Recs at Reasonable Scale

NVIDIA RecSys Summit 2022

Jacopo Tagliabue

Coveo

Ciao!

Entrepreneur (co-founder of Tooso), now Director of AI (TSX:CVO).

OS contributor (1000+ Github stars) and evangelist on “reasonable scale” ML and e-commerce tech.

Researcher (25+ research papers in top ML venues), Prof. of MLSys at NYU, Co-organizer of SIGIR eCom.

Anyone Can Cook

Anyone Can ~~Cook~~ Build Great RecSys

README.md

You Don't Need a Bigger Boat

An end-to-end (Metaflow-based) implementation of an intent prediction (and session recommendation) flow for kids who can't MLOps good and wanna learn to do other stuff good too.

This is a WIP - check back often for updates.

Philosophical Motivations

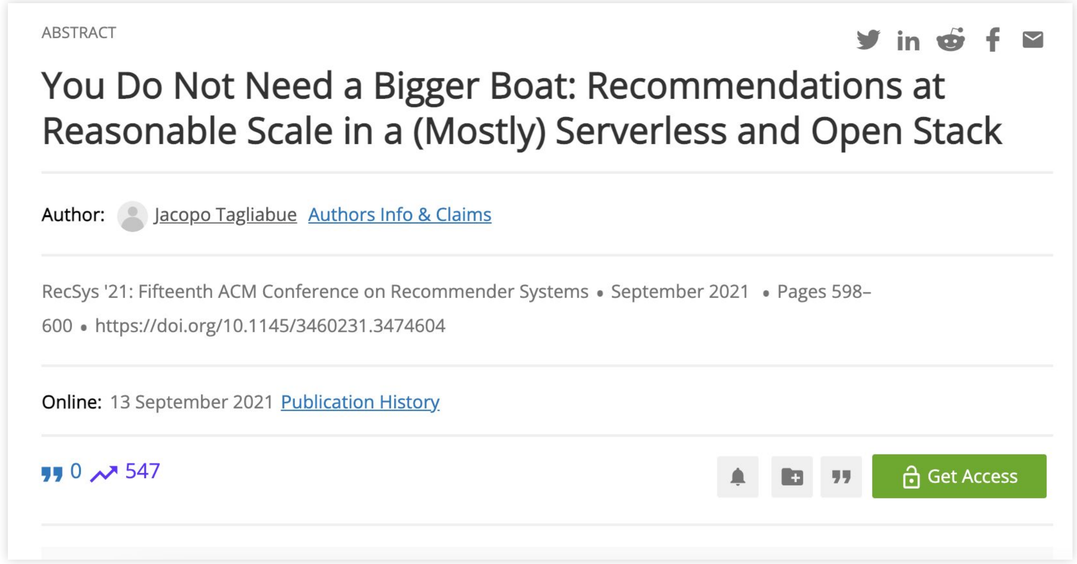
There are plenty of tutorials and blog posts around the Internet on data pipelines and tooling. However:

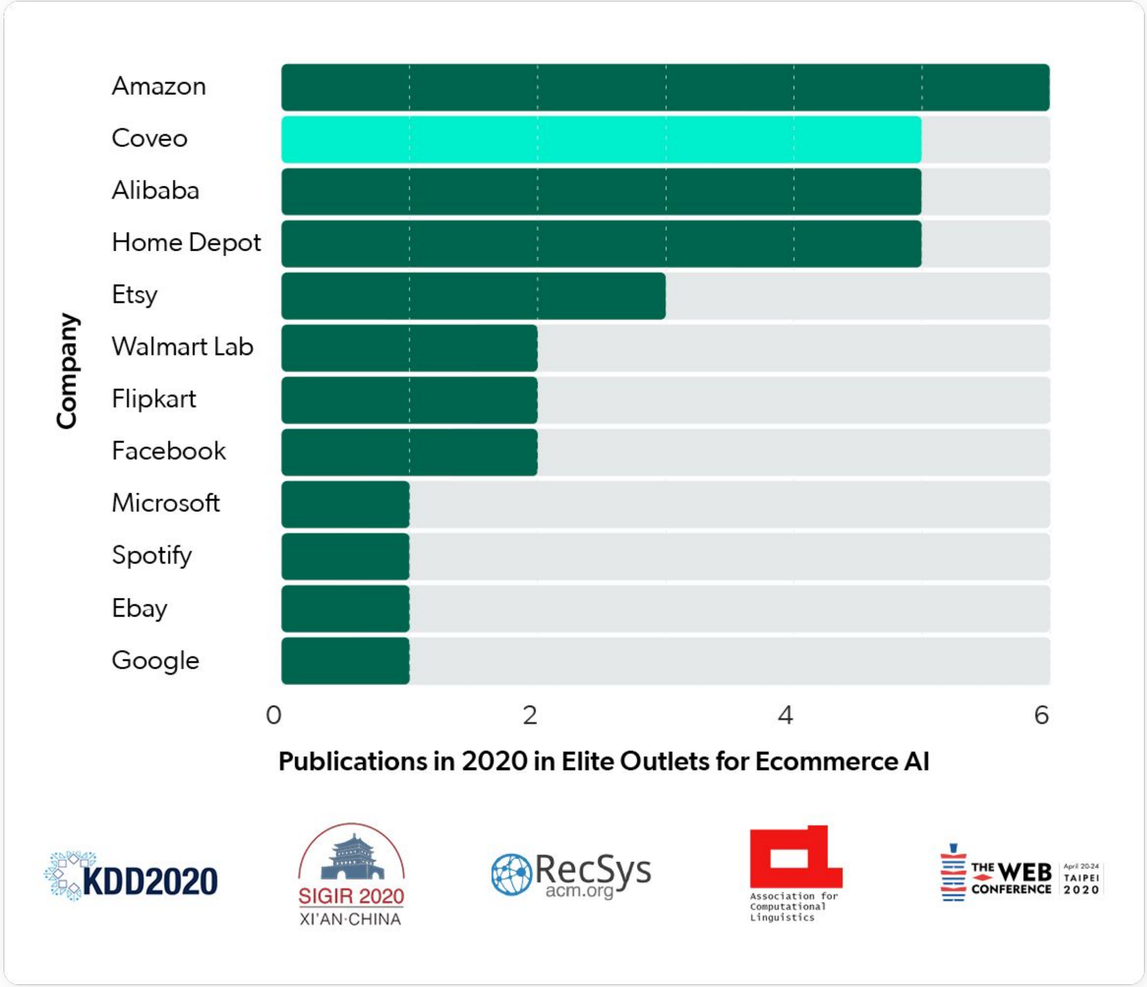
- they (for good pedagogical reasons) tend to focus on *one* tool / step at a time, leaving us to wonder how the rest of the pipeline works;
- they (for good pedagogical reasons) tend to work in a toy-world fashion, leaving us to wonder what would happen when a real dataset and a real-world problem enter the scene.

