

A Bayesian approach to Expert Gate Incremental Learning

*Original*

A Bayesian approach to Expert Gate Incremental Learning / Mieuli, Valerio; Ponzio, Francesco; Mascolini, Alessio; Macii, Enrico; Ficarra, Elisa; Di Cataldo, Santa. - (2021), pp. 1-7. ( 2021 International Joint Conference on Neural Networks (IJCNN) Shenzhen (China) 18-22 July 2021) [10.1109/IJCNN52387.2021.9534204].

*Availability:*

This version is available at: 11583/2898058 since: 2022-05-09T08:22:41Z

*Publisher:*

IEEE

*Published*

DOI:10.1109/IJCNN52387.2021.9534204

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

IEEE postprint/Author's Accepted Manuscript

©2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

# A Bayesian approach to Expert Gate Incremental Learning

Valerio Mieuli, Francesco Ponzio, Alessio Mascolini, Enrico Macii, Elisa Ficarra and Santa Di Cataldo  
Dept. of Control and Computer Engineering, Politecnico di Torino, Italy  
Email: name.surname@polito.it

**Abstract**—Incremental learning involves Machine Learning paradigms that dynamically adjust their previous knowledge whenever new training samples emerge. To address the problem of multi-task incremental learning without storing any samples of the previous tasks, the so-called Expert Gate paradigm was proposed, which consists of a Gate and a downstream network of task-specific CNNs, a.k.a. the Experts. The gate forwards the input to a certain expert, based on the decision made by a set of autoencoders. Unfortunately, as a CNN is intrinsically incapable of dealing with inputs of a class it was not specifically trained on, the activation of the wrong expert will invariably end into a classification error. To address this issue, we propose a probabilistic extension of the classic Expert Gate paradigm. Exploiting the prediction uncertainty estimations provided by Bayesian Convolutional Neural Networks (B-CNNs), the proposed paradigm is able to either reduce, or correct at a later stage, wrong decisions of the gate. The goodness of our approach is shown by experimental comparisons with state-of-the-art incremental learning methods.

## I. INTRODUCTION

In recent years, the extensive research carried on Deep Learning (DL), with particular emphasis on Convolutional Neural Networks (CNNs), has made this class of supervised algorithms the undisputed state-of-the-art approach to image classification tasks. Conventionally, a CNN elaborates a large dataset of images, learning how to extract relevant features to properly classify the training samples, and then applies the obtained model to unseen images. In this scenario, the classification task is well-known from the beginning, with a fixed number of pre-specified class instances. Moreover, all the training data (images and corresponding class labels) are available at the same time and can be accessed in any order during the learning.

As the field of deep learning evolves, the dynamic nature of many real-world situations, where we hardly have all data and information gathered at once, are fostering the development of more adaptable learning strategies. A peculiar example is humanoid robot vision, where a robot progressively improves its visual understanding capabilities based on a continuous interaction with humans and surroundings. In such scenario, the robot should be able to process new training examples that become available over time and to accommodate new class categories that were not originally considered, modifying the learned knowledge accordingly. On top of that, the computational and memory requirements of the learning algorithm should remain bounded. This goes by the name of *Incremental Learning* (IL).

Traditional deep architectures are intrinsically unfit to learning incrementally. When new data is presented to a CNN, the update on the net parameters affects the model

globally, typically destroying existing features learned from earlier data. This is the so-called *catastrophic forgetting* [1]. To avoid that, when training on new data, all the previous examples must be fed again to the network, and the model must be retrained from scratch on both the old and the new data. This makes the learning unfeasible, especially in applications where computational and memory resources are constrained.

A recent approach to address the problem of catastrophic forgetting in a multi-task learning scenario exploits the so-called *Expert Gate* paradigm [2, 3, 4]. In this paradigm, the classification problem is split into a number of disjoint tasks, where each task involves a certain number of classes. The system consists of two consecutive layers, respectively the *Gate* and the *Experts*. The experts are task-specific models (typically, CNNs), one per each classification task. At inference time, the gate decides which expert should be activated on the input image, and the chosen expert provides the final class. The learning is incremental, in that the architecture can accommodate new classification tasks added at a later time, without being retrained on the earlier examples: a new expert trained on the new training data can be sequentially added to the experts layer, and the gate layer can be extended accordingly [4].

While the Expert Gate strategy has shown advantages over alternate incremental solutions in terms of both performance and computational requirements [4], it also has an inherent weakness. As it is universally acknowledged, a CNN is intrinsically incapable of dealing with inputs of a class it was not specifically trained on. Hence, the activation of the wrong expert will invariably end into a classification error, no matter how well the experts were designed and trained.

