to which BP-free algorithms perform compared to forward-only adaptation, as presented in Table I.

1) Weight Transport: The weight transport issue relates to the fact that supervisory/teaching signals are propagated backwards along a neural pathway, transported by the weights used in the system's the forward pass, as illustrated in Fig. 7 (a). We regard the case involving this transport (Fig. 7 (a)) through weights as "weight transport". In contrast, the case without using the transport of weights is labeled "weight-transport-free". Based on this, we group the aforementioned biologically-plausible BP-free algorithms as presented in the two columns of "Weight-Transport-Free" in Table I. Two ✘ in the two columns denote that an algorithm has the issue of weight transport, whereas one ✔ in either column denotes that it is weight-transport-free. BP suffers from the issue of weight transport because BP has ✘ in both columns.

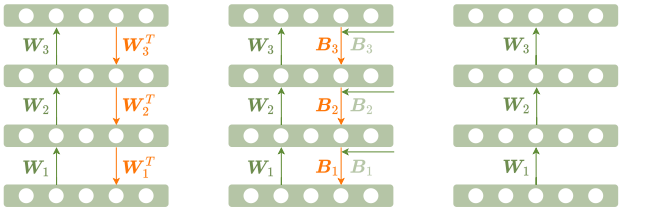
Moreover, we divide weight-transport-free into two subclasses based on the way how it avoids exploiting symmetric weights. Fig. 7 (b.1) shows that weight transport can be avoided by introducing other (feedback) matrices or through random projection with fixed matrices. uSF [16] uses the sign of weights to replace the transport of weights; CHL (in some variants) employs random feedback [75], FA [53],

[1]Since the backprop free version of LL [59] shows deteriorated performance, we consider the standard version of LL [59] here.

TABLE I

THE EXTENT TO WHICH THE BIOLOGICALLY PLAUSIBLE BP-FREE ALGORITHMS PERFORM, AS COMPARED TO FORWARD-ONLY ADAPTATION. THIS ANALYSIS SHOWS THAT FORWARD-ONLY ADAPTATION (DENOTED BY †) STANDS OUT AS A PROMISING AREA TO STUDY.

Algorithms	Weight-Transport-Free		Incompletely Frozen		Locality		Unlocking		
	Others	None	Part-Frozen	Non-Frozen	Single-Layer	Multi-Layer	Backwads	Update	Forward
Fig.	7 (b.1)	7 (b.2)	8 (b.1)	8 (b.2)	9 (b.1)	9 (b.2)	10 (b)	10 (c)	10 (d)
BP [4]	✘	✘	✘	✘	✘	✘	✘	✘	✘
Recirculation [49]	✘	✘	✔			✔	✔	✔	
CHL [69]	✘	✘	✘	✘		✔	✔		
DTP [51]	✔		✘	✘	✘	✘	✘	✘	✘
Kickback [52]	✔		✘	✘	✘	✘	✔		
FA [53]	✔		✘	✘	✘	✘	✘	✘	✘
DFA [54]	✔		✘	✘	✘	✘	✔		
uSF [16]	✔		✘	✘	✘	✘	✘	✘	✘
ADMM-NN [90]	✘	✘	✔			✔	✔	✔	
EP [55]	✘	✘	✘	✘		✔	✔		
DNI [15]	✘	✘		✔	✔		✔	✔	✔
DDG [56]	✘	✘	✘	✘	✘	✘	✔	✔	
LEL [57]	✔			✔	✔		✔	✔	
LRA-E [100]	✔		✘	✘	✘	✘	✘	✘	✘
LL [59]	✘	✘		✔	✔		✔	✔	
WM [60]	✔		✘	✘	✘	✘	✘	✘	✘
DGL [61]		✔		✔	✔		✔	✔	✔
DRTP [62]†	✔			✔	✔		✔	✔	
GEVB [63]		✔	✘	✘	✘	✘	✔		
PC [94]	✘	✘	✘	✘		✔	✔		
FF [20]†		✔		✔	✔		✔	✔	
FGD [64]†		✔		✔	✔		✔	✔	
PEPITA [65]†		✔	✘	✘	✔		✔		
Sigprop [66]†		✔		✔	✔		✔	✔	
LFP [118]	✘	✘	✘	✘	✘	✘	✘	✘	✘
SoftHebb [67]†		✔		✔	✔		✔	✔	
CCL [68]	✔		✘	✘		✔	✘	✘	✘
TinyFoA [21]†	✔			✔	✔		✔	✔	



(a) Symmetric Weights (b.1) Other Matrices (b.2) No Feedback Weights

Fig. 7. A visual comparison between learning algorithms that make use of: **(a)** symmetric weights (backprop-like learning), **(b.1)** other (feedback) matrices; and, **(b.2)** no feedback weights. The last two setups would be considered to be "weight-transport-free".

DFA [54], LEL [57], LRA-E [100], WM [60], DRTP [62], $\mathbf{F}^3$ [116], Bio-FO [127], and TinyFoA [21] use a separate random feedback matrix instead, where, in the cases of LRA-E [100] and WM [60], the feedback is trainable. In addition, DTP [51] approximates the inverse of weights in the feedback mapping while Kickback [52] uses global error and local truncated feedback. CCL [68] exploits a dual network architecture, and the weights are different in its forward pass and feedback pass.

Fig. 7 (b.2) represents the case where no feedback weights are needed. DGL [61] synthesize the error for updating weights in the forward pass, by the auxiliary network. FGD [64] only has one forward pass and calculates the supervisory signal by directional derivative with automatic differentiation, without any usage of feedback weights. In addition, FF [20] (and PFF [117]), and PEPITA [65] only use the weights themselves in the second forward pass, while Sigprop [66] makes use of two forward passes. SoftHebb [67] does not need feedback, targets, or error signals in order to train a DNNs. GEVB [63] directly broadcasts the error to all the hidden layers, without any feedback parameters. Overall, the subclass where no feedback weights are used (Fig. 7 (b.2)) results in lower parameter count and, consequently, lower memory overhead compared to the subclass that employs feedback matrices (Fig. 7 (b.1)).

2) Frozen Activities: The frozen activities problem relates to the fact that activations are explicitly stored and exploited so as to update the network parameters. We regard the case involving storing all of the activations of the model as "completely frozen". In contrast, storing only partial activations or only the corresponding layer is labeled as "incompletely frozen". Using this grouping, we mark the aforementioned biologically plausible BP-free algorithms under the columns of "Incompletely Frozen" from Table I. Two ✘ in the two columns denote that an algorithm operates with completely frozen activities, whereas one ✔ in either column denotes that it is based on incompletely frozen activities. BP is completely frozen because BP has ✘ in both columns.

Fig. 8 (a) represents the situation where the activations from all the layers must be stored to update network parameters, i.e., completely frozen, where uSF [16], BP [4], DTP [51], Kickback [52], FA [53], DFA [54], EP [55], DDG [56], CHL

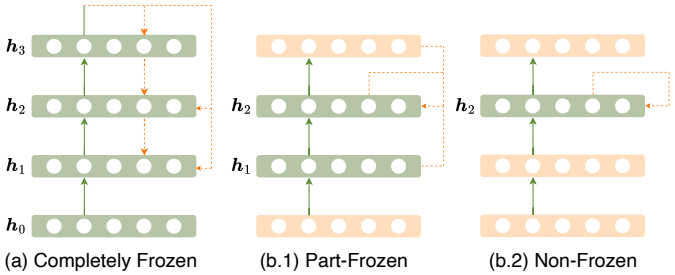


Fig. 8. A visualization of: completely frozen versus incompletely frozen activity values. The last two setups (b.1 and b.2) would be considered "incompletely frozen".

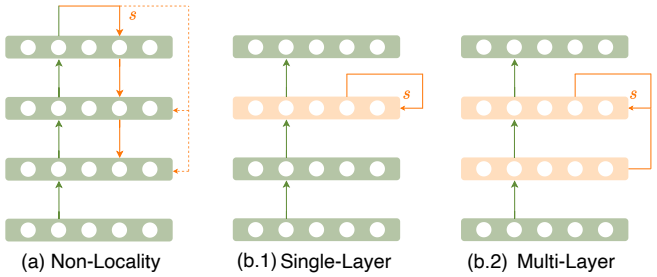


Fig. 9. A visualization of the learning setups for non-locality and locality. $\boldsymbol{S}$ denotes the supervisory signal. The last two setups (b.1 and b.2) would be considered "locality".

[69], [75], LRA [58], LRA-E [100], PC [94], WM [60], GEVB [63], PEPITA [65][2], CCL [68], and LFP [118] belong to this sub-class. On the contrary, in the context of incompletely frozen, Fig. 8 (b.1) represents that only part of all activations must be stored (such as the activations of the preceding or/and subsequent layers); note that recirculation [49], ADMM-NN [90], and the related work of [106] belong to this sub-class. In addition, Fig. 8 (b.2) represents that only the corresponding layer activations are needed to compute an update, without explicitly storing activations. DNI [15], LEL [57], Sigprop [66], DGL [61], DRTP [62], FF [20] (and PFF [117]), FGD [64], $\mathbf{F}^3$ [116], SoftHebb [67], Bio-FO [127], TinyFoA [21] and LL [59] are grouped under non-frozen activities. Overall, approaches based on "non-frozen" (Fig. 8 (b.2)) have less memory overhead as compared to schemes dependent on "part-frozen" (Fig. 8 (b.1)) and "completely frozen" (Fig. 8 (a)).

3) Non-Locality: The issue of non-locality describes the fact that supervisory/teaching signals are backpropagated, along a global backward pass or what is referred to as the global feedback pathway [12]. In contrast, when supervisory signals are generated locally (without a global pathway), a learning algorithm is regarded as local. As a result of this, we next group BP-free algorithms under the columns of "Locality" in Table I. Two ✘ in the two columns denote an algorithm as one that exhibits non-locality, whereas one ✔ in either column denotes that it exhibits locality. BP suffers from the issue of non-locality because BP has ✘ in both columns.

