Online Knowledge Acquisition with the Selective Inherited Model

Xiaocong Du, Shreyas Kolala Venkataramanaiah, Zheng Li†, Jae-sun Seo, Frank Liu‡, Yu Cao
School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ, USA
†School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ, USA
‡CSMD, Oak Ridge National Lab, Oak Ridge, TN, USA
Email: {xiaocong, skvenka5, zhengl11, jseo28, ycao}@asu.edu, liufy@ornl.gov.

***Abstract*—Continual learning, which updates machine learning models according to streaming data, is increasingly needed in the dynamic systems. Such a scenario requires both the preservation of previous knowledge, as well as the adaptation to new observations, with high computational and memory efficiency at the edge. Previous approaches attempt to learn the knowledge class by class from scratch, using either regularization based or memory replay-based methods. However, they still suffer from severe accuracy drop, a.k.a catastrophic forgetting, during this incremental process. Moreover, as the entire model is involved in each updating, their computation cost is too expensive for edge computing. In this work, we propose a novel brain-inspired paradigm named acquisitive learning (AL). Different from previous approaches that focus only on model adaptation, AL emphasizes the importance of both knowledge inheritance and acquisition: the model is first pre-trained and selected in the cloud (the selective inherited model) and then adapted to new knowledge (the acquisition). The quality of the inherited model is monitored by the landscape of the loss function, while the acquisition is realized by segmented training. The combination of both steps reduces accuracy drop by $>10\times$ on the CIFAR-100 dataset. Furthermore, AL benefits edge computing with $5\times$ reduction in latency per training image on FPGA prototype and $150\times$ reduction in training FLOPs.**

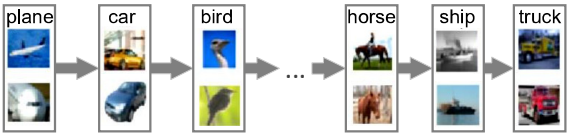
***Index Terms*—Continual learning, acquisitive learning, deep neural networks, brain inspiration, model adaptation, knowledge inheritance, knowledge acquisition**

I. Introduction

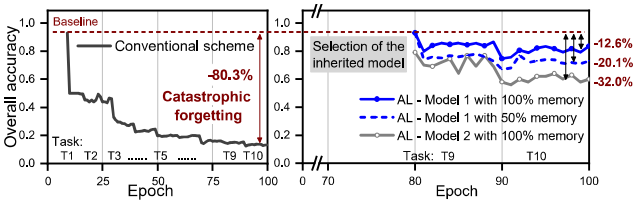
The rapid development of machine learning algorithms and computing hardware has accelerated the implementation of many modern edge applications, such as autonomous vehicles, surveillance drones, and robots. These emerging edge devices are required to handle more complicated and dynamic scenarios locally and in real-time, as compared to conventional edge devices such as mobile phones. One of the critical demands is the capability to learn from a data stream over time, *i.e.* the capability of *continual learning* [1]–[5]. Such a capability requires the system to learn from new observations without interfering or overwriting previous learned knowledge (*i.e.* model parameters). Furthermore, the learning should be bounded by computation and energy resources, including but not limited to the model size, the computation cost and storage, while still completing the process in real-time.

Today the biggest challenge in continual learning is known as *catastrophic forgetting* [6]. When a model is updated to a sequence of new tasks with very limited or even no access to previous input data, previously acquired knowledge is deteriorated, leading to severe accuracy drop (*i.e. forgetting*). While there have been multiple attempts to mitigate catastrophic forgetting [1], [3], [4], [7]–[10], they all follow a conventional procedure of continual learning: updating the model task by task, from scratch, as shown in Fig. 1(a). To be specific, when the learning system starts to learn new knowledge from a data stream, there is no prior knowledge embedded in this model. In this scenario, the network parameters are randomly initialized, and the learning process only focuses on model adaptation. Such a conventional learning flow is suffering from severe accuracy drop, as shown in Fig. 1(b)(left), and excessive computation cost [5]. Moreover, focusing only on model adaption is not the complete picture in biology. It is observed that the brain inherits knowledge in specific neuro-physiological structures (*i.e.* hardwired), through a long and careful evolution process [11]–[13]. Besides model adaptation, the hardwired model that is selected and inherited is also critical to the quality of intelligence.

To overcome the above limitations of conventional continual learning scheme, we propose acquisitive learning (AL), as



(a) Conventional continual learning: incrementally learn one class after another from scratch.



(b) Conventional scheme vs. Acquisitive Learning.

