



17-645 MLiP Group Project:

# Movie Recommendation Service with *Scikit-Surprise*

---

**Aarushi Sinha   Advaith Sridhar   Kaipeng Wang   Shih-Lun Wu   Cara Yi**

Team 13: **The Tensor Titans** (TA: Ritika Dhiman)

*Heinz College & LTI, Carnegie Mellon University*

May 4th, 2023

Carnegie Mellon University  
**HeinzCollege**



# Team Tensor Titans



# Team Tensor Titans

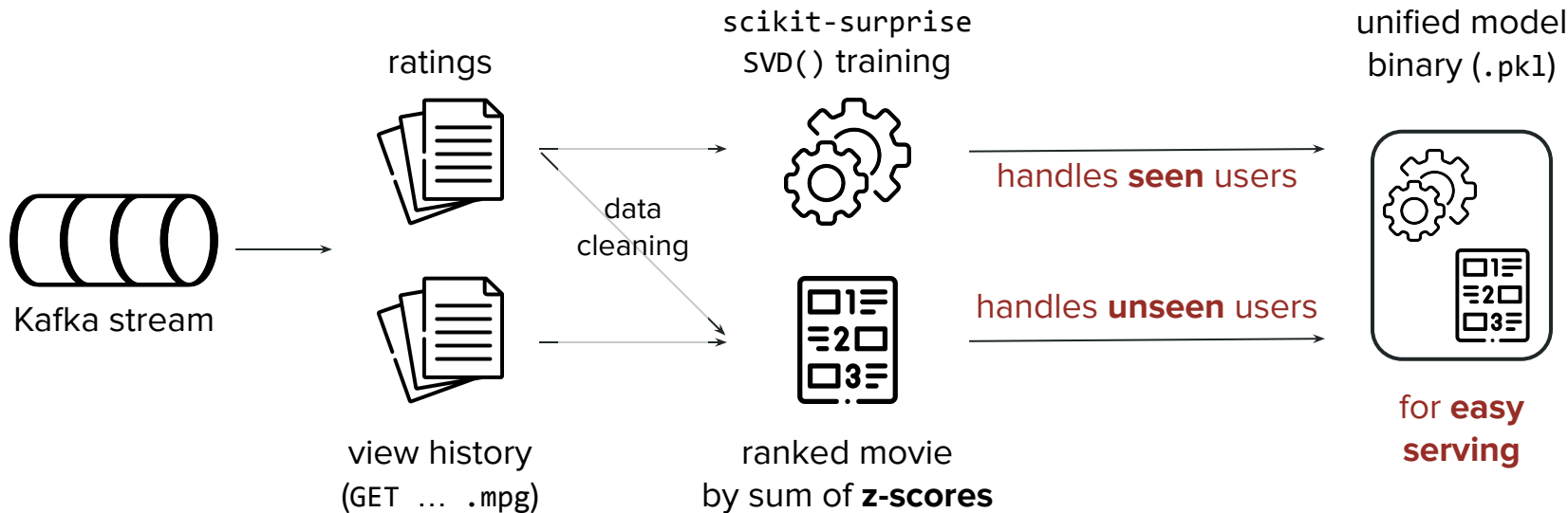


# Outline

- Our Recommendation System
  - Model & Service
  - Evaluation
  - Monitoring
  - Feedback Loop Detection
- Reflections
  - Implementation
  - Teamwork