To address this issue, in this work we propose a probabilistic extension of the classic Expert Gate paradigm [4], leveraging Bayesian Convolutional Neural Networks (B-CNNs). Differently from their deterministic counterparts, B-CNNs provide a statistically significant estimation of the level of uncertainty of the classification outcome. In a probabilistic *Expert Gate* paradigm, this inherent capability can be exploited at two different levels: (i) to improve the robustness of the gate, reducing the probability of activating the wrong expert; (ii) to make the experts identify, and possibly correct, wrong decisions of the gate. To demonstrate the goodness of our approach, the two incremental strategies (with *Bayesian Gate* and with *Bayesian Experts*, respectively) are compared with their state-of-the-art deterministic counterparts on two state-of-

the-art public benchmarks: ImageNet Large Scale Visual Recognition Challenge (ILSVRC 2012) and CIFAR-100 data.

The rest of the paper is structured as follows. Section II presents the background of incremental learning and Bayesian networks, that are the backbone of our solution. Section III describes the proposed methodologies. Section IV presents and discusses the experimental results on the CIFAR and ImageNet datasets. Finally, Section V concludes the paper and presents future works.

## II. BACKGROUND

### A. Incremental learning

IL approaches can be categorized into three main groups [3]:

- 1) *Replay based methods* store samples from the previous tasks, either in their raw format [5] or in the form of pseudo-samples obtained by a generative model [6], and replay them while learning a new task to avoid catastrophic forgetting. These samples/pseudo-samples can be either used for rehearsal, which implies the joint training of previous and current tasks, or to constrain the optimization of the model on the new data [7]. Most frequent issues are typically two. Methods storing previous samples typically require unconstrained memory resources. On the other hand, pseudo-samples obtained with generative models may suffer from approximation errors.
- 2) *Regularization-based methods* do not require the storage of previous examples, as they try to avoid catastrophic forgetting by adding some extra regularization term in the loss function of the new data, with the aim of consolidating features learnt on the previous tasks [8, 9]. Main issue with this category of methods is typically finding a good trade-off between the optimization of the current and earlier information, which is increasingly difficult at larger number of tasks.
- 3) *Parameter isolation-based methods* dedicate a specific subset of the model parameters to each independent task in order to completely prevent catastrophic forgetting without storing any previous sample [10, 4]. This can be done by either extending the network each time a new task is added while masking the weight updates, by creating new independent branches for new tasks, or by making a complete copy of the model all-together [3].

In this work we focus on a parameter isolation-based method, the Expert Gate, that differently from other approaches in the same category does not require any previous knowledge of the task at inference time [3]. Fig. 1 shows a schematic representation of the implementation proposed by [4], that is taken as a reference.

As anticipated in Section I, it is made of two consecutive layers, the gate and the network of experts. The gate consists of a set of task-specific autoencoders, and the experts of a set of task-specific classifiers (more specifically, CNNs). During the training, each autoencoder is trained

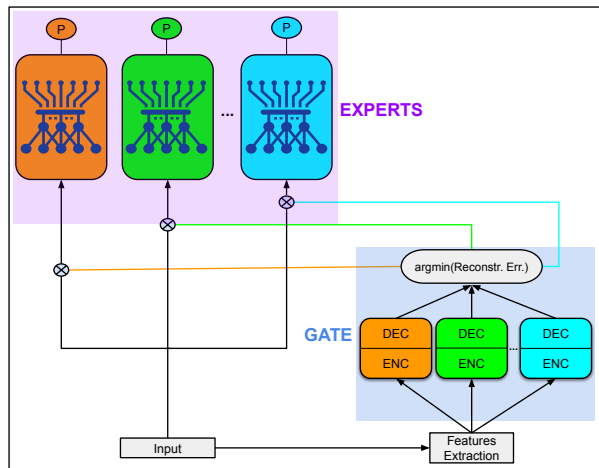


Fig. 1. Expert Gate architecture.

simultaneously with the corresponding expert, that is in charge of the class prediction ( $P$ , in Figure 1). By ensuring that the autoencoders are independent from each other, one can assume that the lower the reconstruction error of the autoencoder, the more similar the input image is to the ones the autoencoder was trained on. Hence, at inference time, the gate decides on which expert to activate based on the autoencoder (and hence, the task) that obtained the minimum reconstruction error. This allows to identify the task and to choose the CNN with the highest chance to predict the correct class.

While this approach was demonstrated to outperform other incremental learning techniques on a number of image classification datasets [4], the deterministic nature of the experts makes them unfit to recognize wrong decisions taken by the gate [11]. On the other hand, the success of the overall method is completely dependent on the gate, and hence on the ability of the autoencoders to properly reconstruct the input image, which is indeed a weakness of the paradigm.

