# Explainable, Transparent, and Steerable Recommender Systems

**[DAT640] Information Retrieval and Text Mining**

**Krisztian Balog**
University of Stavanger

October 29, 2024

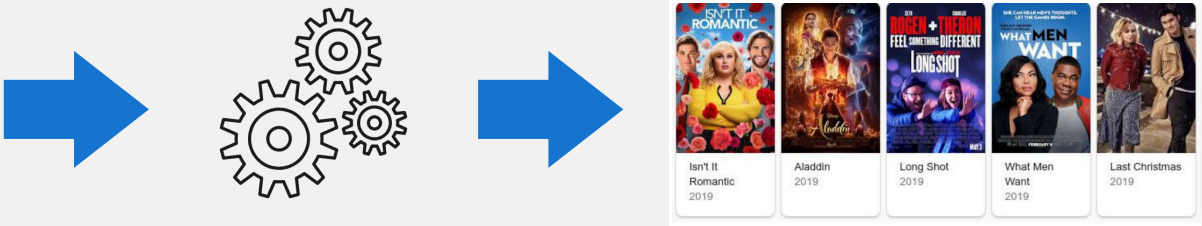
# Explaining Preferences and Recommendations

based on:
K. Balog, F. Radlinksi, and S. Arakelyan. Transparent, Scrutable and Explainable User Models for Personalized Recommendation. *SIGIR '19.*

# Motivation

## Classic recommendation

| | |
|---|---|
| Alien | ☆☆☆☆☆ |
| Sleepless in Seattle | ☆☆☆ |
| Star Wars IV | ☆☆☆☆☆ |
| Pride and Prejudice | ☆☆☆☆ |
| … 147 more movies … | |
| Crazy Rich Asians | ☆☆☆☆ |
| Mamma Mia | ☆☆☆☆ |

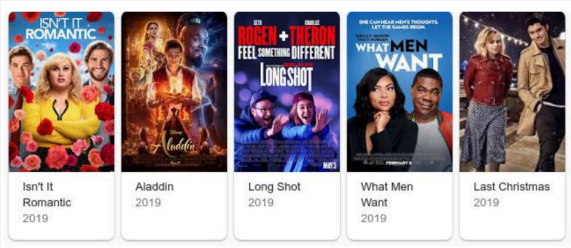


# Motivation

## Transparent and explainable user model

| | |
|---|---|
| Alien | ☆☆☆☆☆ |
| Sleepless in Seattle | ☆☆☆ |
| Star Wars IV | ☆☆☆☆☆ |
| Pride and Prejudice | ☆☆☆☆ |
| … 147 more movies … | |
| Crazy Rich Asians | ☆☆☆☆ |
| Mamma Mia | ☆☆☆☆ |

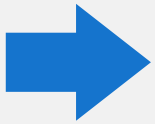
You like science fiction movies, especially thrillers
You like romantic comedies, unless they star Meg Ryan
You like movies with space battles
You like dramas if they are set in England



# Motivation

## Transparent and explainable user model

| | |
|---|---|
| Alien | ☆☆☆☆☆ |
| Sleepless in Seattle | ☆☆☆ |
| Star Wars IV | ☆☆☆☆☆ |
| Pride and Prejudice | ☆☆☆☆ |
| … 147 more movies ... | |
| Crazy Rich Asians | ☆☆☆☆ |
| Mamma Mia | ☆☆☆☆ |

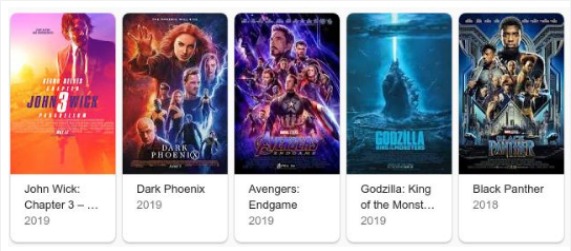


You like science fiction movies, especially thrillers

~~You like romantic comedies, unless they star Meg Ryan~~

You like movies with space battles

You like dramas if they are set in England



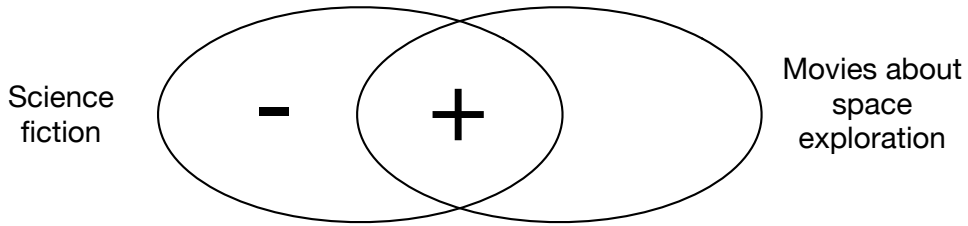
# How to explain user preferences and recommendations?