This repository (and soon-to-be-drafted written tutorial) aims to fill these gaps. In particular:

So, why almost nobody does?

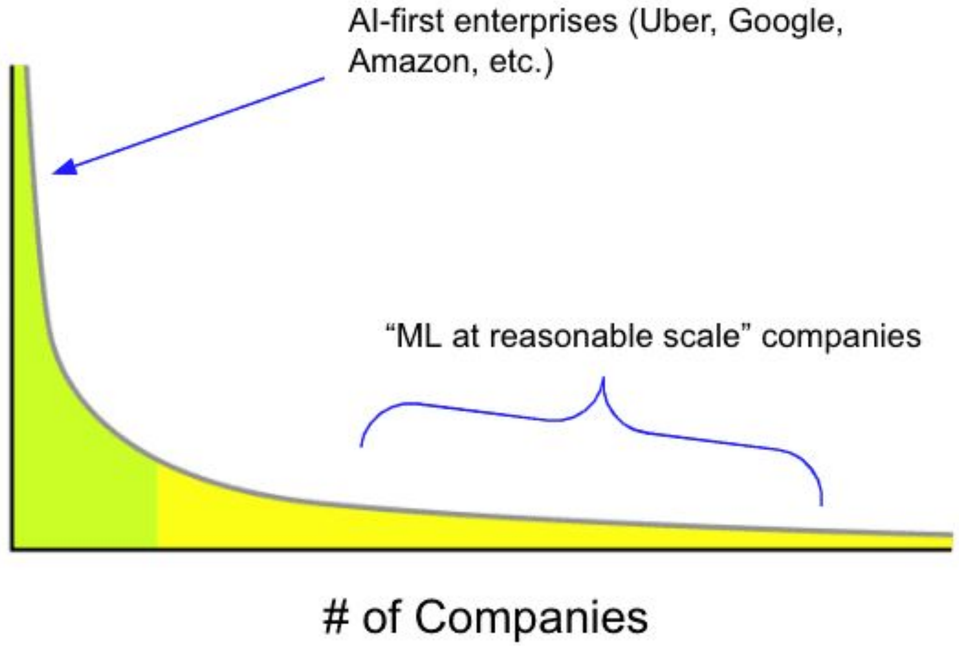
- The majority of innovation in e-commerce tech comes from very few, large, public B2C players (with **one exception**).
- Why? And what can we do about it?





Barriers to entry are **high**

- Implementing cutting-edge recSys algorithms is hard.
- Models are only one part of the challenge, MLops is hard too (GPU provisioning, scheduling, versioning, etc.).
- Most literature is skewed towards the resources, needs, and the problems of few companies.



If you think getting a model into production needs:

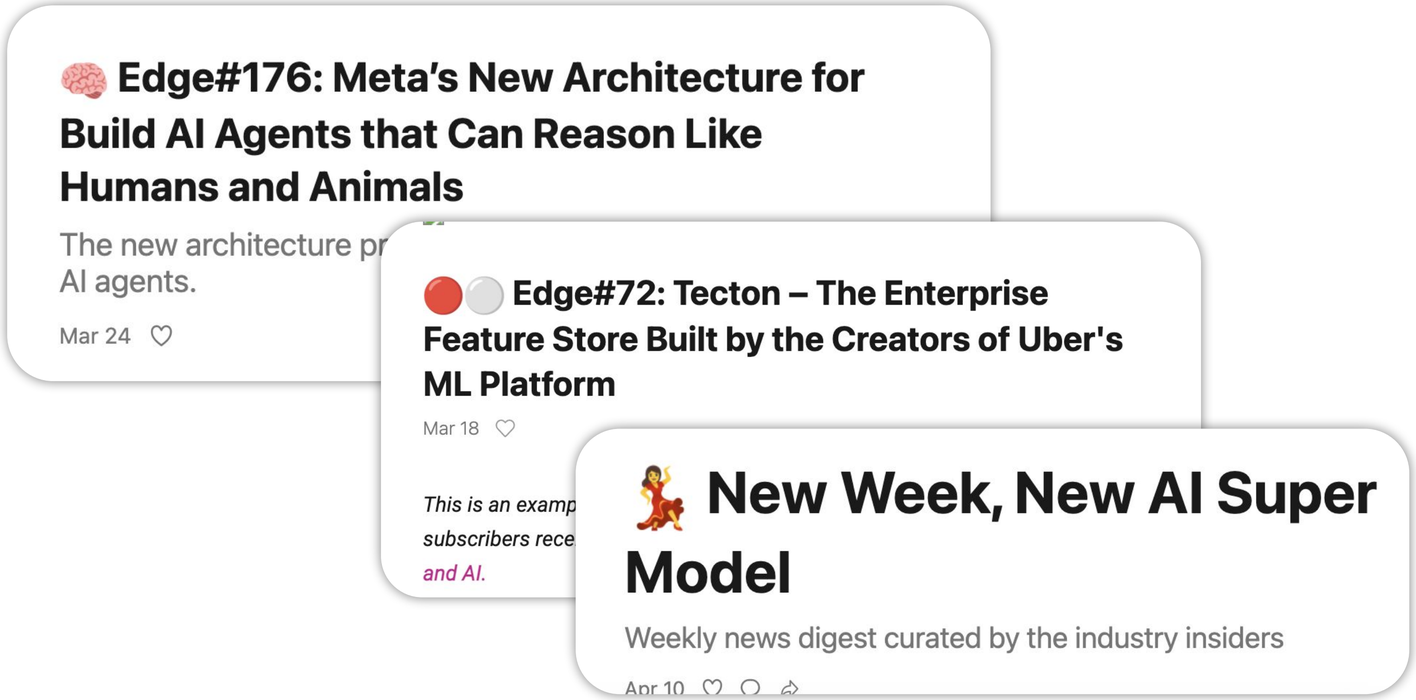
- a data engineer for the ETL
- a data scientist for the modelling
- a machine learning engineer for MLOps
- a software engineer for production

