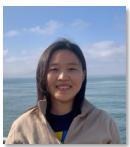
CIKM 2024
BOISE
OCTOBER 21-25

Calibration-Disentangled Learning and Relevance-Prioritized Reranking for Calibrated Sequential Recommendation


Hyunsik Jeon, Se-eun Yoon, Julian McAuley

UC San Diego

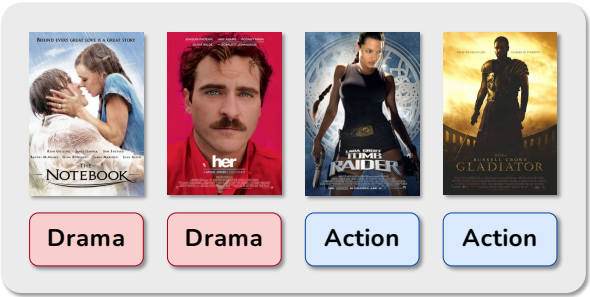
Introduction

Proposed Method

Experiments

Conclusion and Future Work

Recommendation

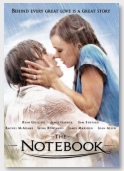


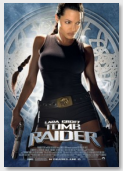
$$argmax\ p(x_t|x_{<t})$$

The Last Samurai | Action

Accurate!

Calibrated Recommendation





Drama | Drama | Action | Action

$argtop_k\ p(x_t|x_{<t})$

Accurate!

Narrow!

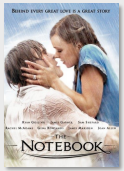
The Last Samurai	Action
Brave Heart	Action
Mad Max	Action
300	Action

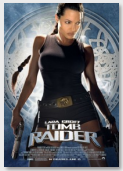
Accurate!

Balanced!

The Last Samurai	Action
Brave Heart	Action
Atonement	Drama
La La Land	Drama

Calibrated Sequential Recommendation





Drama | Drama | Action | Action

Time

$argtop_k\ p(x_t|x_{<t})$

Accurate!

Statically balanced!

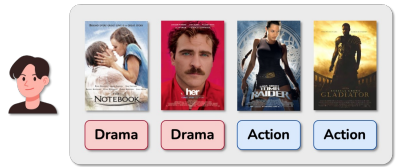
The Last Samurai	Action
Brave Heart	Action
Atonement	Drama
La La Land	Drama

Accurate!

Sequentially Balanced!

The Last Samurai	Action
Brave Heart	Action
Atonement	Drama
Mad Max	Action

Calibrated Sequential Recommendation



The Last Samurai	Action
Brave Heart	Action
Mad Max	Action
300	Action

Miscalibration

The Last Samurai	Action
Brave Heart	Action
Atonement	Drama
La La Land	Drama

Static Calibration

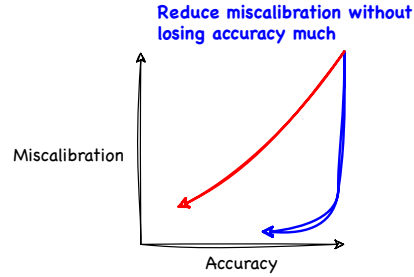
Desired

The Last Samurai	Action
Brave Heart	Action
Atonement	Drama
Mad Max	Action

Sequential Calibration

Problem Definition

- Given user u's interactions:
$$\mathcal{S}^u = (s_1^u, s_2^u, \dots, s_T^u)$$
  - s_t^u is user u's interacted item at t
- The goal is to recommend an item list at $T+1$:
$$\mathcal{R}^u = (r_1^u, r_2^u, \dots, r_K^u)$$
  - r_k^u is kth recommended item for user u
- Desired performance
  - High accuracy (i.e., nDCG)
  - Low miscalibration



Category distribution in user history (recent interactions are more weighted)

$$\mathcal{S}_{KL}(u) = KL(p\|\tilde{q}) = \sum_{c \in C} p(c|u) \log \frac{p(c|u)}{\tilde{q}(c|u)}$$

Category distribution in recommendation

Limitations of Previous Methods

- Most of previous methods have focused only on reranking
  - CaliRec: a greedy approach
  - MIP: a mixed integer programming
  - MCF: a minimum-cost flow algorithm

What if the backbone model's training is not aligned with the reranking objectives? → Improving calibration may lead to a **significant loss in accuracy**.

