Predicting Missing Ratings

Fictitious Case Study: MovieReviews.com

Key Facts (5ws)

- MovieReview.com has a database of 600 users reviewing a minimum of 20 reviews per user.
    - **600 users**, each wrote > **20 reviews (**static) (9,177 **Movies**)

- Interested in improving their current recommender system that provides recommendations by popularity:
    - **Recommender system** = User Preferences & Associations
    - Dataset: Movie, User, Review
        - **100,00**/Movies x **600**/Users x **20**/MinReviews

- **We don’t have a cold start problem.**
- **User data is rich (600 Users/ min 20 Movies)**
- **The data is someone sparse and not dense.**

- Hypothesis:
  - Improve current model from 2% CTR to 15% CTR to the shopping cart

Dataset Overview

- **Movie IDs** - 9,717
- **User Ids** - 610
- **Rating**: 0.5 - 5.0
- **Total Ratings**: 100,856
- **Ratings per user**: Min 20

Predicting Rating

Vivian

Joe B.

Alphonso

Joe M

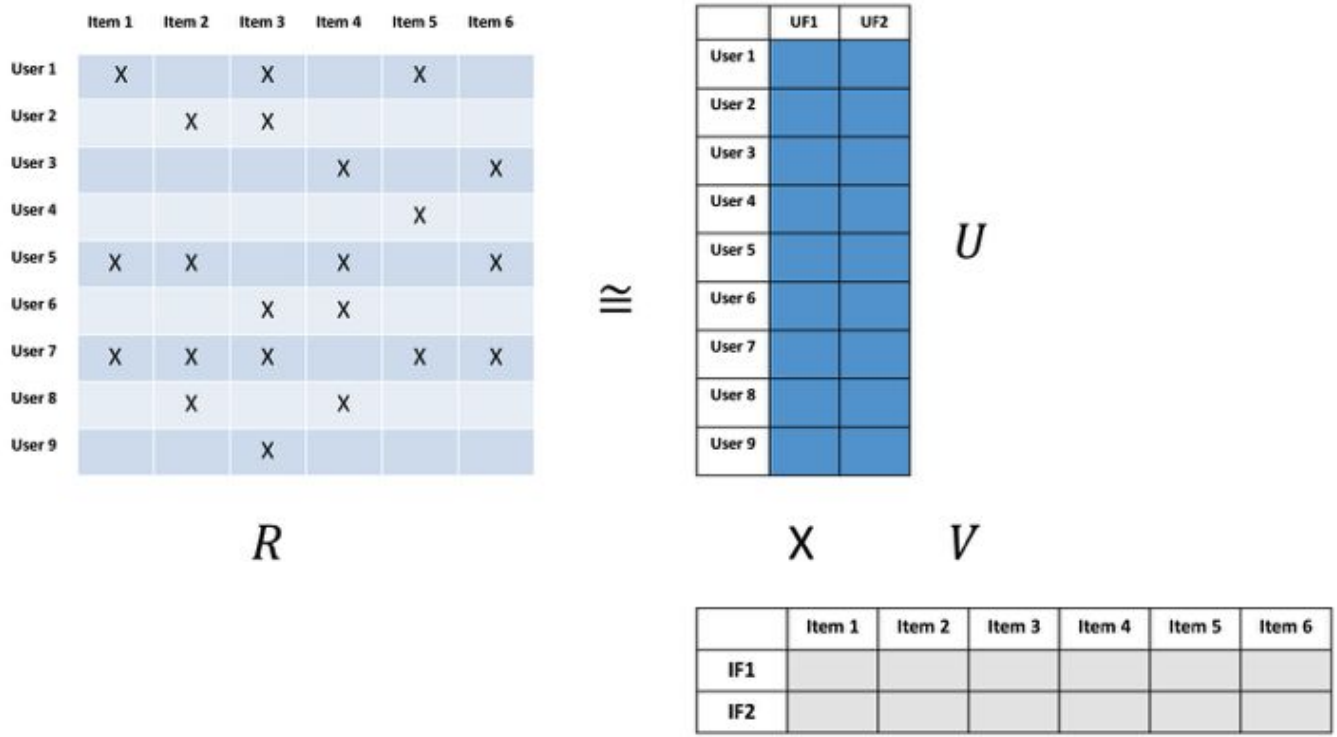
Ammar

Marisa

Model Results

- Our recommendation system accurately predicted a user's preference for an item
- **77%** accuracy in user-item preference

How?