Fig. 1. When learning from a data stream of CIFAR-10, conventional continual learning suffers from catastrophic forgetting, while the proposed acquisitive learning successfully mitigates such forgetting by >6X on CIFAR-10. A well selective inherited model and memory replay all contribute to the accurate learning. Among them, the quality of the inherited model is more vital than the amount of memory used to replay. Model 1 refers to ResNet-56 with better landscape; model 2 refers to ResNet-56-NS with worse landscape.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

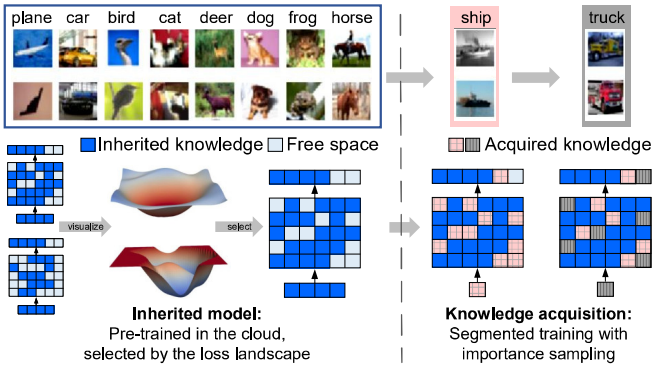


Fig. 2. The flow of acquisitive learning emphasizes both the importance of knowledge inheritance and knowledge acquisition.

shown in Fig. 2. Inspired by the inherited brain model, AL emphasizes both the importance of knowledge inheritance and acquisition: the majority of knowledge is first pre-trained and preserved in the inherited model, and then the model is adapted to new streaming data (the acquisition). More important, we confirm the vital correlation between *the quality of the inherited model* and its *acquisition capacity* on new knowledge. Though pre-training feature extraction layers of a model has been previously explored in transfer learning [14], such a model is still suffering from accuracy drop when the feature space rarely overlaps between old and new data. Such an accuracy drop is because the one-shot pre-trained model is too stochastic to generalize better for new observations.

In this paper, we claim that the pre-trained inherited model should be elaborately selected to optimize future learning performance. We propose one selection method via visualizing the loss landscape [15] and measuring its roughness with quadratic linear regression. That being said, we believe the selection criteria should not be limited to what is proposed in this paper. For the acquisition step, we leverage importance sampling from Progressive Segmented Training (PST) [5] to identify and freeze important parameters for the inherited model, and only train the secondary parameters to acquire new knowledge. In this process, a small and bounded memory set is used to retrieve the previous knowledge.

In summary, model inheritance and selection, knowledge acquisition with importance sampling and memory replay all contribute to the final accuracy in the learning from streaming data. The combination of these techniques reduces the accuracy drop due to catastrophic forgetting by $6.6\times$ on CIFAR-10 and $11.5\times$ on CIFAR-100 dataset. Among these techniques, the selective inherited model plays an indispensable role in maintaining the accuracy, while memory replay plays a complementary role, as verified by the results in Fig. 1(b) and in future sections. Further more, AL is efficient in computation cost. AL reduces the latency per training image by $5\times$ and overall training FLOPs by $150\times$ as benchmarked by FPGA prototype.

To summarize, the contribution of this paper is as below:

- We propose a brain-inspired scheme for learning from streaming data, namely acquisitive learning (AL). Different from conventional continual learning that only focuses on model adaptation, AL emphasizes the importance of both knowledge inheritance and acquisition.
- With experiments on various deep neural networks and datasets, we demonstrate that the proposed AL effectively reduces catastrophic forgetting when learning from streamed data.
- Experiments show that the acquisition is strongly related to the quality of the inherited model and thus, the inherited model should be elaborately selected rather than being one-shot attained. In this paper, we leverage landscape visualization and roughness measurement to select the model.
- We further implement the training of AL with FPGA prototype and benchmark the significant reduction in computation cost, which enables continual learning at the edge.

II. Preliminaries

This section presents the terminology, previous work the and biology background.

A. Terminology

A deep neural network (DNN) such as VGG-Net [16] and ResNet [17] usually consists of a feature extractor $\varphi : \mathcal{X} \rightarrow \mathbb{R}^d$ and classification weight vectors $w \in \mathbb{R}^d$, also known as convolutional layers and fully-connected layers. The network parameters Θ (φ and w) keep being updated according to input data $\mathcal{X}$, and calculating output $\mathcal{Y} = w^\top \varphi(\mathcal{X})$ in order to predict labels $\mathcal{Y}^*$.

When learning the first task with input data $\{X^1, \ldots, X^{s-1}\}$, DNN tries to minimize the loss $\mathcal{L}(\mathcal{Y}; \mathcal{X}_{s-1}; \Theta)$ of this $(s-1)$-class classifier. When a new task with input data $\{X^s, \ldots, X^t\}$ arrives, DNN tries to minimize $\mathcal{L}(\mathcal{Y}; \mathcal{X}_t; \Theta)$ of this t-class classifier by updating Θ. Usually, after the input data of the new task $\{X^s, \ldots, X^t\}$ arrives, the input data of previous task $\{X^1, \ldots, X^{s-1}\}$ is no longer available, except a small subset stored as the memory set $\mathcal{P} = (P_1, \ldots, P_{s-1})$.