# Approach overview

1. Model user preferences based on sets (tags)
2. Generate a scrutable natural language summary of user preferences
3. Recommend items based on associated tags

# Modeling user preferences

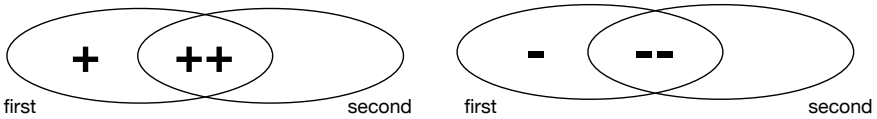
- The user model is a weighted set of tags
- The weight of a tag is the average rating of movies with that tag
- However... single tags are not rich enough to capture realistic user interests → set interactions



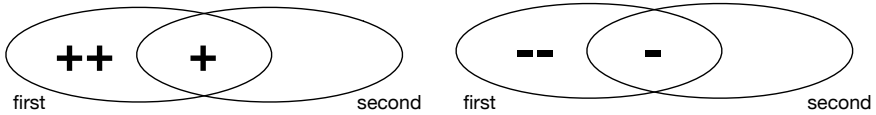

*“You don’t like science fiction movies unless they are about space exploration.”*

# Pairwise set interactions

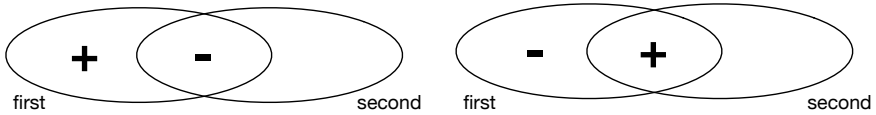
You (don’t) like ***first*** especially if ***second***.



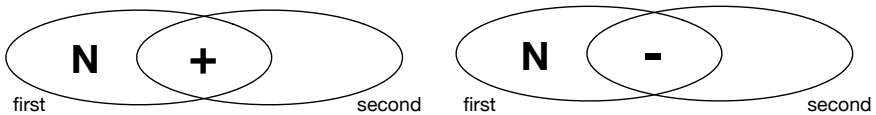

You (don’t) like ***first*** especially if not ***second***.




You (don’t) like ***first*** unless ***second***.




You (don’t) like ***first*** if ***second***.



# Generating a summary of user preferences

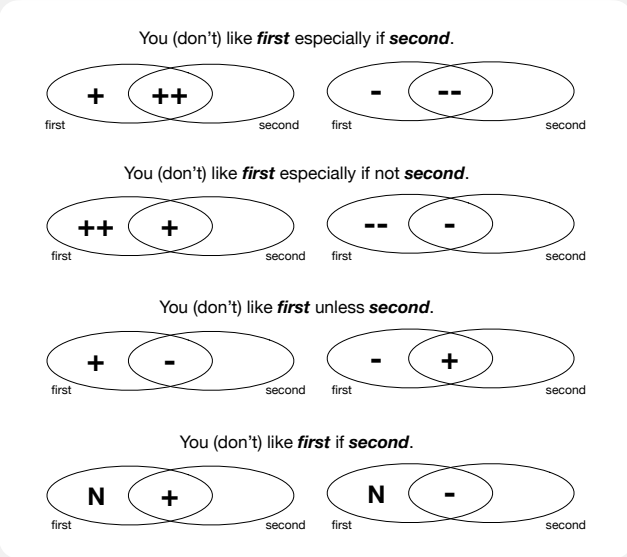
- An explainable user model (*summary of preferences*) is constrained by size
- Need to select which tags (and pairwise tags) should be included in this summary
  - Candidate statements:
    - Tag must apply to a min. number of items (i.e., generalizable)
    - The set of ratings for the tag is statistically significantly different from zero (neutral rating)
  - Select top-k statements based on their utility
    - Utility = the weight of the tag in the user model, corrected for coverage and significance

| Tag | Weight |
|---|---|
| cult film | 0.83 |
| cult film + dark | 0.16 |
| action | 0.44 |
| action + predictable | -0.11 |
| action + suspense | 0.62 |
| violence | -0.76 |
| … | |

*Example of preferences in the user model*

# Generating textual representations

- Using the templates from pairwise set interactions
- Only two grades: “like” and “don’t like”
  - Intensity could be accentuated (“love,” “hate,” etc.)
- Additionally select a representative example, to clarify what we mean by that tag




Example:
- You like movies that are tagged as ‘thought-provoking’, especially those that are tagged as ‘twist ending’, such as Fight Club.
- You like movies that are tagged as ‘heist’, such as Ocean’s Eleven.
- You like movies that are tagged as ‘based on a comic’, such as Iron Man.
- You don’t like movies that are tagged as ‘cheesy’, such as Who Framed Roger Rabbit?