In the case of non-locality, the orange solid line in Fig. 9 (a) represents that the supervisory signals are backpropagated recursively layer by layer in the backward pass or global feedback pathway. This means that uSF [16], BP [4], DTP [51], FA [53], DDG [56], LRA [58], LRA-E [100], WM [60], and LFP [118] belong to this group. The orange dashed line in Fig. 9 (a) represents that the supervisory signals are backpropagated directly to the corresponding layers, where Kickback [52], DFA [54], GEVB [63], and $\mathbf{F}^3$ [116] belong to this group. These algorithms, based on recursive non-locality, require more memory and computational overhead than those that exhibit direct non-locality.

Furthermore, within the locality group, there are two sub-classes; namely, single-layer locality as shown in Fig. 9 (b.1) and multi-layer locality as shown in Fig. 9 (b.2). Single-layer locality means that the supervisory signals are generated only within a corresponding single layer whereas multi-layer locality means that the supervisory signals are generated by the preceding and/or subsequent layers, but not by the entire network. DNI [15], LEL [57], Sigprop [66], DGL [61], DRTP [62], FF [20] (and PFF [117]), PEPITA [65], FGD [64], SoftHebb [67], Bio-FO [127], TinyFoA [21] and LL [59] are grouped in single-layer locality. In contrast, recirculation [49], CHL [69], [75], ADMM-NN [90], EP [55], PC [94], CCL [68] and the work of [106] are grouped under multi-layer locality. Overall, it is important to observe that the subclass of single-layer locality (Fig. 9 (b.1)) requires less memory and computational overhead than multi-layer locality (Fig. 9 (b.2)).

Notably, single-layer locality shown in Fig. 9 (b.1) is not the same as non-frozen activities. For example, PEPITA [65] belongs to single-layer locality, but they need to store all the activations while waiting for a global supervisory signal. Meanwhile, frozen activities are not equivalent to non-locality. For instance, CHL [69] and CCL [68] need to store all of the activation values, but the supervisory/teaching signal is generated locally.

4) Update Locking: The locking issue in BP is denoted as the new input is locked until the previous input finishes its training. Consistent with the existing literature [15], we extend the definition of locking issues to four different groups: backwards locking, backwards unlocking, update unlocking, and forward unlocking, as presented in the columns of "Unlocking" from Table I. Three ✘ in the three columns denote that there is no unlocking, i.e., the system is backwards locked. The degree of unlocking is represented by the number of ✔ symbols across the three columns, where a greater number of ✔ symbols indicates a higher degree of unlocking. BP is backwards locked because BP has ✘ in all three columns.

- *1)* Backwards locking shown in Fig. 10 (a): the entire network can only be updated after all of the layers have been executed in the forward and backward passes. In other words, the new input must wait for the previous input to execute all the layers in the forward and backward passes. Typically, BP [4], uSF [16], DTP [51], FA [53], LRA [58], LRA-E [100], WM [60], LFP [118], and CCL [68] suffer from the backwards locking;
- *2)* Backwards unlocking shown in Fig. 10 (b): the net-

[2]To address the constraint of storing the activations in the standard forward pass, PEPITA-TL [30] is proposed, which exhibits a major degradation in accuracy compared to the original PEPITA [65]; for example, the test accuracy on MNIST is decreased to 92.78% in PEPITA-TL.

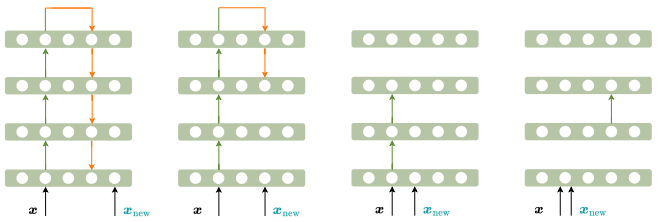


Fig. 10. A visualization of the learning setups for the cases of locking and unlocking (in terms of activity values and updates).

work can be updated before all the layers have been executed in the forward and backward passes. In other words, the new input does not need to wait for the previous input to execute all the layers in the forward and backward passes. Typically, DFA [54], PC [94], CHL [69], EP [55], recursive LRA [101], kickback [52], PEPITA [65] (PEPITA partially addresses the update locking issue because of the direct global error feedback after the execution of the first forward pass.), and GEVB [63] belong to the class of backwards unlocking;

- *3)* Update unlocking shown in Fig. 10 (c): the network can be updated before all the layers have been executed in the forward pass. In other words, the new input does not need to wait for the previous input to execute all the layers in the forward pass. Typically, recirculation [49], LEL [57], FF [20] (and PFF [117]), DRTP [62], DDG [56], FGD [64], ADMM-NN [90], Sigprop [66], SoftHebb [67], $\mathbf{F}^3$ [116], Bio-FO [127], TinyFoA [21], and LL [59] belong to the class of update unlocking.
- *4)* Forward unlocking shown in Fig. 10 (d): the network can be updated before the preceding layers to execute in the forward pass. In other words, forward unlocking allows asynchronous updates without regard to predecessor and dependent layers. Typically, DNI [15], DGL [61], and AsyncFGD [113] belong to the class of forward unlocking.

Notably, non-locality presented in Fig. 9 (a) is not equal to backwards locking or update locking. For example, Kickback [52], DFA [54], DDG [56], GEVB [63] belong to non-locality, but they are backwards unlocking. In addition, $\mathbf{F}^3$ [116] belongs to non-locality, but it is update unlocking. Overall, forward unlocking (Fig. 10 (d)) requires the least memory and energy overhead, followed by update unlocking (Fig. 10 (c)), then backwards unlocking (Fig. 10 (b)), and backwards locking (Fig. 10 (a)) requires the most.

5) Comprehensive Clusters: We next comprehensively cluster the aforementioned bio-plausible BP-free algorithms in Table I, the results of which are shown in Fig. 11. Typically, uSF [16], DTP [51], Kickback [52], FA [53], DFA [54], LRA-E [100], WM [60], and GEVB [63] only address the issue of weight transport. In addition, Recirculation [49], CHL [69], ADMM-NN [90], PC [94], EP [55], and a variant of FF [106] address the issue of non-locality. Moreover, PEPITA [65], CCL [68], and CHL with random feedback [75] address the issue of weight transport and non-locality. Besides, DNI [15] and

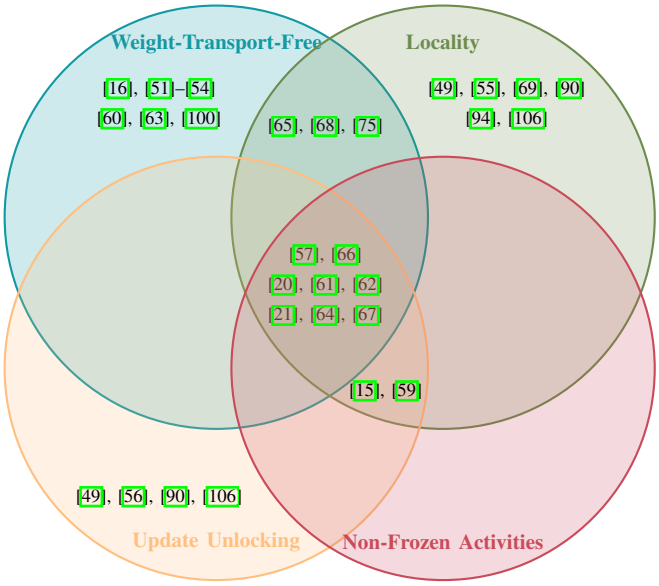


Fig. 11. The comprehensive clusters formed by our biologically-plausible algorithm taxonomy, where forward-only adaptation exhibits the potential to address the four central issues of backprop.

LL [59] address the issue of update locking, non-locality, and frozen activities. Furthermore, LEL [57], Sigprop [66], DGL [61], DRTP [62], FF [20], FGD [64], SoftHebb [67], TinyFoA [21] exhibit a progression from relying less on (separate) feedback pathways in general and more on what could be computed with forward propagation mechanisms alone. This is an important move to more directly leaning on forward propagation as the central driver of credit assignment itself, which has the potential to address the issue of weight transport, frozen activities, update locking, and non-locality. The results show the investigation of the extent to which these BP-free algorithms perform as compared to forward-only adaptation, *making forward-only adaptation stand out as a critical and prospective area of study in the world of BP-free algorithms.*

C. Scalability

We next move to further establish a taxonomy of biologically plausible BP-free algorithms in terms of scalability; dimensions along this property include the implemented architecture, widely-used dataset, and evaluated performance.

1) Implemented Architecture: We mainly consider locally-connected network (LC), fully-connected network (FC), convolutional neural network (CNN), recurrent neural network (RNN) [129], as well as ResNet [88], and transformer [130] structures/architectures. We note that the CNN is not biologically-plausible itself, since it requires forms of long-range communication, e.g., when distributing derivatives during learning. Moreover, weight sharing in the CNN does not exist in biological brains [20], [80].[3]

As shown in Fig. 12 (a), we have considered 42 papers in terms of implemented architecture. There are 37 out of

[3]In FC and LC, the parameters, partially the weights, consume substantially more memory than the intermediate activations. This finding contrasts with CNN-based networks trained with BP (such as MobileNetV2 [131] and ResNet-50 [88]), where activation memory is the primary bottleneck [132].

