Unsupervised Replay Strategies for Continual Learning with Limited Data

Anthony Bazhenov*
Del Norte High School
San Diego, CA, USA
anthonyb11c234@gmail.com

Pahan Dewasurendra*
Del Norte High School
San Diego, CA, USA
pahan.dewa@gmail.com

Giri P. Krishnan
Department of Medicine
University of California, San Diego
La Jolla, CA, USA
giri.prashanth@gmail.com

Jean Erik Delanois
Department of Computer Science & Engineering
University of California, San Diego
La Jolla, CA, USA
jdelanois@ucsd.edu

2024 International Joint Conference on Neural Networks (IJCNN) | 979-8-3503-5931-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/IJCNN60899.2024.10650116

***Abstract*—Artificial neural networks (ANNs) show limited performance with scarce or imbalanced training data and face challenges with continuous learning, such as forgetting previously learned data after new tasks training. In contrast, the human brain can learn continuously and from just a few examples. This research explores the impact of 'sleep' — an unsupervised phase incorporating stochastic network activation with local Hebbian learning rules — on ANNs trained incrementally with limited and imbalanced datasets, specifically MNIST and Fashion MNIST. We discovered that introducing a sleep phase significantly enhanced accuracy in models trained with limited data. When a few tasks were trained sequentially, sleep replay not only rescued previously learned information that had been forgotten following new task training but also often enhanced performance in prior tasks, especially those trained with limited data. This study highlights the multifaceted role of sleep replay in augmenting learning efficiency and facilitating continual learning in ANNs.**

***Index Terms*—Neural Networks, limited training data, enhance memory, sleep, continual learning, unsupervised replay**

I. Introduction

Artificial neural networks have demonstrated the ability to excel beyond human levels in various domains. However, they encounter difficulties when training data are low, imbalanced, or presented sequentially where they tend to prioritize new tasks at the expense of previous tasks, a phenomenon called catastrophic forgetting [1], [2]. In contrast, humans and animals possess an extraordinary capacity to learn continuously from few examples, effortlessly assimilating new information into their existing knowledge base.

Different methods have been proposed to overcome these limitations. For low training data scenarios, these include data augmentation [3], pre-training on other datasets [4] or alternative architectures such as neural tangent kernel [5]. However, these approaches do not address the fundamental question of how to make overparameterized deep learning networks learn to generalize from small datasets without overfitting.

§*Equal contribution

Existing approaches to prevent catastrophic forgetting generally fall under two categories: rehearsal and regularization methods [6]. Rehearsal methods combine previously learned data, either stored or generated, with current task data to avoid forgetting [7]–[12]. Regularization approaches [13], [14] aim to modify plasticity rules by incorporating additional constraints on gradient descent such that important weights from previously trained tasks are maintained. All these methods have significant limitations (reviewed in [1], [2]). Importantly, methods to prevent catastrophic forgetting are not always compatible with methods aimed at improving low data performance.

The human brain demonstrates the ability to learn continuously and quickly from just a few examples. It has been suggested that memory replay during biological sleep can strengthen memories learned during wakefulness [16], [17]. During sleep, neurons are spontaneously active without external input and generate complex patterns of synchronized activity across brain regions [18], [19]. Two critical components which are believed to underlie memory consolidation during sleep are spontaneous replay of memory traces and local unsupervised synaptic plasticity that restricts synaptic changes to relevant memories [16], [20], [21]. During sleep, replay of recently learned memories along with relevant old memories [22]–[27] enables the network to form stable orthogonal memory representations [24] and reduces competition

TABLE I
Neural network model and training parameters

Model Details	
Arch. Size	784, 1200, 1200, 10
Learning Rate (Training / Fine-tune)	0.06 / 0.02
Epochs (Training / Fine-tune)	5 (10, 50) / 1
Dropout	0.25

Algorithm 1 Sleep Replay Consolidation [15]

procedure SLEEP($nn, I, scales, thresholds$) ▷ I is input
  Initialize v (voltage) $= 0$ vectors for all neurons
  for $t \leftarrow 1$ to Ts **do** ▷ Ts - Number of time steps during sleep
    $\mathbf{S} \leftarrow 0s$
    $\mathbf{S}(1) \leftarrow$ Convert input I to Poisson-distributed spiking activity
    $\mathbf{S}$ = ForwardPass($\mathbf{S}, v, \mathbf{W}, scales, thresholds$)
    $\mathbf{W}$ = BackwardPass($\mathbf{S}, \mathbf{W}$)
  end for
end procedure
procedure FORWARDPASS($S, v, W, scales, threshold$)
  for $l \leftarrow 2$ to n **do** ▷ n - number of layers
    $\alpha \leftarrow scales(l-1)$
    $\beta \leftarrow threshold(l)$
    $v(l) \leftarrow \lambda v(l) + (\alpha * \mathbf{W}(l, l-1) * \mathbf{S}(l-1))$ ▷ W(l,l-1) - weights / λ - decay rate
    $\mathbf{S}(l)_i \leftarrow 1 \; \forall \; i$ where $v(l)_i > \beta$ ▷ Propagate spikes
    $v(l)_i \leftarrow 0 \; \forall \; i$ where $v(l)_i > \beta$ ▷ Reset spiking voltages

  end for
return S
end procedure
procedure BACKWARD PASS(S, W)
  for $l \leftarrow 2$ to n **do** ▷ n - number of layers
    $\mathbf{W}(l, l-1)_{i,j} \leftarrow \begin{cases} \mathbf{W}(l, l-1)_{i,j} + inc & \forall \text{ i,j where } \mathbf{S}(l)_j = 1 \;\&\; \mathbf{S}(l-1)_i = 1 \\ \mathbf{W}(l, l-1)_{i,j} - dec & \forall \text{ i,j where } \mathbf{S}(l)_j = 1 \;\&\; \mathbf{S}(l-1)_i = 0 \\ \mathbf{W}(l, l-1)_{i,j} & \text{else} \end{cases}$ ▷ STDP
  end for
return W
end procedure

between memory representations to enable the coexistence of competing memories within overlapping populations of neurons [28].

The idea of replay has been explored in machine learning to enable continual learning (reviewed in [1], [2]). However, spontaneous unsupervised replay found in the biological brain is significantly different compared to explicit replay of past inputs as implemented in machine learning rehearsal methods. Unlike the standard global optimization methods used in machine learning (such as backpropagation), local plasticity allows synaptic changes to affect only relevant memories. Research from neuroscience [15], [21], [28], [29] suggest that applying sleep replay principles to ANNs may enhance memory representations and, consequently, improve the performance of machine learning models trained continuously on a limited or unbalanced datasets.

In this new study, we apply the previously proposed sleep replay consolidation (SRC) method [15] to scenarios when the model is trained sequentially on several tasks with limited data. We found that sleep-like replay can both improve the performance of models trained with limited data as well as rescue previously trained task performance that was damaged by new training.