- A recent method, DACSR, utilized an end-to-end approach, but the calibration is optimized for the entire items

It cannot guarantee the calibration performance for **top-k recommendations**

Introduction

Proposed Method

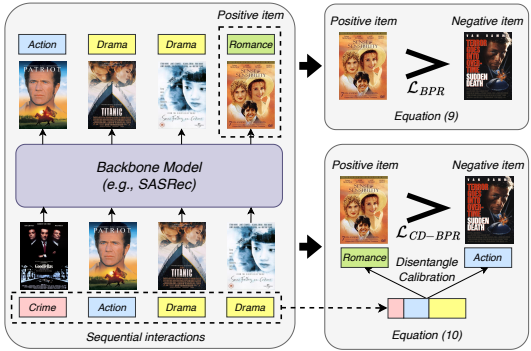
Experiments

Conclusion and Future Work

Overall Process

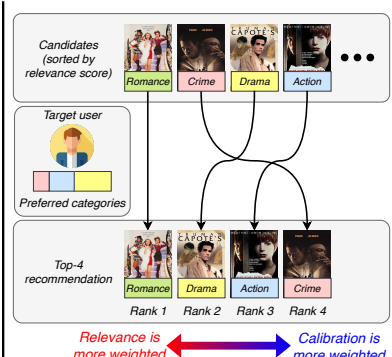
- We propose **LeapRec** (Calibration-Disentangled Learning and Relevance-Prioritized Reranking)

Learning process



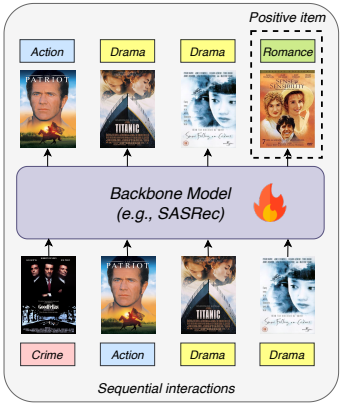
(a) Calibration-Disentangled Learning-to-Rank

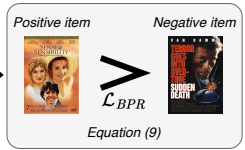
Reranking process



(b) Relevance-Prioritized Reranking

Learning to Rank





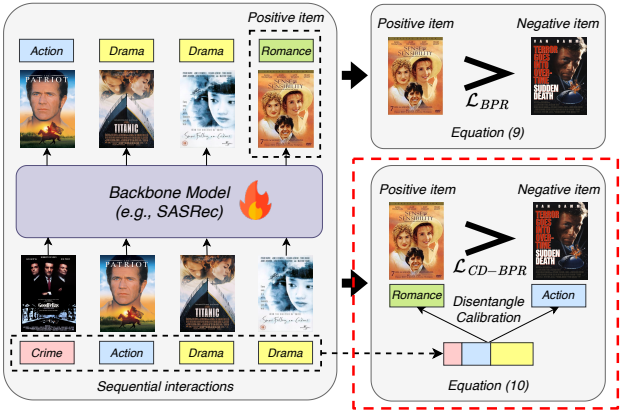
$$p(s_{T+1}^u = v | \mathcal{S}^u),$$

where $\mathcal{S}^u = (s_1^u, s_2^u, \dots, s_T^u)$ is a sequential history and v is an item

What if the calibration is applied in the reranking phase?