Where does it stop?

What's the minimum team size to get started?

Big Tech fetish is real

… and we are not making it easy to start.



Big Tech fetish is real

… and we are not making it easy to start.

🧠 **Edge#176: Meta's New Architecture for Build AI Agents that Can Reason Like Humans and Animals**

The new architecture pr
AI agents.

Mar 24

🔴⚪ **Edge#72: Tecton – The Enterprise Feature Store Built by the Creators of Uber's ML Platform**

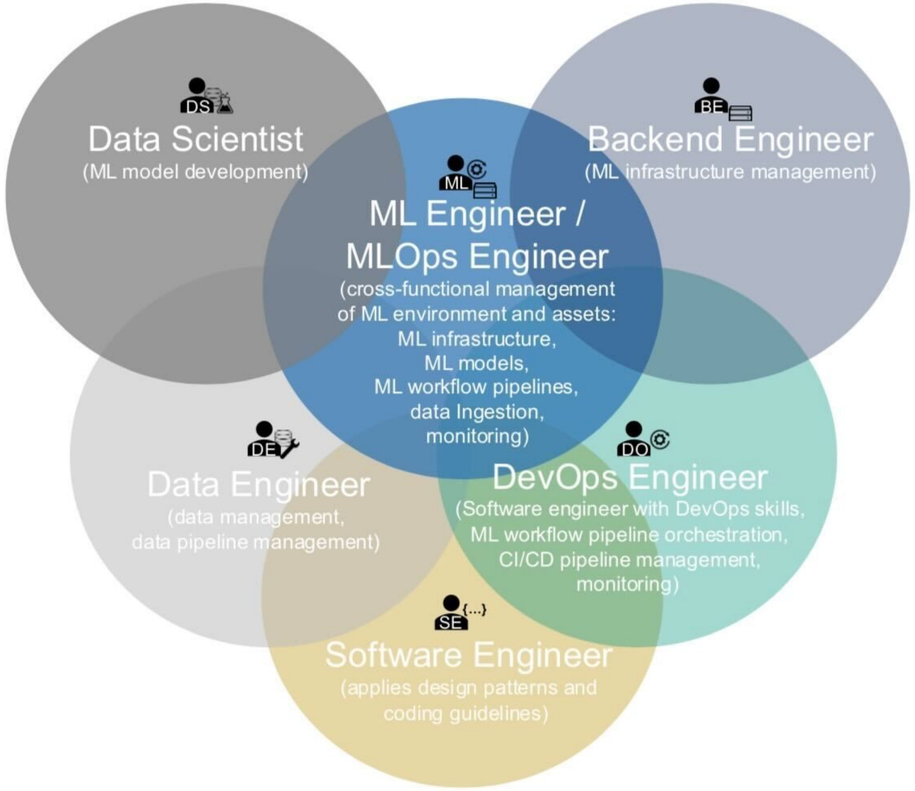
Mar 18

This is an examp
subscribers rece
and AI.

💃 **New Week, New AI Super Model**

Weekly news digest curated by the industry insiders

Apr 10



Big Tech fetish is real

🧠 **Edge#176: Meta's New Architecture for Build AI Agents that Can Reason Like Humans and Animals**

The new architecture pr
AI agents.

Mar 24

🔴⚪ **Edge#72: Tecton – The Enterprise Feature Store Built by the Creators of Uber's ML Platform**

