AutoML
Towards automation of machine learning systems maintainability

Poenaru-Olaru, Lorena

DOI
10.1145/3491087.3493674

Publication date
2021

Document Version
Final published version

Published in
Middleware 2021 Doctoral Symposium - Proceedings of the 22nd International Middleware Conference

AutoML: Towards Automation of Machine Learning Systems Maintainability

Lorena Poenaru-Olaru
L.Poenaru-Olaru@tudelft.nl
Distributed Systems, TU Delft
Netherlands

ABSTRACT

Machine learning systems both gained significant interest from the academic side and have seen adoption in the industry. However, one aspect that has received insufficient attention so far is the study of the lifecycle of such systems. This aspect is particularly important due to various ML systems' strong dependency on data, which is constantly evolving–and, therefore, changing–over time. The focus of my PhD research is the study of the implications of these dynamics on the ML systems' performance. Concretely, I propose a method of detecting changes caused by drift in the data early. Furthermore, I discuss possibilities for automating large parts of the ML lifecycle management, to ensure a better and more controllable maintenance process.

CCS CONCEPTS

• **Automated Machine Learning** → **Maintainance**; • **Concept Drift** → *Detection.*

KEYWORDS

AutoML, Concept Drift Detection, Data Shift

ACM Reference Format:
Lorena Poenaru-Olaru. 2021. AutoML: Towards Automation of Machine Learning Systems Maintainability. In *22nd International Middleware Conference Doctoral Symposium (Middleware '21 Doctoral Symposium), December 6–10, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3491087.3493674

1 INTRODUCTION

AutoML Introduction

Automated Machine Learning (AutoML) is a research field that has gathered plenty of attention in the previous years. The main idea behind this concept is to take the human out of the machine learning (ML) processes in order to make it more accessible for people with limited background in AI. Besides this, its purpose is to accelerate the ML deployment, by replacing traditionally trying different methods with automated models capable of making the right decision based on the input data.

Literature Gap

Limited attention has been paid to automating processes that are related to the ML model lifecycle. A study done by Google researchers [6] highlights the high costs of maintaining ML models and concludes that investing in maintainability automation is a crucial strategy to ensure the long-term viability of the technology in practice. Therefore, significant attention needs to be paid to making ML less reliant on human intervention once deployed into production and, therefore, automating them.

Problem Formulation

One clear reason why current productizing ML models comes with constant maintenance is the data. ML models' performances are strongly correlated with the data they are using. Most of them are created under the assumption that the distribution of the data used to learn is the same as one of the data used for evaluation. Not meeting this assumption has as consequence a low performance of the analysed model. In real-life ML applications, the data is affected by a multitude of external factors and it is, thereby, unlikely that it will not change over time [8]. In order to ensure high performance of the deployed models, the solution that is generally used is constant maintenance.

Within this research, we are looking into creating an ML system which is robust to data changes. Since we target ML automation our research is part of the AutoML systems research field. Our aim is to enrich the AutoML systems research field by exploring options to generally make ML models less maintenance-intensive.

2 STATE OF THE ART AUTOML AND CONCEPT DRIFT DETECTION

In order to create an AutoML system, scientists have initially analysed the main parts of an ML process that could be automated. The main components that researchers have identified are: *data preprocessing*, *feature engineering and selection*, *model selection and evaluation*, *hyperparameter optimisation* [7]. Despite the technology being relatively robust [7], the study of Celik et al. [2] shows that existing AutoML systems struggle to deal with the changes in data over time, which happen often. The reason behind this is the evolving data. *Concept drift*, also known as *data shift*, is a term which was initially used in the data streams research field in order to refer to changes in data distribution over time [1, 3]. With the purpose of ensuring smooth monitoring of the evolving data, scientists developed drift detectors, which are able to identify the moment in which the data change takes place.

In their survey, Lu et al.[5] distinguished three types of drift detectors, namely the *error rate-based drift detectors* (ERDD) the *data distribution-based drift detectors* (DDDD) and the and *multiple hypothesis test drift detectors* (MHTDD). However, they mention that the last group is a combination of the first 2 and, therefore, we will further focus on those.

Middleware '21 Doctoral Symposium, December 6–10, 2021, Virtual Event, Canada

ACM ISBN 978-1-4503-9155-9/21/12.
https://doi.org/10.1145/3491087.3493674

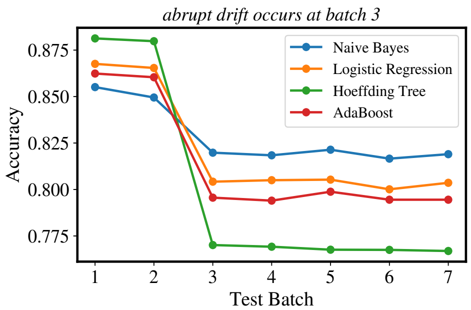

Figure 1: Performance of different classifiers in the presence of concept drift.