B. Conventional approach of continual learning

The conventional approach of continual learning starts from a set of fresh, randomly initialized network parameters Θ, and each incoming task updates entire Θ or partial Θ. They leverage different techniques such as regularization [1], [3], parameter isolation [7], [8], memory replay [4], [9], or network expansion [10], [18] to mitigate catastrophic forgetting.

Regularization-based approaches add penalty term in the loss function to regularize the parameter updating space. Parameter isolation approaches assign a subset of parameters to specific task updating. Memory replay approaches train the model with a small subset of previously seen data. Network expansion approaches expand network by adding new branches or parameters to include new knowledge. However, as the network is not inheriting any prior knowledge, each new task

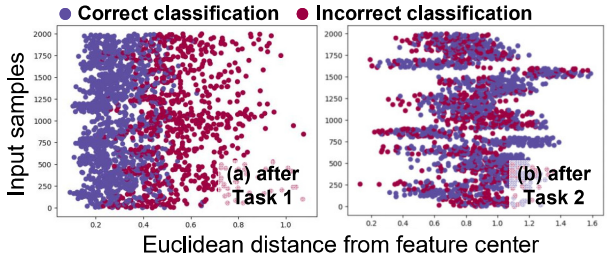


Fig. 3. The main reason of catastrophic forgetting is the drift in the feature space. Visualizing the Euclidean distance between $\varphi(\mathcal{X})$ of each input image in Task 1 and the feature center (*i.e.* the normalized $\varphi(\mathcal{X})$ of the current task): (a) after learning 10 classes (Task 1) from CIFAR-100 with ResNet-56, wrongly classified samples are relatively further from the feature center; (b) after learning another 10 classes (Task 2) from CIFAR-100, the feature center drifts so that the correlation between Euclidean distance and classification is deteriorated.

can easily update the weight distribution and cause feature drifting, as shown in Fig. 3, and thus causing catastrophic forgetting. In other words, conventional approaches focus more on model adaptation to new tasks, without inheriting any prior knowledge. In contrast to them, acquisitive learning emphasizes both knowledge inheritance and acquisition.

C. Difference from transfer learning

It is worth some words here to differentiate transfer learning with the proposed acquisitive learning. Transfer learning (or domain adaptation) is a method where a network developed for one task is reused to learn a new task. It can be formulated as follows: for a new task with input data $\{X^s, \ldots, X^t\}$, DNN tries to minimize $\mathcal{L}(\mathcal{Y}; \mathcal{X}_{s\ldots t}; \Theta)$ of this $(t-s+1)$-class classifier by reusing network φ pre-trained on $\{X^1, \ldots, X^{s-1}\}$ and fine-tuning classification weight vectors w. Thus, the differences between transfer learning and the proposed AL are: (1) transfer learning only focuses on the learning of the new domain while AL requires to learn new tasks and to remember the old tasks; (2) transfer learning is usually one-shot domain transfer, while AL requires to learn a sequential of tasks; (3) transfer learning usually freezes entire feature extractor φ and only fine-tune classification layers, limiting the acquisition of new knowledge. In AL, we only freeze selected parameters to help remember previous knowledge and leave enough Θ to acquire new knowledge; (4) transfer learning does not select pre-trained model, but directly uses the one-shot pre-trained model without quality evaluation.

D. Biology background: Moravec's paradox

There have been increasing evidences [11]–[13] showing that the brain inherits knowledge in specific neurophysiological structures, through a long and careful evolution process, while the capability to adapt in the field is comparatively much more challenging. This was identified as the Moravec's paradox [13], and has led to research outcomes that support the hardwired model of learning. Indeed, the intelligence in nature may be determined more by the long-term genetics and inheritance rather than the short-term adaptation [11]. Therefore, to successfully learn new knowledge, the selective inherited model and knowledge acquisition is both critical.

III. Acquisitive Learning

With preliminaries defined in the previous section, we describe acquisitive learning from two perspectives: model inheritance and knowledge acquisition.

A. Model inheritance

1) Prepare inherited model: In this subsection, we explain how to prepare the inherited model. Throughout this paper, we refer to a network that has been well pre-trained and selected as the inherited model.

Acquisitive learning first trains the network with randomly selected classes from a dataset, and then samples crucial learning units (convolution filters and fully-connected neurons) for the current task. The importance sampling is based on an important score that has been proven in [5], [19], [20]. The score is used to measure how important a filter/neuron is to the loss function.