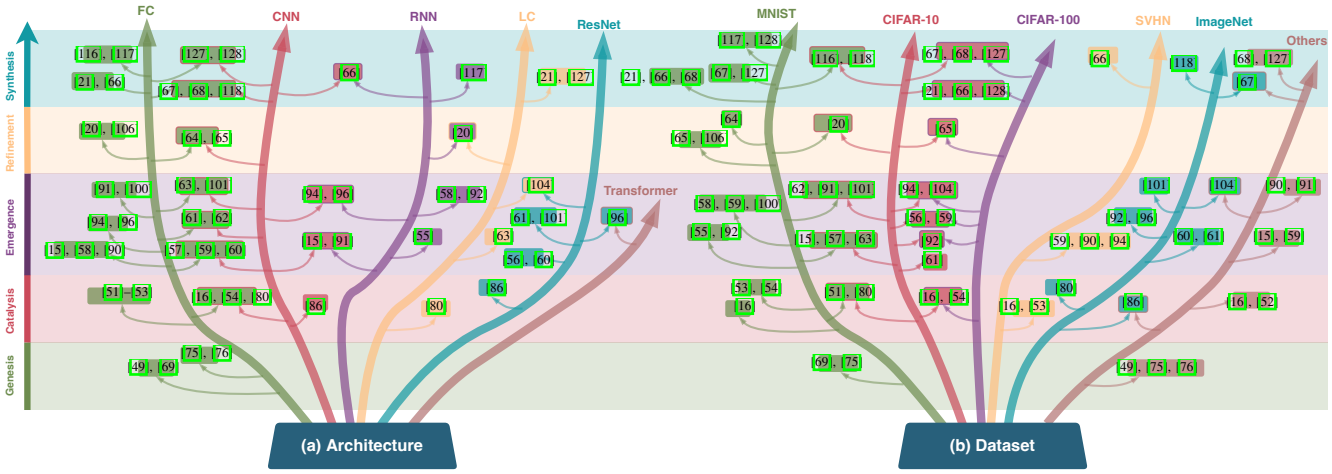


Fig. 12. Scalability for implemented architecture and widely-used dataset. The analysis results imply that more exploration is needed for biologically plausible BP-free algorithms implemented on advanced architectures such as LC, ResNet, Transformer, and applied on more advanced datasets such as ImageNet [87].

42 papers implemented for FC, while 23 out of 42 papers are implemented for CNN. In other words, 88% of these papers have FC implementations and 55% of these papers have CNN implementations. However, there are 6 out of 42 papers implemented for LC, which amounts to 14%. For the RNN, there are only 10 out of 42 papers (24%). For ResNet, there are only 7 out of 42 papers (16%), and for the transformer, there are only 1 out of 42 papers (2%). The analysis results imply that more exploration is needed for biologically plausible BP-free algorithms that are implemented for and applied to advanced architectures including the LC, RNN, ResNet, and transformer formats.

2) Widely-Used Dataset: Here, we primarily consider the MNIST [133], CIFAR-10 [81], CIFAR-100 [81], SVHN [134], and ImageNet [87] databases, as well as other datasets/tasks. These datasets are described as follows:

- **MNIST [133]**: The MNIST dataset is comprised of images of handwritten digits, presented as 28×28 grayscale pixel grids, across 10 different classes (each representing one of the 10 digits). The dataset contains 60,000 training images and 10,000 testing images.
- **CIFAR-10 [81]**: The CIFAR-10 dataset, designed for object recognition tasks, features 32×32 color images categorized into 10 classes. The dataset contains 50,000 training images and 10,000 testing images.
- **CIFAR-100 [81]**: In contrast to CIFAR-10, the CIFAR-100 dataset encompasses 100 classes (though the same image data constitutes it). For each class, there are 500 training images and 100 testing images.
- **SVHN [134]**: The SVHN dataset is a real-world image dataset, obtained from house numbers extracted from Google Street View images. It features 32×32 color images categorized into 10 classes. The dataset contains 73,257 training images and 26,032 testing images.
- **ImageNet [87]**: The ImageNet dataset is a large visual database designed for use in object recognition research. ImageNet-1K contains 1,281,167 training images, 50,000 validation images, and 100,000 test images. The full original dataset is referred to as ImageNet-21K, which contains 14,197,122 images divided into 21,841 classes.
- **Other Datasets/Tasks**: Two synthetic tasks – 'Copy' and 'Repeat Copy' tasks [135] and Penn Treebank dataset (language modeling) [136] are used with DNI [15]. Two robotics datasets, SARCOS [137], [138] and Barrett WAM [139] are evaluated in the work on kickback [52]. Additionally, 15 datasets ranging from machine learning tasks to face identification, such as Caltech 101 [140], are evaluated in the work of [16]. The Microsoft COCO dataset [141] is evaluated using uSF [86]. Moreover, the Higgs dataset [142] is evaluated in work including [90], [91]. The Fashion-MNIST dataset [143] and STL-10 dataset [144] are evaluated in CCL [68] and LL [59] (LRA [58] and LRA-E [100] were also studied on Fashion-MNIST). Meanwhile, the STL-10 dataset [144] and ImageNette dataset [145] are evaluated in the work related to SoftHebb [67]. The mini-ImageNet dataset [146] is evaluated in cascaded forward (CaFo) [128] and Bio-FO [127]. Finally, the TinyImageNet dataset [147] is evaluated in BioCNN [104].

As shown in Fig. 12 (b), we have considered the same 42 papers as discussed in Fig. 12 (a), in terms of datasets used for evaluation. There are 31 out of 42 papers performing their evaluations on MNIST, and 26 out of 42 papers perform their evaluations on CIFAR-10. As such, 74% of these papers are evaluated on MNIST, and over half (62%) of these papers are evaluated on CIFAR-10. Furthermore, only 14 out of 42 papers (33%) are evaluated on CIFAR-100, and only 6 out of 42 papers (14%) are evaluated on SVHN. Similarly, only 10 out of 42 papers (24%) are evaluated on ImageNet. The analysis results imply that more exploration is needed for biologically plausible BP-free algorithms that are applied on datasets with real-world use and noise, such as SVHN [134], as well as more advanced/complex datasets, such as ImageNet [87].

3) Evaluated Performance: Here, we mainly consider the classification performance, namely, accuracy, as the metric of evaluated performance comparison on MNIST [133], CIFAR-10 [81], CIFAR-100 [81], and ImageNet [87] datasets. Considering the different experiment setups across studies, we take

TABLE II
A HOLISTIC VIEW OF EVALUATION ACCURACY (%) OF BP-FREE ALGORITHMS FROM DIFFERENT ERAS.

Algorithms In Different Eras	MNIST [133]	CIFAR-10 [81]	CIFAR-100 [81]	ImageNet [87]
BP [80]	98.9	68.1	-	59.8
CHL [69]	85.0*	-	-	-
DTP [51]	98.1	50.7	-	-
FA [53]	97.9	-	-	-
DFA [54]	98.9	73.1	41.0	-
uSF [16]	99.4	81.5	49.2	-
DNI [15]	99.3	80.5	-	-
DDG [56]	-	93.4	71.4	-
LEL [57]	98.7	80.5	-	-
LRA-E [100]	98.2	-	-	-
WM [60]	-	-	-	76.6
DGL [61]	-	93.5	-	92.0
DRTP [62]	98.5	68.9	-	-
GEVB [63]	98.2	66.3	-	-
BioCNN [104]	-	87.4	58.7	83.1
PC [94]	-	58.0*	22.0*	-
Rec-LRA [101]	98.2	93.6	-	87.9
FF [20]	98.7	59.0	-	-
PEPITA [65]	98.3	56.3	27.5	-
LG-FG [111]	97.5	69.3	-	41.6
AsyncFGD [113]	95.5	47.3	-	-
Sigprop [66]	98.9	91.6	65.7	-
SoftHebb [67]	99.4	80.3	56.0	27.3
CwComp [125]	99.4	78.1	51.2	
CCL [68]	98.1	82.9	56.3	-
TinyFoA [21]	98.4	54.9	25.4	-

the highest accuracy that an algorithm achieves as the final accuracy for the MNIST, CIFAR-10, and CIFAR-100 datasets, as presented in Table II[4]. Furthermore, we normally take the Top-5 accuracy as the final accuracy [5] for the ImageNet dataset. Moreover, we group and order the aforementioned biologically-plausible BP-free algorithms from different eras based on Fig. 2 and the representative works selected in Section III, to create a holistic view of their evaluated performance.

Furthermore, by jointly considering the implemented architecture described in Section IV-C1 with the widely-used dataset described in Section IV-C2, we derive additional insights. By exploiting more advanced architectures, the work of [86] could extend the sign-symmetry algorithm [16] to large-scale datasets, e.g., ImageNet [87], approaching BP-trained performance. By adding lateral connectivity and Hebbian learning, a LC model can obtain similar performance compared to a CNN even on large networks and hard tasks [104], including CIFAR-10 [81], CIFAR-100 [81], TinyImageNet dataset [147], and ImageNet [87]. By incorporating linear bottleneck layers and unsupervised pre-training with adding deformations, FC achieves performance close to the range of that achieved by a BP-trained CNN [148], on CIFAR-10 [81], and the Higgs dataset [142]. Moreover, training an FC model with local connections bridges the gap of classification performance between FC and CNN [149], on the CIFAR-10 [81], CIFAR-100 [81], and SVHN [134] benchmarks.

[4]The accuracy for BP [4] is taken from the work of [80]. The symbol $*$ means the accuracy is manually read from figures of the corresponding paper.

[5]Note that WM [60] and SoftHebb [67] only report Top-1 accuracy.

V. ADVANTAGES AND APPLICATIONS

In this section, we present the advantages of biologically-plausible BP-free algorithms and their utilization in real-world applications. As we mentioned in Section I, in order to train GPT-3, OpenAI used a supercomputer with over 285,000 central processing units (CPU), 10,000 graphics processing units (GPUs), and 400 Gigabits per second of network connectivity for each GPU server. These LLM models consume considerable energy, much to the detriment of the environment [150]. Given this context, an important place that biologically-plausible BP-free algorithms, particularly those centered around forward-only adaptation, stand to play the most important role are in low-data learning scenarios and training in resource-constrained environments, such as in Internet of things (IoT) devices [34].