In this work, we try to address this inherent weakness of the Expert Gate architecture by making the involved architectures able to provide a reliable measure of how confident they are about their decisions. For this purpose, our solution is built on top of Bayesian neural networks.

### B. Bayesian Convolutional Neural Networks

Recently, researchers have shown an increased interest in quantifying uncertainty for DL predictions. Speaking about CNNs, the traditional softmax probability is based on a single set of network parameters, and hence prone to be overconfident [12]. This makes traditional networks incapable of communicating their uncertainty about a prediction. Conversely, a fully probabilistic treatment would consider a distribution over network parameters instead of a point estimate, providing a probability distribution also over predictions. This *predictive distribution* can be obtained by integrating over all possible parameters settings for the model, and exploited to communicate uncertainty.

Bayesian probability theory offers a mathematically well-founded system to investigate model uncertainty. Let  $X =$

$\{x_1, x_2, \dots, x_N\}$  be our training set of  $N$  input samples and  $Y = \{y_1, y_2, \dots, y_N\}$  their matching output labels. We want to approximate a function from our observations  $y = f(x)$ , able to generalize the data. We define a prior distribution over the space of functions  $p(f)$ , expressing a prior belief about which functions are more or less likely with respect to the observed data. The posterior distribution over the space of functions, given our dataset  $(X, Y)$ , can be written as

$$p(f|X, Y) \propto p(Y|X, f)p(f) \quad (1)$$

We can now consider a CNN to put (1) into effect, assuming the net to be completely described by a finite set of random variables  $\omega \in \Omega$ . Here  $\Omega$  is the space of all possible bias and weights parameters. In a classification fashion, we are interested in the predictive distribution for a new input  $x^*$ , given by

$$p(y^*|x^*, X, Y) = \int_{\Omega} p(y^*|x^*, \omega)p(\omega|X, Y)d\omega, \quad (2)$$

where  $y^*$  is the predicted label. As it can be gathered from (2), the integration of  $p(\omega|X, Y)$  with respect to the whole parameters space  $\Omega$ , makes the predictive posterior of a CNN really hard to be analytically computed. To overcome this limitation, MacKay introduced the Laplace approximation to the posterior computation, with the drawback of a introducing a poorly reliable approximation [13]; on the other hand, Neal proposed the Markov chain Monte Carlo method to sample the posterior distribution without directly computing it, but with a prohibitive computational cost [14].

More recent studies proposed to find approximating solutions for (2) via *variational inference* [15]. In this sense, the Bayesian posterior is approximated by the variational distribution  $q_{\theta}(\omega)$ , defined by a variational parameters  $\theta$ . The optimal variational distribution among the family  $Q = \{q_{\theta}(\omega)\}$  is the one closest to the posterior, where closeness is evaluated in terms of the Kullback-Leibler (KL) divergence between  $q_{\theta}(\omega)$  and  $p(\omega|X, Y)$

$$KL\{q_{\theta}(\omega)||p(\omega|X, Y)\} = \int_{\Omega} q_{\theta}(\omega) \log \frac{q_{\theta}(\omega)}{p(\omega|X, Y)} d\omega \quad (3)$$

Minimizing KL divergence is known to be equivalent to maximizing the so-called evidence lower bound (ELBO) [16], given by

$$\int_{\Omega} q_{\theta}(\omega) \log p(y|x, \omega) d\omega - KL\{q_{\theta}(\omega)||p(\omega)\} \quad (4)$$

Maximizing (4) with respect to the approximating distribution  $q_{\theta}(\omega)$  produces two different effects. The first term maximizes the likelihood of the training data, and it is, as in traditional CNNs, a model fit term. The second term takes care of approximating the true distribution by the variational one.

More recently, a key insight from Gal and Ghahramani demonstrated that the KL term in (4) corresponds exactly to a L2-regularization term in dropout networks [12]. It follows that obtaining model uncertainty for a given image is as simple as keeping the dropout mechanism switched on at inference time and performing multiple predictions for

the same input. This method has been referred in literature as *Monte Carlo (MC) dropout*.

In a later work by [17], uncertainty was decomposed into two main components: *aleatoric* uncertainty, which captures the noise of the observation, and *epistemic* uncertainty, which stems from the model's parameters and architecture. It follows that the total uncertainty of a prediction can be measured by averaging the results over a number of stochastic forward passes of the inputs through the model, which is also the approach used for our implementation.

While the theory of Bayesian inference and uncertainty estimation for DL models is well-established, the link with the incremental paradigm has received to date very little attention. In our work, uncertainty estimations in Bayesian models with MC dropout are exploited to improve the accuracy and robustness of the Expert Gate incremental architecture.