For a convolution filter $\Theta_l^o \in \mathbb{R}^{I_l \times K \times K}$, the score is formulated as:

$$|\Delta\mathcal{L}(\Theta_l^o)| \simeq |\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^o}\Theta_l^o| = \sum_{i=0}^{I_l}\sum_{m=0}^{K}\sum_{n=0}^{K}|\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^{o,i,m,n}}\Theta_l^{o,i,m,n}|, \tag{1}$$

where $\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^{o,i,m,n}}$ is the gradient of the loss function with respect to the weight pixel $\Theta_l^{o,i,m,n}$.

For a neuron $\Theta_l^t \in \mathbb{R}^{1\times I_l}$, the score is formulated as:

$$|\Delta\mathcal{L}(\Theta_l^t)| \simeq |\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^t}\Theta_l^t| = \sum_{i=0}^{I_l}|\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^{t,i}}\Theta_l^{t,i}|, \tag{2}$$

where $\frac{\partial\mathcal{L}(\mathcal{Y};\mathcal{X};\Theta)}{\partial\Theta_l^{t,i}}$ is the gradient of the loss with respect to the parameter $\Theta_l^{t,i}$.

Based on the importance score, we sort the learning units layer by layer and identify the top β units for the inherited model. We following the same setting in [5] for β: it should be roughly proportional to the amount of the inherited knowledge. In the following adaptation, these important units are not updated but kept unchanged, in order to preserve inherited knowledge.

2) Landscape visualization and roughness measurement: Following the above-mentioned method, we are able to obtain inherited models with consolidated knowledge. We leverage landscape visualization tool [15] to visualize the minima of the loss function and then calculate the roughness using linear regression. In [15], filter normalization is used to remove the scaling effect, and a 3-dimension matrix (x, y, z, where x, y are the coordinates and z is the loss function) is finally extracted and plotted for visualization. To further quantify the roughness of the landscape, we fit this 3D data using quadratic

linear regression and obtain mean square error (MSE) of this fitting model to represent the roughness:

$$\hat{z}_j = w_{j4}x_j^2 + w_{j3}y_j^2 + w_{j2}x_j + w_{j1}y_j + w_{j0}, \quad (3)$$

$$\hat{w} = \underset{w_j}{\operatorname{argmin}} \frac{1}{n}\sum_{j=0}^{n}(z_j - \hat{z}_j)^2, \quad (4)$$

where w_j represent the fitted coefficients. We denote the roughness as $\text{MSE}(z; x^2, y^2, x, y; \hat{w})$. Models with smaller MSE are more flat and smooth, and vice versa.

B. Knowledge acquisition

With the inherited model fully pre-trained and important units selected, acquisitive learning leverages techniques proposed in PST [5] to learn new observations. PST techniques include importance sampling, model segmentation, memory-assisted training and balancing (we omit model reinforcement step in PST for simplicity). When a new observation arrives, only the secondary parameters (*i.e.* those are not frozen) in the inherited model are updated while the important parameters are frozen. In other words, the model is segmented to the inherited part and the acquisition part. Meanwhile, a small subset of data containing uniformly and randomly sampled images per class from all the trained classes (*i.e.* each class in $\{X^1, \ldots, X^t\}$ contains the same number of images) so far is mixed with new observations to train and balance the model.

By using techniques including importance sampling, model segmentation, memory-assisted training and balancing, the acquisitive learning scheme is able to acquire new knowledge based on an inherited model. It is worth mentioning that the techniques used to consolidate inherited knowledge and to acquire new knowledge are flexible. In this paper, we focus more on the acquisitive learning methodology.

IV. Experimental Results

In this section, we develop various experiments to verify the efficacy of the proposed acquisitive learning flow.

A. Experiment setup

The experiments in Section IV B-D are performed with PyTorch [21] on one NVIDIA GeForce RTX 2080 platform. We use stochastic gradient descent with momentum of 0.9 and weight decay of 0.0005. For each experiment, we shuffle the class order and run 5 times to report the average accuracy. In Section IV-E, Intel Stratix-10 GX equipped with the 4Gb DDR3 with 17Gb/s bandwidth was used as the target hardware. Latency was measured using functional simulation of the CNN training accelerator [22] at 240MHz.

a) Datasets: The CIFAR dataset [23] consists of 50,000 training images and 10,000 testing images in color with size 32×32. CIFAR-10 consists of 10 classes, and CIFAR-100 consists of 100 classes. In the following experiments, we first train a subset of dataset to produce the inherited model, and then we treat the unseen classes as new knowledge that needs to be acquired. The balanced memory set contains 200 and 20 images for each class for CIFAR-10 and CIFAR-100, respectively, so that the total memory size is bounded within 2,000 images for both datasets, aligning with previous work [4], [5].