1) *Parallelism and Asynchronicity*: BP-free learning, especially forward-only adaptation, is ideal for parallel/asynchronous training of layers or modules [15], [61], [113]. Therefore, parallelizing the forward pass in forward-only algorithms offers utility, offering greater gain than parallelizing the backward pass (where the backward time is about twice the forward time [56], [64]).
2) *Resource Efficiency*: The efforts reviewed here [56], [62], [64], [66], [76], [90], [91], [111] show that the biologically plausible BP-free algorithms usually have a faster training speed as compared to BP. The fast convergence requires less energy overhead in principle. Moreover, these schemes [57], [59], [62], [114], [127] present the reduction of energy consumption of biologically plausible BP-free algorithms in the context forward-only adaptation. Several studies [21], [115], [151] demonstrate the reduction of memory consumption of biologically plausible forward-only BP-free algorithms. In addition, forward-only algorithms also reduce the energy consumption [109] at inference-time and are orthogonal to other inference-relief techniques [152].
3) *On-Device Training*: The training of deep learning models on embedded systems is still challenging mainly due to the low amount of memory, available energy, and computing power they offer [21], [153]; this significantly makes the use of traditional training algorithms such as BP [154] nearly impossible. BP-free algorithms, notably those in the class of forward-only adaptation, enable in-the-loop training on neuromorphic devices and edge devices [127], bringing about hardware efficiency [155]–[157]. Moreover, forward-only algorithms show promise for integration into LLMs fine-tuning with reduced memory consumption [27], [28] and supporting efficient tuning and deployment [158], [159].
4) *Reliability and Stability*: Biologically-plausible BP-free algorithms often perform much better than BP when the entire training dataset is not supplied [34]. Notably, in very low data settings, certain BP-free algorithms outperform gradient descent [160]; the work of [35] shows that the discovered biologically plausible plasticity rule improves the online training of a DNNs in the low data regime. Additionally, BP-free algorithms are found

to be robust against different kinds of noise [161], [162] and, furthermore, learn much quicker and converge to a stable accuracy in far fewer training epochs than BP [34]. Moreover, the work of [163] demonstrates how a BP-free model suffers significantly less forgetting (in the context of lifelong learning) by adapting synapses in a biologically plausible fashion when processing data streams.

5) *Real-World Applications*: Biologically plausible forward-only BP-free algorithms have been applied to dynamically adjust the encoding process in hyperdimensional (HD) computing [164], extracting features for skin cancer classification [165], wildlife monitoring [166], hyperspectral image classification [167], melanoma image classification [168], molecular property prediction [169], analyzing biomedical images [170] and vision transformer [171]. In addition, forward-only algorithms have also been applied to usage in IoT devices [21], [115], [127], [172]; the work of [173] describes the new generation of neuromorphic computing technologies. [174] applies forward-only BP-free methods to physical neural networks, while [175] exploits such a scheme to physics-informed neural networks. It is important to observe that forward-only algorithms demonstrate potential for integration into federated learning (FL) [176], test-time adaptation (TTA) [177], and diffusion models [178], possibly further serving as an alternative learning framework for SNNs [20], [179]–[181].

Overall, biologically-plausible BP-free algorithms, particularly those within the class of forward-only adaptation, mimic (in some ways) the learning processes of the human brain, giving us powerful clues to understanding of the underlying mechanisms of real biological neuronal systems/circuits. Concurrently, these BP-free algorithms offer the potential to solve the difficult problems that BP suffers from, e.g., such as getting stuck in local minima, experiencing vanishing/exploding gradients, overfitting, and slower convergence, in the context of non-convex optimization as well as engender resource efficiency. Last, but not least, biologically plausible forward-only BP-free schemes could offer a great asset to future developments in fundamental machine learning research [182].

VI. Limitations and Future Directions

In this section, we examine some of the inherent limitations of current (forward-only) biologically plausible BP-free procedures and, then, sketch out a pathway and potential future for forward-only adaptation.

1) *Required Differentiability*: The requirement for iteratively passing gradients means that the operations of DNN architectures must be differentiable, which is a limitation [91]. Among the aforementioned biologically plausible BP-free algorithms, for instance, these efforts [3], [51], [65], [69], [76], [90], [91], [118], [183] adjust synaptic weights without the need for iteratively passing gradients and, as a result, without requiring differentiability. In addition, local representation alignment [58], [100] and target-prop [51] schemes can be made to work for non-differentiable (and even stochastic) architectures. Thus, future research could investigate how to make biologically plausible BP-free algorithms, particularly forward-only processes, to operate for general architectures, when employing non-differentiable operations; potentially useful candidates include sub-gradient methods [184], zeroth-order gradient-based techniques [185], [186], evolutionary algorithms [187], gradient guessing that utilizes the special low-dimensional structure of neural networks [188], as well as other (as-of-yet under-explored) classical gradient-free techniques.
2) *Challenging Optimization*: Gradient-based optimization approach, especially stochastic gradient descent (SGD), can result in unstable training behavior (e.g., saddle points, poor conditioning, vanishing/exploding gradients, and local minima [91]) as well as high sensitivity to hyperparameters such as the learning rate [118] and weight initialization strategies [58] employed. Among the aforementioned biologically-plausible BP-free algorithms, for example, these efforts [76], [90], [91], [189] do not generally use the SGD at all. At the same time, there are algorithms [3], [65], [69], [118] that do not require iteratively passing gradients and yet still use an SGD-like scheme. Furthermore, investigating ways of computing an analytic solution, instead of an iterative one [190], is an interesting alternative research direction to explore in the context of stable optimization. In essence, there is a great demand for designing suitable optimization tools (bolstered by relevant classical mechanisms/ideas) for biologically-plausible (forward-only) BP-free algorithms.
3) *Dependent Supervision*: Most of the aforementioned biologically-plausible BP-free processes rely on supervisory signals except for certain efforts; for instance, these studies do not rely on them [20], [29], [49], [67], [67], [122], [181], [191]. Among these unsupervised BP-free algorithms, Hebbian theory [3] still stands as an important model/method. Nevertheless, this update rule has neither achieved high accuracy performance compared to BP nor made the training procedure simple [192], [193] thus far. Additionally, for FF [20], [117], the work of [29] provides a simple unsupervised way to generate negative samples with contrastive learning while the paper [194] exploits unsupervised learning models – such as autoencoders and generative adversarial networks (GANs) – but still requires labels to train the classification model. Overall, strongly generalizing biologically plausible BP-free (forward-only) schemes to unsupervised learning remains an important potential research gap.
4) *Limited Scalability*: Current biologically-plausible BP-free algorithms (including forward-only ones), have limited implementation in complicated or biologically-plausible architectures as well as limited evaluation on more complex datasets, as discussed in Section IV-C. Therefore, making BP-free algorithms well-suited for complicated architectures (such as graph neural networks (GNN) [195]) and more complex tasks/problems offers another interesting direction. The paper [80] poses several questions about whether new architectures and algorithms

are required in order to scale biologically-motivated deep learning schemes. Today, certain studies [85], [111], [196]–[202] make efforts towards scalability, with these efforts [83], [86], [99], [101], [111] are trying to scale BP-free algorithms to ImageNet [87].

VII. Conclusion

This survey presents a comprehensive overview and study to foster progress in BP-free learning in the context of emerging forward-only adaptation. We categorized BP-free algorithms across different historical stages according to their technical evolution, as well as analyzed the core principles underlying the representative works. Moreover, we established the taxonomy/organization of these types of algorithms in the context of supervisory signals used, biological-plausibility, and scalability for these BP-free algorithms. We further investigated the extent to which various BP-free algorithms perform compared as well as comprehensively examined the extent to which BP-free algorithms eliminate or approximate the backward pass (and how this resulted in varying levels of resource efficiency and biological plausibility), with the goal of replicating the efficient learning mechanisms of the human brain. Furthermore, we discussed the advantages, practical applications, the current limitations, and, finally, potential future directions of BP-free and forward-only learning.

References

[1] Henry Markram, Joachim Lübke, Michael Frotscher, and Bert Sakmann, "Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps," *Science*, vol. 275, no. 5297, pp. 213–215, 1997.

[2] Jeffrey C Magee and Christine Grienberger, "Synaptic plasticity forms and functions," *Annual review of neuroscience*, vol. 43, pp. 95–117, 2020.

[3] DO Hebb, *The organization of behavior. A neuropsychological theory*, John Wiley, 1949.

[4] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[5] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[6] Stefano Savazzi, Vittorio Rampa, Sanaz Kianoush, and Mehdi Bennis, "An energy and carbon footprint analysis of distributed and federated learning," *IEEE Transactions on Green Communications and Networking*, 2022.

[7] Jordan Aljbour, Tom Wilson, and Poorvi Patel, "Powering intelligence: Analyzing artificial intelligence and data center energy consumption," *EPRI White Paper no. 3002028905*, 2024.

[8] Jeremy Hsu, "Ibm's new brain [news]," *IEEE spectrum*, vol. 51, no. 10, pp. 17–19, 2014.

[9] Vijay Balasubramanian, "Brain power," *Proceedings of the National Academy of Sciences*, vol. 118, no. 32, pp. e2107022118, 2021.

[10] Advait Madhavan, "Brain-inspired computing can help us create faster, more energy-efficient devices — if we win the race," National Institute of Standards and Technology, 2024, URL [Accessed: 2024-10-29].

[11] Francis Crick, "The recent excitement about neural networks.," *Nature*, vol. 337, no. 6203, pp. 129–132, 1989.

[12] Alexander G Ororbia, "Brain-inspired machine intelligence: A survey of neurobiologically-plausible credit assignment," *arXiv preprint arXiv:2312.09257*, 2023.

[13] Kendra S Burbank and Gabriel Kreiman, "Depression-biased reverse plasticity rule is required for stable learning at top-down connections," *PLoS computational biology*, vol. 8, no. 3, pp. e1002393, 2012.

[14] James CR Whittington and Rafal Bogacz, "Theories of error back-propagation in the brain," *Trends in cognitive sciences*, vol. 23, no. 3, pp. 235–250, 2019.

