

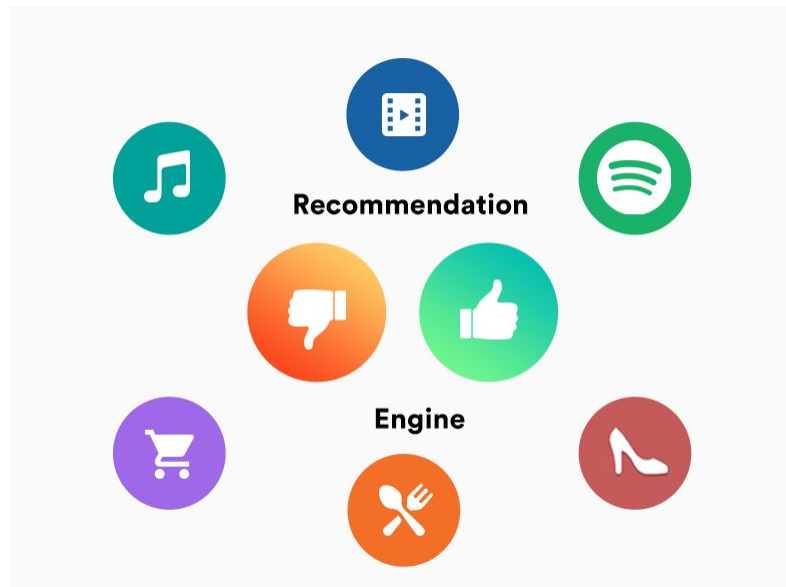


RecXPlainer: Post-Hoc Attribute-Based Explanations for Recommender Systems

Sahil Verma, Anurag Beniwal, Narayanan Sadagopan, Arjun Seshadri

TEA Workshop @ NeurIPS 2022

Goal: Explain recommendations



Explainability provides enhanced **trustworthiness, scrutability, and persuasiveness**

Kinds of Recommender Systems

- Content and Collaborative filtering are the two broad approaches of generating recommendations
- Collaborative filtering (CF) is more widely used
- CF methods pose difficulty in understanding recommendations

Previous techniques

Most previous explainability approaches provide explanations in two formats:

- **User-based**: Explains an item as it was liked by a similar user
- **Item-based**: Explains an item as it is similar to a previously liked item

However, neither of these formats capture the preference of a user over the **attributes of recommended item** -- which is how users inherently think [1, 2].

1. Julian McAuley and Jure Leskovec. 2013. *Hidden factors and hidden topics: understanding rating dimensions with review text*. In *Proceedings of the 7th ACM conference on Recommender systems (RecSys '13)*
2. X. Wang, Y. Chen, J. Yang, L. Wu, Z. Wu and X. Xie, "A Reinforcement Learning Framework for Explainable Recommendation," *2018 IEEE International Conference on Data Mining (ICDM), 2018*

Attribute-based Explanations

A recommendation that is explained in terms of the attributes that a user prefers, for e.g., ‘This movie is recommended because you like *Crime and Horror* movies’, or ‘This shirt is recommended because you like *black shirts* and *Adidas products*’.

Such explanations are personalized to the user and hence enhance the persuasiveness and effectiveness of the recommender system further.

Constraints: **Post-Hoc** and **Model Agnostic**

We want the method to be *post-hoc* and *model-agnostic* because:

- No need to **retrain or re-architecture** the recommender system
- No **accuracy-interpretability trade-off** debate
- **Applicability** to a broad class of recommender systems

Hence such explainability techniques are highly desirable in industrial settings.

Summarizing Goals

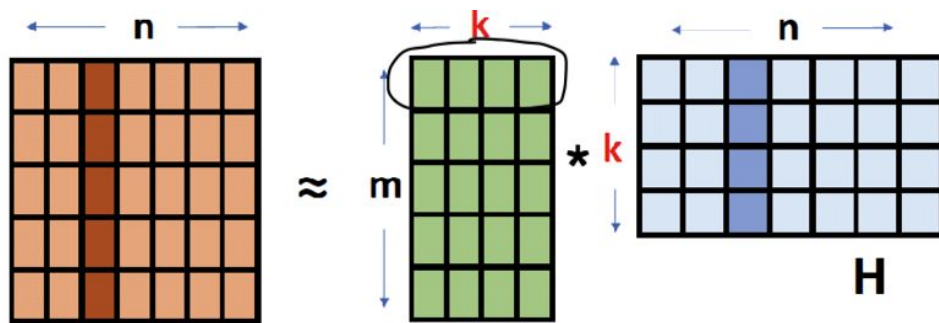
We aim to develop an explainability technique that:

- Generates attribute-based explanations
- Is post-hoc
- Is model-agnostic

RecXplainer

RecXplainer: Inputs

1. User embedding
2. Item attributes



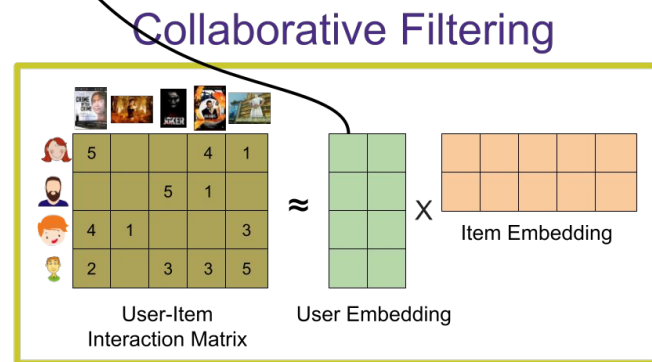
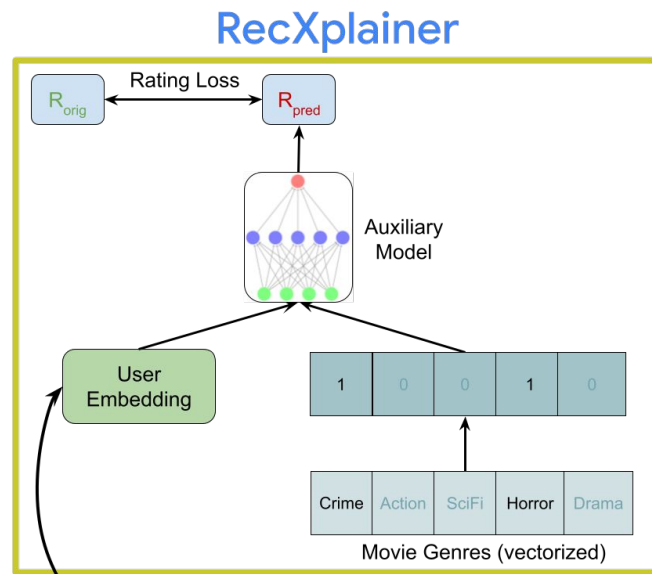
Consider the case of movie recommendation where all movies belong to one or more genres from a list of genres



RecXplainer: Architecture

We train an auxiliary model that takes the two inputs and is trained over all the training datapoints.

The goal is produce ratings as close as possible to what the user originally rated.



RecXplainer: Generating Explanations


Once trained, we use the **leave-one-out approach** to assess attribute impact

- Zero-out one attribute at a time and measure drop in model rating (its impact)
- Sort the attributes by their impact

RecXplainer: Generating Explanations

Consider a movie whose genres are *Horror*, *Crime*, and *Drame*

For this user, *Crime* is most important, followed by *Horror*, and *Drama*

Attribute Zeroed	Predicted Rating	
None	4.2	--
<i>Horror</i>	3.7	0.5
<i>Drama</i>	3.9	0.3
<i>Crime</i>	3.2	1.0

Providing Explanations

RecXplainer generates two kinds of explanations:

- **Specific preferences:** These are the attributes sorted in order of impact for a specific user-item pair
- **General preferences:** These are the attributes that describe the general preference of a user over all the attributes in the dataset

General preferences

Liked Movies	Horror	Drama	Crime	...
<i>Movie #1</i>	0.5	0.3	1.0	
<i>Movie #2</i>	0.3	0.5	0.7	
<i>Movie #3</i>	0.8	1.0	0.4	
...	0.1	0.1	1.1	
<i>Average</i>	0.4	0.5	0.8	

Experiments

Experiments: Metrics

We evaluate RecXplainer over the following metrics:

- *Test coverage*: How many liked test items can the top-3 general preferences explain?
- *Top-k recommendations coverage*: How many top-k recommendations can the top-3 general preferences explain?
- *Personalization of the explanations*: How personalized are the explanations?

Personalization of the explanations

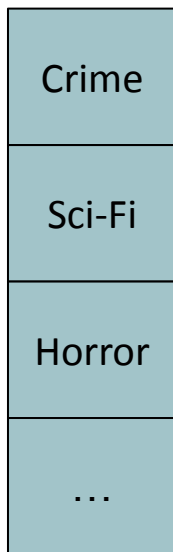
We use two proxies for user attribute preferences:

- Conditional probability of liking an item given an attribute is present
- Odds of liking v.s. disliking an item given an attribute is present

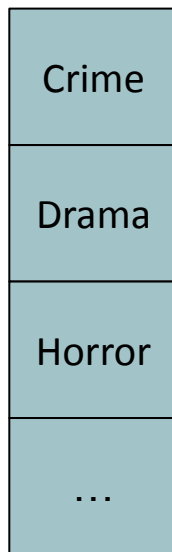
Given the two proxies for attribute preferences, we measure the coverage and ranking for the specific preferences of the liked items and general preferences of a user (for top-3 attributes) -- a total of 8 metrics