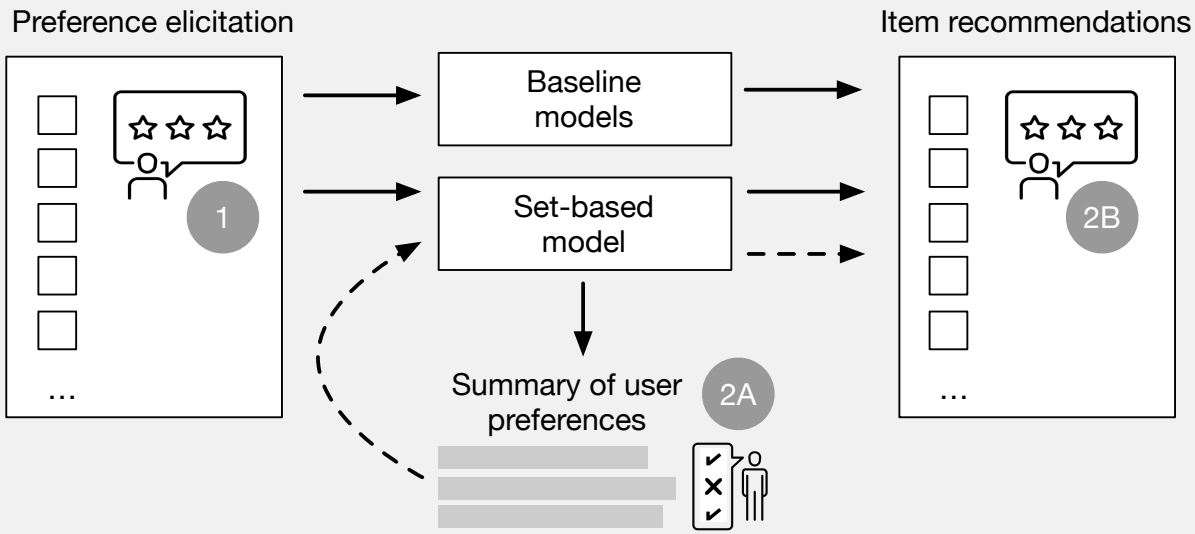
# Recommendation model

- Simple content-based approach that matches tags assigned to items against tag preferences of the user
  - We lose data by throwing away individual ratings, but smoothing should help
- Additionally, incorporate how many users liked/disliked a given item (collaborative filtering element)
- Why a given recommendation is made translates directly into a subset of the user model

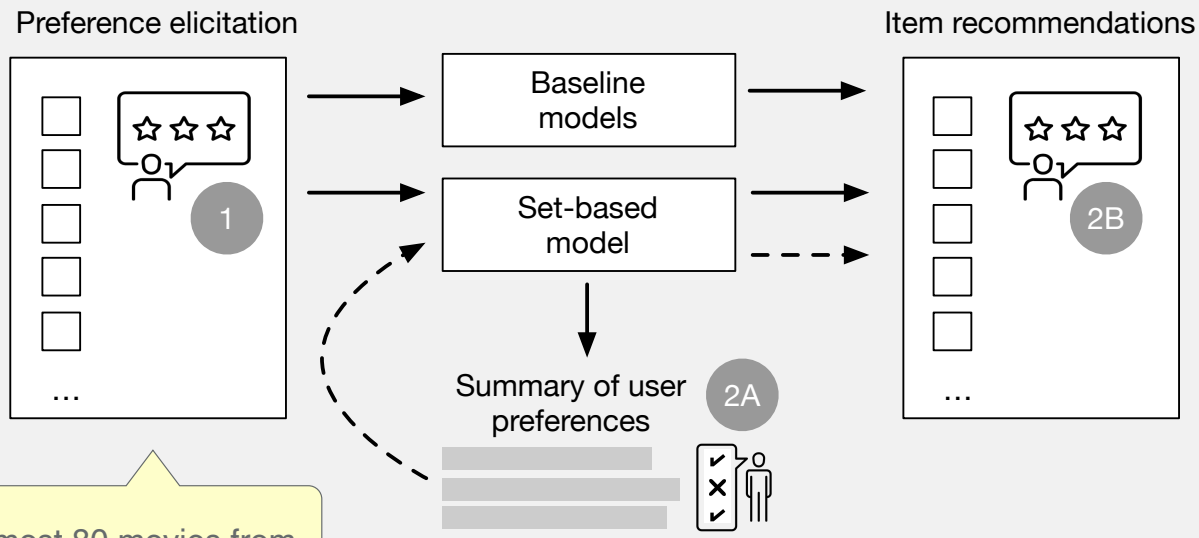
Recommendation: Inception (2010)

User model:
- You like movies that are tagged as ‘thought-provoking’, especially those that are tagged as ‘twist ending’, such as Fight Club.
- You like movies that are tagged as ‘heist’, such as Ocean’s Eleven.
- You like movies that are tagged as ‘based on a comic’, such as Iron Man.
- You don’t like movies that are tagged as ‘cheesy’, such as Who Framed Roger Rabbit?