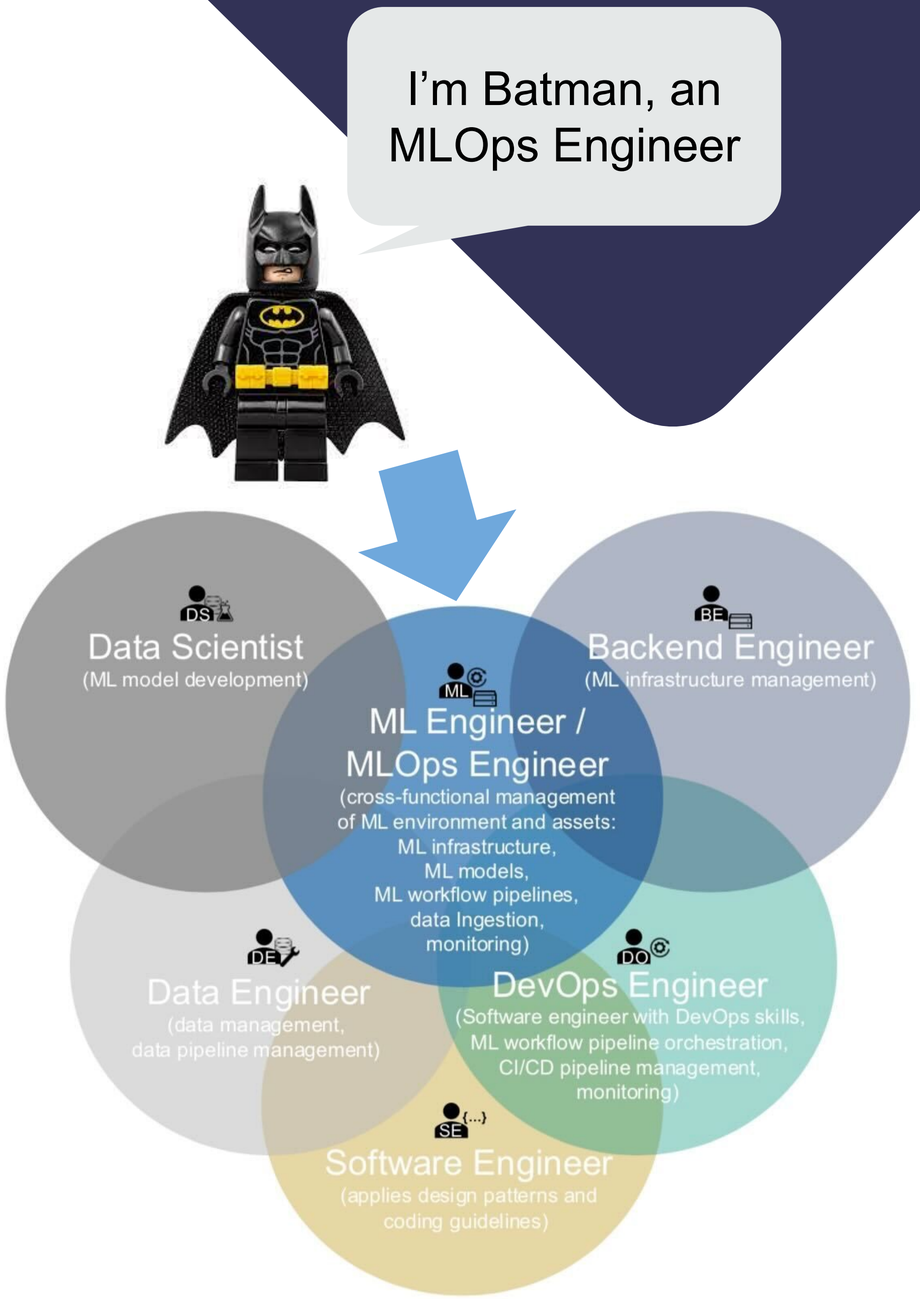
Mar 18

This is an examp
subscribers rece
and AI.

💃 **New Week, New AI Super Model**

Weekly news digest curated by the industry insiders

Apr 10



Myth-Busting RecSys

- RecSys algorithms are hard.
  Yes, but good abstractions mean we don’t start from scratch anymore.
- MLops is hard too.
  Yes, but a new ecosystem means we can buy what we don’t want to maintain / build.
- Literature is skewed towards the problems of few companies.
  Yes, but sometimes you and Big Tech have the same problem (e.g. testing).

Talk is cheap, show me the code

README.md

recs-at-resonable-scale

Recommendations at "Reasonable Scale": joining dataOps with deep learning recSys with Merlin and Metaflow

Overview

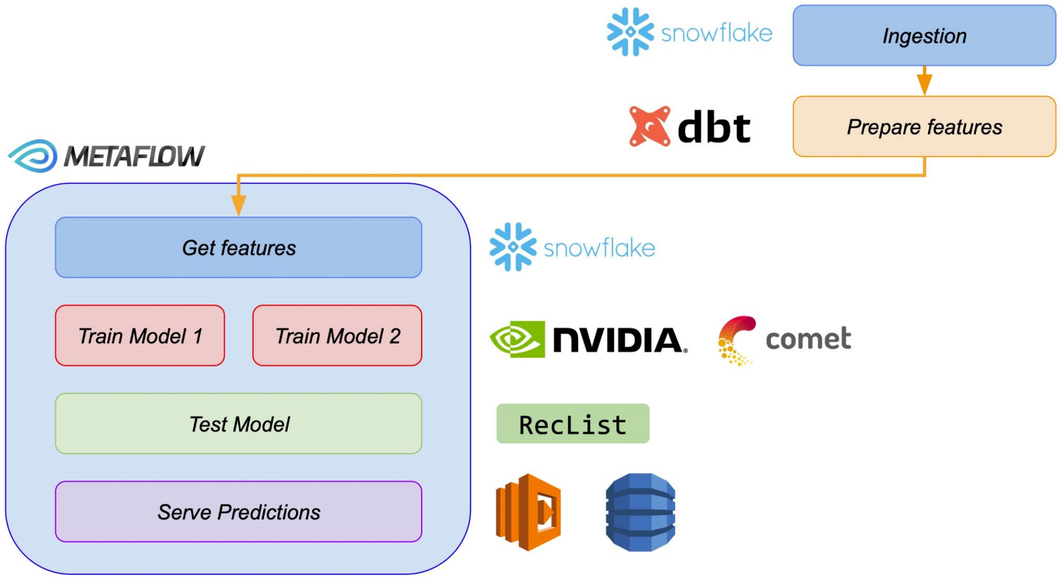
July 2022: this is a WIP, come back often for updates, a blog post and my NVIDIA talk (FORTHCOMING)!