[15] Max Jaderberg, Wojciech Marian Czarnecki, Simon Osindero, Oriol Vinyals, Alex Graves, David Silver, and Koray Kavukcuoglu, "Decoupled neural interfaces using synthetic gradients," in *International conference on machine learning*. PMLR, 2017, pp. 1627–1635.

[16] Qianli Liao, Joel Leibo, and Tomaso Poggio, "How important is weight symmetry in backpropagation?," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016, vol. 30.

[17] Timothy P Lillicrap, Adam Santoro, Luke Marris, Colin J Akerman, and Geoffrey Hinton, "Backpropagation and the brain," *Nature Reviews Neuroscience*, vol. 21, no. 6, pp. 335–346, 2020.

[18] Blake A Richards and Timothy P Lillicrap, "Dendritic solutions to the credit assignment problem," *Current opinion in neurobiology*, vol. 54, pp. 28–36, 2019.

[19] Jordan Guerguiev, Timothy P Lillicrap, and Blake A Richards, "Towards deep learning with segregated dendrites," *Elife*, vol. 6, pp. e22901, 2017.

[20] Geoffrey Hinton, "The forward-forward algorithm: Some preliminary investigations," *arXiv preprint arXiv:2212.13345*, 2022.

[21] Baichuan Huang and Amir Aminifar, "Tinyfoa: Memory efficient forward-only algorithm for on-device learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

[22] Jiong Zhang, Hsiang-Fu Yu, and Inderjit S Dhillon, "Autoassist: A framework to accelerate training of deep neural networks," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. 2019, vol. 32, Curran Associates, Inc.

[23] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al., "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[24] Rohan Anil et al., "Palm 2 technical report," *arXiv preprint arXiv:2305.10403*, 2023.

[25] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al., "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[26] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al., "Deepseek-v3 technical report," *arXiv preprint arXiv:2412.19437*, 2024.

[27] Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D Lee, Danqi Chen, and Sanjeev Arora, "Fine-tuning language models with just forward passes," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[28] Yihua Zhang, Pingzhi Li, Junyuan Hong, Jiaxiang Li, Yimeng Zhang, Wenqing Zheng, Pin-Yu Chen, Jason D. Lee, Wotao Yin, Mingyi Hong, Zhangyang Wang, Sijia Liu, and Tianlong Chen, "Revisiting zeroth-order optimization for memory-efficient LLM fine-tuning: A benchmark," in *Forty-first International Conference on Machine Learning*, 2024.

[29] Xing Chen, Dongshu Liu, Jérémie Laydevant, and Julie Grollier, "Self-contrastive forward-forward algorithm," *Nature Communications*, vol. 16, 2025, Published 1 July 2025.

[30] Ravi Francesco Srinivasan, Francesca Mignacco, Martino Sorbaro, Maria Refinetti, Avi Cooper, Gabriel Kreiman, and Giorgia Dellaferrera, "Forward learning with top-down feedback: Empirical and analytical characterization," in *The Twelfth International Conference on Learning Representations*, 2024.

[31] Thomas Miconi, "Biologically plausible learning in recurrent neural networks reproduces neural dynamics observed during cognitive tasks," *Elife*, vol. 6, pp. e20899, 2017.

[32] Blake A Richards, Timothy P Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy de Berker, Surya Ganguli, et al., "A deep learning framework for neuroscience," *Nature neuroscience*, vol. 22, no. 11, pp. 1761–1770, 2019.

[33] Mufeng Tang, Yibo Yang, and Yali Amit, "Biologically plausible training mechanisms for self-supervised learning in deep networks," *Frontiers in Computational Neuroscience*, vol. 16, pp. 789253, 2022.

[34] Manas Gupta, Sarthak Ketanbhai Modi, Hang Zhang, Joon Hei Lee, and Joo Hwee Lim, "Is bio-inspired learning better than backprop? benchmarking bio learning vs. backprop," *arXiv preprint arXiv:2212.04614*, 2022.

[35] Navid Shervani-Tabar and Robert Rosenbaum, "Meta-learning biologically plausible plasticity rules with random feedback pathways," *Nature Communications*, vol. 14, no. 1, pp. 1805, 2023.

[36] Johannes Feldmann, Nathan Youngblood, C David Wright, Harish Bhaskaran, and Wolfram HP Pernice, "All-optical spiking neurosynaptic networks with self-learning capabilities," *Nature*, vol. 569, no. 7755, pp. 208–214, 2019.

[37] Junnan Li, Caiming Xiong, and Steven Hoi, "Mopro: Webly supervised learning with momentum prototypes," in *International Conference on Learning Representations*, 2020.

[38] Harit Vishwakarma, Heguang Lin, Frederic Sala, and Ramya Korlakai Vinayak, "Promises and pitfalls of threshold-based auto-labeling," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[39] Julian Büchel, Dmitrii Zendrikov, Sergio Solinas, Giacomo Indiveri, and Dylan R Muir, "Supervised training of spiking neural networks for robust deployment on mixed-signal neuromorphic processors," *Scientific reports*, vol. 11, no. 1, pp. 23376, 2021.

[40] Gehua Ma, Rui Yan, and Huajin Tang, "Exploiting noise as a resource for computation and learning in spiking neural networks," *Patterns*, vol. 4, no. 10, 2023.

[41] Alexander Ororbia, Ankur Mali, Adam Kohan, Beren Millidge, and Tommaso Salvatori, "A review of neuroscience-inspired machine learning," *arXiv preprint arXiv:2403.18929*, 2024.

[42] Samuel Schmidgall, Rojin Ziaei, Jascha Achterberg, Louis Kirsch, S Hajiseyedrazi, and Jason Eshraghian, "Brain-inspired learning in artificial neural networks: a review," *APL Machine Learning*, vol. 2, no. 2, 2024.

[43] Fahad Sarfraz, Elahe Arani, and Bahram Zonooz, "A study of biologically plausible neural network: The role and interactions of brain-inspired mechanisms in continual learning," *Transactions on Machine Learning Research*, 2023.

[44] Gabriele Lagani, Fabrizio Falchi, Claudio Gennaro, and Giuseppe Amato, "Synaptic plasticity models and bio-inspired unsupervised deep learning: A survey," *arXiv preprint arXiv:2307.16236*, 2023.

[45] Danilo Pietro Pau, Prem Kumar Ambrose, and Fabrizio Maria Aymone, "A quantitative review of automated neural search and on-device learning for tiny devices," *Chips*, vol. 2, no. 2, pp. 130–141, 2023.

[46] Licheng Jiao, Zhongjian Huang, Xiaoqiang Lu, Xu Liu, Yuting Yang, Jiaxuan Zhao, Jinyue Zhang, Biao Hou, Shuyuan Yang, Fang Liu, Wenping Ma, Lingling Li, Xiangrong Zhang, Puhua Chen, Zhixi Feng, Xu Tang, Yuwei Guo, Dou Quan, Shuang Wang, Weibin Li, Jing Bai, Yangyang Li, Ronghua Shang, and Jie Feng, "Brain-inspired remote sensing foundation models and open problems: A comprehensive survey," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 10084–10120, 2023.

[47] Licheng Jiao, Mengru Ma, Pei He, Xueli Geng, Xu Liu, Fang Liu, Wenping Ma, Shuyuan Yang, Biao Hou, and Xu Tang, "Brain-inspired learning, perception, and cognition: A comprehensive review," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2024.

[48] Xiao-Long Zou, Tie-Jun Huang, and Si Wu, "Towards a new paradigm for brain-inspired computer vision," *Machine Intelligence Research*, vol. 19, no. 5, pp. 412–424, 2022.

[49] Geoffrey E Hinton and James McClelland, "Learning representations by recirculation," in *Neural information processing systems*, 1987.

[50] Conrad C Galland and Geoffrey E Hinton, "Deterministic boltzmann learning in networks with asymmetric connectivity," in *Connectionist models*, pp. 3–9. Elsevier, 1991.

[51] Dong-Hyun Lee, Saizheng Zhang, Asja Fischer, and Yoshua Bengio, "Difference target propagation," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2015, Porto, Portugal, September 7-11, 2015, Proceedings, Part I 15*. Springer, 2015, pp. 498–515.

[52] David Balduzzi, Hastagiri Vanchinathan, and Joachim Buhmann, "Kickback cuts backprop's red-tape: Biologically plausible credit assignment in neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2015, vol. 29.

[53] Timothy P Lillicrap, Daniel Cownden, Douglas B Tweed, and Colin J Akerman, "Random synaptic feedback weights support error backpropagation for deep learning," *Nature communications*, vol. 7, no. 1, pp. 13276, 2016.

[54] Arild Nøkland, "Direct feedback alignment provides learning in deep neural networks," *Advances in neural information processing systems*, vol. 29, 2016.

[55] Benjamin Scellier and Yoshua Bengio, "Equilibrium propagation: Bridging the gap between energy-based models and backpropagation," *Frontiers in computational neuroscience*, vol. 11, pp. 24, 2017.

[56] Zhouyuan Huo, Bin Gu, Heng Huang, et al., "Decoupled parallel backpropagation with convergence guarantee," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2098–2106.

[57] Hesham Mostafa, Vishwajith Ramesh, and Gert Cauwenberghs, "Deep supervised learning using local errors," *Frontiers in neuroscience*, vol. 12, pp. 608, 2018.

[58] Alexander G Ororbia, Ankur Mali, Daniel Kifer, and C Lee Giles, "Conducting credit assignment by aligning local representations," *arXiv preprint arXiv:1803.01834*, 2018.

[59] Arild Nøkland and Lars Hiller Eidnes, "Training neural networks with local error signals," in *International conference on machine learning*. PMLR, 2019, pp. 4839–4850.

[60] Mohamed Akrout, Collin Wilson, Peter Humphreys, Timothy Lillicrap, and Douglas B Tweed, "Deep learning without weight transport," *Advances in neural information processing systems*, vol. 32, 2019.