# User study

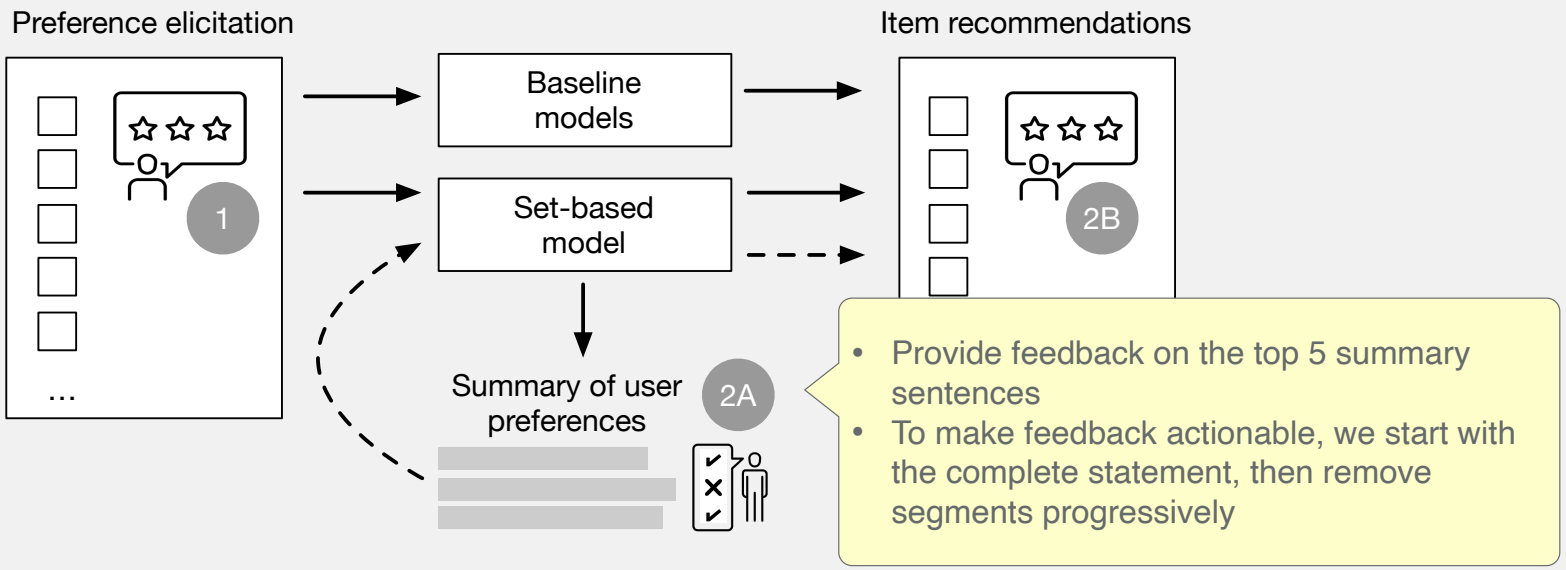

Preference elicitation
Item recommendations
Baseline models
Set-based model
Summary of user preferences
1
2A
2B
...
...

# User study




- Rate at least 20 and at most 80 movies from an elicitation set (400 movies)
- Three-point scale (dislike, neutral, like)
- Also requested to specify what they liked/ disliked about the movie

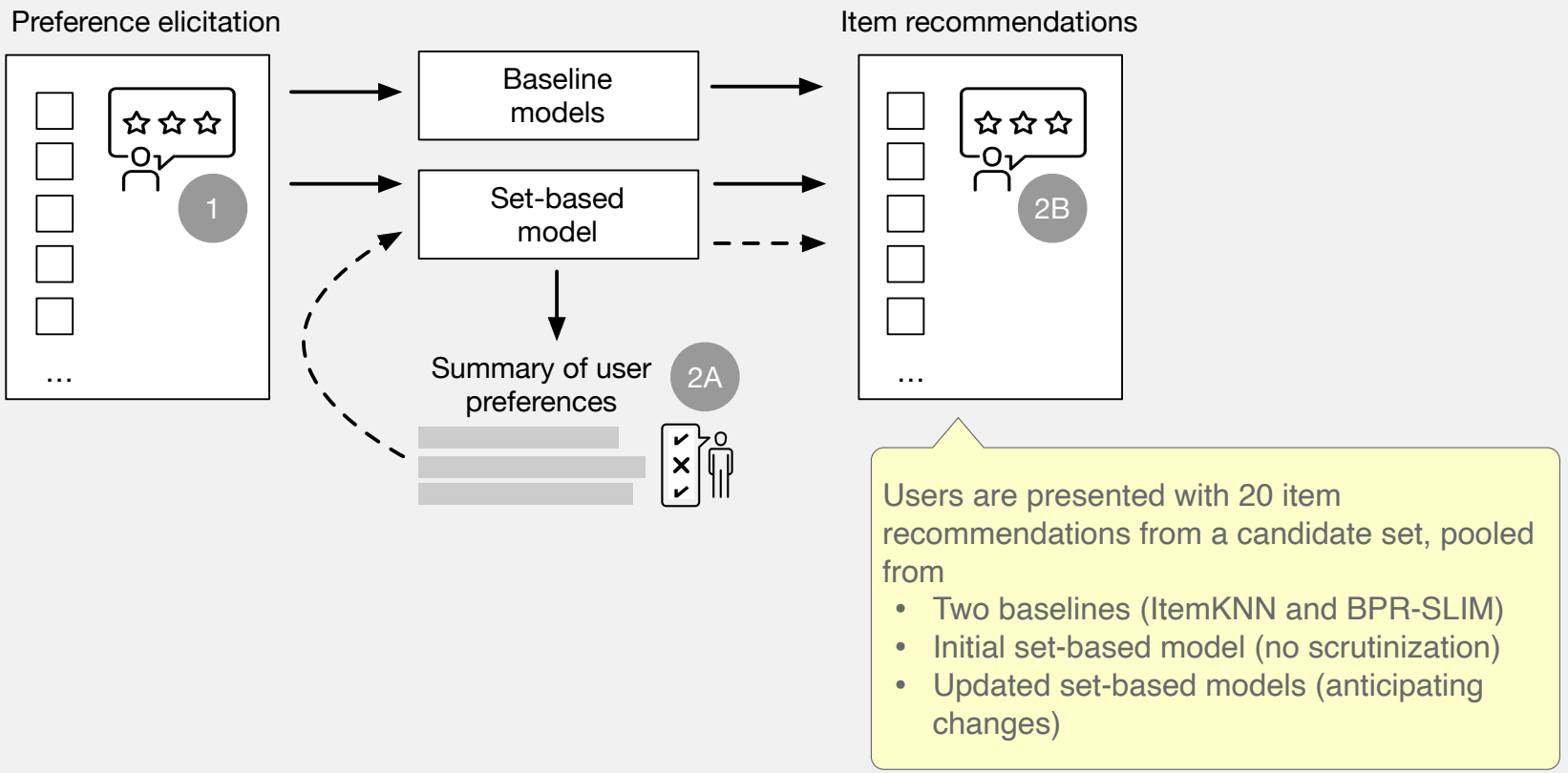
# User study




Select the first statement that best describes your movie preferences.