This project is a collaboration with the Outerbounds and NVIDIA Merlin teams, in an effort to release as open source code a realistic data and ML pipeline for cutting edge recommender systems "that just works". Anyone can ~~cook~~ do great ML, not just Big Tech, if you know how to pick and choose your tools.

TL;DR: (after setup) a single ML person is able to train a cutting edge deep learning model (actually, several versions of it in parallel), test it and deploy it without any explicit infrastructure work, without talking to any DevOps person, without using anything that is not Python or SQL.

As a use case, we pick a popular RecSys challenge, user-item recommendations for the fashion industry: given the past purchases of a shopper, can we train a model to predict what he/she will buy next? In the current V1.0, we target a typical offline training, cached predictons setup: we prepare in advance the top-k recommendations for our users, and store them in a fast cache to be served when shoppers go online.

Our goal is to build a pipeline with all the necessary real-world ingredients:



Talk is cheap, show me the code

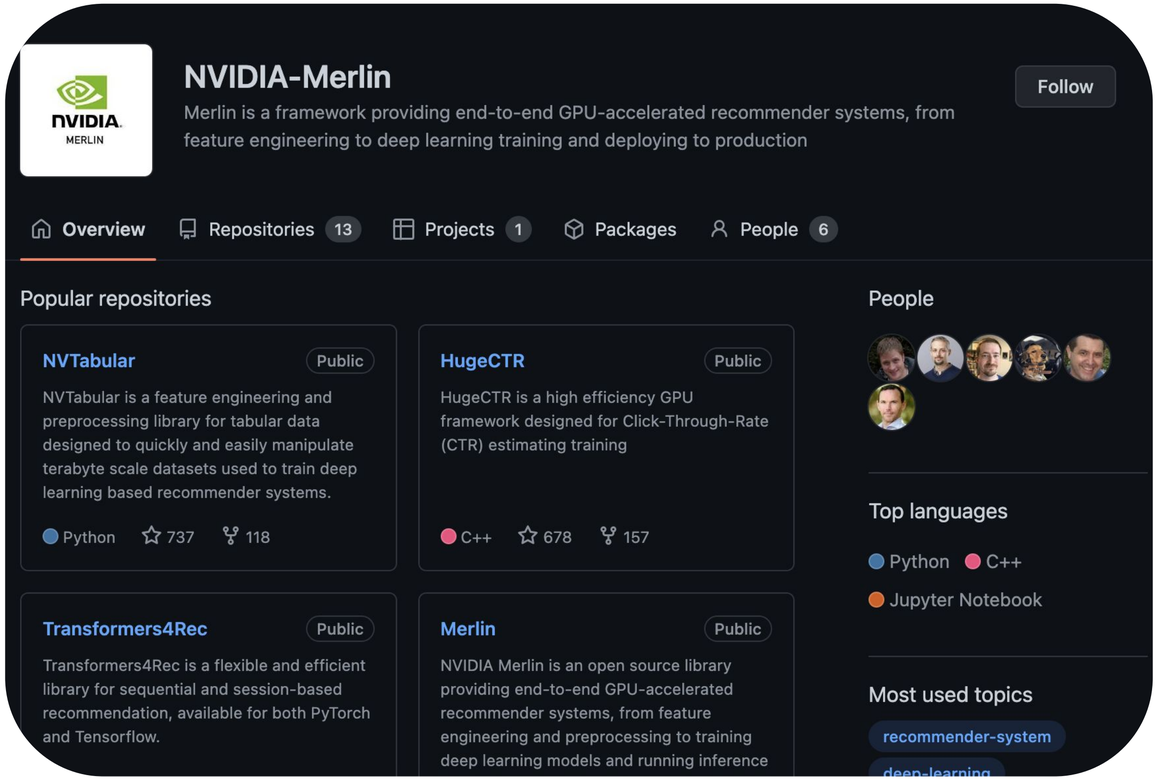
“After a 10 minutes setup, one MLE can run an entire RecSys pipeline - from data to API - with no explicit infrastructure work.”

“What one fool **can do**, another can.”

RecSys that just works

RecSys as an algorithmic challenge

- **The “HuggingFace” moment of RecSys:** the field has converged to a set of best practices and building blocks that are effective in most cases.
- **NVIDIA Merlin** is an open source library for DL-based RecSys “with an easy-to-use API” exposing common architectures. **How easy?**



RecSys as an algorithmic challenge

- **The “HuggingFace” moment of RecSys:** the field has converged to a set of best practices and building blocks that are effective in most cases.
- **NVIDIA Merlin** is an open source library for DL-based RecSys “with an easy-to-use API” exposing common architectures. **How easy?**