[61] Eugene Belilovsky, Michael Eickenberg, and Edouard Oyallon, "Decoupled greedy learning of cnns," in *International Conference on Machine Learning*. PMLR, 2020, pp. 736–745.

[62] Charlotte Frenkel, Martin Lefebvre, and David Bol, "Learning without feedback: Fixed random learning signals allow for feedforward training of deep neural networks," *Frontiers in neuroscience*, vol. 15, pp. 629892, 2021.

[63] David Clark, LF Abbott, and SueYeon Chung, "Credit assignment through broadcasting a global error vector," *Advances in Neural Information Processing Systems*, vol. 34, pp. 10053–10066, 2021.

[64] Atılım Güneş Baydin, Barak A Pearlmutter, Don Syme, Frank Wood, and Philip Torr, "Gradients without backpropagation," *arXiv preprint arXiv:2202.08587*, 2022.

[65] Giorgia Dellaferrera et al., "Error-driven input modulation: solving the credit assignment problem without a backward pass," in *International Conference on Machine Learning*. PMLR, 2022, pp. 4937–4955.

[66] Adam Kohan, Edward A Rietman, and Hava T Siegelmann, "Signal propagation: The framework for learning and inference in a forward pass," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[67] Adrien Journé, Hector Garcia Rodriguez, Qinghai Guo, and Timoleon Moraitis, "Hebbian deep learning without feedback," in *The Eleventh International Conference on Learning Representations*, 2023.

[68] Chia Hsiang Kao and Bharath Hariharan, "Counter-current learning: A biologically plausible dual network approach for deep learning," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[69] Xiaohui Xie and H Sebastian Seung, "Equivalence of backpropagation and contrastive hebbian learning in a layered network," *Neural computation*, vol. 15, no. 2, pp. 441–454, 2003.

[70] Seppo Linnainmaa, "The representation of the cumulative rounding error of an algorithm as a taylor expansion of the local rounding errors," M.S. thesis, Master's Thesis (in Finnish), Univ. Helsinki, 1970.

[71] Paul Werbos, "Beyond regression: New tools for prediction and analysis in the behavioral sciences," *PhD thesis, Committee on Applied Mathematics, Harvard University, Cambridge, MA*, 1974.

[72] Randall C O'Reilly, "Biologically plausible error-driven learning using local activation differences: The generalized recirculation algorithm," *Neural computation*, vol. 8, no. 5, pp. 895–938, 1996.

[73] Javier R Movellan, "Contrastive hebbian learning in the continuous hopfield model," in *Connectionist models*, pp. 10–17. Elsevier, 1991.

[74] David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowski, "A learning algorithm for boltzmann machines," *Cognitive science*, vol. 9, no. 1, pp. 147–169, 1985.

[75] Georgios Detorakis, Travis Bartley, and Emre Neftci, "Contrastive hebbian learning with random feedback weights," *Neural Networks*, vol. 114, pp. 1–14, 2019.

[76] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, 2006.

[77] Yann Le Cun, "Learning process in an asymmetric threshold network," in *Disordered systems and biological organization*, pp. 233–240. Springer, 1986.

[78] Alexander Meulemans, Francesco Carzaniga, Johan Suykens, João Sacramento, and Benjamin F Grewe, "A theoretical framework for target propagation," *Advances in Neural Information Processing Systems*, vol. 33, pp. 20024–20036, 2020.

[79] Tatsukichi Shibuya, Nakamasa Inoue, Rei Kawakami, and Ikuro Sato, "Fixed-weight difference target propagation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023, vol. 37, pp. 9811–9819.

[80] Sergey Bartunov, Adam Santoro, Blake Richards, Luke Marris, Geoffrey E Hinton, and Timothy Lillicrap, "Assessing the scalability of biologically-motivated deep learning algorithms and architectures," *Advances in neural information processing systems*, vol. 31, 2018.

[81] A. Krizhevsky, "Learning multiple layers of features from tiny images," Tech. Rep., Department of Computer Science, University of Toronto, Toronto, ON, Canada, 2009.

[82] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al., "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, pp. 211–252, 2015.

[83] Maxence M Ernoult, Fabrice Normandin, Abhinav Moudgil, Sean Spinney, Eugene Belilovsky, Irina Rish, Blake Richards, and Yoshua Bengio, "Towards scaling difference target propagation by learning backprop targets," in *International Conference on Machine Learning*. PMLR, 2022, pp. 5968–5987.

[84] Maria Refinetti, Stéphane d'Ascoli, Ruben Ohana, and Sebastian Goldt, "Align, then memorise: the dynamics of learning with feedback alignment," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8925–8935.

[85] Julien Launay, Iacopo Poli, François Boniface, and Florent Krzakala, "Direct feedback alignment scales to modern deep learning tasks and architectures," *Advances in neural information processing systems*, vol. 33, pp. 9346–9360, 2020.

[86] Will Xiao, Honglin Chen, Qianli Liao, and Tomaso Poggio, "Biologically-plausible learning algorithms can scale to large datasets," in *International Conference on Learning Representations*, 2019.

[87] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[88] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[89] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[90] Gavin Taylor, Ryan Burmeister, Zheng Xu, Bharat Singh, Ankit Patel, and Tom Goldstein, "Training neural networks without gradients: A scalable admm approach," in *International conference on machine learning*. PMLR, 2016, pp. 2722–2731.

[91] Anna Choromanska, E Tandon, Sadhana Kumaravel, Ronny Luss, Irina Rish, Brian Kingsbury, Ravi Tejwani, and Djallel Bouneffouf, "Beyond back-prop: Alternating minimization with co-activation memory," *stat*, vol. 1050, pp. 24, 2018.

[92] Axel Laborieux and Friedemann Zenke, "Improving equilibrium propagation without weight symmetry through jacobian homeostasis," in *The Twelfth International Conference on Learning Representations*, 2024.

[93] James CR Whittington and Rafal Bogacz, "An approximation of the error backpropagation algorithm in a predictive coding network with local hebbian synaptic plasticity," *Neural computation*, vol. 29, no. 5, pp. 1229–1262, 2017.

[94] Beren Millidge, Alexander Tschantz, and Christopher L Buckley, "Predictive coding approximates backprop along arbitrary computation graphs," *Neural Computation*, vol. 34, no. 6, pp. 1329–1368, 2022.

[95] Rajesh PN Rao and Dana H Ballard, "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects," *Nature neuroscience*, vol. 2, no. 1, pp. 79–87, 1999.

[96] Tommaso Salvatori, Yuhang Song, Zhenghua Xu, Thomas Lukasiewicz, and Rafal Bogacz, "Reverse differentiation via predictive coding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, vol. 36, pp. 8150–8158.

[97] Alexander Ororbia and Daniel Kifer, "The neural coding framework for learning generative models," *Nature communications*, vol. 13, no. 1, pp. 2064, 2022.

[98] Wojciech Marian Czarnecki, Grzegorz Świrszcz, Max Jaderberg, Simon Osindero, Oriol Vinyals, and Koray Kavukcuoglu, "Understanding synthetic gradients and decoupled neural interfaces," in *International Conference on Machine Learning*. PMLR, 2017, pp. 904–912.

[99] Eugene Belilovsky, Michael Eickenberg, and Edouard Oyallon, "Greedy layerwise learning can scale to imagenet," in *International conference on machine learning*. PMLR, 2019, pp. 583–593.

[100] Alexander G Ororbia and Ankur Mali, "Biologically motivated algorithms for propagating local target representations," in *Proceedings of the aaai conference on artificial intelligence*, 2019, vol. 33, pp. 4651–4658.

[101] Alexander G Ororbia, Ankur Mali, Daniel Kifer, and C Lee Giles, "Backpropagation-free deep learning with recursive local representation alignment," in *Proceedings of the AAAI conference on artificial intelligence*, 2023, vol. 37, pp. 9327–9335.

[102] Sunghyeon Woo, Jeongwoo Park, Jiwoo Hong, and Dongsuk Jeon, "Activation sharing with asymmetric paths solves weight transport problem without bidirectional connection," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29697–29709, 2021.

[103] Alexander Meulemans, Matilde Tristany Farinha, Javier García Ordóñez, Pau Vilimelis Aceituno, João Sacramento, and Benjamin F Grewe, "Credit assignment in neural networks through deep feedback control," *Advances in Neural Information Processing Systems*, vol. 34, pp. 4674–4687, 2021.

[104] Roman Pogodin, Yash Mehta, Timothy Lillicrap, and Peter E Latham, "Towards biologically plausible convolutional networks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 13924–13936, 2021.

[105] Guy Lorberbom, Itai Gat, Yossi Adi, Alexander Schwing, and Tamir Hazan, "Layer collaboration in the forward-forward algorithm," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024, vol. 38, pp. 14141–14148.

[106] Fabio Giampaolo, Stefano Izzo, Edoardo Prezioso, and Francesco Piccialli, "Investigating random variations of the forward-forward algorithm for training neural networks," in *2023 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2023, pp. 1–7.

[107] Niccolo Tosato, Lorenzo Basile, Emanuele Ballarin, Giuseppe De Alteriis, Alberto Cazzaniga, and Alessio ansuini, "Emergent representations in networks trained with the forward-forward algorithm," in *High-dimensional Learning Dynamics 2024: The Emergence of Structure and Reasoning*, 2024.

[108] Riccardo Scodellaro, Ajinkya Kulkarni, Frauke Alves, and Matthias Schroeter, "Training convolutional neural networks with the forward-forward algorithm," *Bulletin of the American Physical Society*, 2024.

[109] Amin Aminifar, Baichuan Huang, Azra Abtahi Fahliani, and Amir Aminifar, "Lightff: Lightweight inference for forward-forward algorithm," in *27th European Conference on Artificial Intelligence, ECAI-2024*. IOS Press, 2024, vol. 392, pp. 1728–1735.

[110] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind, "Automatic differentiation in machine learning: a survey," *Journal of machine learning research*, vol. 18, no. 153, pp. 1–43, 2018.