- You like movies that are tagged as `action', especially those that are tagged as `sword fight', such as The Princess Bride.
- You like movies that are tagged as `action', especially those that are tagged as `sword fight'.
- You like movies that are tagged as `action', such as The Princess Bride.
- You like movies that are tagged as `action'.
- You like The Princess Bride.

# User study


Preference elicitation
Baseline models
Set-based model
Item recommendations
1
2B
2A
Summary of user preferences
...
...
Users are presented with 20 item recommendations from a candidate set, pooled from
• Two baselines (ItemKNN and BPR-SLIM)
• Initial set-based model (no scrutinization)
• Updated set-based models (anticipating changes)

# Results

| Method | MRR | MAP | NDCG@5 | NDCG@10 |
|---|---|---|---|---|
| MostPopular | 0.882 | 0.515 | 0.721 | 0.628 |
| Item-kNN | 0.479 | 0.245 | 0.452 | 0.364 |
| BPR-MF | 0.709 | 0.378 | 0.624 | 0.511 |
| *Set-based model* | | | | |
| Full transparency, no priors | 0.710 | 0.393 | 0.543 | 0.499 |
| Full transparency, priors | 0.835 | 0.529 | 0.748 | 0.643 |
| Partial transparency, no priors | 0.748 | 0.516 | 0.663 | 0.648 |
| Partial transparency, priors | 0.866 | 0.554 | 0.782 | 0.670 |

Full transparency: recommendations are based on a user model comprised of at most five statements

Partial transparency: recommendations are based on all candidate statements

Key takeaway

It is possible to make a recommender system explainable and transparent without sacrificing effectiveness

# Summary

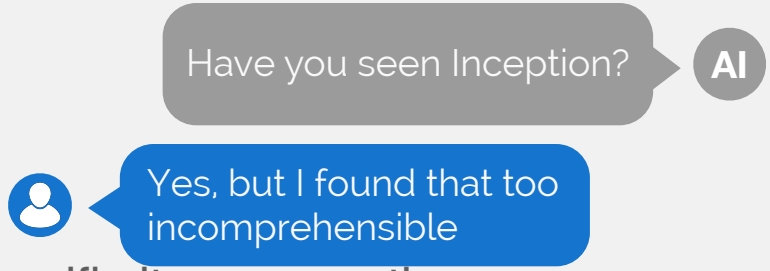
- It is possible to be  explainable and transparent without sacrificing recommendation accuracy
- Future work
  - From template-based to model-based generation of explanations
  - Generalization to other domains and ultimately to arbitrary interests and preferences in a PKG
  - Better utilization of user feedback

# Natural Language Critiques

based on:
K. Balog, F. Radlinksi, and A. Karatzoglou. On Interpretation and Measurement of Soft Attributes for Recommendation. *SIGIR '21.*

# Motivation

- Incorporate natural language feedback (critiques) on items, which often come in the form of *soft attributes*
  - For example, *originality* of a movie plot, *noisiness* of a venue, or the *complexity* of a recipe
  - Unconstrained natural language, without limiting to specific item properties




> *We define a soft attribute as a property of an item that is not a verifiable fact that can be universally agreed upon, and where it is meaningful to compare two items and say that one item has more of the attribute than another.*

# Soft attributes vs. social tags

- Soft attributes resemble social tags, BUT
  - Tags are binary labels that don’t allow for partial ordering of items
  - Most tagging approaches intentionally bias users towards a consistent vocabulary

| Examples of soft attributes | |
|---|---|
| artsy | feels real |
| light-hearted | predictable |
| incomprehensible | intense |
| simplistic script | violent |

**How to interpret and measure nuanced natural language critiques (*soft attributes*)?**

# Main contributions

- Development of a reusable test collection using a novel multi-stage crowd labeling mechanism
- (Quantification of the subjectivity (or "softness") of soft attributes)
- Methods for *soft attribute-based critiquing*:
  Ranking items relative to a given anchor item, w.r.t. a given soft attribute

# Creating a reusable test collection

Process

1. Sampling soft attributes
2. Obtaining ground truth item orderings for each attribute
3. Developing an appropriate evaluation measure (for ranking items w.r.t. a soft attribute)

Two types of evaluation collections

- Based on social tags → MovieLens Attribute Collection
- Based on soft attributes in a conversational dataset → Soft Attribute Collection
- Items: Top 300 most popular movies in the MovieLens-20M collection

# MovieLens Attribute Collection

- Soft attributes: Top 100 most frequent MovieLens tags
  - Excluding tags that are named entities, refer to adult content, or contain coarse language

| Examples of soft attributes | | | |
|---|---|---|---|
| action | comedy | depressing | dystopia |
| romance | sci-fi | superhero | suspense |