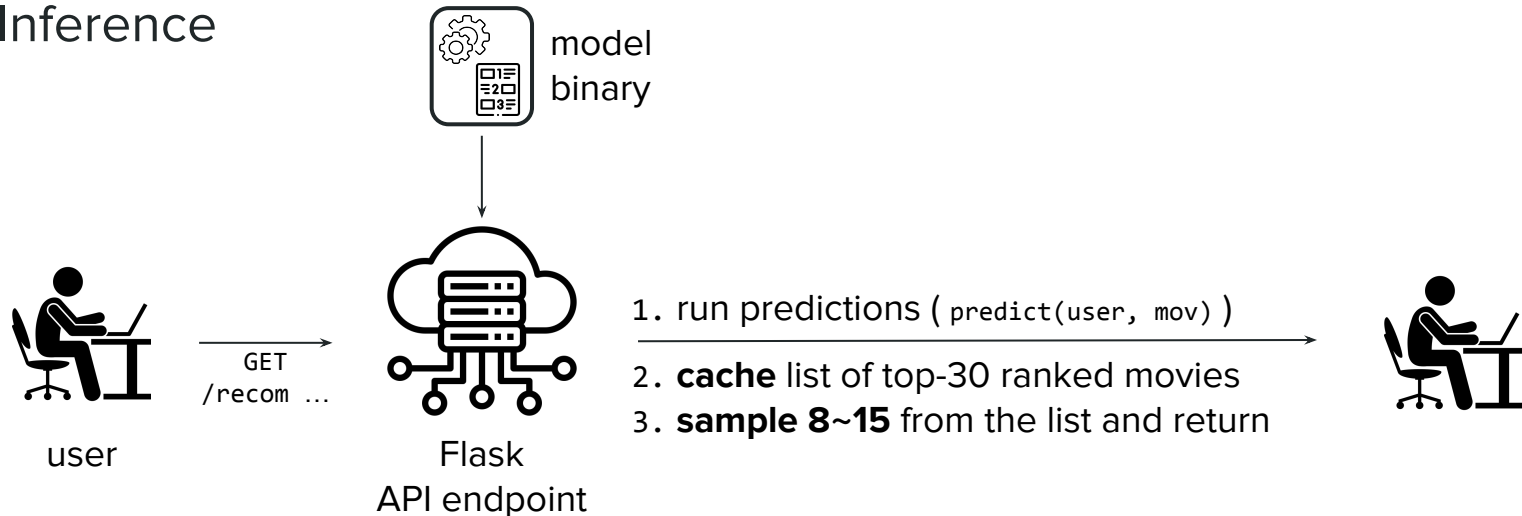
# Model & Service (1/2)

- Training



# Model & Service (2/2)

- Inference

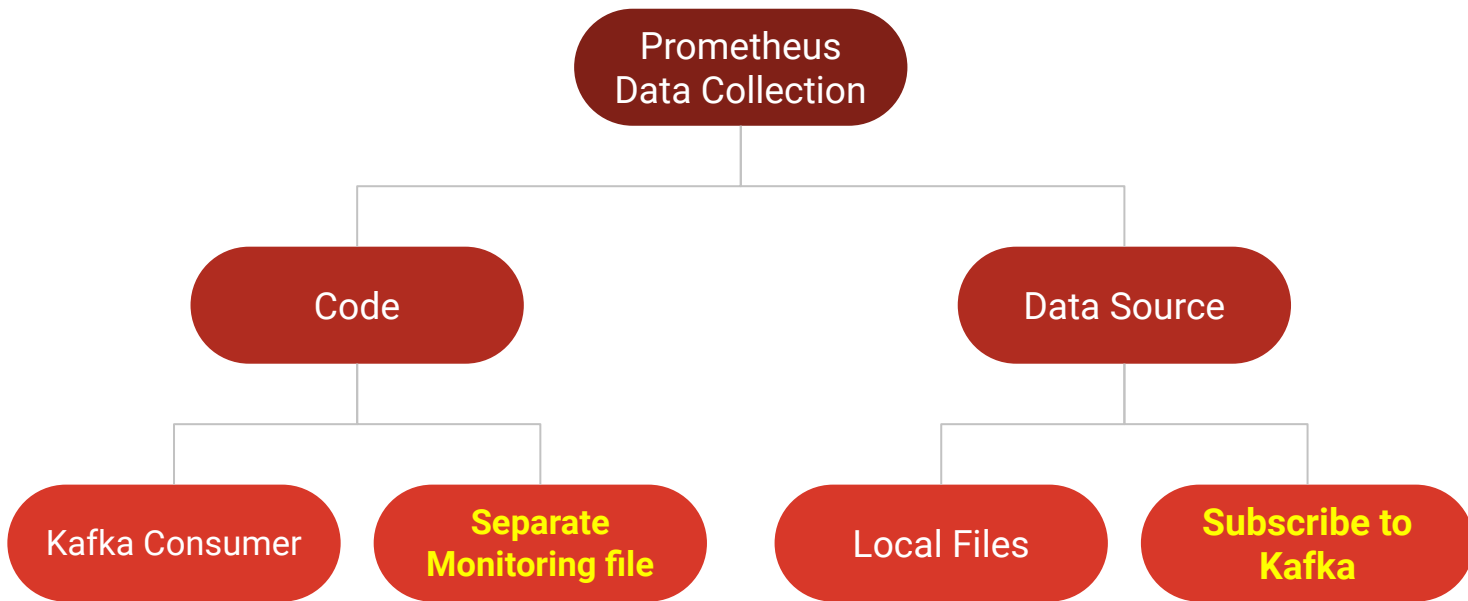


- cache → **improve latency** on repeated requests from same user
- sample → **ensure novelty** for each request