[111] Mengye Ren, Simon Kornblith, Renjie Liao, and Geoffrey Hinton, "Scaling forward gradient with local losses," in *The Eleventh International Conference on Learning Representations*, 2023.

[112] Florian Bacho and Dominique Chu, "Low-variance forward gradients using direct feedback alignment and momentum," *Neural Networks*, vol. 169, pp. 572–583, 2024.

[113] Xiaohan Zhao, Hualin Zhang, Zhouyuan Huo, and Bin Gu, "Accelerated on-device forward neural network training with module-wise descending asynchronism," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[114] Danilo Pietro Pau and Fabrizio Maria Aymone, "Suitability of forward-forward and pepita learning to mlcommons-tiny benchmarks," in *2023 IEEE International Conference on Omni-layer Intelligent Systems (COINS)*. IEEE, 2023, pp. 1–6.

[115] Baichuan Huang and Amir Aminifar, "Binary forward-only algorithms," *IEEE Design & Test*, pp. 1–1, 2025.

[116] Katharina Flügel, Daniel Coquelin, Marie Weiel, Charlotte Debus, Achim Streit, and Markus Götz, "Feed-forward optimization with delayed feedback for neural networks," *arXiv preprint arXiv:2304.13372*, 2023.

[117] Alexander Ororbia and Ankur A Mali, "The predictive forward-forward algorithm," in *Proceedings of the Annual Meeting of the Cognitive Science Society*, 2023, vol. 45.

[118] Leander Weber, Jim Berend, Alexander Binder, Thomas Wiegand, Wojciech Samek, and Sebastian Lapuschkin, "Layer-wise feedback propagation," *arXiv preprint arXiv:2308.12053*, 2023.

[119] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PloS one*, vol. 10, no. 7, pp. e0130140, 2015.

[120] Danilo Pau, Prem Kumar Ambrose, Andrea Pisani, and Fabrizio M Aymone, "Tinyrce: Forward learning under tiny constraints," in *2023*

IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE). IEEE, 2023, pp. 295–300.

[121] Danilo Pietro Pau, Andrea Pisani, Fabrizio M Aymone, and Gianluigi Ferrari, "Tinyrce: Multi purpose forward learning for resource restricted devices," *IEEE Sensors Letters*, 2023.

[122] Timoleon Moraitis, Dmitry Toichkin, Adrien Journé, Yansong Chua, and Qinghai Guo, "Softhebb: Bayesian inference in unsupervised hebbian soft winner-take-all networks," *Neuromorphic Computing and Engineering*, vol. 2, no. 4, pp. 044017, 2022.

[123] Bariscan Bozkurt, Cengiz Pehlevan, and Alper Erdogan, "Correlative information maximization: A biologically plausible approach to supervised deep neural networks without weight symmetry," *Advances in Neural Information Processing Systems*, vol. 36, pp. 34928–34941, 2023.

[124] João Sacramento, Rui Ponte Costa, Yoshua Bengio, and Walter Senn, "Dendritic cortical microcircuits approximate the backpropagation algorithm," *Advances in neural information processing systems*, vol. 31, 2018.

[125] Andreas Papachristodoulou, Christos Kyrkou, Stelios Timotheou, and Theocharis Theocharides, "Convolutional channel-wise competitive learning for the forward-forward algorithm," in *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*. 2024, AAAI'24/IAAI'24/EAAI'24, AAAI Press.

[126] Marco P. E. Apolinario, Arani Roy, and Kaushik Roy, "Lls: Local learning rule for deep neural networks inspired by neural activity synchronization," in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, February 2025, pp. 7796–7805.

[127] Baichuan Huang and Amir Aminifar, "Efficient on-device machine learning with a biologically-plausible forward-only algorithm," *Proceedings of Machine Learning and Systems*, 2025.

[128] Gongpei Zhao, Tao Wang, Yi Jin, Congyan Lang, Yidong Li, and Haibin Ling, "The cascaded forward algorithm for neural network training," *Pattern Recognition*, vol. 161, pp. 111292, 2025.

[129] Larry R Medsker and LC Jain, "Recurrent neural networks," *Design and Applications*, vol. 5, no. 64-67, pp. 2, 2001.

[130] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[131] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[132] Han Cai, Chuang Gan, Ligeng Zhu, and Song Han, "Tinytl: Reduce memory, not parameters for efficient on-device learning," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, Eds. 2020, vol. 33, pp. 11285–11297, Curran Associates, Inc.

[133] Yann LeCun, "The mnist database of handwritten digits," *http://yann. lecun. com/exdb/mnist/*, 1998.

[134] Netzer Yuval, "Reading digits in natural images with unsupervised feature learning," in *Proceedings of the NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[135] Alex Graves, "Neural turing machines," *arXiv preprint arXiv:1410.5401*, 2014.

[136] Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini, "Building a large annotated corpus of english: the penn treebank," *Comput. Linguist.*, vol. 19, no. 2, pp. 313–330, June 1993.

[137] Sethu Vijayakumar and Stefan Schaal, "Locally weighted projection regression: An o (n) algorithm for incremental real time learning in high dimensional space," in *Proceedings of the seventeenth international conference on machine learning (ICML 2000)*. Morgan Kaufmann Burlington, MA, USA, 2000, vol. 1, pp. 288–293.

[138] "The sarcos data," https://gaussianprocess.org/gpml/data/, Accessed: 2024-02-22.

[139] Duy Nguyen-Tuong, Matthias Seeger, and Jan Peters, "Model learning with local gaussian process regression," *Advanced Robotics*, vol. 23, no. 15, pp. 2015–2034, 2009.

[140] Li Fei-Fei, Rob Fergus, and Pietro Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *2004 conference on computer vision and pattern recognition workshop*. IEEE, 2004, pp. 178–178.

[141] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 2014, pp. 740–755.

[142] Pierre Baldi, Peter Sadowski, and Daniel Whiteson, "Searching for exotic particles in high-energy physics with deep learning," *Nature communications*, vol. 5, no. 1, pp. 4308, 2014.

[143] Han Xiao, Kashif Rasul, and Roland Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[144] Adam Coates, Andrew Ng, and Honglak Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 215–223.

[145] Jeremy Howard and Sylvain Gugger, "Fastai: a layered api for deep learning," *Information*, vol. 11, no. 2, pp. 108, 2020.

[146] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al., "Matching networks for one shot learning," *Advances in neural information processing systems*, vol. 29, 2016.

[147] Yann Le and Xuan Yang, "Tiny imagenet visual recognition challenge," *CS 231N*, vol. 7, no. 7, pp. 3, 2015.

[148] Zhouhan Lin, Roland Memisevic, and Kishore Konda, "How far can we go without convolution: Improving fully-connected networks," in *International Conference on Learning Representations (ICLR) Workshop*, 2016.

[149] Behnam Neyshabur, "Towards learning convolutions from scratch," *Advances in Neural Information Processing Systems*, vol. 33, pp. 8078–8088, 2020.

[150] Sophia Chen, "How much energy will ai really consume? the good, the bad and the unknown," *Nature*, vol. 639, no. 8053, pp. 22–24, 2025.

[151] Aditya Somasundaram, Pushkal Mishra, and Ayon Borthakur, "Learning using a single forward pass," *Transactions on Machine Learning Research (TMLR)*, 2025.

[152] Manuele Rusci, Alessandro Capotondi, and Luca Benini, "Memory-driven mixed low precision quantization for enabling deep network inference on microcontrollers," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 326–335, 2020.

[153] Baichuan Huang, Azra Abtahi, and Amir Aminifar, "Energy-aware integrated neural architecture search and partitioning for distributed internet of things (iot)," *IEEE Transactions on Circuits and Systems for Artificial Intelligence*, vol. 1, no. 2, pp. 257–271, 2024.

[154] Fabrizio De Vita, Rawan MA Nawaiseh, Dario Bruneo, Valeria Tomaselli, Marco Lattuada, and Mirko Falchetto, "μ-ff: On-device forward-forward training algorithm for microcontrollers," in *2023 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2023, pp. 49–56.

[155] Charlotte Frenkel, Jean-Didier Legat, and David Bol, "A 28-nm convolutional neuromorphic processor enabling online learning with spike-based retinas," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2020, pp. 1–5.

[156] Yequan Zhao, Hai Li, Ian Young, and Zheng Zhang, "Poor man's training on mcus: A memory-efficient quantized back-propagation-free approach," *ACM Transactions on Design Automation of Electronic Systems*, 2024.

[157] Chen Feng, Jay Zhuo, Parker Zhang, Ramchalam Kinattinkara Ramakrishnan, Zhaocong Yuan, and Andrew Zou Li, "Stepping forward on the last mile," *Advances in Neural Information Processing Systems*, vol. 37, pp. 94851–94870, 2024.

[158] Tianxiang Sun, Yunfan Shao, Hong Qian, Xuanjing Huang, and Xipeng Qiu, "Black-box tuning for language-model-as-a-service," in *International Conference on Machine Learning*. PMLR, 2022, pp. 20841–20855.

[159] Tianxiang Sun, Zhengfu He, Hong Qian, Yunhua Zhou, Xuan-Jing Huang, and Xipeng Qiu, "Bbtv2: Towards a gradient-free future with large language models," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022, pp. 3916–3930.

[160] Akhilan Boopathy and Ila Fiete, "How to train your wide neural network without backprop: An input-weight alignment perspective," in *International Conference on Machine Learning*. PMLR, 2022, pp. 2178–2205.

[161] Matilde Tristany Farinha, Thomas Ortner, Giorgia Dellaferrera, Benjamin Grewe, and Angeliki Pantazi, "Efficient biologically plausible adversarial training," *arXiv preprint arXiv:2309.17348*, 2023.

[162] Matilde Tristany Farinha, Thomas Ortner, Giorgia Dellaferrera, Benjamin Grewe, and Angeliki Pantazi, "Intrinsic biologically plausible adversarial robustness," *arXiv preprint arXiv:2309.17348*, 2023.