We cannot assure that the positive item remains in higher ranking

Calibration-Disentangled Learning-to-Rank

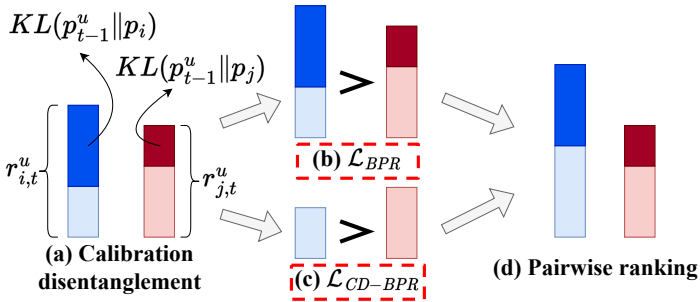


This interaction indicates that the user prefers the positive item over the negative item, **even considering their categories**

We disentangle miscalibration scores, and learn-to-rank using them as well

Calibration-Disentangled Learning-to-Rank

- Disentangle the calibration term and learn pairwise rankings



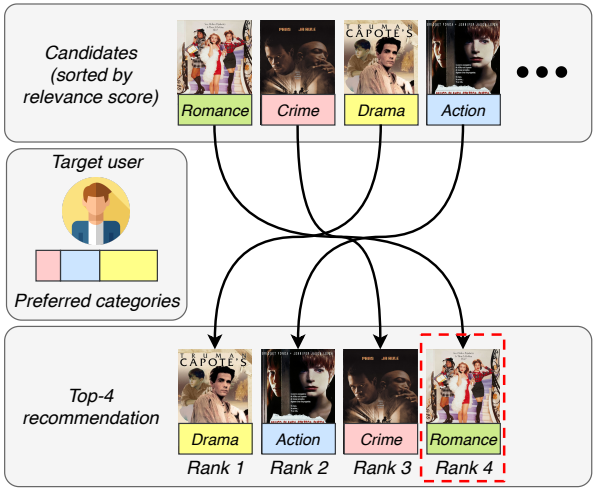
$$\mathcal{L}_{BPR} = r^u_{i,t} > r^u_{j,t}$$

$$\mathcal{L}_{CD-BPR} = r^u_{i,t} - KL(p^u_{t-1}||p_i) > r^u_{j,t} - KL(p^u_{t-1}||p_j)$$

$$\mathcal{L} = \mathcal{L}_{BPR} + \gamma\mathcal{L}_{CD-BPR}$$

$r^u_{i,t}$: score between user u and item i at step t

Reranking



$\lambda \in (0, 1)$

Linear interpolation

$$\max_{\mathcal{R}^u, |\mathcal{R}^u|=K} \left(\overline{(1-\lambda)} \sum_{i \in \mathcal{R}^u} r^u_{i,T+1} - \overline{\lambda} \mathcal{S}_{KL}(u) \right)$$

Greedily select top-k items

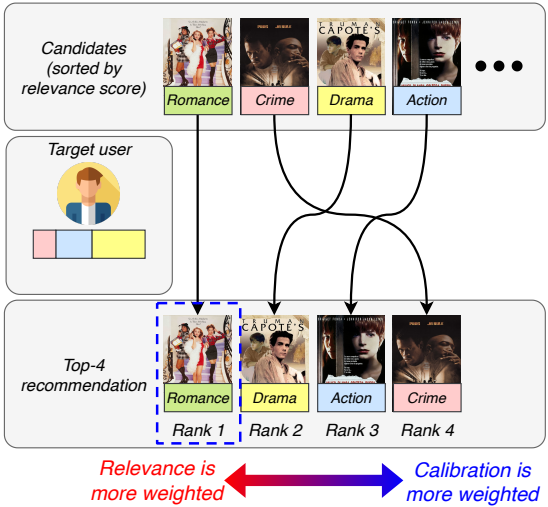
Relevance term

Calibration term

CaliRec (Steck)

Highly relevant items might be low -ranked because of the calibration

Relevance-Prioritized Reranking



$\lambda \in (0, 1)$ Relevance-prioritized interpolation

$$\max_{\mathcal{R}^u, |\mathcal{R}^u|=K} \left(\overline{(1-\lambda^{1/k})} \sum_{i \in \mathcal{R}^u} r^u_{i,T+1} - \overline{\lambda^{1/k}} \mathcal{S}_{KL}(u) \right)$$

Greedily select top-k items

Relevance term

Calibration term

In higher ranks, we prioritize relevance over calibration

Introduction

Proposed Method

Experiments

Conclusion and Future Work

Experimental Settings

- **Datasets**

Dataset	# Users	# Items	# Categories	# Interactions	Avg. sequence len.	User-item density	Avg. # categories
ML-1M	6,038	3,883	18	575,281	95.28	0.0245	1.6503
Goodreads	16,765	25,474	10	954,958	56.96	0.0022	3.6269
Grocery	54,882	39,853	26	438,681	7.99	0.0002	1.0000
Steam	242,223	14,419	22	2,732,749	11.29	0.0008	2.6242