### III. METHODS

Our work addresses multi-task incremental learning, building upon the Expert Gate paradigm shown in Figure 1. As anticipated in Section I, this strategy has a weakness: as a consequence of the deterministic nature of all the involved models, a wrong decision of the gate will activate an expert that is inherently incapable of identifying the correct class for the given input, ending into a classification error.

In our probabilistic version of the Expert Gate paradigm, we exploit Bayesian deep models, that are able to communicate a measure of their prediction uncertainty on a given input. This can be exploited at two different levels:

- 1) at the *Gate* level, as a substitute of the reconstruction error of the autoencoders. The rationale of this approach, referred to as *Bayesian Gate*, is that the gate should activate the task that is identified with the lowest level of uncertainty. This will possibly improve the robustness of the task identification, and hence the chances of a correct classification.
- 2) at the *Experts* level, to make the task-specific classifiers able to identify at inference images of a task they were not specifically trained for. This approach, referred to as *Bayesian Experts*, can be exploited to correct wrong decisions of the gate at a later stage, possibly improving the classification performance.

In the following, we describe the implementation details of the two strategies.

#### A. Bayesian Gate strategy

The first methodology is schematically represented in Fig. 2(a). This solution differs from the one in Fig. 1 because the gate consists of a set of task-specific B-CNNs instead of autoencoders.

To design each B-CNN we put [12] into effect, leveraging MC dropout both during training and inference. More specifically, we used a VGG16 model pre-trained on ImageNet and inserted a Dropout layer with a 0.25 rate after each convolutional, pooling and fully connected layer. Our choice stems from the necessity of obtaining a robust

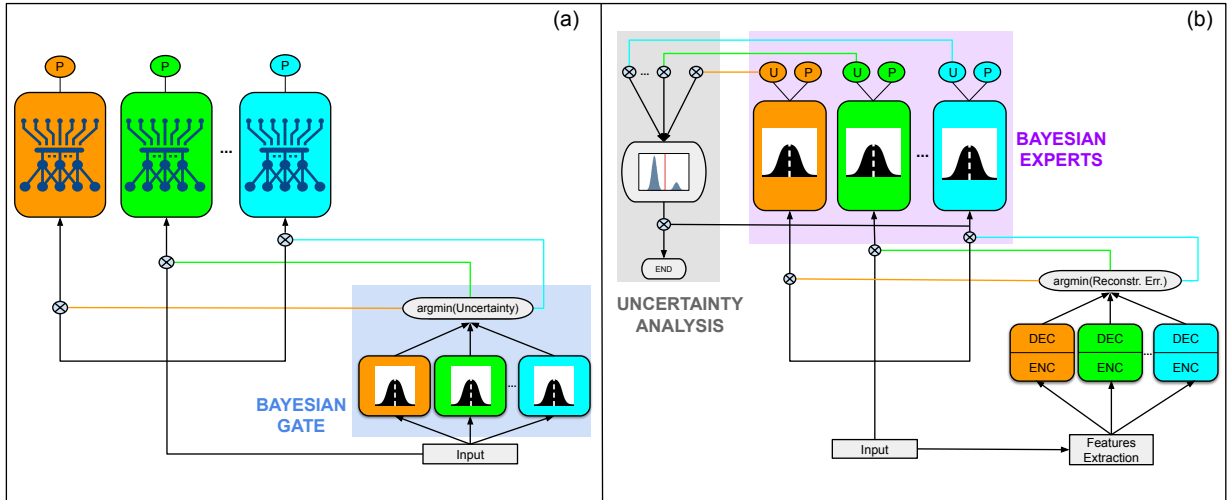


Fig. 2. Proposed Expert Gate solutions (a) with Bayesian Gate (b) with Bayesian Experts.

image representation with a limited number of iterations: indeed, from our preliminary experiments we found that the weights initialization on the ImageNet makes the training of the B-CNN much more straightforward and faster than a standard random initialization.

Before training each model on the dataset of the corresponding task, we pre-processed the samples by zero-centred normalization. ImageNet is known to be a good approximation of the distribution of general purpose images. Hence, we used the statistics based on ImageNet to perform the normalization, as in [4].

Each task-specific B-CNN was trained with AdaGrad optimizer [18], setting  $\epsilon$  parameter to  $10^{-8}$ , weight decay to 0.005 and learning rate to 0.001. As a result, seen  $N$  tasks, we obtain a gate made of  $N$  Bayesian decision-makers (see the schematic representation in Fig. 2(a)).

At inference time, the input image is given to all the  $N$  Bayesian models to compute the respective uncertainty values. Finally, a decision is made based on the task that provided the lowest uncertainty value. Based on this decision, the input is forwarded to the corresponding expert.

