Curriculum Learning

Yoshua Bengio[1], Jérôme Louradour[1], Ronan Collobert[2] and Jason Weston[2]
(1) Dept. IRO, U. Montreal, (2) NEC Laboratories America

Abstract. Humans and animals learn much better when the examples are not randomly presented but organized in a meaningful order which illustrates gradually more concepts, and gradually more complex ones. Here, we formalize such training strategies in the context of machine learning, and call them "curriculum learning". In the context of recent research studying the difficulty of training in the presence of non-convex training criteria (for deep deterministic and stochastic neural networks), we explore curriculum learning in various set-ups. The experiments show that significant improvements in generalization can be achieved. We hypothesize that curriculum learning has both an effect on the speed of convergence of the training process to a minimum and, in the case of non-convex criteria, on the quality of the local minima obtained: curriculum learning can be seen as a particular form of continuation method (a general strategy for global optimization of non-convex functions).

Introduction. By choosing which examples to present and in which order to present them to the learning system, one can *guide* training and remarkably increase the speed at which learning can occur. This idea is routinely exploited in *animal training* where it is called **shaping** (Skinner, 1958).

Previous research (Elman, 1993; Rohde and Plaut, 1999) at the intersection of cognitive science and machine learning has raised the following question: can machine learning algorithms benefit from a similar training strategy? The idea of training a learning machine with a curriculum can be traced back at least to (Elman, 1993). The basic idea is to *start small*, learn easier aspects of the task or easier sub-tasks, and then gradually increase the difficulty level. Such conclusions are important for developmental psychology, because they illustrate the adaptive value of starting, as human infants do, with a simpler initial state, and then building on that to develop more and more sophisticated representations of structure. (Elman, 1993) makes the statement that this strategy could make it possible for humans to learn what might otherwise prove to be unlearnable. However, these conclusions have been seriously questioned in (Rohde and Plaut, 1999). These authors did a similar set of learning simulations in which pseudo-languages were learned better without a curriculum than with a curriculum.

Therefore it remains to be shown clearly when a learning algorithm can benefit (if at all) from a curriculum or "starting small" strategy. Here we contribute to this question by showing several cases - involving vision and language tasks - in which very simple multi-stage curriculum strategies give rise to improved generalization and faster convergence. We also contribute with the introduction of a hypothesis which may help to explain some of the advantages of a curriculum strateg: it can act as a continuation method (Allgower and Georg, 1980), i.e., help to find better local minima of a non-convex training criterion. In addition, the experiments reported here suggest that (like other strategies recently proposed to train deep deterministic or stochastic neural networks) the curriculum strategies appear on the surface to operate like a regularizer, i.e., their beneficial effect is most pronounced on the test set. Furthermore, experiments on convex criteria also show that a curriculum strategy can speed the convergence of training towards the global minimum.

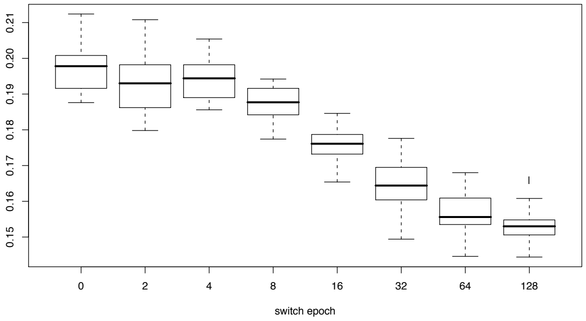
At an abstract level, a curriculum can also be seen as a sequence of training criteria, each being associated with a different set of weights on the training examples, or more generally, on a reweighting of the training distribution. Initially, the weights favor the "easiest" examples, or examples illustrating the simplest concepts, that can be learned most easily. The next training criterion involves a slight change in the weights that increases the probability of sampling slightly more difficult examples. At the end of the sequence, the reweighting of the examples is uniform and we train on the target training set or the target training distribution.

To test the hypothesis that a curriculum strategy could help to find better local minima of a highly non-convex criterion, we consider in particular deep architectures (Hinton, Osindero and Teh, 2006; Bengio, 2009), which have been shown to involve good local minima that are almost impossible to find by random initialization (Erhan et al., 2009), but can be found with unsupervised pre-training, i.e. unsupervised learning at each layer, to guide training and initialize it much better than with marginally random initialiation of the parameters.