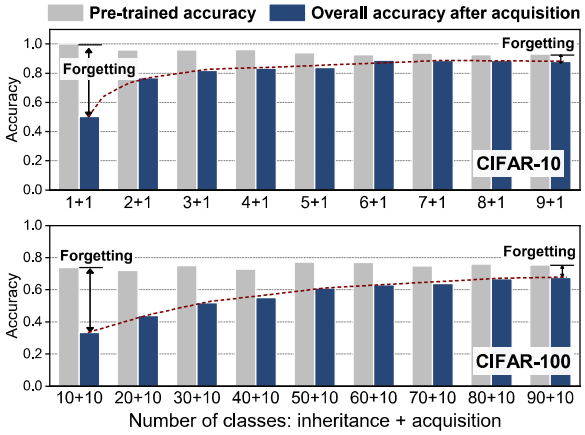


Fig. 4. Accuracy drop is minimized with the an increasing amount of knowledge in the inherited model. Top: VGG-16 on CIFAR-10 dataset. Bottom: ResNet-56 on CIFAR-100 dataset.

b) Network structure: The network structures of VGG-16 [16], ResNets [17], DenseNets [24] used in the following experiment are standard structures following [15]. Since the total number of classes is unknown in a real-world application, we leave $1.2\times$ space at the final classification layer in the following experiments, *i.e.* 12 outputs for CIFAR-10 and 120 outputs for CIFAR-100. Note that the extra space reserved at the final classification layer does not affect the evaluation since there is no feedback from vacant outputs.

c) Evaluation protocol: 'Pre-trained accuracy' or 'accuracy of the inherited model' refers to the testing accuracy of $(s-1)$-class classifier if input data is $\{X^1, \ldots, X^{s-1}\}$. 'Accuracy on the new task' refers to the testing accuracy of $(t-s+1)$-class classifier for input data $\{X^s, \ldots, X^t\}$ as new observations. 'Overall accuracy' refers to the testing accuracy of t-class classifier on all the data seen so far. 'Accuracy forgetting' refers to the accuracy drop from pre-trained accuracy to overall accuracy during continual learning.

B. Amount of inherited knowledge

We first explore whether and how the amount of inherited knowledge impacts the acquisition capacity. We mimic the different amount of inherited knowledge using different numbers of pre-trained classes and plot the results in Fig. 4. 'X+Y' refers to the scenario when X classes are pre-trained in the inherited model and Y classes need to be acquired. There is no overlapping of classes in X and Y. The inherited model size (frozen filters/neurons) is proportional to the number of classes in X across experiments. For the new task Y, we use the same number of classes across experiments and keep the number of active filters/neurons for this new task the same. In Fig. 4 (top) for '1+1' case on CIFAR-10 with VGG-16 network, the accuracy drops from 100.0% to 50.5% (49.5% forgetting); but for '9+1' case, the accuracy drops from 93.0% to 88.0% (5.0% forgetting). In Fig. 4 (bottom), the accuracy

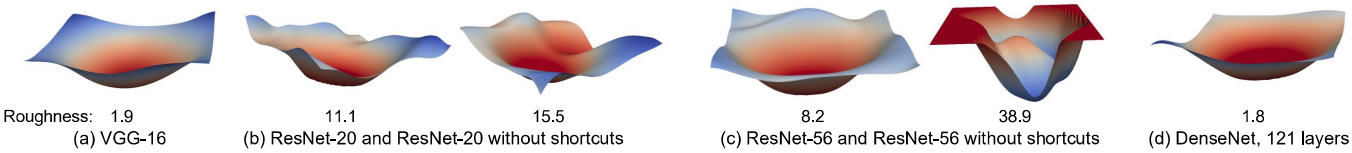


Fig. 5. Landscape visualization of the loss function for 6 models. Shallow models (like VGG-16) have smooth landscapes. Deep models with shortcuts have smoother landscapes than the ones without shortcuts.

TABLE I
ACQUISITION CAPACITY FOR DIFFERENT MODELS. '9+1' EXPERIMENT WITH CIFAR-10 DATASET IS PRESENT HERE.

Network	VGG-16	ResNet-20	ResNet-20-NS	ResNet-56	ResNet-56-NS	DenseNet-121
Pre-trained accuracy	0.927	0.915	0.901	0.923	0.790	0.935
Accuracy on the new task	0.865	0.851	0.810	0.860	0.597	0.883
Accuracy drop	0.062	0.064	0.091	0.063	*0.193*	**0.052**
Roughness ($\times 10^{-3}$)	1.9	11.1	15.5	8.2	*38.9*	**1.8**

forgetting is 40.5% for '10+10' case but only 7.7% for '90+10' case. It is concluded that, with more knowledge embedded in the inherited model, less forgetting is observed for acquisition, and such a trend gradually saturates.

