

MovieLens

Introduction to Deep Learning

Qiuyu Huang

2025-09

CONTENT

01

Project Background &
Objectives

Deep learning
recommender on sparse
MovieLens dataset

02

Methodology & Workflow

Data prep, models,
and evaluation
pipeline

03–08

Notebook Overview

Splits & Samples →
Baselines (EDA) →
Deep Models →
Training & Evaluation
→ Ablations &
Debiasing → Error
Analysis & Conclusion

09

Final Takeaways


Key results and
insights

01 Project Background & Objectives

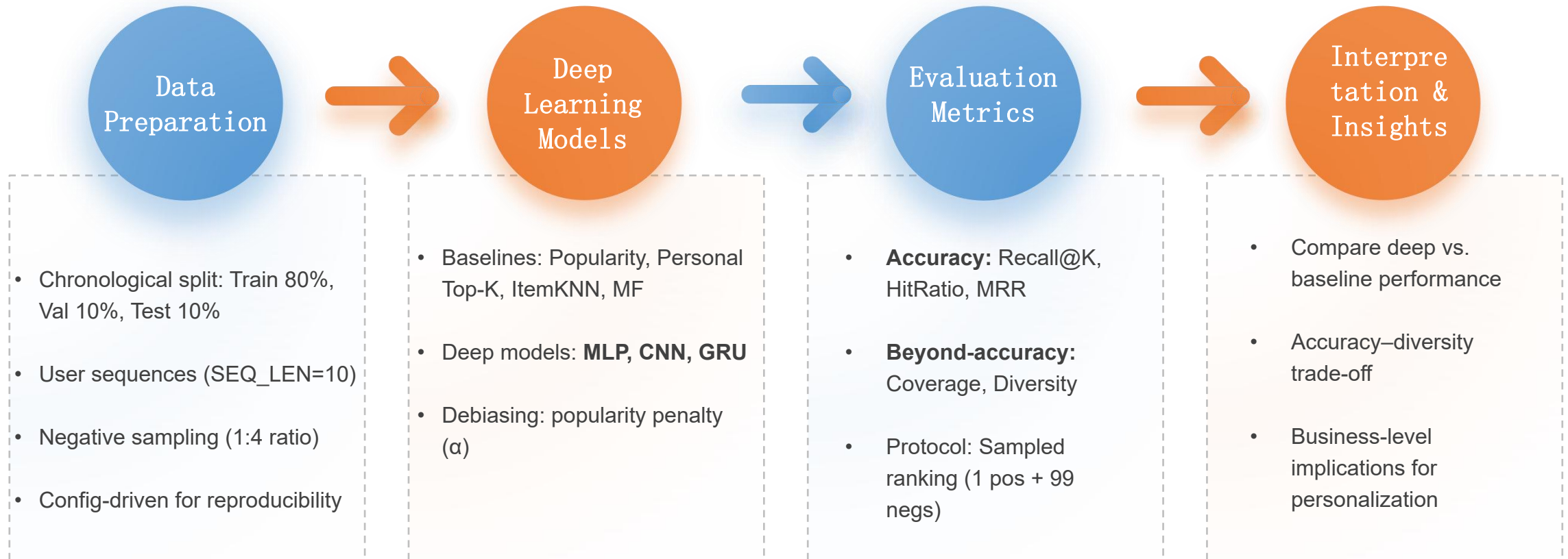
Background:

- MovieLens dataset with **implicit feedback only** (watch history, no ratings)
- Highly sparse user–item matrix (~97% empty)
- Key question: under such extreme sparsity, can we still make effective recommendations?

Objectives:

- Compare **deep sequential models** (MLP, CNN, GRU)
 - Evaluate **popularity bias** in recommendations
 - Explore **debiasing strategies** to improve diversity & personalization
- 

02 Methodology & Workflow



03-08 Instructions

Sections 03–08 are shown directly in the Jupyter Notebook:

03

Splits & Samples

Data overview,
train/val/test split,
sequence construction,
negative sampling

04

Baselines (EDA)

Popularity, Personal
Top-K, ItemKNN, MF
performance

05

Deep Models

MLP, CNN, GRU results
and comparisons

06

Training & Evaluation

Learning rates,
sampled ranking
protocol, metrics
(Recall@K, HR, MRR)

07

Ablations & Debiasing

SEQ_LEN, NEG_RATIO,
embedding size;
popularity penalty α

08

Error Analysis & Conclusion

Popular vs long-tail
distribution; final
findings and
implications

09 Final Takeaways

Key Findings

- GRU = best balance (recall + interpretability)
- Debiasing ($\alpha=0.1$) \uparrow coverage to 82% w/o loss
- Long-tail share 77.6% vs. 22.4% popular

Business Value

- Beyond popularity bias
- Broader personalization
- Long-tail boosts retention

Future Directions

- Balance accuracy, diversity, fairness

Deep learning improves accuracy while expanding long-tail coverage.