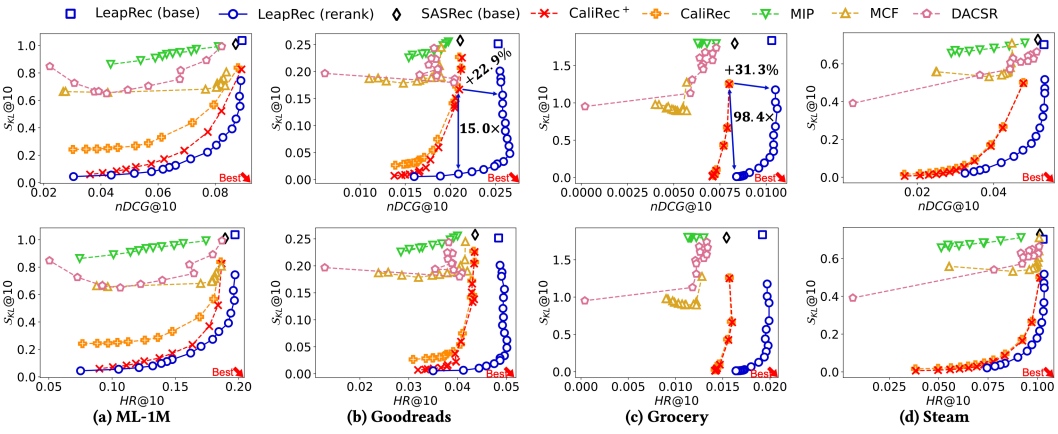
- **Evaluation Metrics**
  - nDCG@10 (↑): for accuracy
  - $\mathcal{S}_{KL}$@10 (↓): for calibration

Experimental Settings

- **Backbone model**
  - SASRec, Caser, and BERT4Rec

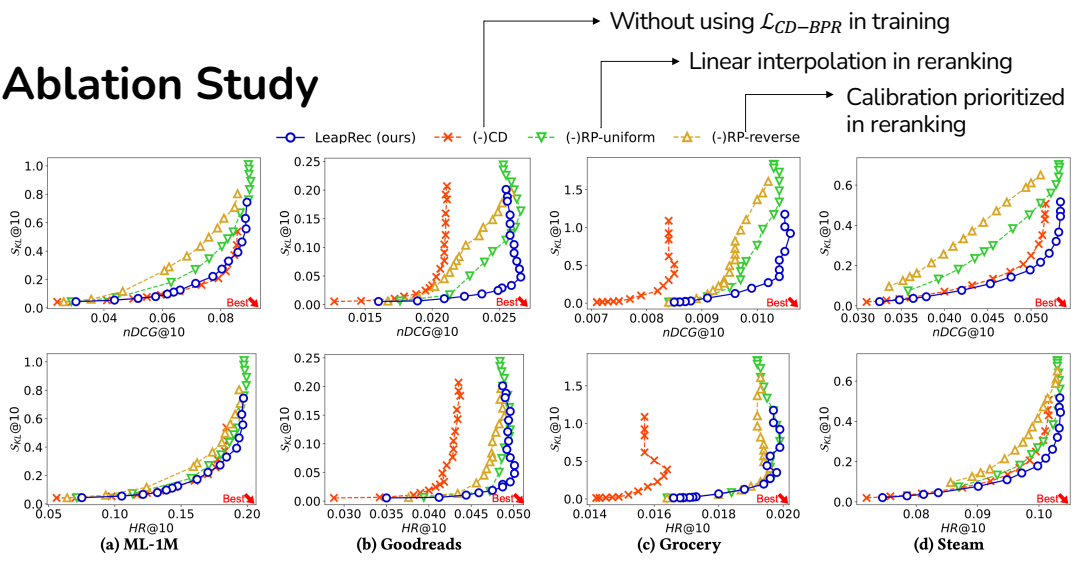
- **Baselines**
  - CaliRec: reranking-only (greedy), static calibration
  - CaliRec^{+}: reranking-only (greedy), sequential calibration
  - MIP: reranking-only (integer programming)
  - MCF: reranking-only (minimum-cost flow)
  - DACSR: end-to-end training

Performance Comparison



LeapRec (ours) outperforms the baselines by drawing way better trade-off curves!

