# AutoDL challenge design and beta tests: towards Automatic Deep Learning

Zhengying Liu*[1,2], Olivier Bousquet[4], André Elisseeff[4], Isabelle Guyon[1,2,3], Adrien Pavao[1,3], Lisheng Sun-Hosoya[2,3], and Sebastien Treguer[5]

[1] Inria, France
[2] Université Paris-Sud, Université Paris-Saclay, France
[3] ChaLearn, USA
[4] Google Zürich, Switzerland
[5] La Paillasse, France

**Abstract.** Despite recent successes of deep learning, designing good neural networks remains difficult and requires practical experience and expertise. This problem is drawing increasing interest, yielding progress towards fully automatic solutions. However it remains difficult to evaluate where we stand at this stage. We present the design of the AutoDL challenge whose objective is to blind test learning machines (with no human intervention whatsoever) on tasks lending themselves to being solved by deep learning techniques (however, using deep learning will not be required). All problems will be multi-class or multi-label **classification problems** coming from various domains including **text processing, speech recognition, image and video processing**. Raw data will be provided WITHOUT PREPROCESSING, but formatted in a uniform manner, to encourage participants to submit generic algorithms. Further, we impose restrictions on training time and resources to push the state-of-the-art further. The objective of our paper is to seek feedback on our challenge protocol from potential participants, attract participation in the challenge, and promote the idea of formatting and sharing data in the format of the challenge as a common resource.

**Keywords:** AutoML, deep learning, machine learning, model selection, hyper-parameter selection, meta-learning, any-time learning.

## 1 Motivation

Over the last few years, deep learning has attracted much attention both from academia and industry due to its success in many difficult tasks [6]. However, for a given task, designing appropriate model architectures is still a very tedious and labor-intensive process. The acceleration of the demand for deep learning solutions naturally gives rise to the need for improving the automation of the design of deep learning solution. Many approaches have been proposed to address this

* Corresponding author: `zhengying.liu@inria.fr`. The authors are in alphabetical order of last name, except the first author.

problem. Earlier works used neuro-evolution strategies [7] and inspired many recent methods using *evolutionary algorithms* in a similar flavor [1,8]. However, methods of this kind are known to scale poorly and may over-fit. *Bayesian optimization* methods provide another possibility but they also have scaling issues when the dimension (number of hyper-parameters) is high [9,5]. Lastly, ideas borrowed from *reinforcement learning* have recently been applied for this problem [10,2]. But almost all of these methods require huge computational resources (e.g. GPU) accessible only to big companies and laboratories.

It is difficult to form an opinion of the relative merit of these various approaches since they are each using different settings, resources, data, metrics, etc. Thus, to fairly evaluate all these methods and help to advance the state of the art in this emerging research area, it is necessary to devise standard benchmarks. This motivates the organization of an AutoDL challenge.

## 2 Data and Challenge Protocol

The main differences between the new proposed AutoDL challenge and prior AutoML challenges [3,4] are:

1. **Raw data:** Data are no longer pre-processed in a uniform feature vector representation; they include all data types representable as spatio-temporal sequences. We will use a generic data format called TFRecords, used by TensorFlow. This format allows us to format any 3D+time data, including text, speech, image, video, etc.[6]
2. **Large scale datasets:** For development, datasets will all be under 3GB, for practical reasons, however, for final testing, datasets of hundreds of thousands of examples will be used.
3. **Any-time/any-resource learning:** the metric of evaluation will force the participants to provide algorithms, which can be stopped at any time (not known in advance), given any memory and computational resources.

One key aspect of this challenge (and other past AutoML challenges (`http://automl.chalearn.org`)) is that this is a challenge with **code submission**. The participants submit code that is trained and tested on the challenge platform without any human intervention on datasets they never see. During a **development period** feed-back will be immediately provided on a leaderboard using **practice datasets**. After the challenge deadline, the code of the participants will be **blind tested** on some new **evaluation datasets** that were never disclosed before to the public. In contrast with previous AutoML challenges [3,4] in which the practice data was distributed to the participants (except for the target values of the validation and test sets), in the AutoDL challenge, neither the **practice data** nor the **evaluation data** will be exposed to the participants directly in any of the challenge phases. However, the participants will be provided with a large number of **public datasets** in the same format for practice

[6] This will not however impose to participants to use deep learning algorithms nor even Tensorflow; we will provide examples of wrappers to scikit-learn and other libraries.

purposes and to encourage research on **meta-learning**. In addition, a repository will be set up so they can exchange among themselves other datasets.