C. Quality of the inherited model

Besides the amount of inherited knowledge, the inherited model itself is also a critical factor in acquisitive learning. As explained in [15], different deep learning models have a different landscape of the loss function, where wide and flat minima generalizes better and sharp minima with many small regions of convexity generalizes poorly. The quality of the landscape is influenced by model depth, model size, batch size, and skip connections (*i.e.* 'shortcuts') between layers. We plot six representative models that are pre-trained on the same 9 classes of CIFAR-10 but with different landscapes and their corresponding roughness measurement in Fig. 5. VGG-16, ResNet-20 with and without shortcuts, ResNet-56 with and without shortcuts, and DenseNet-121. Among them, VGG-16, ResNet-20, ResNet-56 and DenseNet-121 have relatively flat landscapes and thus lower roughness; ResNet-20 without shortcuts (ResNet-20-NS) and ResNet-56 without shortcuts (ResNet-56-NS) have relatively sharp landscapes and higher roughness. The landscape of ResNet-56-NS is the most rough one. Note that these six models exhibit the same amount of inherited knowledge (9 classes) but show different quality in acquisition.

For each of these six inherited models, we add one new class to acquire and report the accuracy in Table I. The first row shows the pre-trained accuracy of the inherited models on 9 classes, and the second row represents the testing accuracy on the new task. We focus more on the relative accuracy between the first row to the second row, shown in the row 'accuracy drop', as this data represents the generalization ability of the pre-trained model on new observations, *i.e.* the acquisition capacity of the inherited model. On ResNet-20-NS and ResNet-56-NS models that have sharper landscapes,

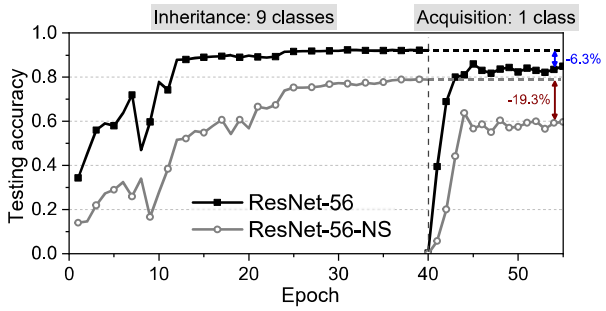
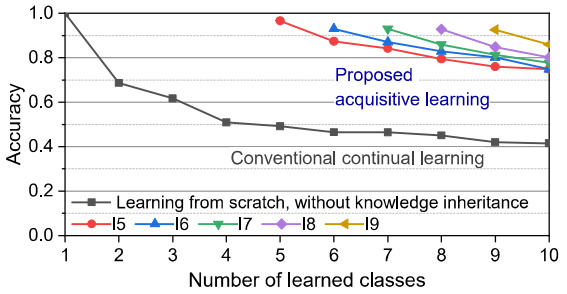


Fig. 6. Learning curve for '9+1' experiment on CIFAR-10 with two models. 6.3% and 19.3% accuracy drop is observed for ResNet-56 and ResNet-56-NS, respectively.

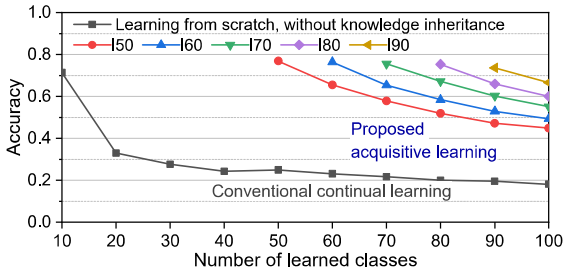
we observe 9.1% and 19.3% accuracy drop, respectively. This drop is more severe as compared to other models, indicating that the knowledge acquisition capacity of these two models are poor. We further zoom in ResNet-56 and ResNet-56-NS in Fig. 6 by plotting the learning curve of '9+1' simulation. ResNet-56-NS has worse acquisition capacity on new tasks than ResNet-56. These results indicate that the quality of the inherited model is another vital factor in knowledge acquisition.

D. Learning from a data stream with AL

We design experiments to verify that acquisitive learning is a more effective approach to learn from streaming data as compared to conventional continual learning scheme. On one side, we simulate the conventional continual learning that starts learning from scratch (*i.e.* no inherited model with pre-trained knowledge is available for knowledge acquisition) and learns each task (1 class from CIFAR-10 or 10 classes from CIFAR-100) in a sequence. We follow the techniques described in Section III to learn new tasks. β is set as 0.1 for the first task. The overall accuracy of conventional method is plotted in gray in Fig. 7(a) and Fig. 7(b). On the other side, assuming inherited

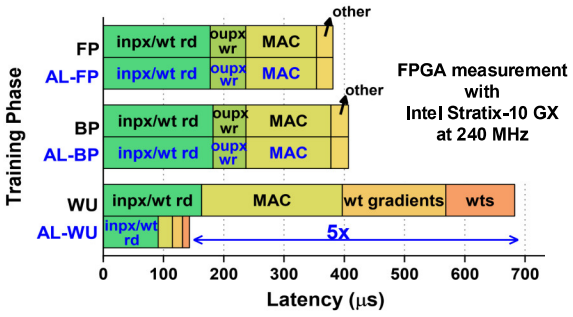


(a) Incrementally learning 1 class of CIFAR-10 with VGG-16.