II. METHODS

A. Neural Network

We used a fully connected feed-forward neural network with 2 hidden layers and ReLU nonlinearities consisting of 1200 nodes each, followed by a soft-max classification layer with 10 output neurons. The model was trained using hidden layer dropout and a binary cross entropy loss with weights modified by a standard stochastic gradient descent optimizer. Each neuron in the network operated without a bias, which aided in the conversion to a spiking neural network during the sleep stage [30].

B. Sleep Replay Consolidation (SRC) algorithm

The sleep replay consolidation (SRC) algorithm was implemented as previously described in [15] on the models used in this work. The intuition behind SRC is that a period of off-line, noisy activity may reactivate network nodes that represent tasks trained in awake. If network reactivation is combined with unsupervised local learning, SRC can then strengthen necessary and weaken unnecessary pathways in the network. Even if a model is under-trained there is information regarding the task encoded in the synaptic weight matrix, SRC can then augment and enhance this previously existing structure.

SRC consists of first mapping of an ANN to SNN, where the network's activation function is replaced by a Heaviside function (threshold function) and weights are scaled by the layer-wise activation maximum observed on the previous training dataset(s), as suggested by [30], to ensure the network maintains reasonable firing activity. Successive forward passes are executed where generated spike trains are fed to the input layer thereby propagating activity (spiking behavior) across the network. For each input vector, the probability of assigning a value of 1 (bright or spiking) to a given element (input pixel) is taken from a Poisson distribution with mean rate calculated from the average element intensity across all previously observed training inputs. Thus, e.g., a pixel that was typically bright in all training inputs would be assigned a value of 1 more often than a pixel with lower mean intensity. Following each forward pass, a backward pass is executed to update synaptic weights. To modify network connectivity

during sleep we use an unsupervised, simplified Hebbian type learning rule which is implemented as follows: a weight is increased between two nodes when both pre- and post-synaptic nodes are activated (i.e., input exceeds the Heaviside activation function threshold), a weight is decreased between two nodes when the post-synaptic node is activated but the pre-synaptic node is not (in this case, presumably another pre-synaptic node is responsible for activity in the post-synaptic node). After running multiple steps of this unsupervised learning during sleep, the final weights are rescaled back (simply by removing the original scaling factor), the Heaviside-type activation function is restored to the ReLU, and testing or further supervised training on new data is performed. The hyperparameters used during SRC are listed in Table II and were determined using a genetic algorithm aimed at maximizing performance on the validation set (results then generalized to the test set).

TABLE II
HYPERPARAMETERS FOR THE SRC ALGORITHM

	MNIST	FMNIST
Time Steps	365	267
Dt	0.001	0.001
Max Firing Rate (Input Generation)	211.9	465.5
Alpha Scale (Augments layer-wise scales)	15.5	23.7
Beta	[24.3, 5.08, 18.42]	[22.60, 14.93, 23.05]
Decay	0.97	0.95
Synaptic Increase	$7.49 * 10^{-4}$	$4.95 * 10^{-4}$
Synaptic Decrease	$-1.87 * 10^{-4}$	$-2.72 * 10^{-4}$

C. Datasets

To evaluate the effects of sleep on under-trained and under-performing Artificial Neural Networks (ANNs) we leveraged two datasets: MNIST [31] and Fashion MNIST (FMNIST) [32]. The MNIST dataset consists of handwritten numbers (0-9) each belonging to its own class while FMNIST consists of 10 classes of Zalando's article images. Together the MNIST and FMNIST datasets are some of the most widely used datasets in machine learning making them good candidates to initially investigate sleep's ability to improve performance in limited data and imbalanced data contexts. Each dataset consists of 60,000 training images and 10,000 testing images.

D. Single task training with limited data

To test the effect of sleep on networks trained with limited data, we trained the neural network on a randomly selected subset (0.1% - 100%) of the full datasets MNIST and FMNIST. Subsequently we applied a sleep replay and fine-tuning stage to see how performance was impacted. The sleep replay stage followed the algorithm explained in Section II-B.

We used 5 epochs of training in the main analysis and we verified results using 10 and 50 epochs of training. The fine-tuning stage following sleep consisted of a single epoch of training with a learning rate of 0.02 with all other hyperparameters remaining the same. The same network architecture and training parameters were used in all simulations.

During our experiment implementation, we found that the number of images in each class can vary significantly when a small fraction of data from MNIST or FMNIST is extracted randomly. This can inadvertently cause preferential performance on over-represented classes and diminished performance on under-represented classes. Therefore, in our experiments we first ensured that each class contained the same exact number of images when a portion of data from MNIST or FMNIST was extracted.

E. Training with limited and imbalanced data

We conducted a second set of experiments to test the effect of sleep on training data imbalance when the overall dataset is limited. This is relevant for many real world scenarios where examples of certain positive or negative classes are not available due to data acquisition constraints.

In these experiments, we first extracted a subset of images (10%) from MNIST. Within this limited subset, we explicitly introduced imbalance by reducing the number of images in one selected class, keeping the number of images for all other classes equal and fixed. In this scenario, after the initial training phase with backpropagation, the ANN became biased towards classes with more training data at the expense of the class with reduced data. We then tested whether sleep, as described in II-B, could recover performance for the reduced data class.

F. Data Limited Continual Learning

In our third set of experiments, we utilized the MNIST and FMNIST datasets to examine the influence of SRC on ANN performance in a data limited continual learning paradigm. Here we leveraged 2 tasks, the initial task involved distinguishing classes 0-4 while the subsequent task focused on classes 5-9. The datasets were then randomly sub-sampled such that 1% to 20% of the original task data remained and used for model training.

The ANN underwent sequential training with initial training on Task 1 (classes 0-4) followed by training on Task 2 (classes 5-9). Each MNIST/FMNIST task training phase consisted of 3/5 epochs with a learning rate of 0.06 and a Binary Cross Entropy loss function optimized by Stochastic Gradient Decent (batch size 64). Following initial Task 1 training, the model learned to differentiate classes 0-4 while performance on the untrained Task 2 digits remained poor. After training the second task, Task 2 performance (classes 5-9) reached a maximum while Task 1 was catastrophically forgotten. These two training phases were followed by SRC (Section II-B), after which the ANN was tested again for both tasks.

The 5 task scenario was trained similarly to the two task scenario consisting of 3/5 epochs with a learning rate of 0.06 and a Binary Cross Entropy loss function optimized by Stochastic Gradient Decent (batch size 64) for MNIST / FMNIST. Here, each task consisted of differentiating two classes (e.g., 0 vs 1, 2 vs 3, 4 vs 5, and so forth) with 5% of the total training data available for the corresponding classes. To explore the robustness of our approach, models were trained on five different random task orderings (which were subsequently renamed to reflect chronological task order) with each task