3 PROPOSED APPROACH AND PROOF OF CONCEPT

The most commonly used technique to preserve the accuracy in the presence of concept drift is model retraining [4]. For large-scale models, retraining becomes expensive and is therefore only done irregularly.

In our approach, the data of an ML model deployed into production could be seen as a data stream in which the training is done on the first part of the stream and the testing data comes in separate data batches. For each testing batch, we could exploit drift detectors to determine whether a concept drift occurs. We propose to employ such techniques to identify the proper moment of concept drift presence and use this as an alarm system which announces the probability of performance decrease. This alarming system is the input to the self-adaptive model, which based on the fact that drift was detected could take appropriate measures to adapt itself in order to preserve the accuracy. In this way, the ML system would become more robust in the presence of concept drift and less dependent on human involvement.

We further prove the fact that concept drift is responsible for lowering the accuracy by generating a synthetic dataset, which contains *abrupt drift*. We trained multiple classifiers on the first part of the data stream (before the concept drift occurs) and tested them on the remaining test batches. We further analyzed the model performance in terms of *accuracy*. In Figure 1 we show how the presence of concept drift can even lead to around 10% decrease in accuracy, which is a significant decline in performance. We also experimented with a *gradual drift*, which could be seen as a slow drift and also observed the same effect.

4 PLANS FOR THE PHD

Creating a fully automated ML system that is able to adapt to concept drift is challenging but exciting and it has the potential of changing the traditional way of designing ML models. Furthermore, current research is lacking methods for AutoML systems that could easily adapt to concept drift in order to preserve the accuracy [2].

We started this research by firstly conducting an extensive analysis regarding the impact of concept drift on classifiers' performances. We further evaluated the performance of each drift detector type in terms of detecting data shifts. Within this study, we aim to compare the 2 drift detectors categories, namely the ERDD and the DDDD. The reason behind this comparison is the fact that ERDD are assuming that labels are available immediately, while the DDDD do not depend on them. For example, there are ML applications which require human annotation to obtain labels and, therefore, the former drift detectors category is expensive to employ. Currently, there is no study to compare the two groups, as most of them focus on comparing different ERDD [1], since their implementation is available in multiple libraries and frameworks, while the implementations of DDDD are hardly available. Therefore, we also worked on implementing those to complete the comparative study.

The next step into fulfilling our goal is to study various types of drift adaptation techniques in different drift scenarios, which is part of my second year. The last years of my PhD will be dedicated to creating and optimizing an ML system that is able to self-adapt to the concept drift and, thereby, preserve the accuracy during data shifts.

5 CONCLUSIONS

ML models are heavily dependent on data and, therefore, data shifts are likely to have tremendous effects on their performance. Once deployed into production, it is not feasible to constantly check the data used for testing or periodically retrain the model in some situations presented above. Therefore, there is a significant need for automating the model maintenance, which is currently insufficiently studied within the AutoML research community. We consider that the most appropriate way to do so is to first detect when concept drift occurs, which could be done through drift detectors and then apply proper adaptation techniques. This research will not only expand the AutoML research field but also the concept drift detection by our comparison study.

ACKNOWLEDGMENTS

This research is done within TU Delft under the supervision of Dr. Jan Rellermeyer and Dr. Dick Epema. Partially funded by the ICAI lab AI for Fintech Research.

REFERENCES

[1] Roberto Barros and Silas Santos. 2018. A Large-scale Comparison of Concept Drift Detectors. *Information Sciences* 451-452 (07 2018).
[2] Bilge Celik and Joaquin Vanschoren. 2020. Adaptation Strategies for Automated Machine Learning on Evolving Data.
[3] João Gama, I. Žliobaitė, A. Bifet, Mykola Pechenizkiy, and A. Bouchachia. 2014. A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)* 46 (2014), 1 – 37.
[4] Akanksha Kavikondala, Vivek Muppalla, Dr. Krishna Prakasha, and Vasundhara Acharya. 2019. Automated Retraining of Machine Learning Models. 8 (10 2019), 445–452.
[5] Jie Lu, Anjin Liu, Fan Dong, Feng Gu, João Gama, and Guangquan Zhang. 2019. Learning under Concept Drift: A Review. *IEEE Transactions on Knowledge and Data Engineering* 31, 12 (2019), 2346–2363.
[6] D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-Francois Crespo, and Dan Dennison. 2015. Hidden Technical Debt in Machine Learning Systems. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2* (Montreal, Canada) *(NIPS'15)*. MIT Press, Cambridge, MA, USA, 2503–2511.
[7] Jonathan Waring, Charlotta Lindvall, and Renato Umeton. 2020. Automated Machine Learning: Review of the State-of-the-Art and Opportunities for Healthcare. *Artificial Intelligence in Medicine* 104 (2020), 101822.
[8] Indrė Žliobaitė, Mykola Pechenizkiy, and João Gama. 2016. *An Overview of Concept Drift Applications*. Vol. 16. 91–114.