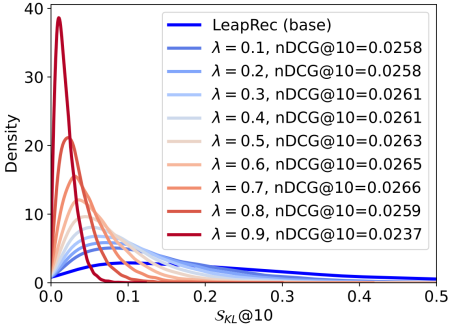
Ablation Study



All ideas in LeapRec (ours) help improve the performance!

Hyperparameter Effect

- Kernel density estimation (KDE) of $\mathcal{S}_{KL}$@10 on Goodreads

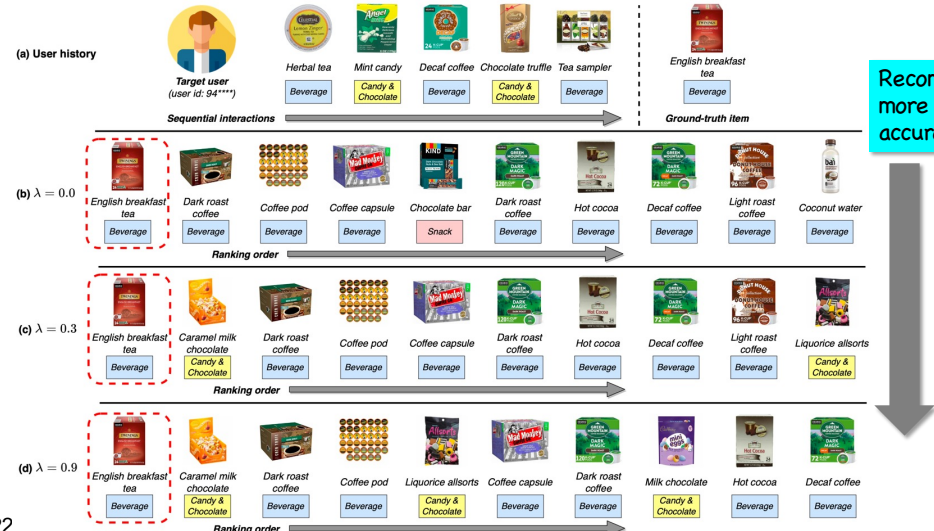


$$\max_{\mathcal{R}^u, |\mathcal{R}^u|=K} \left((1 - \lambda^{1/k}) \sum_{i \in \mathcal{R}^u} r^u_{i,T+1} - \lambda^{1/k} \mathcal{S}_{KL}(u) \right)$$

As λ increases, recommendations become more uniformly calibrated for entire users

The accuracy also increases until λ reaches 0.7

Case Study



Recommendation gets more calibrated while the accuracy is maintained

Case Study

Effectively reflects relevance even for an item that is from a non-interac ted category

Introduction

Proposed Method

Experiments

Conclusion and Future Work 

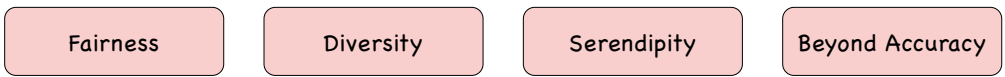
Summary

- **Calibrated sequential recommendation**: we focus on the problem that is practically crucial but not widely studied before
- **LeapRec (ours)**: the proposed method outperforms previous methods in extensive experiments
- **Further analysis**: we showed our main ideas help improve the performance and showed case studies to verify how it is practical

Future Directions

- Our reranking-aware learning approach could be a general solution to multi-objective recommendations

- Relevance priority is also important in multi-objective recommendations



Thanks!


https://arxiv.org/pdf/2408.02156


https://github.com/jeon185/LeapRec


https://www.linkedin.com/in/jeon185