Personalization of the explanations

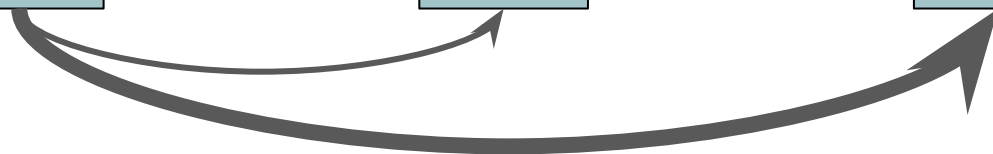
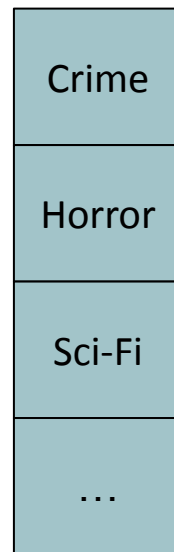
Odds Liking v.s.
Disliking



Specific
Preference



General
Preference



Experiments: Baselines

Previous techniques:

- **LIME-RS**: The only previous attribute-based post-hoc explainability technique
- **AMCF**: Another attribute-based explanation technique (not post-hoc)
- **AMCF-PH**: AMCF technique adapted to be post-hoc

We also compare to the often overlooked popularity approaches:

- **Global Popularity** of attributes
- **User-specific popularity** of attributes

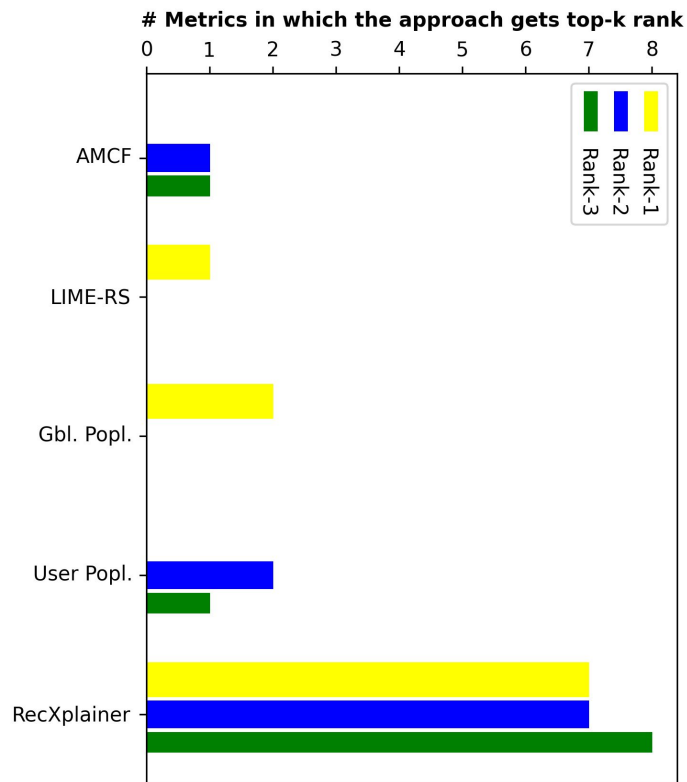
And a **random** baseline for control.

