# An Empirical Study of Challenges in Machine Learning Asset Management

Authors: Zhimin Zhao, Yihao Chen, Abdul Ali Bangash, Bram Adams, Ahmed E. Hassan (2024)

Presenter: Jacie Jermier

11/4/2025

UNIVERSITY OF WATERLOO

## Problem

- ML applications rely on assets (models, datasets, code, configs) for development, training, and deployment.
- ML systems run in quickly-changing environments (data shifts, requirements evolve, models need continuous updates) which make versioning, traceability, and reproducibility difficult.
- Traditional tools (ie Git) lack specific ML support; modern tools. Modern tools like MLflow and DVC exist, but face their own challenges like library coupling, and management issues.
- Gap: Limited empirical studies on real-world user challenges and solutions across tools.



## New Idea

Approach

- Analyze user experiences from Q&A posts (15,065) to identify challenges, solutions, and forum differences.
- **Knowledge inquiries:** Posts asking how to use a tool/feature, what a concept means, or best practices. Not reporting a failure.
  - ~40% of posts
- **Problem inquiries:** Posts reporting errors, bugs, crashes, or unexpected behavior. The tool is not working as expected.
  - ~60% of posts

Methods

- Classify posts: Knowledge inquiries vs. Problem inquiries
- BERTopic on all 15,065 posts: Extract **133 problem topics** (ie MLflow UI not loading)
  - 3 researchers then grouped the 133 topics into 16 macro-topics
  - Found nearly 1 in 5 posts dealt with environment setup
- **Key finding**: **133 more granular topics** to **16 high-level categories** to Top 3 macro-topics:
  - Software Environment & Dependency (**18.89%**)
  - Model Deployment & Serving (**10.59%**)
  - Model Creation & Training (**9%**)
- Open card sorting + BERTopic for solutions in solved posts only (4,758): 79 topics into 18 macro-topics. Top 3: **Environment & Dependency Fixes** (**23.31%**), **Feature & Component Development** (**15.35%**), **File & Directory Management** (**9.64%**)



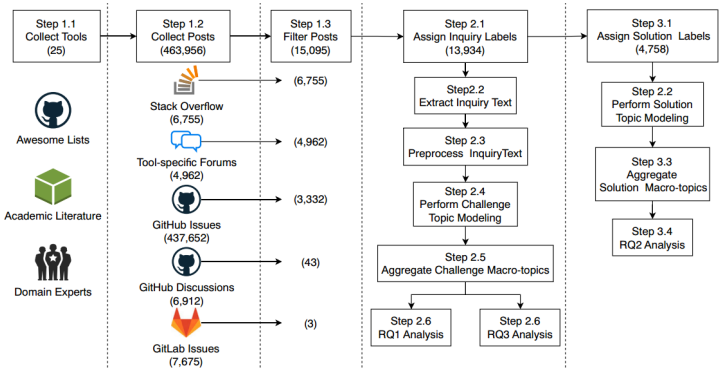


**Fig. 2** Study workflow to analyze Q&A posts related to ML asset management from developer discussion forums

# Research Questions and Findings

- RQ1: Challenge Topics (133) - What topics are most frequently discussed related to machine learning asset management?
  - Findings: 1. **Environment & Dependency Mgmt** (18.89%), 2. **Model Deployment & Serving** (10.59%), 3. **Model Creation & Training** (9%). Problem inquiries more common than knowledge inquires.
- RQ2: Solution Topics (79) - What topics of solutions exist for the challenges related to machine learning asset management?
  - Findings: 1. **Environment & Dependency Mgmt** (23.31%), 2. **Feature/Component Dev** (15.35%), 3. **File/Dir Mgmt** (9.64%). **Knowledge inquiries** tend to be self-resolving. **Problem-type posts**, 62.5%, tend to be fixed using environment/dependency solutions where cross-domain solutions are required.
- RQ3: What are the commonalities and differences between developer forums in their discussion related to machine learning asset management?
  - Findings: SO contains nearly half of all Q&A posts. Where tool forums come in second contributing to ~35%. Posts related **to software environment and dependency** are the most prevalent in most forums.



# Summary

- SO seems to act like a central hub for ML tool help
- Software environment mgmt and dependency seem to be a **universal pain point** across forums
- Depending on the ML tool being used – it may be more useful to go to specific forum (ie DVC to github but MLflow to SO).




# Positives

- Comprehensive dataset: **15,065** posts from a diverse pool of sources
- Method: **BERTopic and open card** sorting identifies granular topics and then evolving those into macro-topics, with clear mappings. (ie heatmaps)
- Practical findings:
  - Can better anticipate **what types of issues may occur** – which can help to better understand what tool may be most useful. (ie better support on multiple forums)
  - Helps developers contributing to improving the tools understand where their users are having issues and how they're solving them. Can **innovate more quickly to help users**. (ie better documentation).
  - Better **education** if this is a missing gap in knowledge for many users.



# Negatives

- Adding Reddit and Discord as they are large developer forums as well. Although they had a sufficient amount of data, it wouldn't have been that much work to add Reddit and Discord.
- Macro-topic aggregation risks bias – there will always be a level of subjectivity here.
- 1,131 unassigned posts which have the potential to miss nuances
- With the focus being on open-sourced tools the study may be overlooking issues or nonissues in proprietary ones (ie does Facebook have a similar pattern of issues found in this article).

## Future Work

- How can education play a role here:
  - In a classroom-based environment - if we know that environment and dependency management is the #1 pain point, can there be insight provided for better knowledge transfer? Is there a "best" practice that would help mitigate challenges for environment management? Or is this inherently challenging, like when so many of us find ourselves in dependency hell?
    - 23% of all fixes are just 'use conda env export' or 'pin your packages.
- A longitudinal study would be very interesting.
  - Over a one-year period: do we see challenges and solution topics changing in their distribution? Do new topics start appearing?



## Rating

- 5/5
  - Clear practical implications
  - Large data set
  - Helpful read for my own research



## Discussion

- Could the root cause be because tools don't enforce standards? Or because docs are scattered frequently? Or because ML is dynamic?
- How can ML tools be used to better automate the management of these top issues? How could AI play a role here?
- Are the solution mappings (ie cross-domain for problems) applicable to non-ML asset management?
- Should academia play a role in standardizing ML asset practices, when has this been done in the past?