To obtain a measure of the uncertainty, we do as follows. As anticipated in Section II-B, the predictive uncertainty of a B-CNN may be worked out as the sum of the predictive variances of each class [19], which can be decomposed into the aleatoric component, able to model the noise of the observation, and the epistemic component, which comes out of the model's parameters and architecture:

$$\underbrace{\frac{1}{T} \sum_{t=1}^T \text{diag}(\hat{p}_t) - \hat{p}_t^{\otimes 2}}_{\text{aleatoric}} + \underbrace{\frac{1}{T} \sum_{t=1}^T (\hat{p}_t - \bar{p})^{\otimes 2}}_{\text{epistemic}} \quad (5)$$

Here  $\bar{p} = \sum_{t=1}^T \hat{p}_t / T$ ;  $\hat{p} = \text{Softmax}f(\omega_t, x^*)$  and  $T$  is the number of forward passes for input  $x^*$ .  $T$  has been empirically set to 100 as the best trade-off between

computational time and reliability of the uncertainty value for the given sample  $x^*$ .

As it can be gathered from Fig. 2(a), the downstream stage of our architecture consists of a network of  $N$  task-specific deterministic experts, as in the classic Expert Gate paradigm. In our solution, each expert is a VGG16 model designed and trained with the very same procedure of the gate, with the only difference of having set the MC dropout layers to zero.

### B. Bayesian Experts strategy

Our second methodology is schematically represented in Fig. 2(b). In this case, the gate includes a set of unregularized one-layer under-complete autoencoders, just like in the classic Expert Gate architecture. For each task, the corresponding autoencoder is trained only on the task-specific data, using the mean squared error criterion as loss function. As shown in [20, 4], this estimates the negative log-likelihood.

At inference time, the reconstruction error  $er_j$  of the  $j$ -th autoencoder is the output of the loss function for the input sample  $x^*$ . A softmax layer receives the reconstruction errors of all the autoencoders for the same input  $x^*$ , and returns a probability value  $p_j$  per each task, computed as follows:

$$p_j = \frac{\exp(-er_j/t)}{\sum_j^N \exp(-er_j/t)}, \quad (6)$$

where  $t$  is the so-called *temperature*, set to 2 as in [4].

Based on equation (6), the gate decides which task-specific expert should be activated. That is, the lower the reconstruction error of the  $j$ -th autoencoder on the input sample  $x^*$ , the higher the probability of activating the  $j$ -th expert.

Differently from the standard Expert Gate architecture, in our approach the downstream network of task-specific classifiers consists of B-CNNs, whose implementation is

similar to the one described in Section III-A. Each expert provides a measure of prediction uncertainty on the input sample ( $U$ , in Fig. 2(b)). This uncertainty is exploited to identify input samples that were inappropriately assigned to that expert, as follows: the uncertainty value  $U_j$  returned by the  $j$ -th expert is compared with an *uncertainty threshold*  $Tu_j$ ; If  $U_j < Tu_j$ , the gate’s decision of activating the expert  $j$  is considered reliable, and the corresponding prediction  $P_j$  is accepted. Otherwise, the input sample  $x^*$  is forwarded back to all the other experts, and the final prediction will be the one provided by the expert with the *lowest* uncertainty value. This procedure is represented by the *Uncertainty Analysis* block in Fig. 2(b).

For a generic task  $j$ , the uncertainty threshold  $Tu_j$  is computed as follows. First, we put equation (5) into effect on the training set of task  $j$ . By doing so, we obtain a distribution of uncertainty values that is typically bimodal, as already observed in previous literature [21] (see the histogram of Fig. 3). In this histogram, the first mode (with the highest peak) is associated to samples classified with a high level of confidence. Conversely, the second mode is associated to low-confidence predictions: this may be due to noise, bad cropping, bad scaling of the corresponding samples, but can also identify samples of a different task.

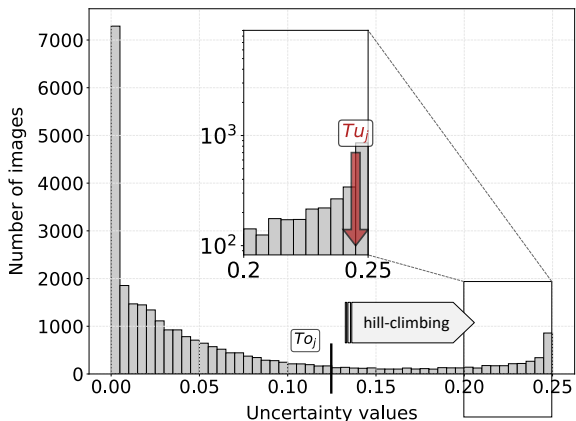


Fig. 3. Computation of the uncertainty threshold.