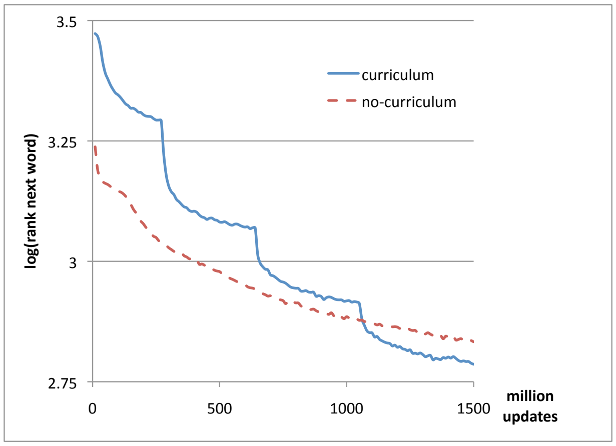
Experimental results. The experiments presented here suggest that pre-training with a curriculum strategy might act similarly to unsupervised pre-training, acting both as a way to find better local minima and as a regularizer.

They also suggest that they help to reach faster convergence to a minimum of the training criterion. The results are the following. **(1) Cleaner Examples May Yield Better Generalization Faster.** A simple toy experiment with Perceptrons (i.e. online learning) shows that training only with "easy" examples that are less noisy yields faster convergence of generalization error. **(2) Introducing Gradually More Difficult Examples Speeds-up Online Training.** In another toy experiment with Perceptrons, examples are presented from from the easiest ones to the more difficult ones, where "easy" here means less irrelevant inputs. Statistically significant improvements in generalization error after a fixed number of training examples (i.e. updates) is obtained using this simple curriculum strategy (vs no curriculum).

(3) Deep Architecture Trained by Recognizing Gradually more Varied Geometric Shapes. We train a deep neural network (3 hidden layers) using a 2-stage curriculum to classify 3 geometric shape categories: in the first phase only circles, squares, and equilateral triangles; in the second phase generalizing circles to ellipses, squares to rectantles, and equilateral triangles to arbitrary triangles. We always train for a total of 256 epochs (over 10,000 training examples), but show only the easier (less varied) examples before the "switch" epoch. The figure on the right shows final test error vs "switch epoch" (i.e. first box-plot is for no curriculum and last one uses only easy examples in the first half of training).



(4) Starting with Baby Language Helps. Following the procedure in (Collobert and Weston, 2008) we train a deep neural network to rank the next word given the previous ones, using the text in Wikepedia as training corpus (631 million examples). The curriculum starts by showing only sentences with words from the most frequent 5k words in the first epoch (270 million examples), then does another epoch with sentences containing the 10k most frequent words, etc. for 4 epochs. The no-curriculum network is trained from the beginning with all the sentences with the 20k most frequent words. The figure on the right shows the ranking loss (average of log of rank of highest-scoring predicted word) as training progresses, with or without curriculum. The curriculum-trained network error quickly dips at the beginning of each epoch as it learns new words. In the last epoch both networks are trained with all sentences with the 20k most frequent words, and we see the curriculum-trained network quickly outperforming the no-curriculum network.



References

Allgower, E. L. and Georg, K. (1980). *Numerical Continuation Methods. An Introduction*. Number 13 in Springer Series in Computational Mathematics. Springer-Verlag.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, to appear.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*.

Elman, J. L. (1993). Learning and development in neural networks: The importance of starting small. *Cognition*, 48:781–799.

Erhan, D., Manzagol, P.-A., Bengio, Y., Bengio, S., and Vincent, P. (2009). The difficulty of training deep architectures and the effect of unsupervised pre-training. In *AISTATS*.

Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554.

Rohde, D. and Plaut, D. (1999). Language acquisition in the absence of explicit negative evidence: How important is starting small? *Cognition*, 72:67–109.

Skinner, B. F. (1958). Reinforcement today. *American Psychologist*, 13:94–99.