Experiments: Results

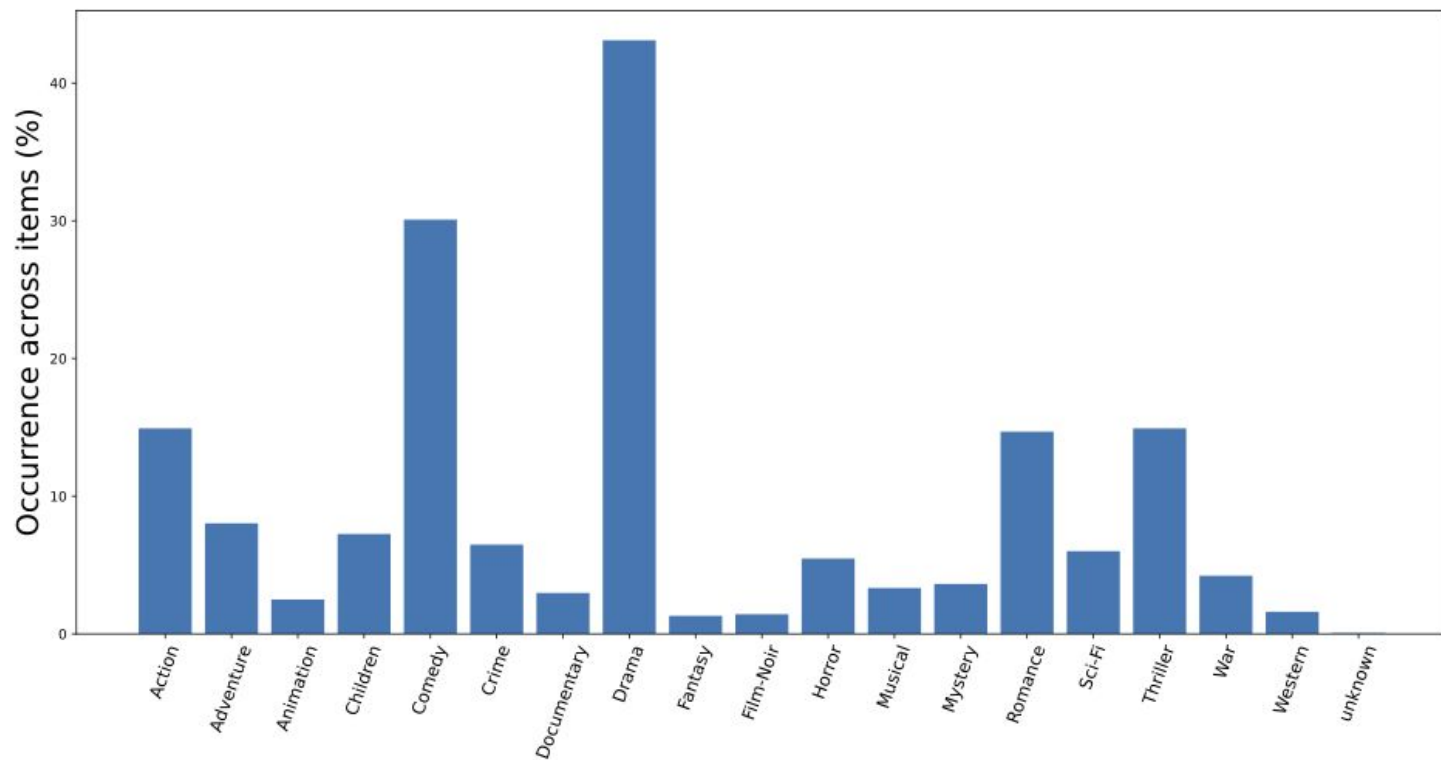
For RecXplainer, we used three auxiliary models: **Linear**, **2-layer NN: MLP1**, and **4-layer NN: MLP2**. These results are for **MovieLens-100K** dataset. This dataset has 100K ratings given by 1000 users for 1700 movies. Each movie has one or more of the total 18 genres in the dataset: *Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western*

Metrics	LIME-RS	AMCF	AMCF-PH	GBL Popl	User Popl.	Random	Our-Linear	Our-MLP1	Our-MLP2
Testset Coverage	8.4	47.2	52.6	84.5	77.9	32.4	60.7	70.8	71.2
Recommendations Coverage	26.0	44.3	46.1	73.2	69.6	30.6	61.3	68.6	67.8
CondProb Generalpref Coverage	50.3	58.2	47.2	29.0	41.8	43.9	62.9	58.1	56.6
CondProb Generalpref Ranking	15.8	15.1	11.2	5.8	9.9	11.0	15.2	14.0	13.8
CondProb Specificpref Coverage	52.5	53.1	52.4	29.0	41.7	44.1	55.3	53.8	53.6
CondProb Specificpref Ranking	14.6	49.6	50.3	6.8	10.4	11.0	52.2	50.7	51.2
Odds Generalpref Coverage	32.6	63.7	60.4	48.5	55.0	43.7	73.4	72.3	70.4
Odds Generalpref Ranking	9.4	19.4	18.0	18.2	26.7	10.6	25.6	27.6	28.2
Odds Specificpref Coverage	37.2	55.8	56.8	48.5	55.0	44.2	59.8	59.2	59.5
Odds Specificpref Ranking	10.2	49.7	50.7	19.6	27.2	11.0	50.8	51.6	51.7

Experiments: Results



Genre Skew



Discussion

- LIME-RS: Trails our approach in 9 out of 10 metrics.
- AMCF and AMCF-PH: Both approaches trail our approach in all 10 metrics.
- Global and user-specific popularity: Global and user-specific popularity perform the best for test and top-k recommendations coverage, however, they are trail our approach in all personalization metrics.

Conclusion and Future Work

We conclude that our approach strikes a great balance between coverage and personalization of the explanations.

We plan to add experiments on various datasets:

- Movielens-1M
- Anime
- Yelp

And architectures:

- Factorization machines
- Auto-encoder based recommender systems

Thank you



vsahil@cs.washington.edu