Starting from this consideration, finding the optimal value of  $Tu_j$  can be approached as if it was a histogram thresholding problem, with following two-steps.

- 1) First, we apply Otsu thresholding algorithm. By doing so, we obtain the uncertainty value  $To_j$  that splits the histogram into two groups with maximum inter-group variance;
- 2) Starting from  $To_j$ , we apply a hill-climbing approach in the direction of growing uncertainty, and stop the search in the point of maximum slope. This corresponds to the final value  $Tu_j$  (see Fig. 3).

#### IV. EXPERIMENTAL RESULTS

In this section we present the experimental validation of our Bayesian incremental IL solutions (Fig. 2) compared to the baseline *Expert Gate* (Fig. 1).

To do so, we compare the classification accuracy of the task-incremental learning methods on the same classifica-

tion tasks, adopting the validation protocol suggested by [5].

- 1) Given a multi-class classification dataset, the available classes are randomly split into  $N$  different tasks, each including a disjoint sub-set of the original classes.
- 2) Each method is trained in a task-incremental way on the corresponding training data. That is, first on task 1 alone, then adding task 2, task 3, etc. up-to task  $N$ .
- 3) Downstream the training phase for the  $j$ -th task, the resulting classifier is evaluated on the test set made up of only the classes of tasks 1 to  $j$ , for which the model has been trained on.

As a result of the validation protocol, for each incremental approach we obtain the mean accuracy of the classifier in a task-by-task fashion.

As introduced in Section II-A, both our probabilistic solutions and the corresponding deterministic baseline belong to the category of the parameter isolation-based methods. To better contextualize our validation, in the following we also provide results of the most representative algorithms of the other two categories: respectively, iCaRL [5] for replay-based methods and LwF-MC [8] for regularization-based methods.

This protocol has been put into effect on two well-established public benchmarks.

##### A. ImageNet benchmark

For the first set of experiments, we used the ImageNet Large Scale Visual Recognition Challenge (ILSVRC 2012). It includes 1,281,167 images for training and 50,000 images for validation. The 1000 available classes were split into 10 incremental tasks of 100 classes each.

The obtained results are shown in Fig. 4, where the y-axis reports the mean accuracy of the classifier, and the x-axis the corresponding number of classes on which this accuracy was computed. Different lines and markers are associated to our proposed solutions (respectively, the one with Bayesian Experts and the one with Bayesian Gate) and to the baseline Expert Gate architecture. Dashed lines refer to iCaRL and LwF-MC, which belong to different categories of IL methods.

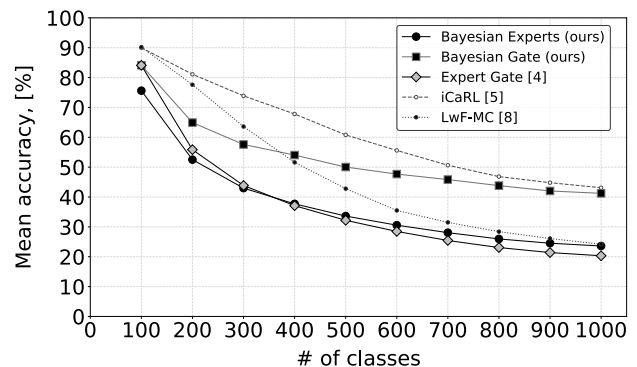


Fig. 4. Task-incremental classification accuracy on ImageNet, at increasing number of classes seen by the model.

