**Exploring the ML Tooling Landscape (Part 3 of 3)**

**The Future of ML Tooling**



Photo by [Pat Krupa](https://unsplash.com/@patkrupa?utm_source=medium&utm_medium=referral) on [Unsplash](https://unsplash.com?utm_source=medium&utm_medium=referral)

The [previous blog post](https://towardsdatascience.com/exploring-the-ml-tooling-landscape-part-2-of-3-8e4f3e511891) in this series considered the current state of the ML tooling ecosystem and how this was reflected in ML adoption in industry. The main takeaway was the widespread use of propriety tooling amongst companies in this field, with a correspondingly diverse and splintered ML tooling market. The post ended by looking at some emerging near-term trends, highlighting the predominance of data observability and related tools, as well as the emergence of MLOps startups. This blog post will pick up from this previous thread to discuss some of the key trends in ML tooling that are likely to dominate in the near future — or at least ones I want to talk about! As indicated in the previous blog post, I want to focus on MLOps, AutoML, and data-centric AI.

In this series of blog post, I aim to address the following questions,

1. What is the level of maturity with regards to ML in industry?
2. What is the current state of ML tooling and adoption?
3. What are likely trends in ML tooling?

This blog post is concerned with the **third** question.

As with all the other posts in this series, the same disclaimer applies: This series of blog posts is by no means meant to be exhaustive — or necessarily even correct in places! I wrote this to try to organise my thinking on the reading I’ve done in recent weeks and I want this to become a jumping off point for further discussion. This is an absolutely fascinating field and I am really keen to learn more about the industry, so please get in touch!

**MLOps**

Although MLOps has been a consistent theme throughout this series, I wanted to dedicate some space to examine what changes MLOps promises to introduce represent more generally to ML.

MLOps aims to address problems specific to deploying and maintaining production-facing ML products. Distinct from more traditional software engineering (SWE), ML models, and the pipelines that power them, are composed of multiple highly integrated components, which depend on historic inputs i.e. are “stateful”; meaning that real-time end-to-end monitoring is the only means of fully addressing problems in production (Shankar, 2021).

“Drift” is an example of a problem that can only really be addressed with MLOps. Drift is an umbrella term that can relate to a range of connected phenomena; in a nutshell, it is generally observed through the performance degradation of a model over time due a mismatch between the data the model was trained on, and the data the model uses for inference. A full remedy to this problem can only be provided by having a truly Agile end-to-end development pipeline in place.

More generally, MLOps at its core represents a shift away from data-led model development to something closer to SWE, where operational logging becomes the basis of monitoring and development (Shankar, 2021).

**AutoML**

Automated machine learning (AutoML) refers to a range of systems designed to eliminate manual work across most to all steps of an ML workflow. The term “AutoML” was originally popularised by Google in 2017 (TWiML, 2019), and received a huge boost in 2018 following the success of Google’s NASNet, which outperformed all existing human-created models (State of AI Report: June 29 2018).

Generally speaking, the goals of AutoML systems are twofold. Firstly, AutoML systems aim to make the use of ML accessible to a wider range of users, who may not have the expertise to use ML otherwise. Secondly, AutoML aims to accelerate the velocity of model development by making the end-to-end development process as seamless as possible (Hutter et al., 2019).

AutoML is essentially a superset of Combined Algorithm Selection and Hyperparameter (CASH) Optimisation. Founded on the understanding that no single ML model performs best for all tasks (Hutter et al., 2019), CASH optimisation treats the data preparation, choice of ML algorithm, and hyperparameter optimisation as a single global optimisation problem (Brownlee, 2020). In addition to these elements, AutoML introduces “pipeline structure search” within a pipeline creation problem i.e. an end-to-end ML problem: Whereas a CASH approach will return a linear pipeline, an AutoML algorithm may generate pipelines with multiple parallel branches, which are then recombined (Zöller, 2021).

Surveys of existing AutoML libraries, that is not part of an end-to-end platform, have generally found that they are outperformed by CASH algorithms; at least in terms of pure model performance. Furthermore, in a survey of classic ML models, i.e. not neural networks, AutoML generated pipelines are typically found to be quite simple, are limited to supervised learning, and do not yet fully address all elements required of a truly automated ML pipeline (Zöller, 2021). Results produced for AutoML applied to neural networks, Neural Architecture Search (NAS), returned similar results, with the observation that NAS could generate novel, albeit simple, architectures (He, 2021). AutoML offerings across cloud vendors, AutoML platforms (H2O, DataRobot etc.) and open-source software, are currently not feature complete, despite the claims of many distributors (Xin, 2021); there is currently no single open-source utility able to create a fully automated MLOps workflow (Ruf, 2021).