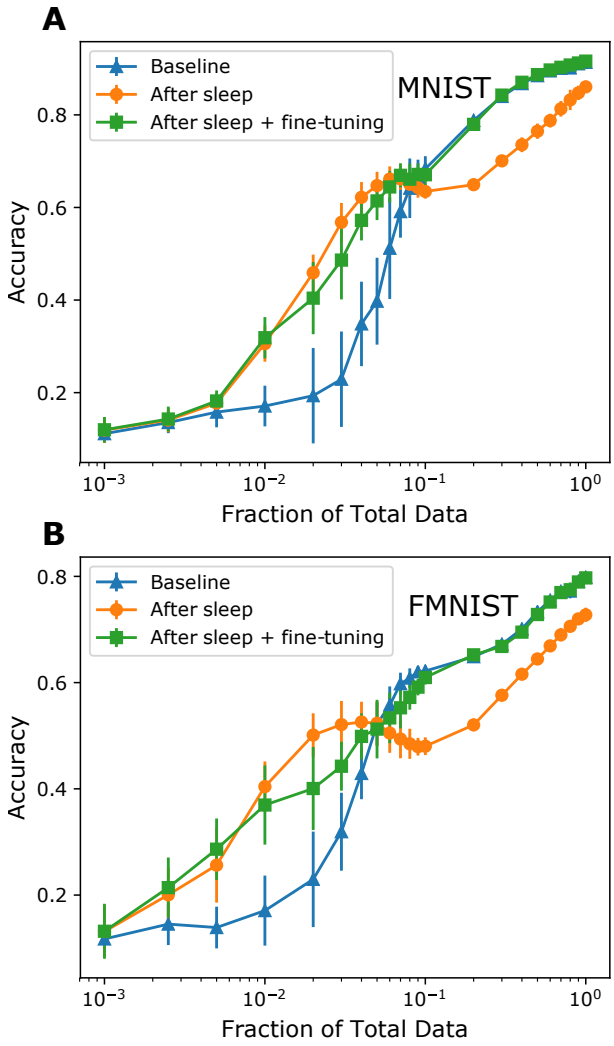

Fig. 1. Accuracy on MNIST **(A)** and FMNIST **(B)** with mean (lines) and standard deviation (error bars) across 10 trials. X-axis - log of the relative amount of data used for training (e.g., 0.01=1% of data). Blue - baseline (after ANN training); Orange - baseline + sleep; Green - baseline + sleep + fine-tuning. Note significant gain in accuracy after the sleep phase on low data. The sleep phase reduced performance on high data but was largely recovered by fine-tuning.

order tested across five trials differing in randomness (initialization, training, and SRC) to ensure generalizable results. Although additional SRC hyperparameters were tuned, single and multi-task performance improvements were still observed.

III. Results

A. Performance with limited training data

When the network was trained on the full dataset, the model achieved over 90% accuracy on both the MNIST and Fashion MNIST datasets. Yet, as illustrated in Fig. 1, accuracy significantly drops (blue line) when only a fraction of training data was used. To evaluate the impact of sleep replay on accuracy, Sleep Replay Consolidation (SRC) was employed following initial training with limited data. Briefly, the ANN was trained on limited data then translated to a Spiking Neural Network (SNN) with an identical architecture (for details, see the Methods section). This SNN was then stimulated by Poisson-distributed spiking inputs with mean firing rates corresponding to mean input element intensity across all datasets used during the preceding training sessions. During the sleep phase, synaptic weights were adjusted through local Hebbian-type plasticity: strengths were increased if presynaptic activity led to a postsynaptic response, and reduced if postsynaptic activity occurred in the absence of presynaptic activation. Following the sleep phase, the SNN was transformed back into an ANN where performance was reassessed either immediately or after an additional fine-tuning on the original limited dataset.

We found a significant improvement in accuracy for models trained with low amounts of data (0.2-10% of the original dataset) for both MNIST and FMNIST after introducing the sleep phase (Fig. 1, orange line). We verified these results using 10 and 50 epochs of training. These long training duration experiments revealed that while increasing the amount of training improved baseline performance in data restricted scenarios, sleep was still capable of further improving accuracy.

Fig. 2 presents the confusion matrices obtained during the MNIST experiment, prior to and following the sleep phase. For these experiments we employed a 3% subset of the MNIST dataset for training while using the training parameters detailed in Table I. Following training, the network demonstrated noteworthy accuracy for only four specific classes, achieving 0.99 for class 0, 0.51 for class 3, 0.71 for class 8, and 0.87 for class 9 (Fig. 2A). The remaining classes were predicted inaccurately and misclassified as class 0, 8, or 9. Overall, the prediction accuracy was very low at 0.32. After SRC, there were significant accuracy improvements for most underperforming classes. Specifically there was notable class-wise enhancement for 1, 2, 4, 6 and 7 (Fig. 2B) which led to an overall improved accuracy of 0.60.

We obtained similar results for FMNIST. Fig. 3 illustrates the confusion matrices generated using a 3% subset of the FMNIST dataset. Prior to the sleep phase, the overall accuracy stood at 0.38, with only classes 0, 6, 8, and 9 being accurately-classified (Fig. 3A). Following sleep, a notable improvement was observed in the accuracy of other classes (2,3,4,7) (Fig. 3B), culminating in an overall accuracy of 0.59.

These results suggest that not only does SRC improve performance in low data contexts but it proportionally aids underperforming classes yielding a more class balanced model.

While performance improved when there was limited training data, we also observed a slight (10-15%) decrease in performance when more than approximately 10% of the data was employed for ANN training. This performance degradation, however, could be mitigated by introducing a short one epoch fine tuning phase after sleep using the original (limited) training data (Fig. 1, green line).

B. Performance with limited and imbalanced data

Next, we systematically tested the effect of SRC in class imbalanced settings, i.e., in addition to training the network

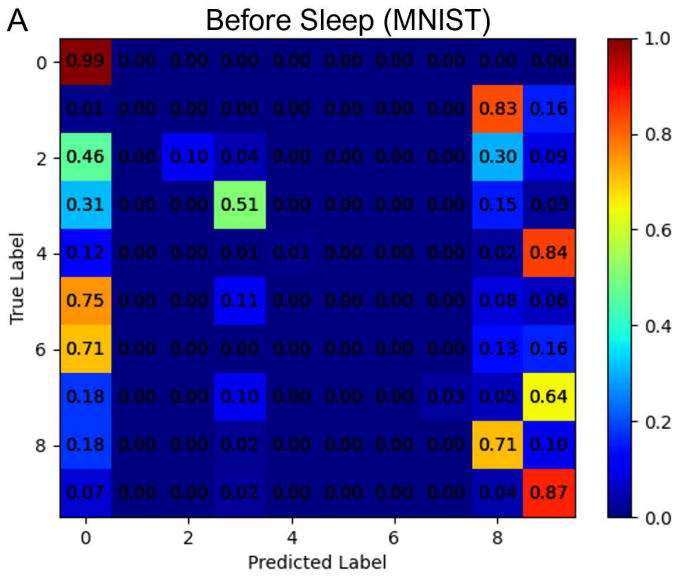

Fig. 2. Confusion matrices before and after sleep for MNIST dataset. A 3% subset of the overall MNIST dataset was used in training. The value in each cell indicates the fraction of images of a given true label that were classified as a given predicted label by the model. **(A)** - before SRC, **(B)** - after SRC.