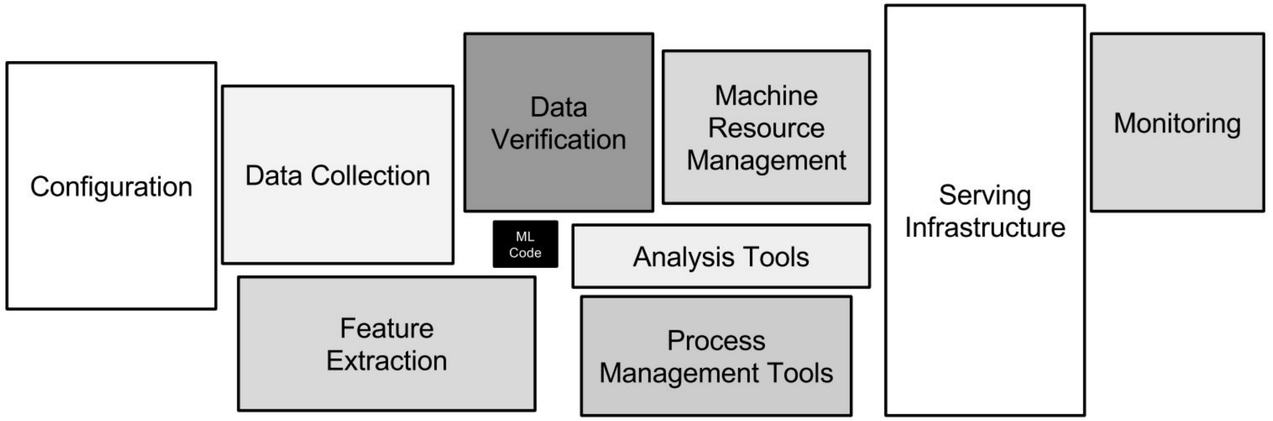
```
# train the model
model = mm.TwoTowerModel(
    train.schema,
    query_tower=mm.MLPBlock([128, 64], no_activation_last_layer=True),
    samplers=[mm.InBatchSampler()],
    embedding_options=mm.EmbeddingOptions(infer_embedding_sizes=True)
)
model.compile(optimizer="adam", run_eagerly=False, metrics=[mm.RecallAt(10), mm.NDCGAt(10)])
model.fit(train, validation_data=valid, batch_size=self.hypers['BATCH_SIZE'], epochs=int(self.N_EPOCHS))
# test the model on validation set and store the results in a MF variable
self.metrics = model.evaluate(valid, batch_size=1024, return_dict=True)
print("\n\n====> Eval results: {}\n\n".format(self.metrics))
```

VERY EASY!

MLOps without much Ops

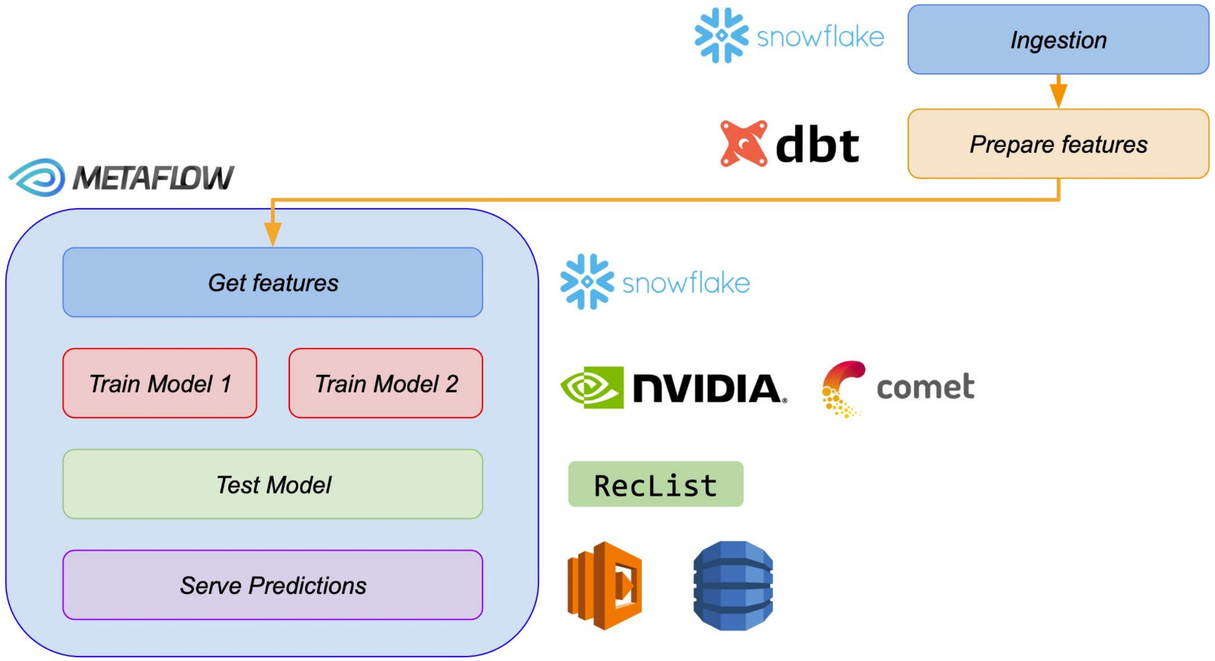
RecSys as an MLOps challenge

- **DataOps (ingestion, preparation, quality) is as important as MLOps**
  - Leverage the Modern Data Stack to provide ultra-scalable data support
- **Infrastructure management is tedious and error prone**
  - Strive for velocity, embrace PaaS / FaaS for non-core activities
  - From “infra as code” to “infra in code”: **Python is all you need**.



RecSys as an MLOps challenge

1. **Add NVIDIA Merlin**, to train cutting-edge models.
2. **Add the MDS**, to provide scalable DataOps.
3. **Add a Python orchestrator**, to provide Python abstractions over GPU training, scheduling, versioning.
4. **Add a tracker,** to track your progress through experiments.
5. **Add a FaaS+SaaS endpoint**, to serve predictions to shoppers.

