Impatient DNNs – Deep Neural Networks with Dynamic Time Budgets

Manuel Amthor
manuel.amthor@uni-jena.de
Erik Rodner
erik.rodner@uni-jena.de
Joachim Denzler
joachim.denzler@uni-jena.de

Computer Vision Group
Friedrich Schiller University Jena
Germany
www.inf-cv.uni-jena.de

Abstract We propose Impatient Deep Neural Networks (DNNs) which deal with dynamic time budgets during application. They allow for individual budgets given a priori for each test example and for anytime prediction, *i.e.* a possible interruption at multiple stages during inference while still providing output estimates. Our approach can therefore tackle the computational costs and energy demands of DNNs in an adaptive manner, a property essential for real-time applications.

Our Impatient DNNs are based on a new general framework of learning dynamic budget predictors using risk minimization, which can be applied to current DNN architectures by adding early prediction and additional loss layers. A key aspect of our method is that all of the intermediate predictors are learned jointly. In experiments, we evaluate our approach for different budget distributions, architectures, and datasets. Our results show a significant gain in expected accuracy compared to common baselines.

Learning Dynamic Budget Predictors In our paper, we develop a framework for learning dynamic budget predictors using risk minimization. We consider inference algorithms f providing predictions $y \in \mathcal{Y}$ for input examples $\boldsymbol{x} \in \mathcal{X}$ at different times $t \in \mathbb{R}$, *i.e.* we have $f : \mathcal{X} \times \mathbb{R} \to \mathcal{Y}$.

Learning the parameters $\boldsymbol{\theta}$ of f is done by minimizing the following regularized risk:

$$\operatorname{argmin}_{\boldsymbol{\theta}} \int_{t\in\mathbb{R}} \int_{y\in\mathcal{Y}} \int_{\boldsymbol{x}\in\mathcal{X}} \mathcal{L}(f(\boldsymbol{x},t;\boldsymbol{\theta}),y) \cdot p(\boldsymbol{x},y,t)\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}y\,\mathrm{d}t + \mathcal{R}(\boldsymbol{\theta}) \quad ,$$

with $\mathcal{L}$ being a suitable loss function, $\mathcal{R}(\boldsymbol{\theta})$ being a regularization term, and $p(\boldsymbol{x},y,t)$ being the joint distribution of an input-output pair $(\boldsymbol{x},y)$ and the available time t.

Our framework leads to a very flexible and simple learning scheme for deep neural networks. A deep neural network with additional prediction layers is well suited for providing a series of prediction models due to its layered architecture. In particular, the resulting architecture of our networks as seen in Figure 2 contains "early prediction layers" directly connected to loss layers. The parameters of all of the layers are learned jointly by minimizing a weighted combination of the loss layers with standard back-propagation and the usual tricks of the trade. The weights used are directly computed using the distribution of time budgets specific for the application.

Experiments and Evaluation We present experimental results for different architectures, such as AlexNet [2] and VGG19 [3], on various object classification datasets to answer the most interesting question: Does our joint training scheme provide superior results compared to learning predictors independently? We compare our approach with different baselines that learn several SVM classifiers based on extracted CNN features [1] at each early prediction layer using an original CNN pre-trained on ImageNet (ORIG) and a pre-trained CNN fine-tuned on the current dataset (FT).

Our joint learning of early prediction layers provides superior results for almost all time budget distributions (*cf*. Table 1). Especially in the case of small time budgets our method benefits from taking the budget distribution during learning into account resulting in an improvement of almost 10% on MIT-67 for an Impatient VGG19 compared to the best performing baseline. The last two columns show the expected test time of a single image for an anytime scenario t_A and a priori known time budget t_B. Please note the significant reduction in inference time compared to a standard VGG19 CNN with a test time of 1.9 ms.

BUDGET SCHEME	ORIG	FT	OURS	$\varnothing t_\mathrm{B}$ [ms]	$\varnothing t_\mathrm{A}$ [ms]
⎯ uniformly distributed time budgets	46.65	48.07	**53.93**	1.11	1.19
◞ large time budgets are likely	62.82	67.07	**69.66**	1.72	1.84
◟ small time budgets are likely	25.63	25.65	**35.11**	0.50	0.51
∩ normal distributed time budgets	47.53	47.90	**55.38**	1.07	1.15

Table 1: Comparison of an Impatient VGG19 with several baselines on MIT-67. Performance is measured by expected accuracy in % based on the particular budget distribution.

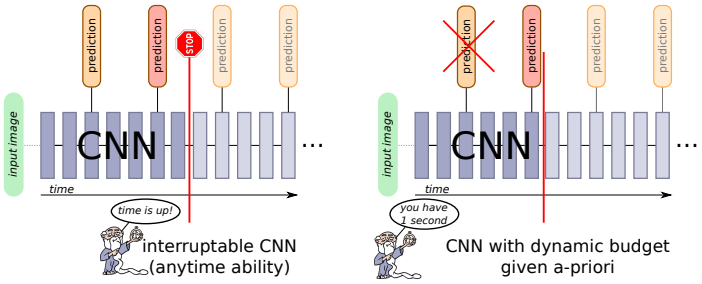

Figure 1: Illustration of convolutional neural network prediction in dynamic budget scenarios: (left) prediction can be interrupted at **any time** or (right) the budget is given **before** each prediction.

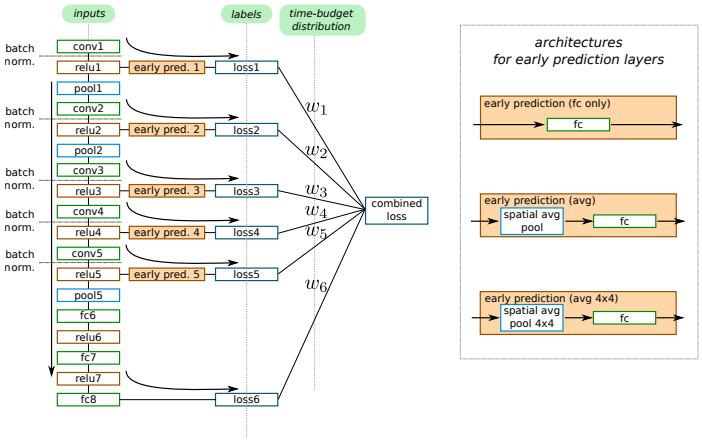

Figure 2: (Left) Modification of AlexNet for dynamic budgets and early predictions. (Right) Possible architectures for early prediction.

Summary In our paper, we present a novel approach for anytime prediction with deep neural networks which can easily be adapted directly to state-of-the-art convolutional neural network architectures. Joint training with weighted losses provides superior results for different time budget distributions compared to independently trained early predictors. Furthermore, we show that the idea of early prediction layers allows for reducing computational costs in the case of being already certain about intermediate classification results.

[1] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013.

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[3] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.