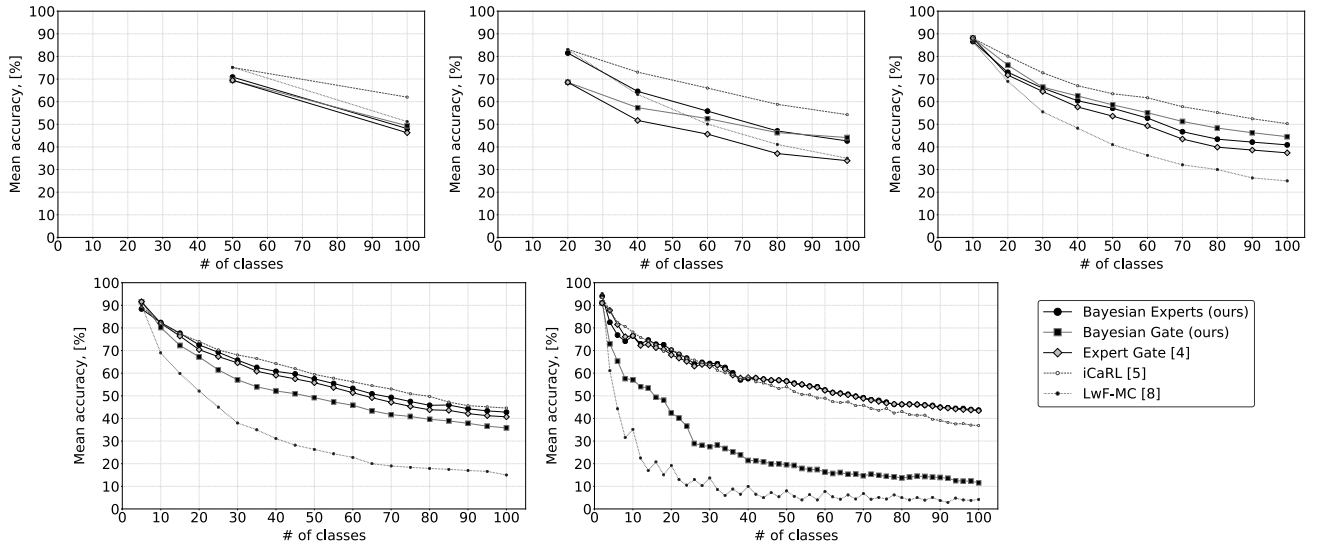


Fig. 5. Task-incremental classification accuracy on CIFAR-100, at increasing number of classes seen by the model. Graphs from left to right and from top to bottom show experiments with with 2, 5, 10, 20 and 50 tasks, respectively.

As it can be observed from the plot, for all the approaches, the mean classification accuracy decreases in a similar-exponential way at increasing number of classes shown to the model. This is consistent with previous literature.

iCaRL overcomes all the other IL solutions: this was expected, as the storage of previous training samples confers a superior capability to defeat catastrophic forgetting [3]. Nonetheless, as already discussed, this strategy suffers from well-known memory limitations compared to the other IL categories.

The proposed Bayesian approaches generally outperform the corresponding deterministic Expert Gate method. The strategy with Bayesian Experts has the lowest improvement (around 4%, but only at higher incremental batches). On the other hand, the Bayesian Gate outperforms the deterministic baseline by around 20%, with a slower decreasing trend at increasing number of classes seen by the model. At higher number of classes, the Bayesian Gate becomes comparable with iCaRL in terms of accuracy and outperforms LwF-MC by around 17%.

### B. CIFAR-100 benchmark

To investigate the accuracy of the incremental approaches at varying experimental conditions, we exploited a smaller dataset, the CIFAR-100 benchmark. It consists of 60,000 32x32 colour images in 100 different classes, with 600 images per class (respectively, 500 for training and 100 for testing purposes).

For this second set of experiments, we repeated the validation protocol at increasing number of tasks: respectively 2, 5, 10, 20 and 50. That is, first we split the 100-classes dataset into 2 tasks of 50 classes each, then 5 of 20 classes each, and so on, up-to a last configuration of 50 binary classification tasks.

Fig. 5 shows the overall results of these experiments. Each plot represents the outcome of the validation at increasing number of tasks (respectively, the top-left plot shows the

experiment with 2 tasks and the bottom-right the one with 50 tasks), where the total number of tasks corresponds to the total number of values that are displayed in the graph.

From the analysis of the obtained results, the following considerations can be made.

- Again, the mean classification accuracy decreases in a similar-exponential way at increasing number of classes shown to the model.
- Again, iCaRL overcomes the accuracy of the other non-replay-based IL strategies. Nonetheless, its performance sensibly decreases in experiments with higher number of tasks.
- When considering experiments with a small number of multi-class tasks, (see first three plots of Fig. 5), the Bayesian approach generally outperforms both the deterministic Expert Gate and LwF-MC. The configuration which shows the best improvement is the one with a total number of 10 tasks, of 10 classes each. In this configuration (third plot), both our Bayesian solutions overcome the other non-replay-based methods. In particular, the Bayesian Gate improves LwF-MC by 15%-20%. This is consistent with the analogous experiment on the ImageNet dataset, where the number of tasks was exactly the same.
- When the total number of tasks is higher (and, conversely, the number of classes per task is small), the two Bayesian strategies behave very differently (see last two plots of Fig. 5). The approach with Bayesian Experts is the one with the highest performance, but with progressively decreased improvement over the deterministic Expert Gate baseline. As regard to LwF-MC, our Bayesian Experts improves by more than 20% and 30%, respectively for experiments with 20 and 50 tasks (last two plots of Fig. 5). In the most extreme configuration (50 tasks of 2 classes each, last plot), the accuracy of the Bayesian Experts is the same as the deterministic Expert Gate, but overcomes iCaRL by almost 10%. On the other hand,

the approach with Bayesian Gate is the one with the worst performance, with accuracy degrading quickly at increasing number of tasks. A possible explanation is that a B-CNN trained on a smaller task tends to be less uncertain of its predictions, and hence the uncertainty level becomes less discriminative in this case.