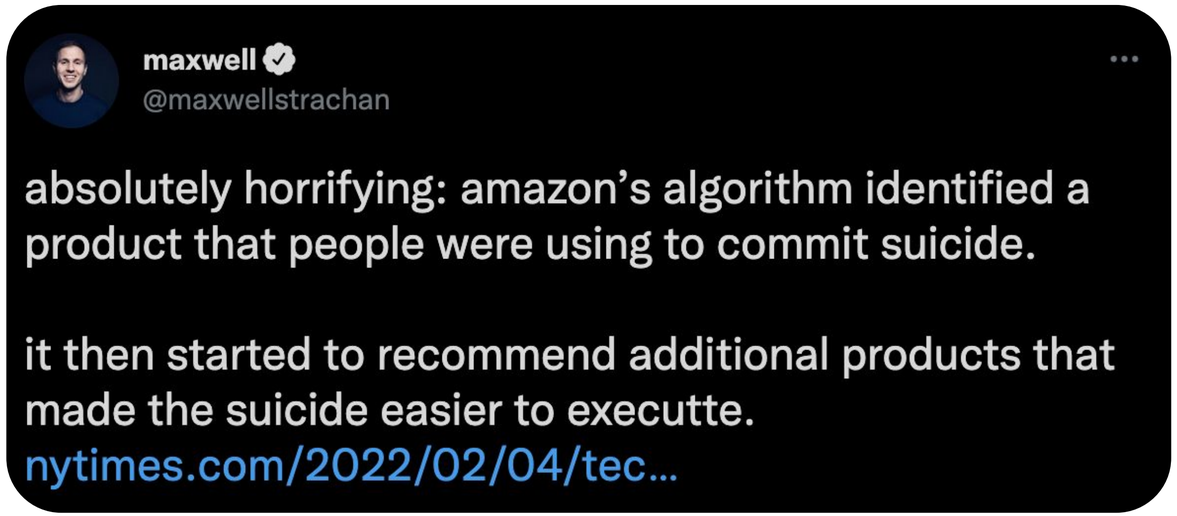


Start with the problem (and the stack)
you have, not the one you wish you had.

Wild Wild Tests

RecSys as a testing challenge

- **RecSys evaluation** is typically a point-wise metric over held-out data points, say HIT RATE, or MRR.
- RecSys are some of the most ubiquitous form of Machine Learning systems, and *even one mistake can have dire consequences*.
- Typical testing protocols fall short of measuring real-world generalization performance.



RecSys as a testing challenge

The ~~heart~~ power-law is deceitful above all things: two models may have the same exact “accuracy” while behaving very differently.

HR: 57 / 110 = 0.52

#: 100, H: 50, HR: 0.5

#: 10, H: 7, HR: 0.7

VS

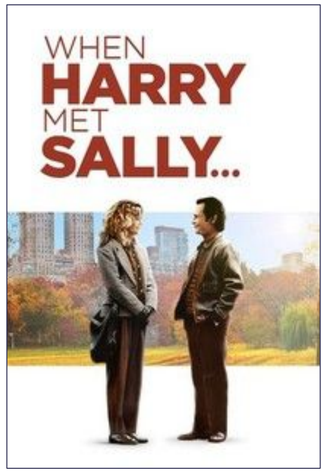
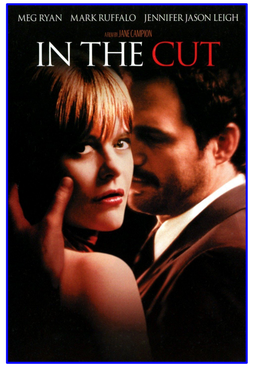
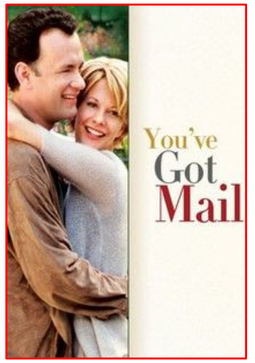
HR: 57 / 110 = 0.52