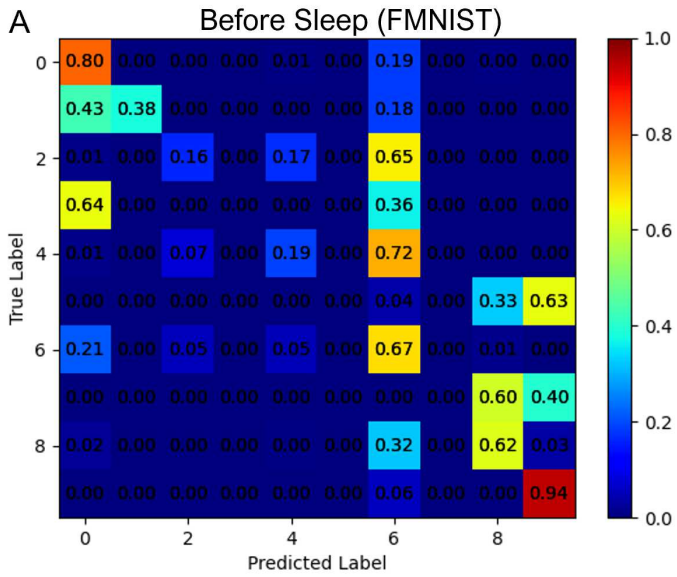

Fig. 3. Confusion matrices before and after sleep for FMNIST dataset. A 3% subset of the overall FMNIST dataset was used in training. The value in each cell indicates the fraction of images of a given true label that were classified as a given predicted label by the model. **(A)** - before SRC, **(B)** - after SRC.

on limited data, one selected class was explicitly underrepresented. Fig. 4 shows an example of such analysis when we used a 10% subset of MNIST training data and further limited the number of images for one selected class during training. In practice, we developed these datasets by first randomly selecting a 10% subset of the MNIST dataset and ensuring each class had the exact same number of images. Then, for one selected "imbalanced" class, we further restricted the number of instances present causing this class to be underrepresented.

The fraction of images in the low data class ranged from 10% to 100% of the original training data subset (i.e. 1% to 10% of the class from the original full MNIST).

We found that class-wise model performance was more robust to data reduction for some classes when compared to others. For example, digit 0 showed high class-wise accuracy when more than 70-80% digit 0's were used for training (i.e., more than 7-8% of digit 0's in the total dataset), while digit 5 had very low performance even when all data were used (i.e., 10% of 5's in the total dataset). After SRC, most classes (except digit 8) showed a positive improvement in class-wise accuracy (Fig. 4). The magnitude of the gain and the range of data reduction where the gain was observed were varied between digits likely because of the different sensitivity to data

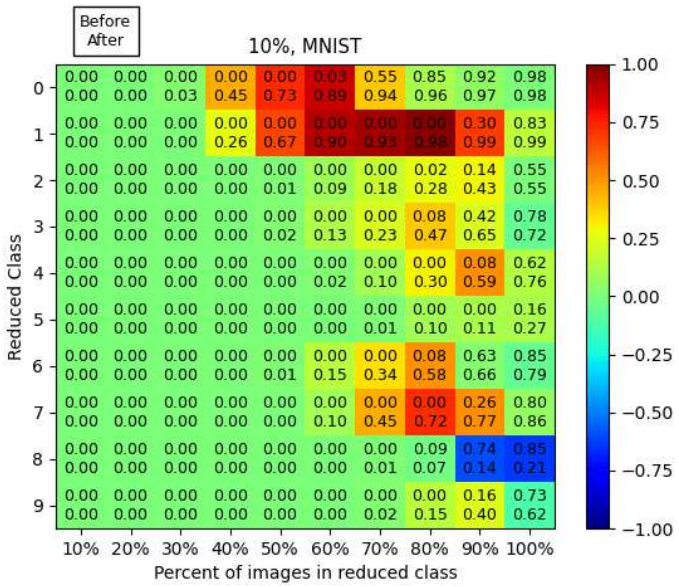

Fig. 4. Imbalanced class accuracy improvement due to sleep. Each row shows experiments with data reduction for one specific class (shown on the left), with the percentage of reduction shown on the horizontal axis. Each cell shows the class-wise accuracy of the underrepresented class before sleep (top value) and after sleep (bottom value). The color map is based on the change in accuracy, Δ = After Sleep - Before Sleep. Reds indicate a positive difference (improvement), while blues indicate a negative difference (drop in accuracy). Note, many red squares showing class-wise improvement with only a few blue squares showing class-wise performance loss.

reduction. Similar results (not shown) were observed with the FMNIST dataset.

C. Performance during sequential learning tasks with limited data

We next tested how sleep replay can affect performance of models trained on sequential data limited tasks. A previous study [15] reported that SRC can protect older tasks damaged by the recent training, however, it remains unknown if SRC can accomplish the same effect when training data are limited. It is also unknown if the same SRC hyper-parameters can both protect old tasks' performance while simultaneously improving recent under-trained task performance.

To test SRC in a sequential learning paradigm, we created 2 tasks (per dataset) for the MNIST and Fashion MNIST where the first task consisted of the first 5 classes and the second task comprised of the last 5. These two tasks (T1 and T2) were trained sequentially followed by a sleep phase. The amount of data used to train each task was varied independently in the range 0-20% of the full datasets.

Fig. 5 shows results for both the MNIST and Fashion MNIST datasets. After T1 training, network accuracy increased for that task (top left plots in Fig. 5A,B), while intuitively the untrained T2 performance was zero (middle left plots in Fig. 5A,B). Subsequent T2 training led to an increase in T2 performance (middle middle plots in Fig. 5A,B) but T1 suffered from catastrophic forgetting causing T1 performance to fall to zero, except when T2 training data was very low (top