When considering different numbers of classes per task, the strategy with Bayesian Experts is to be preferred: at best, it compensates wrong decisions of the gate, improving the overall performance of the incremental learner; at worst, it is as good as the corresponding deterministic method.

## V. CONCLUSIONS AND FUTURE WORKS

As demonstrated by our experiments, in a Bayesian Expert Gate paradigm, prediction uncertainty can be exploited to make the gate more robust, as well as to identify and correct wrong decisions of the gate at the experts level. Future works will focus on (i) improving the robustness of the Bayesian Gate for small classification tasks. This includes the exploration of adaptive dropout policies, where the dropout rate (and hence, the probabilistic behaviour of the model) is automatically adapted to the number of classes; (ii) integrating the strategies with Bayesian Gate and Bayesian Experts; (ii) experimenting on different deep architectures as base-learners.

## REFERENCES

- [1] Ian J. Goodfellow et al. *An Empirical Investigation of Catastrophic Forgetting in Gradient-Based Neural Networks*. 2013. arXiv: 1312.6211 [stat.ML].
- [2] Iasonas Kokkinos. “Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 6129–6138.
- [3] Matthias De Lange et al. *Continual learning: A comparative study on how to defy forgetting in classification tasks*. 2019. arXiv: 1909.08383 [cs.CV].
- [4] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. “Expert gate: Lifelong learning with a network of experts”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 3366–3375.
- [5] Sylvestre-Alvise Rebuffi et al. “icarl: Incremental classifier and representation learning”. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2017, pp. 2001–2010.
- [6] Hanul Shin et al. “Continual learning with deep generative replay”. In: *Advances in Neural Information Processing Systems*. 2017, pp. 2990–2999.
- [7] David Lopez-Paz and Marc’Aurelio Ranzato. “Gradient episodic memory for continual learning”. In: *Advances in neural information processing systems*. 2017, pp. 6467–6476.
- [8] Zhizhong Li and Derek Hoiem. “Learning without forgetting”. In: *IEEE transactions on pattern analysis and machine intelligence* 40.12 (2017), pp. 2935–2947.
- [9] James Kirkpatrick et al. “Overcoming catastrophic forgetting in neural networks”. In: *Proceedings of the national academy of sciences* 114.13 (2017), pp. 3521–3526.
- [10] Chrisantha Fernando et al. *PathNet: Evolution Channels Gradient Descent in Super Neural Networks*. 2017. arXiv: 1701.08734 [cs.NE].
- [11] Anh Nguyen, Jason Yosinski, and Jeff Clune. “Deep neural networks are easily fooled: High confidence predictions for unrecognizable images”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 427–436.
- [12] Yarin Gal and Zoubin Ghahramani. “Dropout as a bayesian approximation: Representing model uncertainty in deep learning”. In: *international conference on machine learning*. 2016, pp. 1050–1059.
- [13] David JC MacKay. “A practical Bayesian framework for backpropagation networks”. In: *Neural computation* 4.3 (1992), pp. 448–472.
- [14] Radford M Neal. “Bayesian learning via stochastic dynamics”. In: *Advances in neural information processing systems*. 1993, pp. 475–482.
- [15] Alex Graves. “Practical variational inference for neural networks”. In: *Advances in neural information processing systems*. 2011, pp. 2348–2356.
- [16] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. “Variational inference: A review for statisticians”. In: *Journal of the American Statistical Association* 112.518 (2017), pp. 859–877.
- [17] Alex Kendall and Yarin Gal. “What uncertainties do we need in bayesian deep learning for computer vision?” In: *Advances in neural information processing systems*. 2017, pp. 5574–5584.
- [18] John Duchi, Elad Hazan, and Yoram Singer. “Adaptive subgradient methods for online learning and stochastic optimization”. In: *Journal of machine learning research* 12.Jul (2011), pp. 2121–2159.
- [19] Yongchan Kwon et al. “Uncertainty quantification using bayesian neural networks in classification: Application to biomedical image segmentation”. In: *Computational Statistics & Data Analysis* 142 (2020), p. 106816.
- [20] Yoshua Bengio et al. “Learning deep architectures for AI”. In: *Foundations and trends® in Machine Learning* 2.1 (2009), pp. 1–127.
- [21] Jan M Köhler, Maximilian Autenrieth, and William H Beluch. “Uncertainty Based Detection and Re-labeling of Noisy Image Labels.” In: *CVPR Workshops*. 2019, pp. 33–37.