Item 1 Item 2 Item 3 Item 4 Item 5 Item 6
User 1 X X X
User 2 X X
User 3 X X
User 4 X
User 5 X X X X
User 6 X X
User 7 X X X X X
User 8 X X
User 9 X
R
≅
UF1 UF2
User 1
User 2
User 3
User 4
User 5
User 6
User 7
User 8
User 9
U
X
V
Item 1 Item 2 Item 3 Item 4 Item 5 Item 6
IF1
IF2

Example:

Titanic = 20% Action + 0% Comedy + 70% Romance + …
Joe M = 15% Action + 0% Comedy + 80% Romance + …
Joe B = 10% Action + 80% Comedy + 5% Romance + ...

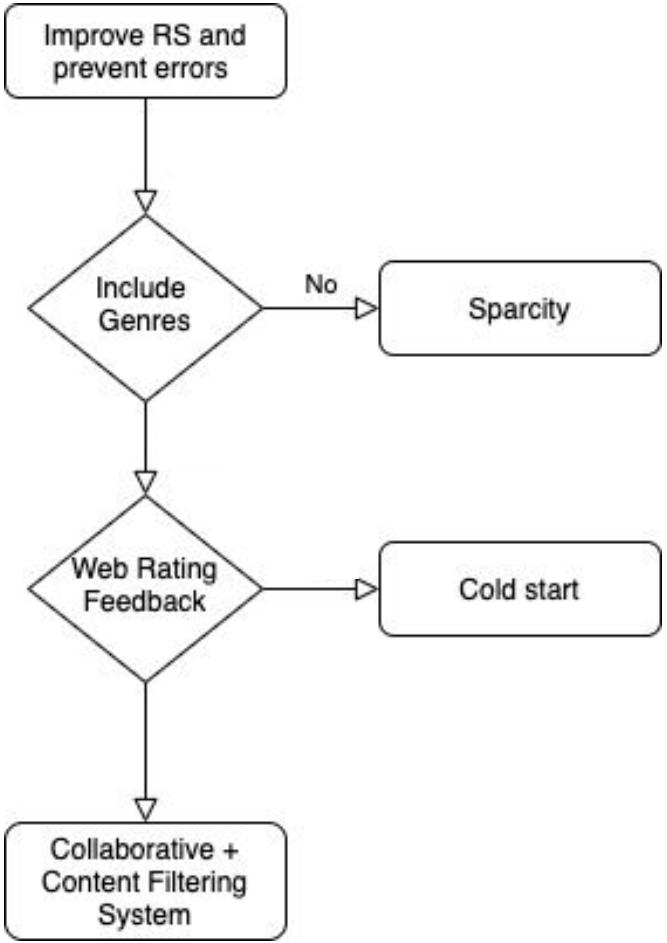
Problems with RSs?

1. **Implementation** will lose users
2. **Sparsity** in data
3. **Cold start**
4. **Low correlation** to others
5. **Algorithmic Development** not worth time/money

We recommending adopting the following functionality to your website:

1. Adding features to website for seamless feedback.
2. Upgrading your Recommender System.

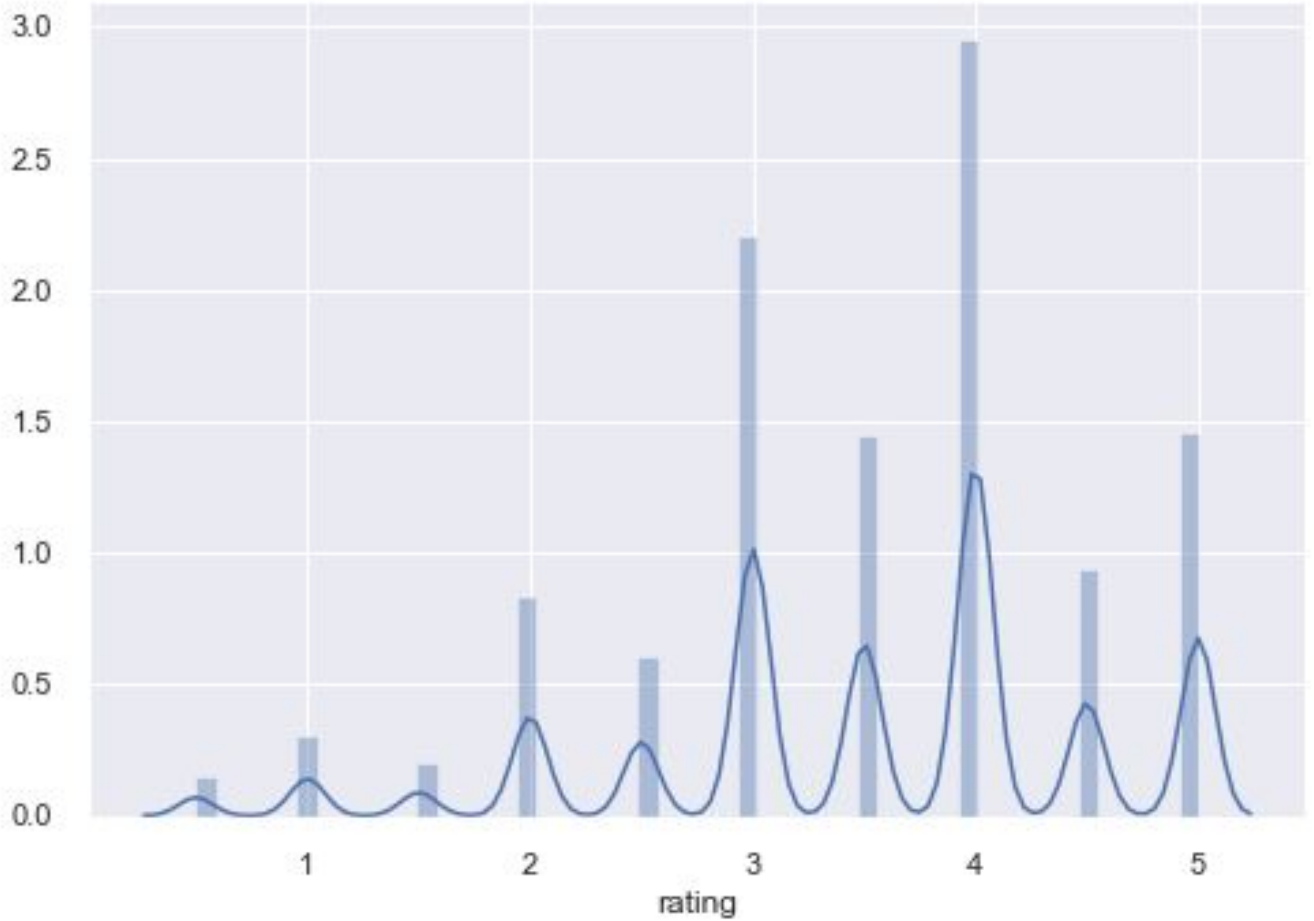
Improvements, Next Steps



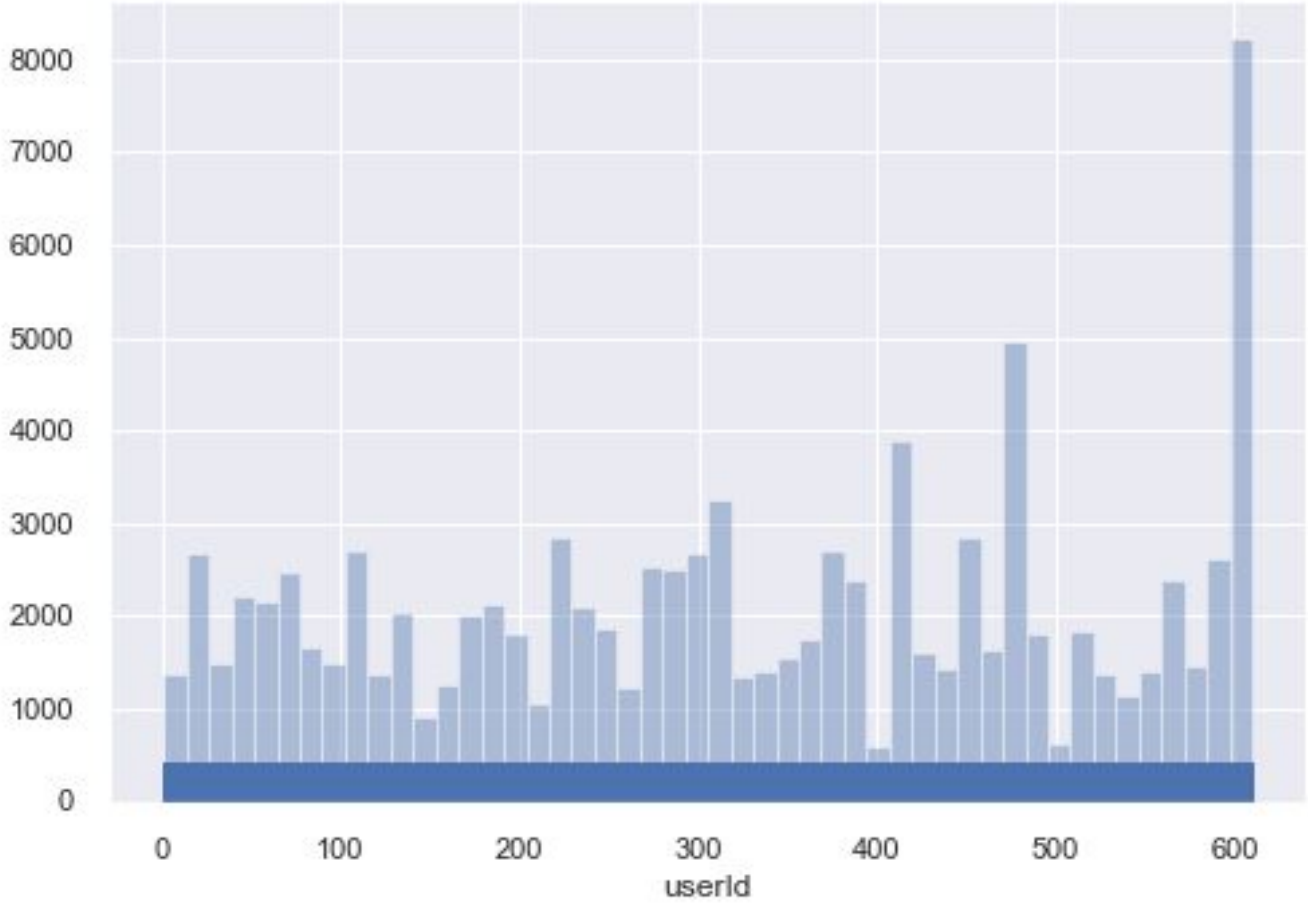
Thank You!