# Evaluation

- Offline
  - Root-mean-square error (**RMSE**) → measures **abs diff** of pred vs. true ratings
  - Pearson's correlation (**r**) → checks if **trend** of pred ratings matches true ones
    - around **0.45~0.48** for our models
- Online
  - Mean rating after deployment → measures **user satisfaction**
    - around **3.95** for SVD( ) model
    - around **4.02** for BaselineOnly() model 🤖

# Monitoring



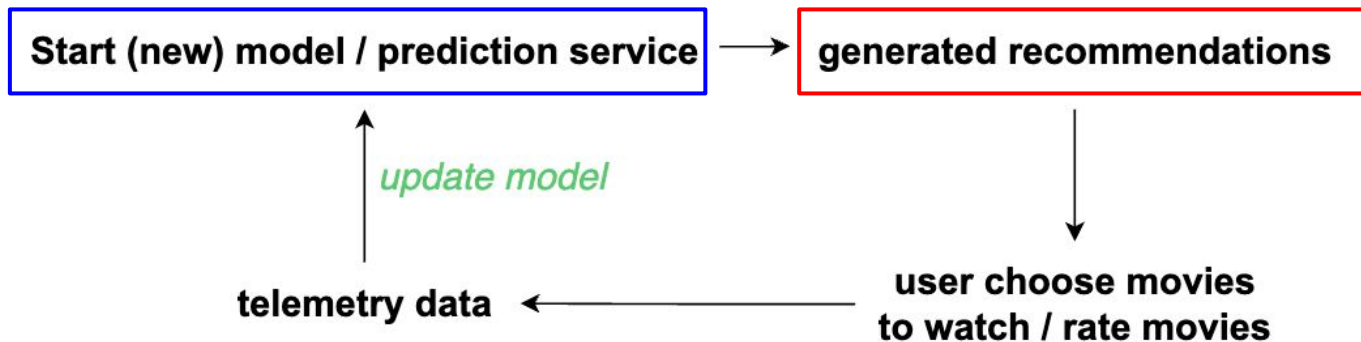
Separation of concerns



Loose coupling with storage  
More real-time



# Anticipated feedback loops



- feedback loop caused by prediction logic
- feedback loop caused by modeling (use new features / metadata)

# Feedback loop detection

Consider potential feedback loops caused by change of prediction logic:

- narrowing the list of candidate movie: **30 → 10**
- number of movies returned: **[8, 15] → [3, 5]**

We collect 2 hours' telemetry data (movie ratings & view history) after each model update:

	after 1st model update	$\Delta$	after 2nd model update	$\Delta$	after 3rd model update	$\Delta$
Total number of unique movies watched	30363 → 31544	<b>+1181</b>	28368 → 30581	<b>+2213</b>	25575 → 27155	<b>+1580</b>
Number of active users	254233 → 255783	<b>+1550</b>	231785 → 249041	<b>+17256</b>	205399 → 200856	<b>-4543</b>
Average movie rating	3.9795 → 3.9802	<b>+0.0007</b>	3.9890 → 4.0353	<b>+0.0463</b>	4.0031 → 4.0313	<b>+0.0282</b>

# Some key design decisions

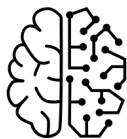
- Not choosing K8s - significantly simplified our deployment but had the cost of \*\*\*\*\* writing our own load balancer and model update script \*\*\*\*\*
- Retraining with only a subset of the data

# What we would change if we did this as a company



Better A/B experiments:

- Calculate average ratings of movies that we actually recommended for model evaluation
- Log down time, load-balancing, etc. to track experiments



Understand why Baseline does better than the more sophisticated SVD



Actually look at Jenkins (was forgotten post M2)

# Reflection on teamwork

Clear task division

Efficient communication & Regular team meeting

Highly self-disciplined and self-motivated team members

Need to improve:

- Try different team-building activities
- Update Github project boards



# Thank You !

---

by team 13: **The Tensor Titans**

## Questions ?

Carnegie Mellon University  
**HeinzCollege**

