An Operational Metrics Framework for ML Data

Anoop K. Sinha [1] **Gunveer Gujral** [2] **Liz Jenkins** [2] **Nicolas Scheffer** [2]

Abstract

Maintainable, high quality, rapidly built, scalable ML datasets have been fundamental for multiple AI production applications that we have worked on. How have we gone about building these ML datasets in a systematic way? Our approach has included defining a set of operational metrics for ML data. Our framework for organizing those metrics focuses on goals that we have: time to launch, effect on model performance, properties of the data, data quality, and tracking dataset and historical changes. In each area, we have defined more detailed metrics and created operational processes to track them. Through disciplined tracking, we have seen the benefits of ML dataset improvements on ML performance improvements in diverse examples.

1. Introduction

We have worked on shipping ML models in multiple industrial production applications: large scale search ranking, recommendation, natural language understanding, speech understanding, and more. Building ML datasets for training and evaluation has been a foundational step in each project. As we worked on these different projects, we saw the need for a systematic approach to creating ML datasets. As we gained experience, we created a framework and a set of linked metrics that has helped us improve ML data and has resulted in ML model improvements.

Our efforts to be systematic about ML data have had hurdles. Significantly more resources have been allocated to ML model development than to ML data development, as others have seen (Sambasivan et al., 2021). ML data is operationally intensive, requiring analytics on datasets, individual fixes to datasets, and more. The overall pipeline from ML project initiation until model serving is full of many data steps as noted in MLOps (Kreuzberger et al., 2022).

[1]FAIR, Meta AI [2]Reality Labs, Meta. Correspondence to: Anoop K. Sinha <anoopsinha@fb.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

Though the impact of the ML data efforts on the quality of the ML models has been undeniable (Halevy et al., 2009), the fraction of improvement from data and the fraction from modeling has been difficult to separate. Individuals working on ML model development are expected to do ML data development work as well, but we have found that we made the most improvement when we have had a dedicated, expert team.

2. Framework

Our framework for ML data metrics has emerged from the following core goals, which are aligned to our business objectives.

- **Time to launch**: One of our core goals has been to increase velocity. We hope to accelerate reaching model quality that will then have an impact on products.
- **Effect on model performance**: We want teams to have a clear understanding of how datasets impact model performance in all aspects: quality, volume, distribution. And we want teams to be metrics-driven about their ML data development.
- **Properties of the data**: We want users of the data to have clarity as to what the data represents and what the distribution of the data is, including what segments may be under or overrepresented, among other properties.
- **Data quality**: Data quality is a special property of data that we call out separately. If any data is labeled, we want to have clear quality metrics into how noisy these labels are.
- **Tracking datasets and historical changes**: We want historical changes in data and datasets to be clearly documented to users of the data, similar to version control and code history. This also includes understanding data distributional shifts over time.

3. Metrics Considerations

In using the framework, specific metrics vary by project. We cover some of the important considerations, challenges, and some example metrics that we have used.

3.1. Time to launch

Improving the end-to-end time from ML dataset development to launch is a fundamental goal. However, this is also one of the most difficult metrics to develop since ML development is iterative. The time to launch for an effort can vary significantly. Oftentimes the new release is improving on an existing model and the starting points vary.

Example Metrics: We do watch progress in our project management tools and strive for continuous improvement. However, we have found that this metric is difficult to instrument in our infrastructure systems. If we cannot capture the measurement automatically, in some instances the time to build each component is tracked manually or we measure this with a developer survey asking: how long was the time from data to ML model, or was there an improvement in time to launch from ML data efforts?

3.2. Effect on model performance

Model performance measurements vary significantly based on the ML domain. Careful design of evaluation sets and coverage of those sets applies to all domains.

Example Metrics: specific metrics for Automatic Speech Recognition (ASR) for example include WER (Word Error Rate), SER (Sentence Error Rate), matrixed by locale and domain

Before launching, we run offline side by side comparisons between production and candidate models as well as A/B comparison where a portion of production traffic is exposed to the new models. Additionally, we monitor end to end offline metrics through scaled user testing to ensure the model works in a production like environment. One critical consideration is to make sure that the metrics are reproducible for each run. Given the high scale of many of our models, variations on each run can affect the metrics.

Another consideration is that the model performance varies over time in production systems that reload models periodically. End-to-end testing and on-going measurements help ensure success.

3.3. Properties of the data

We emphasize multiple properties of the data itself to help ensure we minimize issues of bias.

This includes keeping a close eye on feature coverage and the most important features in the dataset. Anomalies tend to come up in certain features, and thus it is very important that we have analysis of features.

Often the features analysis is done with statistical metrics. We need to be cognizant of drift in distribution of the features that can come over time in the datasets from various factors.

We have started to investigate methods that identify the value of specific data for improving model accuracy.

Example Metrics: in ASR for example, we measure demographics, environment, topic, domain, locale, hours of data, distinct participants, percentage completion.

3.4. Data quality

For any labeled data, the accuracy of the datasets is often difficult to ensure and measure. In critical projects, we add significant resources to help review datasets and do multiple reviews, which while expensive, have been essential for improving quality.

Metrics such as the multi-review agreement rate, the agreement rate on audit, and the agreement rate on "golden test sets" are all core metrics that we track.

We use case-specific automated checks to measure data correctness against expectations by domain to identify anomalies.

Example Metrics: in ASR for example, we measure against golden set: accuracy, precision, recall, precision/recall for slots, CER (character error rate), WER, SER, punctuation issues, null data, out of total utterances, rater disagreement, reasons for mistakes.

3.5. Tracking datasets and historical changes

We have built special tools for retention management, data discovery, and reuse.

To help ensure we have retained or deleted all data as expected by privacy policy, we use automated checks with specific policies.

Given the cost of development of datasets, we measure efficiency and utilization. We provide tooling where engineers can easily find existing datasets and generate new features from them.

Additionally, in management of datasets it is critical for us to understand the history of the datasets for reproducibility. Data lineage requires significant infrastructure investments but has proved useful in model debugging flows.

Example Metrics: in ASR for example, we measure number of data changes, source of those changes, dataset purpose, and authors, again by domain, locale, and product.

4. Conclusions

By using this framework for ML data, we have helped increase the speed and quality of machine learning development. Introducing this framework systematically has had

hurdles, but in the areas we have made investments, we have seen return on investment.

There are multiple additional enhancements that we plan in the future. Those include more integrated instrumentation of the metrics across domains. We also want tooling that helps encode best practices around data properties and quality. Some of the properties that we hope to measure, such as bias and quality, are still active areas of research and so our goal is to develop flexible metrics and systems that can change as our needs change.

References

Halevy, A., Norvig, P., and Pereira, F. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24:8–12, 2009. URL http://www.computer.org/portal/cms_docs_intelligent/intelligent/homepage/2009/x2exp.pdf.

Kreuzberger, D., Kühl, N., and Hirschl, S. Machine learning operations (MLOps): Overview, definition, and architecture, 2022. URL https://arxiv.org/abs/2205.02302.

Sambasivan, N., Kapania, S., Highfill, H., Akrong, D., Paritosh, P. K., and Aroyo, L. M. "Everyone wants to do the model work, not the data work": Data cascades in high-stakes AI. 2021.