#: 100, H: 56, HR: 0.56

#: 10, H: 1, HR: 0.1

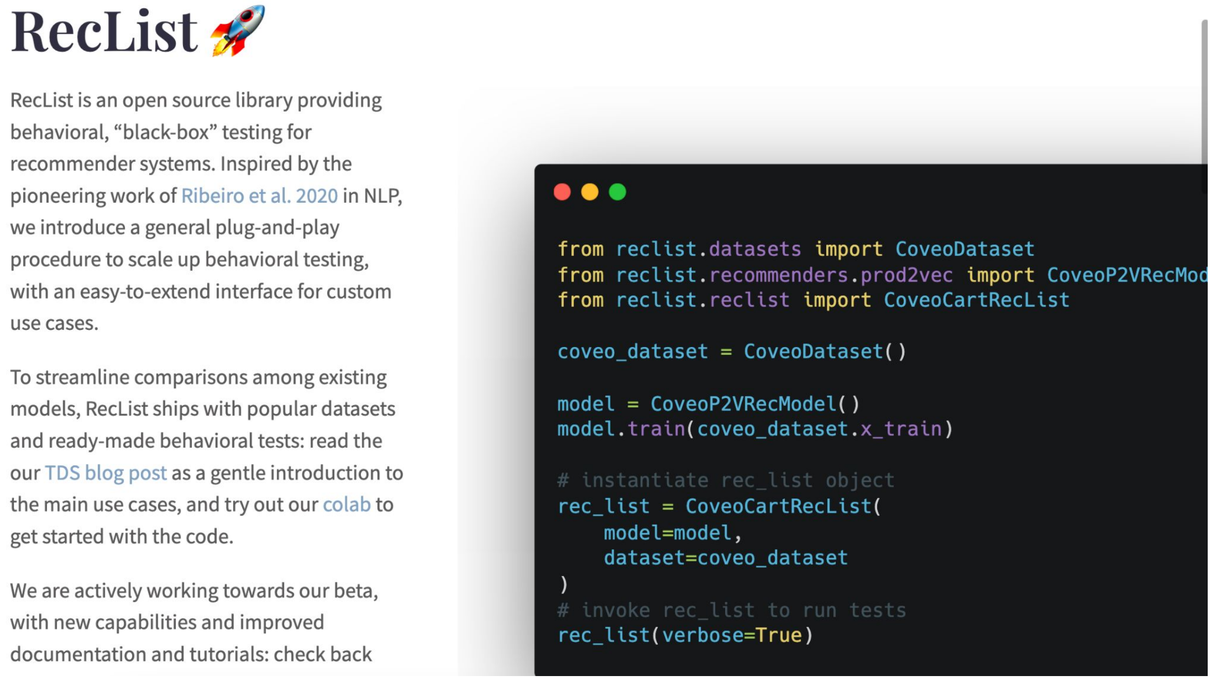
RecSys as a testing challenge

Not all predictions are wrong in the same way: in particular, some mistakes are worse than others.

Test Truth	Model #1	Model #2	
			*This is a reasonable mistake!*

RecSys as a testing challenge

- **Introducing RecList**, an open source package for behavioral testing.
- RecList provides abstractions over datasets and best practices to help with testing and monitoring!
- RecList is a community project supported by forward-looking companies: **check them out**!



https://reclist.io/

comet neptune.ai GANTRY

Bonus: Join us at CIKM!

CIKM 22 Data Challenge:
https://reclist.io/cikm2022-cup/

The future at Reasonable Scale

A roadmap between RecSys and MLOps

1. **Training and Optimizing Offline**: recommendations are often more similar to *interventions* than predictions.
2. **Deployment**: while “simple” deployment is a commodity, more complex models are still hard to handle.
3. **Testing and Monitoring**: performance on test sets is a small part of a rounded evaluation process.

Ask *not* what the RS can do for you…

Open Data

README.md

SIGIR eCOM 2021 Data Challenge Dataset

Public Data Release 1.0.0

Overview

Coveo hosts the 2021 SIGIR eCom Data Challenge and this repository contains utility scripts and information about data preparation and testing for the Challenge and the paper introducing the Challenge is available as a draft on arxiv.

README.md

Shopper Intent Prediction from Clickstream E-Commerce Data with Minimal Browsing Information

Public Data Release 1.0.0

Overview

This repo contains the description of the data released in conjunction with our *Nature Scientific Reports* paper Shopper Intent Prediction from Clickstream E-Commerce Data with Minimal Browsing Information.

Data Download

The dataset is available for *research and educational* purposes here. To obtain the dataset, you are required to fill out a form with information about you and your institution, and agree to the *Terms And Conditions* for fair usage of

Open Source

README.md

You Don't Need a Bigger Boat

An end-to-end (Metaflow-based) implementation of an intent prediction flow for kids who can't MLOps good and wanna learn to do other stuff good too.

This is a WIP - check back often for updates

README.rst

RecList

pypi v0.2.2 | docs passing | Python package passing | contributors 6 | License MIT | downloads 1k | youtube video

RecList

README.md

The Post-Modern Stack

Joining the modern data stack with the modern ML stack

README.md

recs-at-resonable-scale

Recommendations at "Reasonable Scale": joining dataOps with deep learning recSys with Merlin and Metaflow

Overview

July 2022: this is a WIP, come back often for updates, a blog post and my NVIDIA talk (FORTHCOMING)!

This project is a collaboration with the Outerbounds, NVIDIA Merlin and Comet teams, in an effort to release as open source code a realistic data and ML pipeline for cutting edge recommender systems "that just works". Anyone can ~~cook~~ do great ML, not just Big Tech, if you know how to pick and choose your tools.

TL;DR: (after setup) a single ML person is able to train a cutting edge deep learning model (actually, several

Open Science

ACL Anthology FAQ Corrections Submissions Search...

Query2Prod2Vec: Grounded Word Embeddings for eCommerce

Federico Bianchi, Jacopo Tagliabue, Bingqing Yu

ABSTRACT

The Embeddings That Came in From the Cold: Improving Vectors for New and Rare Products with Content-Based Inference

Authors:

Language in a (Search) Box: Grounding Language Learning in Real-World Human-Machine Interaction

Federico Bianchi, Ciro Greco, Jacopo Tagliabue

"Does it come in black?" CLIP-like models are zero-shot recommenders

Patrick John Chia, Jacopo Tagliabue, Federico Bianchi, Ciro Greco, Diogo Goncalves

Abstract

Product discovery is a crucial component for online shopping. However, item-to-item recommendations today do not allow users to explore changes along selected dimensions: given a query item, can a model suggest something similar but in a different color? We consider item recommendations of the comparative nature (e.g. "something darker") and show how CLIP-based models can support this use case in a zero-shot manner. Leveraging a large model built for fashion, we introduce GradREC and its industry potential, and offer a first rounded assessment of its strength and weaknesses.

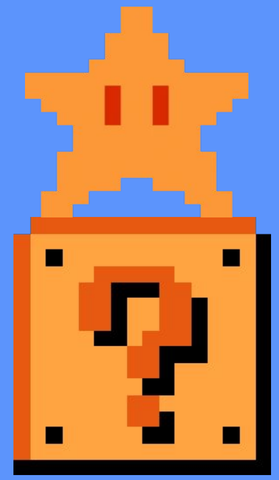
PDF

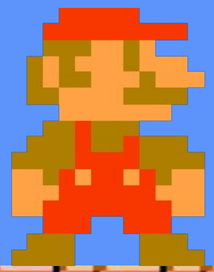
Cite

Search

Check out / share / star our open source repositories!

Wanna work with us? Get in touch!





“We can only see a short distance ahead, but we can see plenty there that needs to be done.”

See you, RecSys cowboys