Although the often stated aims of AutoML adoption are given in terms of wider enablement of ML capabilities, I could imagine, at least in its current form, due to the resource requirements and black box nature of the tooling, AutoML may, ironically, better serve companies with established ML competencies. In particuler, if AutoML tooling hopes to enable less technically-able users to make use of ML, it places a hard requirement on parallel improvements in explainable AI (XAI) technologies. Regardless of these issues, some observers have suggested a more limited, piecemeal approach, whereby AutoML merely aids elements of an ML workflow where customsability and generalisability are not key concerns. For strategic reasons, there may always remain an argument for manual tasks to persist (Xin, 2021).

**Data-centric AI**

The specifics of what term “data-centric AI” implies can be difficult to pin down exactly. However, it can generally be taken to mean advocating for a holistic approach that places emphasis on the infrastructure, processes, and culture that support model performance, including of course the data itself, rather than optimising a model in isolation (Kurlansik, 2021). A more specific cause of the increased relevancy of data-centric AI recently, may be due in part to automated labelling, and plateauing architecture performance (State of AI Report: October 12, 2021).

Even if data-centric AI may not pertain to any exclusive tools or processes, it does place added emphasis on technologies that fit under the DataOps umbrella, including data observability, data lineage, data quality, and data governance tools. To elaborate on some of these areas a little more, data observability can be considered to be more than typical software monitoring and instead empowers users to perform ad hoc analysis on historical data (Hu, 2022).

Another such area pertains to data quality, which is generally maintained through some kind of data validation tool. There are two broad approaches taken: using a set of manually maintained declarative rules, like Great Expectations, or using some layer of ML (perhaps on top of a ruleset) to automatically detect data quality issues, like Anomalo. The two approaches have their pros and cons. On the one hand, although maintaining a set of declarative rules may be laborious, it ensures that expectations of the data are made explicit and helps to clarify understanding — similar to how code tests aid documentation. On the other hand, it may be difficult for stakeholders to articulate what they require of the data, hence the need for automation (Data Driven NYC, 2021).

**Summary**

From the [first blog post](https://towardsdatascience.com/exploring-the-ml-tooling-landscape-part-1-of-3-53f8c39e6e4e) in this series, we found both that the vast majority of companies applying ML principles are fairly immature in the field, and for all companies the initial steps related to data collection and processing remain the main stumbling blocks. On this basis, we would expect that tooling and solutions that gain traction would be expected to directly address these issues, evidenced by the interest in data-centric AI. Although this blog post treated MLOps, AutoML, and data-centric AI somewhat separately, the former two broadly fit under the data-centric umbrella: taken together, the goal is to drive the business value derived from ML/AI. In addition to these topic areas, there is a case to be made for the use of AI to enhance data migration and data cleansing.

Over time, we are seeing more companies getting involved in ML/AI, but as a proportion total ML/AI ability has remained about the same as the number of newcomers outweigh more established players. Despite this, it is generally argued that innovation in this area is being driven by the needs and experiences of mature ML companies, which also likely means a degree of fragmentation will persist, as we saw in [part two](https://towardsdatascience.com/exploring-the-ml-tooling-landscape-part-2-of-3-8e4f3e511891) of this series.

It should also be considered the extent to which, or indeed if at all, the problems faced by companies in this field can ever be adequately addressed by another tooling solution: In many cases, issues arise due to more fundamental, systemic problems (Brinkmann & Rachakonda, 2021). Although it may be unfair to most startups in this field, in many cases they have their beginnings in addressing problems the founders have themselves previously experienced, which as we would come to expect now, are likely not representative of wider industry (Brinkmann & Rachakonda, 2022). Although past experience can form the basis of an effective proof-of-concept, it is not uniformly applicable, and can run the risk of becoming what is termed a “Solution in Search of a Problem” (Friedman, 2021). The next era of ML adoption may also see a reassertion of fundamentals that have demonstrably proved to deliver real business value, noteably valuing structured data over unstructured, and process over tooling (Brinkmann & Rachakonda, 2021). This would also likely mean that data-specific tooling will continue to dominate the landscape in the near future.