- Ground truth:
  - $\mathcal{X}^-$ : Items that have not been assigned that tag by any user
  - $\mathcal{X}^+$ : Items where a significant portion (0.15) of users who assigned any tag to the item assigned the given tag
  - Only items that are tagged by at least 50 users
- Evaluation measure: Goodman and Kruskal's gamma
  - Ranges from -1 (perfect inversion) to +1 (perfect agreement)

$$G = \frac{N_s - N_d}{N_s + N_d}$$

$N_s$: number of concordant pairs with respect to the ground truth ordering

$N_d$: number of discordant pairs with respect to the ground truth ordering

# Soft Attribute Collection

- Soft attributes: Sampled from a conversational movie dataset (CCPE-M [3])
  - Over 500 English dialogs between a user and an assistant discussing movie preferences in natural language
  - Dialogues are annotated with expressions of preferences; we filter for "more", "less", "too"
  - Soft attributes are extracted manually from those (173 in total — only 47% present as tags in MovieLens, 60 sampled for evaluation)

| Examples of soft attributes | | | |
|---|---|---|---|
| artsy | feels real | incomprehensible | intense |
| light-hearted | predictable | simplistic script | violent |

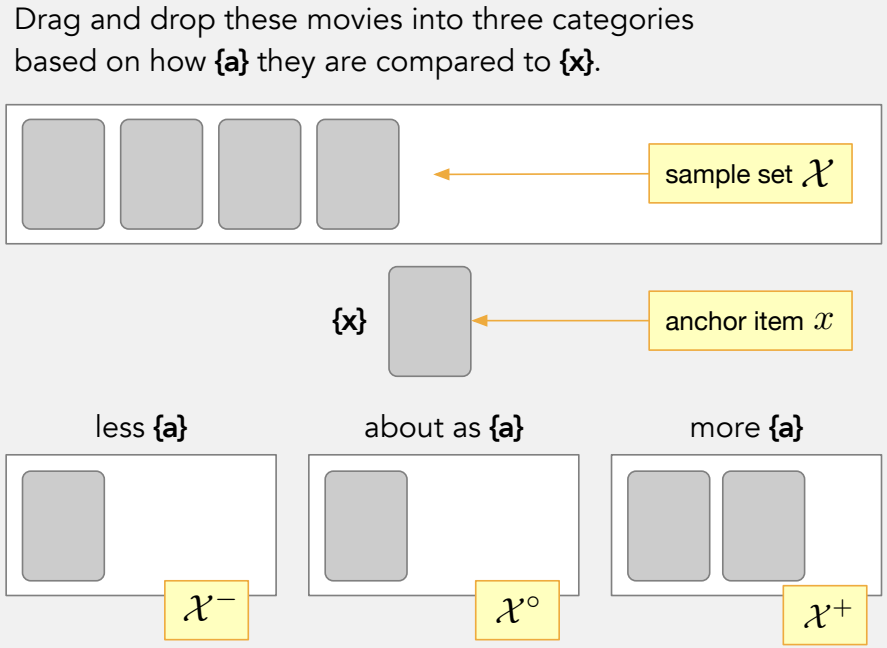
*Example partial exchange between a user and an agent about the user's movies interests*

| | |
|---|---|
| USER | Zodiac's one of my favorite movies. |
| USER | Zodiac the movie about a serial killer from the '60s or '70s, around there. |
| ASSISTANT | Zodiac? Oh wow ok, what do you like about that movie? |
| USER | And I just think serial killers, in general, are interesting, so the movie was really good. And it was just neat to see that world. Like it went really in-depth. It was like almost 3 hours long, so you got to really feel like you were a part of that world and time period. And see what the detectives and the police did and the investigators. |
| ASSISTANT | So you feel like you were part of that world ? |
| USER | Yeah. It was really an immersive movie. |
| ASSISTANT | If you were in the movie what character do you feel most relatable |
| USER | Probably the main character, Robert Graysmith, just cuz he's awkward and bumbly. So, I guess that's the one I would be the closest to. |
| ASSISTANT | What scene do you like the best? |
| USER | Probably the most memorable one is the murder at the lake. Just cuz it's really vivid and horrific to watch. But it's very memorable. |

[3] Radlinksi et al. Coached Conversational Preference Elicitation: A Case Study in Understanding Movie Preferences. *SIGDIAL 2018.*

# Soft Attribute Collection

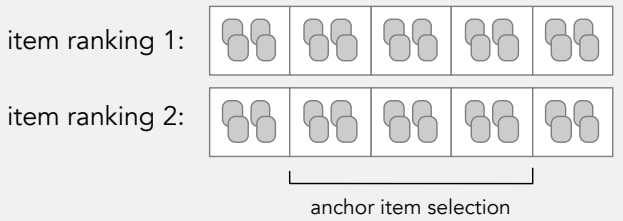
- Ground truth: Two-stage annotation procedure via crowdsourcing
  - Focus: efficient collection of relative preferences over pairs of items for the same soft attribute
  - Stage 1: Workers are asked to indicate which movies they have seen from a pool of items
  - Stage 2: Each worker is presented with a specific ordering task for each soft attribute



# Soft Attribute Collection

- The way items are selected influences the number of pairwise item preferences that can be obtained!