[163] Alexander G. Ororbia, Ankur Mali, C. Lee Giles, and Daniel Kifer, "Lifelong neural predictive coding: learning cumulatively online without forgetting," in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2022, NIPS '22, Curran Associates Inc.

[164] Hyunsei Lee, Jiseung Kim, Seohyun Kim, Hyukjun Kwon, Mohsen Imani, Ilhong Suh, and Yeseong Kim, "Efficient forward-only training for brain-inspired hyperdimensional computing," in *2024 IEEE 42nd International Conference on Computer Design (ICCD)*. IEEE, 2024, pp. 707–714.

[165] Abel Reyes-Angulo and Paheding Sidike, "The forward-forward algorithm as a feature extractor for skin lesion classification: A preliminary study.," in *LatinX in AI Workshop at ICML 2023 (Regular Deadline)*, 2023.

[166] Hamad AlHammadi, Meriam Mkadmi, Rohan Mitra, and Imran Zualkernan, "Exploring the forward-forward algorithm to train neural networks for camera trap images on the edge," in *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)*. IEEE, 2024, pp. 741–746.

[167] Abel A Reyes-Angulo and Sidike Paheding, "Forward-forward algorithm for hyperspectral image classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 3153–3161.

[168] Maria Frasca, Jianyi Lin, and Davide La Torre, "Comparing forward-forward and backpropagation in u-net for melanoma image classification," in *2024 International Conference on Decision Aid Sciences and Applications (DASA)*, 2024, pp. 1–5.

[169] Ali Kianfar, Parvin Razzaghi, and Zahra Asgari, "Integrating convolutional layers and biformer network with forward-forward and backpropagation training," *Scientific Reports*, vol. 15, no. 1, pp. 7230, 2025.

[170] Riccardo Scodellaro, Jana Zschüntzsch, Anna-Kathrin Hell, and Frauke Alves, "A first explainable-ai-based workflow integrating forward-forward and backpropagation-trained networks of label-free multiphoton microscopy images to assess human biopsies of rare neuromuscular disease," *Computer Methods and Programs in Biomedicine*, p. 108733, 2025.

[171] Hossein Aghagolzadeh and Mehdi Ezoji, "Contrastive forward-forward: A training algorithm of vision transformer," *arXiv preprint arXiv:2502.00571*, 2025.

[172] Saleh Baghersalimi, Alireza Amirshahi, Tomas Teijeiro, Amir Aminifar, and David Atienza, "Layer-wise learning framework for efficient dnn deployment in biomedical wearable systems," in *2023 IEEE 19th International Conference on Body Sensor Networks (BSN)*. IEEE, 2023, pp. 1–4.

[173] Giorgia Dellaferrera, Stanisław Woźniak, Giacomo Indiveri, Angeliki Pantazi, and Evangelos Eleftheriou, "Introducing principles of synaptic integration in the optimization of deep neural networks," *Nature Communications*, vol. 13, no. 1, pp. 1885, 2022.

[174] Ali Momeni, Babak Rahmani, Matthieu Malléjac, Philipp Del Hougne, and Romain Fleury, "Backpropagation-free training of deep physical neural networks," *Science*, vol. 382, no. 6676, pp. 1297–1303, 2023.

[175] Yequan Zhao, Xinling Yu, Zhixiong Chen, Ziyue Liu, Sijia Liu, and Zheng Zhang, "Tensor-compressed back-propagation-free training for (physics-informed) neural networks," *arXiv preprint arXiv:2308.09858*, 2023.

[176] Seonghwan Park, Dahun Shin, Jinseok Chung, and Namhoon Lee, "Fedfwd: Federated learning without backpropagation," in *Federated Learning and Analytics in Practice: Algorithms, Systems, Applications, and Opportunities*, 2023.

[177] Shuaicheng Niu, Chunyan Miao, Guohao Chen, Pengcheng Wu, and Peilin Zhao, "Test-time model adaptation with only forward passes," in *Forty-first International Conference on Machine Learning*, 2014.

[178] Ziwei Luo, Fredrik K Gustafsson, Jens Sjölund, and Thomas B Schön, "Forward-only diffusion probabilistic models," *arXiv preprint arXiv:2505.16733*, 2025.

[179] Mohammadnavid Ghader, Saeed Reza Kheradpisheh, Bahar Farahani, and Mahmood Fazlali, "Backpropagation-free spiking neural networks with the forward-forward algorithm," *arXiv preprint arXiv:2502.20411*, 2025.

[180] Guangzhi Tang, Neelesh Kumar, Ioannis Polykretis, and Konstantinos P Michmizos, "Biograd: biologically plausible gradient-based learning for spiking neural networks," *arXiv preprint arXiv:2110.14092*, 2021.

[181] Alexander G Ororbia, "Contrastive signal–dependent plasticity: Self-supervised learning in spiking neural circuits," *Science Advances*, vol. 10, no. 43, pp. eadn6076, 2024.

[182] Anthony Zador, Sean Escola, Blake Richards, Bence Ölveczky, Yoshua Bengio, Kwabena Boahen, Matthew Botvinick, Dmitri Chklovskii, Anne Churchland, Claudia Clopath, et al., "Catalyzing next-generation artificial intelligence through neuroai," *Nature communications*, vol. 14, no. 1, pp. 1597, 2023.

[183] David Silver, Anirudh Goyal, Ivo Danihelka, Matteo Hessel, and Hado van Hasselt, "Learning by directional gradient descent," in *International Conference on Learning Representations*, 2022.

[184] Dimitri Bertsekas, *Convex optimization algorithms*, Athena Scientific, 2015.

[185] Aochuan Chen, Yimeng Zhang, Jinghan Jia, James Diffenderfer, Konstantinos Parasyris, Jiancheng Liu, Yihua Zhang, Zheng Zhang, Bhavya Kailkhura, and Sijia Liu, "Deepzero: Scaling up zeroth-order optimization for deep model training," in *The Twelfth International Conference on Learning Representations*, 2024.

[186] Tao Ren, Zishi Zhang, Jinyang Jiang, Guanghao Li, Zeliang Zhang, Mingqian Feng, and Yijie Peng, "Flops: Forward learning with optimal sampling," in *The Thirteenth International Conference on Learning Representations*, 2025.

[187] Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever, "Evolution strategies as a scalable alternative to reinforcement learning," *arXiv preprint arXiv:1703.03864*, 2017.

[188] Utkarsh Singhal, Brian Cheung, Kartik Chandra, Jonathan Ragan-Kelley, Joshua B. Tenenbaum, Tomaso A Poggio, and Stella X. Yu, "How to guess a gradient," in *OPT 2023: Optimization for Machine Learning*, 2023.

[189] Felipe Petroski Such, Vashisht Madhavan, Edoardo Conti, Joel Lehman, Kenneth O Stanley, and Jeff Clune, "Deep neuroevolution: Genetic algorithms are a competitive alternative for training deep neural networks for reinforcement learning," *arXiv preprint arXiv:1712.06567*, 2017.

[190] Ke Wang, Binghong Liu, Pandi Liu, Yungao Shi, Ping Guo, Yafei Li, and Mingliang Xu, "Bi-pil: Bidirectional gradient-free learning scheme for multilayer neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

[191] Saratha Sathasivam and Wan Ahmad Tajuddin Wan Abdullah, "Logic learning in hopfield networks," *Modern Applied Science*, vol. 2, no. 3, 2008.

[192] Manas Gupta, ArulMurugan Ambikapathi, and Savitha Ramasamy, "Hebbnet: A simplified hebbian learning framework to do biologically plausible learning," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 3115–3119.

[193] James L McClelland, "How far can you go with hebbian learning, and when does it lead you astray," *Processes of change in brain and cognitive development: Attention and performance xxi*, vol. 21, pp. 33–69, 2006.

[194] Taewook Hwang, Hyein Seo, and Sangkeun Jung, "Employing layer-wised unsupervised learning to lessen data and loss requirements in forward-forward algorithms," *arXiv preprint arXiv:2404.14664*, 2024.

[195] Namyong Park, Xing Wang, Antoine Simoulin, Shuai Yang, Grey Yang, Ryan A Rossi, Puja Trivedi, and Nesreen K Ahmed, "Forward learning of graph neural networks," in *The Twelfth International Conference on Learning Representations*, 2023.

[196] Andrii Krutsylo, "Scalable forward-forward algorithm," *arXiv preprint arXiv:2501.03176*, 2025.

[197] Thomas Dooms, Ing Jyh Tsang, and Jose Oramas, "The trifecta: Three simple techniques for training deeper forward-forward networks," *Transactions on Machine Learning Research*, 2024.

[198] Chenxiang Ma, Jibin Wu, Chenyang Si, and KC Tan, "Scaling supervised local learning with augmented auxiliary networks," in *The Twelfth International Conference on Learning Representations*, 2024.

[199] Yulin Wang, Zanlin Ni, Shiji Song, Le Yang, and Gao Huang, "Revisiting locally supervised learning: an alternative to end-to-end training," in *International Conference on Learning Representations*, 2021.

[200] Liang Sun, Yang Zhang, Weizhao He, Jiajun Wen, Linlin Shen, and Weicheng Xie, "Deeperforward: Enhanced forward-forward training for deeper and better performance," in *The Thirteenth International Conference on Learning Representations*, 2025.

[201] Bernd Illing, Wulfram Gerstner, and Johanni Brea, "Biologically plausible deep learning—but how far can we go with shallow networks?," *Neural Networks*, vol. 118, pp. 90–101, 2019.

[202] Changze Lv, Jingwen Xu, Yiyang Lu, Xiaohua Wang, Zhenghua Wang, Zhibo Xu, Di Yu, Xin Du, Xiaoqing Zheng, and Xuanjing Huang, "Dendritic localized learning: Toward biologically plausible algorithm," in *Forty-second International Conference on Machine Learning*, 2025.