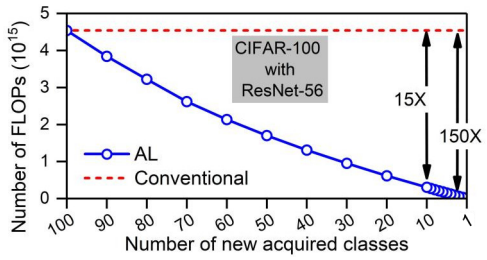


(b) Incrementally learning 10 classes of CIFAR-100 with ResNet-20. To align with previous work [4], [5], we use ResNet-20 here though it is not the smoothest model.

Fig. 7. The comparison of overall accuracy between conventional continual learning and the proposed acquisitive learning on two datasets. In the figure, 'I5' means that AL starts training from a model that is pre-trained on 5 classes. Similarly, 'I90' means the inherited model is pre-trained on 90 classes.



(a) Latency per training image



(b) Number of FLOPs needed for acquisition of various number of new classes, derived from the FPGA result

Fig. 8. Comparison on computation cost between conventional continual learning and the proposed acquisitive learning.

knowledge contains much more classes than new observations, we prepare inherited models that are pre-trained on 5 to 9 classes for CIFAR-10 dataset and then incrementally train the network with 1 class from the rest of dataset (Fig. 7(a)), following the techniques described in Section III. β is set as 0.5 for the inherited model in 'I5' experiment, and 0.9 for the inherited model in 'I9' experiment, and so on. Similarly, we pre-train 50 to 90 classes on the CIFAR-100 dataset and then incrementally learn 10 classes from the rest of the dataset (Fig. 7(b)).

The results of acquisitive learning starting from different inherited models are plotted in colors in Fig. 7. For CIFAR-10, with the conventional scheme, the final overall accuracy for 10 classes is 41.5%, while AL achieves 83.8% accuracy. The accuracy forgetting is 58.5% for the conventional scheme and 8.8% for AL, reducing the accuracy forgetting by $6.6\times$. For CIFAR-100, conventional scheme forgets 81.9% accuracy after learning 100 classes, while acquisitive learning forgets only 7.1% accuracy, reducing the accuracy forgetting by $11.5\times$.

E. Computation cost and FPGA prototyping

We benchmark the computation cost, including latency, number of floating-point operations (FLOPs), and energy efficiency of both the conventional scheme and the proposed AL. For AL, the computation flow of the FPGA training accelerator remains unchanged during forward pass and backward pass, which is the same as the conventional continual learning. However, during the weight update phase, the computation of gradient and weight update is only performed for the selected weights and thus, largely improving computation efficiency. The proposed learning approach was evaluated based on a FPGA training accelerator [22]. Details of the selected weights in each layer were given as an input to the accelerator. With the segmented training in AL, the FPGA control logic completely skips the DRAM access of the frozen weights thereby significantly reducing the off-chip communication and latency during the weight gradient computation. During the entire weight update phase, the frozen weights in DRAM remain untouched.

Fig. 8(a) shows the latency breakdown of ResNet-20 for Forward Pass (FP), Backward Pass (BP) and Weight Update (WU) of training, for both the conventional scheme and AL. The bar graph highlighted with blue colored text shows the latency of the AL scheme to acquire one class of CIFAR-10, while the bar with black colored text refers to a conventional scheme. Using AL, we achieve $5\times$ reduction in latency for WU phase per training image by only updating the selected weights ('AL-WU' in Fig. 8(a)), compared to the conventional scheme.

Fig. 8(b) shows the number of training FLOPs. As AL only needs to acquire a few classes with the main model segmented, the training FLOPs is largely reduced as compared to a conventional training. Learning 1 class ('99+1' scheme) and 10 classes ('90+10' scheme) from CIFAR-100 with AL reduces FLOPs by $15\times$ and $150\times$, respectively. Based on FPGA values, Table II further derives the simulated throughput (TFLOPs/s) required for training different numbers of new acquired classes, on CIFAR-10, CIFAR-100 and ImageNet [25] with ResNet-56. We assume a typical hardware platform (such as FPGA and GPU) that manages the input image stream at

TABLE II
REQUIRED THROUGHPUT (TFLOPS/SECOND) AND THE NUMBER OF HARDWARE PLATFORMS* NEEDED TO LEARN VARIOUS NUMBER OF CLASSES WITH AL.