- Sampling methodology to increase the likelihood of meaningful comparisons between items (same pair of items ordered for a given soft attribute by multiple users):
  - Rank all seen items using two baselines and partition them into M=5 equal-sized bins
  - Anchor item should not be selected from the first or last bin
  - Anchor item selection is further biased towards popular items
  - $x_i$ is a stratified sample of one item from each of the M bins for each baseline ranking, biased towards more popular items



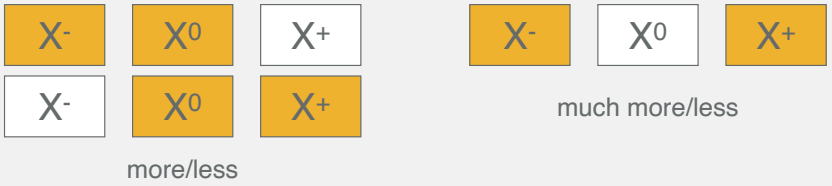
# Soft Attribute Collection

- Evaluation measure: Based on the number of pairs that are ordered in agreement with the ground truth
  - With three classes, we are able to measure agreement with a stronger metric
  - Extended version of Goodman and Kruskal's gamma rank correlation, differentiating between more/less and *much* more/less
  - Ranges -1 (perfect anti-correlation) to +1 (perfect correlation)

$$G' = \frac{(N_s - N_d) + 2(N_{ss} - N_{dd})}{(N_s + N_d) + 2(N_{ss} + N_{dd})}$$

number of concordant/discordant pairs with respect to the ground truth ordering

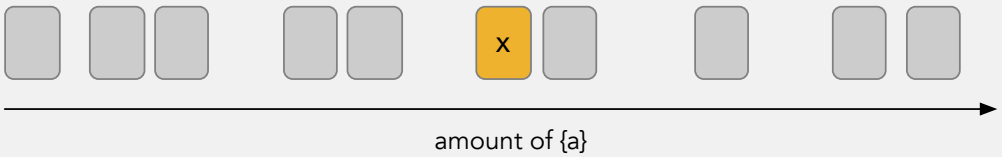
#identically/differently ranked items in X+ and X- compared to the ground truth



# How to apply critiques in terms of soft attributes in a recommendation setting?

# Soft attribute-based critiquing

- Critiquing we aim to support: Given a suggestion fo a particular item x made by the recommender system, the user may respond with *"show me more/less {a} than this"*
- Core idea:
  - It is sufficient to determine the relative ordering of items with respect to a soft attribute
  - Then, we can locate the anchor item in the ranking and move in the desired direction



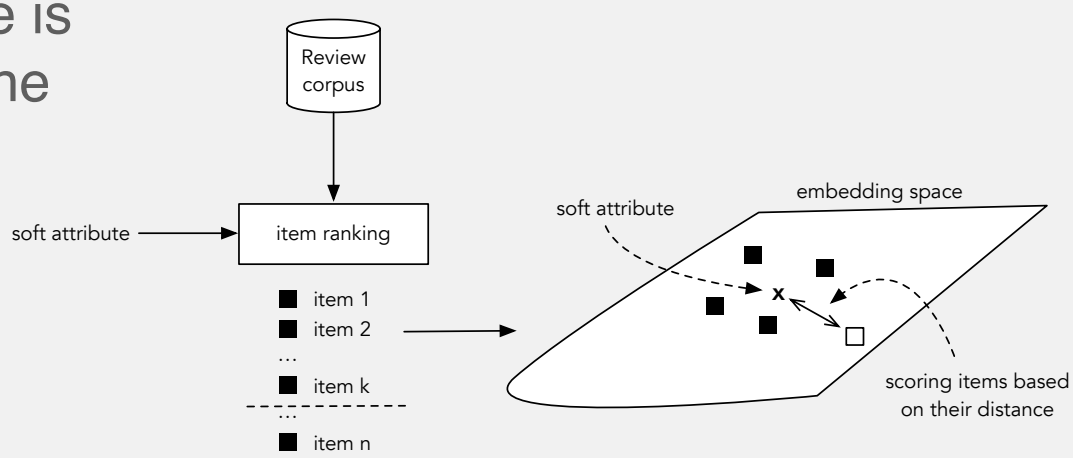

- Formal objective: devise a scoring function that can be applied to any item and soft attribute pair

# Approach

- Core question: How to represent items and soft attributes?
- Items are represented in a latent space by learning item embeddings from item-rating data (using matrix factorization)
- Main assumption: Soft attributes can be represented in the same space (either as a point or as a direction)
  - Learning representations for soft attributes in three ways: unsupervised, weakly supervised, fully supervised

# Unsupervised

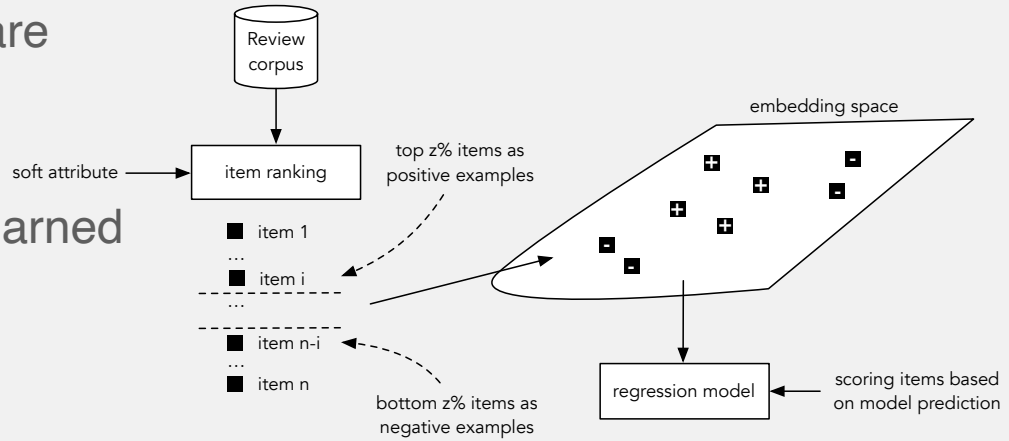
- No explicit training data, based on implicit signals from item reviews
- Term-based: Established models from entity retrieval [4]
  - Two evidence aggregation strategies: item-centric and review-centric
- Embedding-based: Soft attribute is represented by the centroid of the top ranked items
  - Other items can be scored by their distance to the centroid