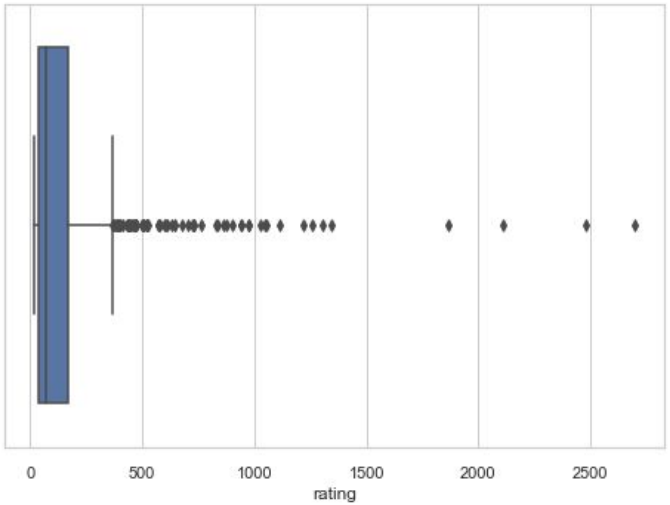
Questions?

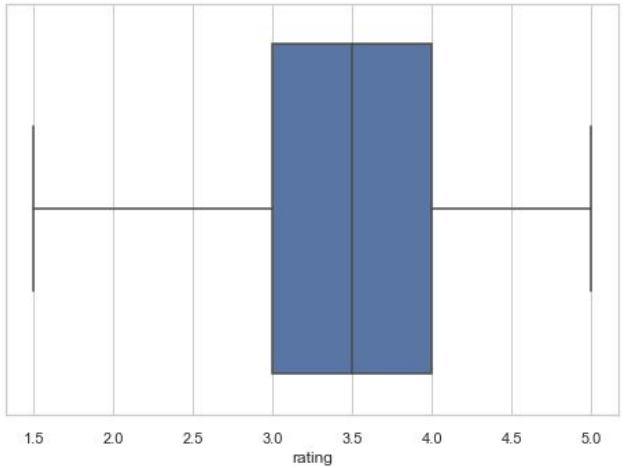
Rating distribution



User distribution




0
500
1000
1500
2000
2500
rating


1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
rating