Number of Classes	1000	500	100	50	10	5	1
CIFAR-10	-	-	-	-	2.7	2.0	1.8
CIFAR-100	-	-	2.7	2.0	1.8	1.8	1.8
ImageNet	22.0	16.6	14.8	14.7	14.7	14.7	14.7

100 platforms | 10 platforms | 2 platforms | 1 platform

*We assume that one hardware platform provides 20 GFLOPs/s/Watt with 100W [28].

30 frames/second [26], exhibits power budget of 100W [27], [28] and energy efficiency of 20 GFLOPs/second/Watt [28] per platform. As AL effectively reduces computation cost, such a platform is able to support the acquisition of as many as 50 classes with one platform for CIFAR-100, or 500 classes with 10 platforms for ImageNet.

V. CONCLUSION

In this paper, we propose a new perspective to mitigate catastrophic forgetting in continual learning: acquisitive learning (AL). Different from previous continual learning that learns from scratch and focuses only on model adaptation, AL addresses both knowledge inheritance and acquisition, inspired by the Moravec's paradox. With AL, the accuracy drop in learning sequential tasks is reduced by $6.6\times$ and $11.5\times$ for CIFAR-10 and CIFAR-100 datasets, respectively, as compared to the conventional scheme. Meanwhile, we confirm that the amount of inherited knowledge and the quality of inherited model are important to the capacity of knowledge acquisition. Furthermore, benefiting from segmented training, the weight update latency is reduced by $5\times$ as benchmarked by FPGA prototype, training FLOPs is reduced by $150\times$, enabling knowledge acquisition at the edge. In the future, we plan to investigate more criteria to select the inherited model, and techniques to automatically generate better models for more accurate and robust acquisition. We will also develop more flexible and efficient hardware techniques for the implementation of AL.

ACKNOWLEDGMENT

This work was supported in part by the Semiconductor Research Corporation (SRC) and DARPA. It was also partially supported by National Science Foundation (NSF) under CCF #1715443.

REFERENCES

[1] 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.

[2] A. Chaudhry, M. Ranzato, M. Rohrbach, and M. Elhoseiny, "Efficient lifelong learning with a-gem," *arXiv preprint arXiv:1812.00420*, 2018.

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

[4] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010.

[5] X. Du, G. Charan, F. Liu, and Y. Cao, "Single-net continual learning with progressive segmented training," in *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, Dec 2019, pp. 1629–1636.

[6] 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.

[7] A. Mallya, D. Davis, and S. Lazebnik, "Piggyback: Adapting a single network to multiple tasks by learning to mask weights," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 67–82.

[8] A. Mallya and S. Lazebnik, "Packnet: Adding multiple tasks to a single network by iterative pruning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7765–7773.

[9] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 6467–6476.

[10] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," *arXiv preprint arXiv:1708.01547*, 2017.

[11] A. M. Zador, "A critique of pure learning and what artificial neural networks can learn from animal brains," *Nature communications*, vol. 10, no. 1, pp. 1–7, 2019.

[12] M. Ingalhalikar, A. Smith, D. Parker, T. D. Satterthwaite, M. A. Elliott, K. Ruparel, H. Hakonarson, R. E. Gur, R. C. Gur, and R. Verma, "Sex differences in the structural connectome of the human brain," *Proceedings of the National Academy of Sciences*, vol. 111, no. 2, pp. 823–828, 2014.

[13] H. Moravec, *Mind children: The future of robot and human intelligence*. Harvard University Press, 1988.

[14] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[15] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," in *Advances in Neural Information Processing Systems*, 2018, pp. 6389–6399.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[18] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[19] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," *arXiv preprint arXiv:1611.06440*, 2016.

[20] X. Du, Z. Li, Y. Ma, and Y. Cao, "Efficient network construction through structural plasticity," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 9, no. 3, pp. 453–464, 2019.

[21] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[22] S. K. Venkataramanaiah, Y. Ma, S. Yin, E. Nurvithadhi, A. Dasu, Y. Cao, and J.-s. Seo, "Automatic compiler based fpga accelerator for cnn training," in *2019 29th International Conference on Field Programmable Logic and Applications (FPL)*. IEEE, 2019, pp. 166–172.

[23] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[24] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[26] X. Long, S. Hu, Y. Hu, Q. Gu, and I. Ishii, "An fpga-based ultra-high-speed object detection algorithm with multi-frame information fusion," *Sensors*, vol. 19, no. 17, p. 3707, 2019.

[27] Z. Li, C. Liu, H. Li, and Y. Chen, "Neuromorphic hardware acceleration enabled by emerging technologies," in *Emerging Technology and Architecture for Big-data Analytics*. Springer, 2017, pp. 217–244.

[28] *https://github.com/karlrupp/cpu-gpu-mic-comparison*.