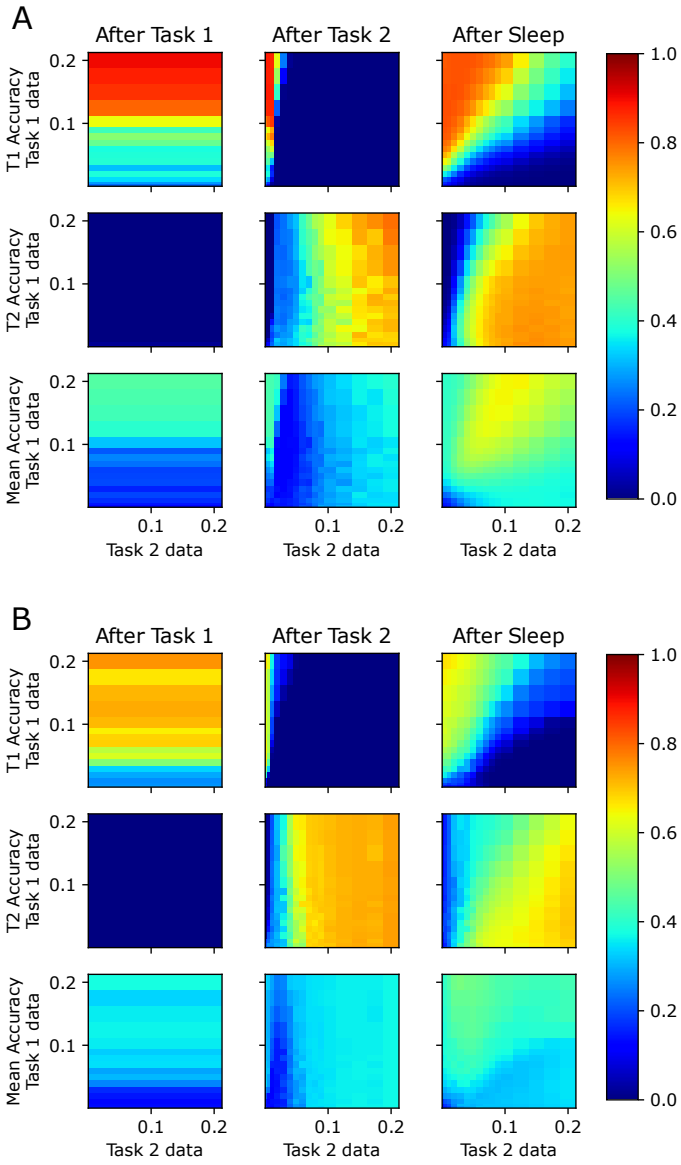

Fig. 5. Heatmaps showing accuracy changes after each phase of sequential learning and SRC (columns) on MNIST **(A)** and FMNIST **(B)**. In each subplot, X-axis represents amount of T2 training data and Y-axis - amount of T1 training data. The rows show accuracy on T1, T2 and the mean accuracy.

middle plots in Fig. 5A,B). Finally, SRC implemented after T2 training rescued T1, except when the T2 dataset significantly exceeded the size of the T1 dataset (top right plots in Fig. 5A,B). Following this period of replay, T2 performance was further increased for MNIST (middle right plot in Fig. 5A) and slightly decreased for FMNIST (middle right plot in Fig. 5B).

Fig. 6 presents the same results but for the specific case of equal amounts of training data for T1 and T2 (i.e., diagonal in Fig. 5) with the Y-axis denoting mean accuracy between two tasks and X-axis denoting amount of data. Overall, we observed an increase in the mean accuracy in the range 15-30% for MNIST and 5-15% for FMNIST with the exact improvement dependent on the amount of training data.

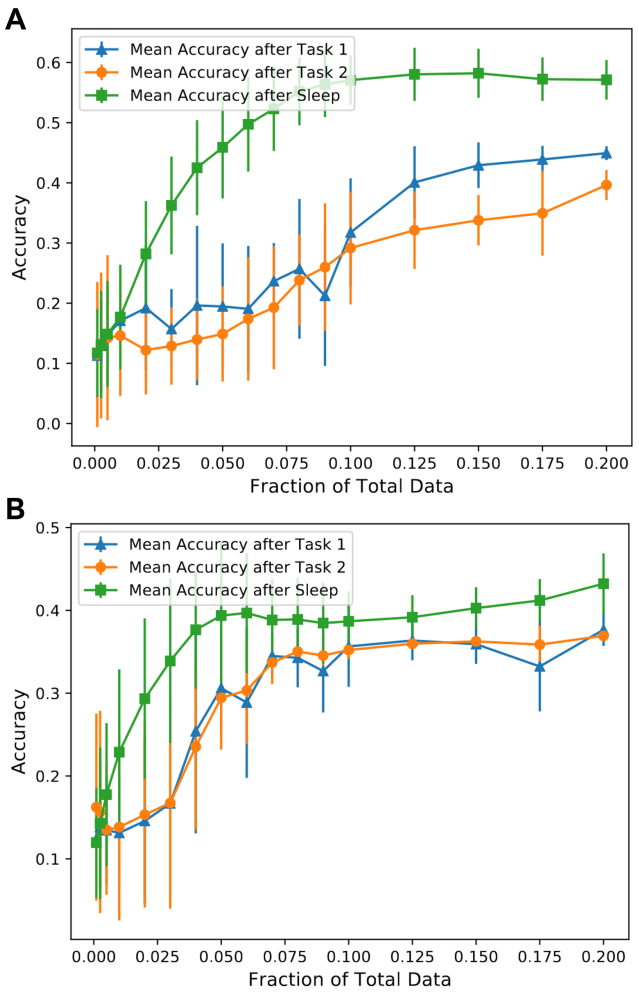

Fig. 6. Accuracy for sequential learning tasks on MNIST **(A)** and FMNIST **(B)** with mean (lines) and standard deviation (error bars) across 10 trials. X-axis - log of the relative amount of data used for training (e.g., 0.01=1% of data). Blue - baseline (after ANN training); Orange - baseline + sleep; Green - baseline + sleep + fine-tuning. Note significant gain in overall accuracy after sleep phase on low data.

To further investigate, we implemented SRC within a multi-task paradigm. The MNIST and FMNIST datasets were each divided into 5 subset tasks, models were then trained on random orders of these tasks either with or without interspersed periods of SRC. Each task consisted of 5% of the total data available for that corresponding subset. Our findings again illustrate that sequential training, even on limited data, invariably leads to complete catastrophic forgetting of prior tasks (Fig. 7A,B). However, when SRC is strategically interleaved between training sessions, there is a marked recovery of previously learned tasks and sometimes the most recently learned task, resulting in a notably elevated mean task performance (Fig. 7C,D). For example, for MNIST data, Task 1 performance is 0.39 after Task 2 training but reaches 0.61 after subsequent sleep (Fig. 7C). Interestingly, we also found that sleep has a protective effect that reduced the amount of damage to old tasks observed immediately after new task training. For example, compare the effect of Task 2 training on Task 1 accuracy without sleep between training stages (Fig. 7A, change from 0.78 to 0) and when sleep is applied in between T1 and T2 training (Fig. 7C, change from 0.82 to 0.39).

D. Network dynamics during SRC

In an attempt to gain further insight as to why SRC was capable of improving under-trained model performance, we analyzed the sleep stage for the single task scenario. We first measured network activity by examining the instantaneous layer-wise firing rates. It can be observed that all layers, excluding the output layer, in both the MNIST and FMNIST networks were highly active (Fig. 8). The lack of output layer activity along with the previously described Hebbian learning rules means the output layer received no modifications during the entirety of SRC, all benefits were therefore due to hidden layer modifications.

Analysis of individual hidden layer neuron activity revealed a progressive decay of activity over the course of the sleep phase (Fig. 9). This was more obvious in the second hidden layer. Thus, task performance improved by SRC reducing the overall level of activity in the network thereby forming sparser, but presumably more contrasting, representations between tasks.

To confirm this prediction, we next examined the magnitude of weight modifications. Fig. 10 shows a histogram of the cumulative weight perturbation each synapse received during sleep. Although a small number of critical synapses showed an increase in strength (positive weight deltas) most synapses decreased their sensitivity (Fig. 10 shows large number of synapses with negative weight deltas; note log scale in Y-axis).