[4] Balog. Entity-Oriented Search. *Springer, 2018.*

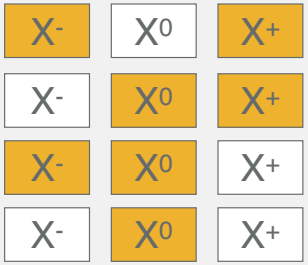
# Weakly supervised

- Still no explicit training data available
- Learn which factors in the embedding space encode a particular soft attribute
- Learn a pointwise regression model (here: Logistic Regression) in a weakly supervised way
  - Top-ranked and bottom-ranked items are taken as positive and negative (pseudo-)training examples
  - Items can be scored by applying the learned model and taking the prediction probabilities as scores



# Fully supervised

- Assumes the availability of pairwise item orderings for a given attribute (facilitated by the data collection methodology discussed earlier)
- Generate training examples and learn a model using full supervision
  - Infer all possible pairwise preferences given a reference item r and labeled sets
    1. $\{i \succcurlyeq j\},\ \forall i \in \mathcal{X}^+, j \in \mathcal{X}^-$
    2. $\{i \succ j\},\ \forall i \in \mathcal{X}^+, j \in \mathcal{X}^\circ \bigcup \{r\}$
    3. $\{i \succ j\},\ \forall i \in \mathcal{X}^\circ \bigcup \{r\}, j \in \mathcal{X}^-$
    4. $\{i \sim j\},\ \forall i, j \in \mathcal{X}^\circ \bigcup \{r\}$




  - Learn a pairwise ranking model (here: rankSVM)
- Score items by applying the learned model

# Evaluation results

| Method | MovieLens (G) | SoftAttr (G') |
|---|---|---|
| Term-based, item-centric (TB-IC) | 0.800 | 0.110 |
| Term-based, review-centric (TB-RC) | 0.733 | 0.136 |
| Centroid-based (w/ TB-IC) | 0.404 | 0.087 |
| Centroid-based (w/ TB-RC) | 0.471 | 0.101 |
| Weakly supervised (w/ TB-IC) | 0.539 | 0.194 |
| Weakly supervised (w/ TB-RC) | 0.517 | 0.200 |
| Fully supervised | | 0.485 |

# Evaluation results

| Method | MovieLens (G) | SoftAttr (G') |
|---|---|---|
| Term-based, item-centric (TB-IC) | 0.800 | 0.110 |
| Term-based, review-centric (TB-RC) | 0.733 | 0.136 |
| Centroid-based (w/ TB-IC) | 0.404 | 0.087 |
| Centroid-based (w/ TB-RC) | 0.471 | 0.101 |
| Weakly supervised (w/ TB-IC) | 0.539 | 0.194 |
| Weakly supervised (w/ TB-RC) | 0.517 | 0.200 |
| Fully supervised | | 0.485 |

**Key takeaway**

A more accurate abstraction of the attribute ranking problem proves to be considerably harder

# Analysis: Performance per soft attribute

**Soft attributes with highest, mid-range, and lowest performance for the fully supervised model.**

| Highest Performing | | | Selected Mid-Range | | | Lowest Performing | | |
|---|---|---|---|---|---|---|---|---|
| Attribute | Mean(G') | Std(G') | Attribute | Mean(G') | Std(G') | Attribute | Mean(G') | Std(G') |
| juvenile humor | 0.676 | (0.248) | fictionalized | 0.573 | (0.396) | incomprehensible | 0.268 | (0.372) |
| serious | 0.674 | (0.302) | mushy mushy | 0.524 | (0.341) | boring | 0.240 | (0.386) |
| believable | 0.671 | (0.248) | exaggerated | 0.495 | (0.351) | entertaining | 0.237 | (0.397) |
| factual | 0.662 | (0.280) | confusing | 0.470 | (0.356) | unique story | 0.211 | (0.368) |
| cartoonish | 0.658 | (0.254) | tearful | 0.433 | (0.375) | overrated | 0.188 | (0.431) |
| action filled | 0.652 | (0.283) | romantic | 0.340 | (0.346) | typical | 0.171 | (0.421) |

High inter-user agreement
(less subjectivity)

Medium inter-user agreement

Low inter-user agreement
(more subjectivity)

Key takeaway

For attributes with high agreement, non-personalized scoring models perform well. For "softer" attributes, there is significant headroom for improvement.

# Summary

- Problem of critiquing based on soft attributes
  - Enabled by the new data collection methodology
  - Unsupervised, weakly supervised, and fully supervised approaches for this task
- Future work concerns personalization
  - Significant headroom for personalized soft attribute scoring models
  - Predicting the "softness" of a soft attribute is also an interesting future direction

# Summary

- New possibilities for more natural interactions
  - Explanation of preferences and recommendations
  - Nuanced interpretation of natural language critiques