Another of our contributions will be to facilitate the exchange of data formatted in the format of the challenge by setting up a data exchange repository.

## 3 Challenge metric and baseline results

We are running a full rehearsal of the challenge this fall, in the form of a hackathon lasting a few days, open internally at Google. The hackathon will be organized using the CodaLab platform, of which a cloned instance was created running on Google Cloud.

The hackathon will feature five public datasets and five practice datasets, selected to presents in each set all 5 types of data we are interested in: tabular data, time series, text, image and video. The participants will be given only 2 hours of computation per dataset. We set up Codalab so all five datasets can be processed in parallel. Additionally, we modified Codalab such that participants can save their results at intervals they choose and incrementally improve their performance, until the time limit is attained. In this way we can plot learning curves: performance as a function of time. We will treat both multi-class and multi-label problems alike. The participants will be asked to make binary predictions of presence or absence of a label in a pattern. We will measure performance with the average over all labels of $balanced\_accuracy = (1/2)(TPR + TNR)$. The ranking score for each dataset will be the area under the learning curve computed by the trapeze method, i.e. the area of $mean\_balanced\_accuracy$ as a function of $log(time)$, where $time$ is the cumulative time of training and testing. The overall ranking will be made by averaging the ranks obtained on the 5 practice datasets. There will be no final blind testing on extra evaluation datasets for this hackathon.

We will present at the workshop the results of this hackathon and open up for discussion with the workshop participants what improvements to the challenge protocol are needed.

#### Acknowledgements

This challenge would not be possible without the work of many people, including contributors to the challenge protocol, starting kit, and datasets: Stephane Ayache (AMU, France), Mahsa Behzadi (Google, Switzerland), Hugo Jair Escalante (IANOE, Mexico and ChaLearn, USA), Sergio Escalera (U. Barcelona, Spain and ChaLearn, USA), Yi-Qi Hu (4paradigm, China), Julio Jacques Jr. (U. Barcelona, Spain), Mehreen Saeed (FAST Nat. U. Lahore, Pakistan), Michele Sebag (U. Paris-Saclay; CNRS, France), Danny Silver (Acadia University, Canada), Wei Wei Tu (4paradigm, China), Jun Wan (Chinese Academy of Sciences, China). The challenge is running on the Codalab platform administered by CKCollab LLC with primary developers Eric Carmichael and Tyler Thomas. Google is the primary sponsor of the challenge. Other institutions of the co-organizers provided in-kind contributions.

## References


1. F. Assunção, N. Lourenço, P. Machado, and B. Ribeiro. Evolving the topology of large scale deep neural networks. In *European Conference on Genetic Programming*, pages 19–34. Springer, 2018.
2. B. Baker, O. Gupta, N. Naik, and R. Raskar. Designing Neural Network Architectures using Reinforcement Learning. *arXiv:1611.02167 [cs]*, Nov. 2016. arXiv: 1611.02167.
3. I. Guyon, K. Bennett, G. Cawley, H. J. Escalante, S. Escalera, Tin Kam Ho, N. Macia, B. Ray, M. Saeed, A. Statnikov, and E. Viegas. Design of the 2015 ChaLearn AutoML challenge. pages 1–8. IEEE, July 2015.
4. I. Guyon, I. Chaabane, H. J. Escalante, S. Escalera, D. Jajetic, J. R. Lloyd, N. Maci, B. Ray, L. Romaszko, M. Sebag, A. Statnikov, S. Treguer, and E. Viegas. A Brief Review of the ChaLearn AutoML Challenge: Any-time Any-dataset Learning Without Human Intervention. In *Workshop on Automatic Machine Learning*, pages 21–30, Dec. 2016.
5. A. Klein, S. Falkner, S. Bartels, P. Hennig, and F. Hutter. Fast Bayesian Optimization of Machine Learning Hyperparameters on Large Datasets. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS 2017)*, volume 54 of *Proceedings of Machine Learning Research*, pages 528–536. PMLR, Apr. 2017.
6. Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436, 2015.
7. K. O. S. a. R. Miikkulainen. Evolving Neural Networks Through Augmenting Topologies. 2002.
8. E. Real, S. Moore, A. Selle, S. Saxena, Y. L. Suematsu, J. Tan, Q. Le, and A. Kurakin. Large-Scale Evolution of Image Classifiers. *arXiv:1703.01041 [cs]*, Mar. 2017. arXiv: 1703.01041.
9. K. Swersky, J. Snoek, and R. P. Adams. Freeze-thaw bayesian optimization. *arXiv preprint arXiv:1406.3896*, 2014.
10. B. Zoph and Q. V. Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*, 2016.