IV. Discussion

Although artificial neural networks (ANNs) have recently begun to approach human performance in various tasks, from intricate games [33] to image classification [34] to language processing [35], they exhibit several limitations. These include catastrophic forgetting, lack of generalization, and dependency on substantial amounts of training data. Catastrophic forgetting [36]–[38] is the phenomenon where a system cannot learn new information without losing previously acquired knowledge. Consequently, even if new data that could enhance a model's performance are available, direct training on these new datasets often results in damage to previously learned information. Additionally, ANNs may develop bias when training data is limited and imbalanced, leading to models that are unfairly prejudiced against certain categories. This poses risks in sensitive applications like healthcare, where it could result in misdiagnoses, or in legal settings, where it might contribute to unequal treatment [39], [40]. Addressing these biases is crucial for creating fair and reliable AI systems.

Here, we have tested an unsupervised sleep replay consolidation (SRC) algorithm on artificial neural networks trained incrementally with limited and imbalanced datasets. We found that SRC effectively mitigates catastrophic forgetting and improves accuracy when only a very small fraction of data is

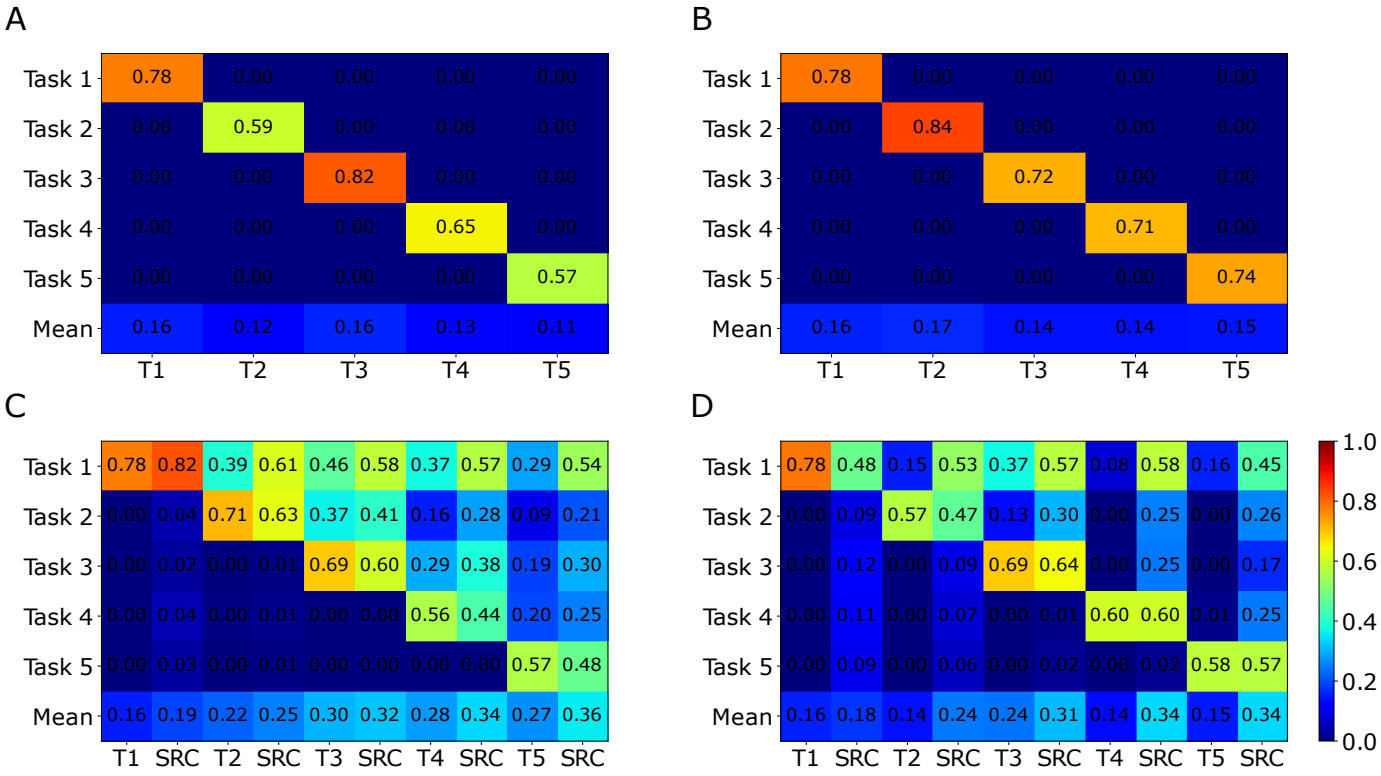

Fig. 7. Accuracy for sequential training across 5 tasks on MNIST ((**A**)) and FMNIST ((**B**)) datasets, alongside interleaved training and SRC on MNIST ((**C**)) and FMNIST ((**D**)). Notably, integrating SRC between task training reduces forgetting and increases average joint task performance.

used for training or when data is class-imbalanced. Effectively, SRC was able to shift the transition point from "low" to "high" accuracy as the amount of training data increases. The relationship between training data and accuracy has been widely studied (e.g., [41]), and our study provides evidence that the transition to high accuracy during learning could be influenced by spontaneously generated unsupervised replay. One advantage of SRC is that it is complementary to fine-tuning and other gradient-based methods and thus can be used in conjunction with these methods.

Existing methods to prevent catastrophic forgetting generally fall into two categories: rehearsal and regularization techniques [6]. Rehearsal methods involve incorporating previously learned data, either stored or generated, alongside new inputs during subsequent training sessions to mitigate forgetting [7]–[12]. While effective, as the number of previously learned tasks increases, this approach may necessitate increasingly complex generative networks capable of reproducing all previously acquired knowledge.

Regularization methods aim to modify plasticity rules, maintaining important weights from previous tasks by adding constraints to gradient descent [13]. Notably, the Elastic Weight Consolidation (EWC), Synaptic Intelligence (SI) and Orthogonal Weight Modification (OWM) techniques penalize weight updates for prior tasks [14], [42], [43]. While these methods show promise in preventing catastrophic forgetting across various tasks, such as MNIST permutations, Split MNIST, and Atari games, they may not perform optimally in class-incremental learning scenarios, where one input class is learned at a time [9]. OWM has demonstrated greater success than EWC and SI in class-incremental learning tasks.

Methods to improve network accuracy when training data are limited include data augmentation [3], pre-training on other datasets [4] or alternative architectures such as neural tangent kernel [5]. It is worth noting, that these methods should be applied independently from those developed to enable continual learning leading to complex models with many hyperparameters to be tuned.

SRC has been proposed as an unsupervised method applicable to solving various tasks, including catastrophic forgetting [15], lack of generalization [44], [45], and low accuracy with limited training data [46]. However, the advantages of SRC were reported in independent studies, each using different sets of hyperparameters tuned to optimize performance for a specific task. Here, we report that SRC can accomplish several independent tasks together in the same model and using the same set of hyperparameters. These results are consistent with the known role of sleep in memory and learning.

Sleep has been shown to play a critical role in memory consolidation, generalization, and the transfer of knowledge in biological systems [17], [47], [48]. During sleep, there is a reactivation of neurons involved in previously learned activity [16]. Because sleep reactivation depends on the previously learned synaptic connectivity matrix, it invokes the same spatio-temporal patterns of neuronal firing as the patterns observed during preceding training in the awake state [20].

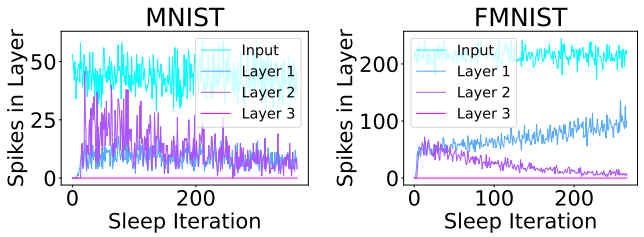

Fig. 8. Layer-wise activity during SRC for MNIST **(left)** and FMNIST **(right)**. It can be seen that all layers are active except the output layer in both the MNIST and FMNIST networks implying the benefits of SRC are due to enhancing the networks ability to extract relevant features.

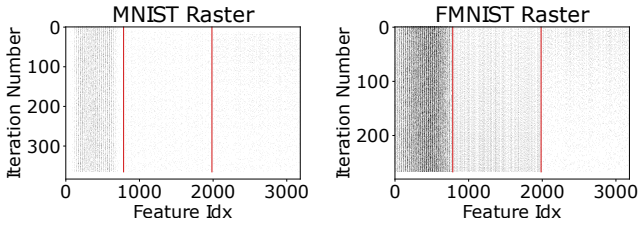

Fig. 9. MNIST **(left)** and FMNIST **(right)** spike rasters for input and hidden layers (denoted by red lines). While generated input remains at a consistent level (left of first red lines), hidden layer activity begins high and slightly decays over the course of sleep (right of first red lines)

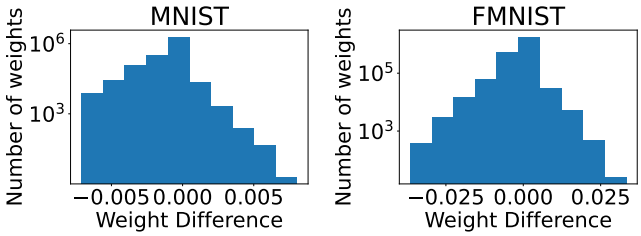

Fig. 10. Weight difference distributions for MNIST **(left)** and FMNIST **(right)**. Plots highlight that many synapses decrease in strength implying performance benefits are a result of suppressing incorrect signals.

Thus, sleep reactivation, or replay, would strengthen synapses involved in a previously learned task. It was reported that plasticity changes during sleep can increase a brain's ability to reduce memory interference [28], to form connections between memories and to generalize knowledge learned during the awake state [49], to link semantic categories based on a single episodic event to enable one-shot learning [50].

Why can noise-driven unsupervised plasticity improve accuracy and successfully recover memories learned during ANN training, as we report here? In biology, sleep-generated brain activity, while spontaneous, still mirrors the synaptic weight structure acquired during training. It was observed in vivo that similar spatial patterns of neuronal activity occur both spontaneously and in response to specific stimuli [51]. Although the noisy input in the SRC algorithm may lack the higher-order structure of the training data, sleep replay extracts such complex interactions, as these are embedded in the synaptic weight patterns. This form of spontaneous replay in the biological brain, as implemented in the SRC algorithm, differs significantly from the explicit replay of past inputs common in machine learning rehearsal methods, reviewed in [1], [2]

Analysis of synaptic weight dynamics revealed that while SRC increased the strength for a subset of presumably critical synapses, many others were weakened. This suggests that the overall increase in accuracy after SRC was a result of sparser responses. Thus, SRC may be improving feature representations by causing hidden layer neurons to be less sensitive to certain stimuli. Intuitively, in a data-limited context, the training distribution may have many outliers among the small number of samples, thereby causing the model to pick up irrelevant features only present in a small percentage of examples. SRC then selectively suppresses the strength of many synapses while maintaining a few, thereby reducing irrelevant feature sensitivity and leading the model to focus on the most common and therefore predictive features. In multiple task scenarios, SRC may reduce weights biased towards the most recent tasks, thus allowing older tasks to be correctly classified as long as information about these tasks is still present in the synaptic weights matrix.

In sum, our study sheds light on a potential synaptic weight dynamics strategy employed by the brain during sleep to enhance memory performance for continual learning when training data are limited or imbalanced. Applied to ANNs, sleep-like replay improves performance in a completely unsupervised manner, requiring no additional data, and can be applied to pre-trained models.

Acknowledgment

Supported by NSF (grants 2323241 and 2223839).

References

[1] D. Kudithipudi, M. Aguilar-Simon, J. Babb, M. Bazhenov, D. Blackiston, J. Bongard, A. P. Brna, S. C. Raja, N. Cheney, J. Clune, A. Daram, S. Fusi, P. Helfer, L. Kay, N. Ketz, Z. Kira, S. Kolouri, J. L. Krichmar, S. Kriegman, M. Levin, S. Madireddy, S. Manicka, A. Marjaninejad, B. McNaughton, R. Miikkulainen, Z. Navratilova, T. Pandit, A. Parker, P. K. Pilly, S. Risi, T. J. Sejnowski, A. Soltoggio, N. Soures, A. S. Tolias, D. Urbina-Melendez, F. J. Valero-Cuevas, G. M. van de Ven, J. T. Vogelstein, F. Wang, R. Weiss, A. Yanguas-Gil, X. Y. Zou, and H. Siegelmann, "Biological underpinnings for lifelong learning machines," *Nature Machine Intelligence*, vol. 4, no. 3, pp. 196–210, 2022.

[2] T. L. Hayes, G. P. Krishnan, M. Bazhenov, H. T. Siegelmann, T. J. Sejnowski, and C. Kanan, "Replay in deep learning: Current approaches and missing biological elements," *Neural Computation*, vol. 33, no. 11, pp. 2908–2950, 2021.

[3] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[4] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[5] S. Arora, S. S. Du, Z. Li, R. Salakhutdinov, R. Wang, and D. Yu, "Harnessing the power of infinitely wide deep nets on small-data tasks," *arXiv preprint arXiv:1910.01663*, 2019.

[6] R. Kemker, M. McClure, A. Abitino, T. L. Hayes, and C. Kanan, "Measuring catastrophic forgetting in neural networks," in *Thirty-second AAAI conference on artificial intelligence*, 2018.

[7] T. L. Hayes, N. D. Cahill, and C. Kanan, "Memory efficient experience replay for streaming learning," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9769–9776.

[8] A. Robins, "Catastrophic forgetting, rehearsal and pseudorehearsal," *Connection Science*, vol. 7, no. 2, pp. 123–146, 1995.

[9] G. M. van de Ven, H. T. Siegelmann, and A. S. Tolias, "Brain-inspired replay for continual learning with artificial neural networks," *Nature communications*, vol. 11, no. 1, pp. 1–14, 2020.

[10] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," in *Advances in Neural Information Processing Systems*, 2017, pp. 2990–2999.

[11] P. Buzzega, M. Boschini, A. Porrello, D. Abati, and S. Calderara, "Dark experience for general continual learning: a strong, simple baseline," *Advances in neural information processing systems*, vol. 33, pp. 15 920–15 930, 2020.

[12] P. Buzzega, M. Boschini, A. Porrello, and S. Calderara, "Rethinking experience replay: a bag of tricks for continual learning," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 2180–2187.

[13] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.

[14] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[15] T. Tadros, G. Krishnan, R. Ramyaa, and M. Bazhenov, "Sleep-like unsupervised replay reduces catastrophic forgetting in artificial neural networks," *Nature Communications*, vol. 13, no. 1, p. 7742, 2022.

[16] R. Stickgold, "Sleep-dependent memory consolidation," *Nature*, vol. 437, no. 7063, pp. 1272–1278, 2005.

[17] P. A. Lewis and S. J. Durrant, "Overlapping memory replay during sleep builds cognitive schemata," *Trends in cognitive sciences*, vol. 15, no. 8, pp. 343–351, 2011.

[18] M. Steriade, D. A. McCormick, and T. J. Sejnowski, "Thalamocortical oscillations in the sleeping and aroused brain," *Science*, vol. 262, no. 5134, pp. 679–685, 1993.

[19] G. P. Krishnan, S. Chauvette, I. Shamie, S. Soltani, I. Timofeev, S. S. Cash, E. Halgren, and M. Bazhenov, "Cellular and neurochemical basis of sleep stages in the thalamocortical network," *Elife*, vol. 5, p. e18607, 2016.

[20] M. A. Wilson and B. L. McNaughton, "Reactivation of hippocampal ensemble memories during sleep," *Science*, vol. 265, no. 5172, pp. 676–679, 1994.

[21] Y. Wei, G. P. Krishnan, and M. Bazhenov, "Synaptic mechanisms of memory consolidation during sleep slow oscillations," *Journal of Neuroscience*, vol. 36, no. 15, pp. 4231–4247, 2016.

[22] P. A. Lewis, G. Knoblich, and G. Poe, "How memory replay in sleep boosts creative problem-solving," *Trends in cognitive sciences*, vol. 22, no. 6, pp. 491–503, 2018.

[23] E. Hennevin, B. Hars, C. Maho, and V. Bloch, "Processing of learned information in paradoxical sleep: relevance for memory," *Behavioural brain research*, vol. 69, no. 1-2, pp. 125–135, 1995.

[24] B. Rasch and J. Born, "About sleep's role in memory," *Physiological reviews*, 2013.

[25] S. C. Mednick, D. J. Cai, T. Shuman, S. Anagnostaras, and J. T. Wixted, "An opportunistic theory of cellular and systems consolidation," *Trends in neurosciences*, vol. 34, no. 10, pp. 504–514, 2011.

[26] K. A. Paller and J. L. Voss, "Memory reactivation and consolidation during sleep," *Learning & Memory*, vol. 11, no. 6, pp. 664–670, 2004.

[27] D. Oudiette, J. W. Antony, J. D. Creery, and K. A. Paller, "The role of memory reactivation during wakefulness and sleep in determining which memories endure," *Journal of Neuroscience*, vol. 33, no. 15, pp. 6672–6678, 2013.

[28] O. C. González, Y. Sokolov, G. P. Krishnan, J. E. Delanois, and M. Bazhenov, "Can sleep protect memories from catastrophic forgetting?" *Elife*, vol. 9, p. e51005, 2020.

[29] R. Golden, J. E. Delanois, P. Sanda, and M. Bazhenov, "Sleep prevents catastrophic forgetting in spiking neural networks by forming joint synaptic weight representations," *PLoS Computational Biology*, vol. 18, no. 11, p. e1010628, 2022.

[30] P. U. Diehl, D. Neil, J. Binas, M. Cook, S.-C. Liu, and M. Pfeiffer, "Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing," in *2015 International Joint Conference on Neural Networks (IJCNN)*. ieee, 2015, pp. 1–8.

[31] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[32] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[33] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel *et al.*, "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018.

[34] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012.

[35] Y. Liu, T. Han, S. Ma, J. Zhang, Y. Yang, J. Tian, H. He, A. Li, M. He, Z. Liu, Z. Wu, L. Zhao, D. Zhu, X. Li, N. Qiang, D. Shen, T. Liu, and B. Ge, "Summary of chatgpt-related research and perspective towards the future of large language models," *Meta-Radiology*, vol. 100017, p. 21, 2023, arXiv:2304.01852v4 [cs.CL]. [Online]. Available: https://doi.org/10.48550/arXiv.2304.01852

[36] J. L. McClelland, B. L. McNaughton, and R. C. O'Reilly, "Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory." *Psychological review*, vol. 102, no. 3, p. 419, 1995.

[37] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in cognitive sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[38] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*. Elsevier, 1989, vol. 24, pp. 109–165.

[39] N. Norori, Q. Hu, F. M. Aellen, F. D. Faraci, and A. Tzovara, "Addressing bias in big data and ai for health care: A call for open science," *Patterns*, vol. 2, no. 10, p. 100347, 2021.

[40] D. V. PS, "How can we manage biases in artificial intelligence systems – a systematic literature review," *International Journal of Information Management Data Insights*, vol. 3, no. 1, p. 100165, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667096823000125

[41] M. Paul, S. Ganguli, and G. K. Dziugaite, "Deep learning on a data diet: Finding important examples early in training," *Advances in Neural Information Processing Systems*, vol. 34, pp. 20 596–20 607, 2021.

[42] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3987–3995.

[43] G. Zeng, Y. Chen, B. Cui, and S. Yu, "Continual learning of context-dependent processing in neural networks," *Nature Machine Intelligence*, vol. 1, no. 8, pp. 364–372, 2019.

[44] T. Tadros, G. Krishnan, R. Ramyaa, and M. Bazhenov, "Biologically inspired sleep algorithm for increased generalization and adversarial robustness in deep neural networks," in *International Conference on Learning Representations*, 2019.

[45] J. E. Delanois, A. Ahuja, G. Krishnan, T. Tadros, and M. Bazhenov, "Improving robustness of convolutional networks through sleep-like replay," in *2023 22nd IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2023.

[46] A. Bazhenov, P. Dewasurendra, G. Krishnan, and J. E. Delanois, "Sleep-like unsupervised replay improves performance when data are limited or unbalanced (student abstract)," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024 in press.

[47] D. Ji and M. A. Wilson, "Coordinated memory replay in the visual cortex and hippocampus during sleep," *Nature neuroscience*, vol. 10, no. 1, pp. 100–107, 2007.

[48] M. P. Walker and R. Stickgold, "Sleep-dependent learning and memory consolidation," *Neuron*, vol. 44, no. 1, pp. 121–133, 2004.

[49] J. D. Payne and et al., "The role of sleep in false memory formation," *Neurobiology of Learning and Memory*, vol. 92, no. 3, pp. 327–334, 2009.

[50] S. Witkowski, E. Schechtman, and K. A. Paller, "Examining sleep's role in memory generalization and specificity through the lens of targeted memory reactivation," *Current Opinion in Behavioral Sciences*, vol. 33, pp. 86–91, 2020.

[51] M. Tsodyks, T. Kenet, A. Grinvald, and A. Arieli, "Linking spontaneous activity of single cortical neurons and the underlying functional architecture," *Science*, vol. 286, no. 5446, pp. 1943–1946, 1999.