Data Sources for Recommendations

Joshua Bernhard

Remember Recommendation Systems

- Collaborative Filtering Recommenders
- Demographic Recommenders
- Content Based Recommenders
- Utility Based Recommenders
- Knowledge Based Recommenders

The data we collect is directly related to the recommendation system we want to use.

Remember Recommendation Systems

- Collaborative Filtering Recommenders
  - Need user-item interactions
- Demographic Recommenders
  - Need user information
- Content Based Recommenders
  - Need content information
- Utility Based Recommenders
  - Any/all of the above data
- Knowledge Based Recommenders
  - Apply user filters to data

Alternatively, we likely have data, which enforces the use of certain recommendation systems.
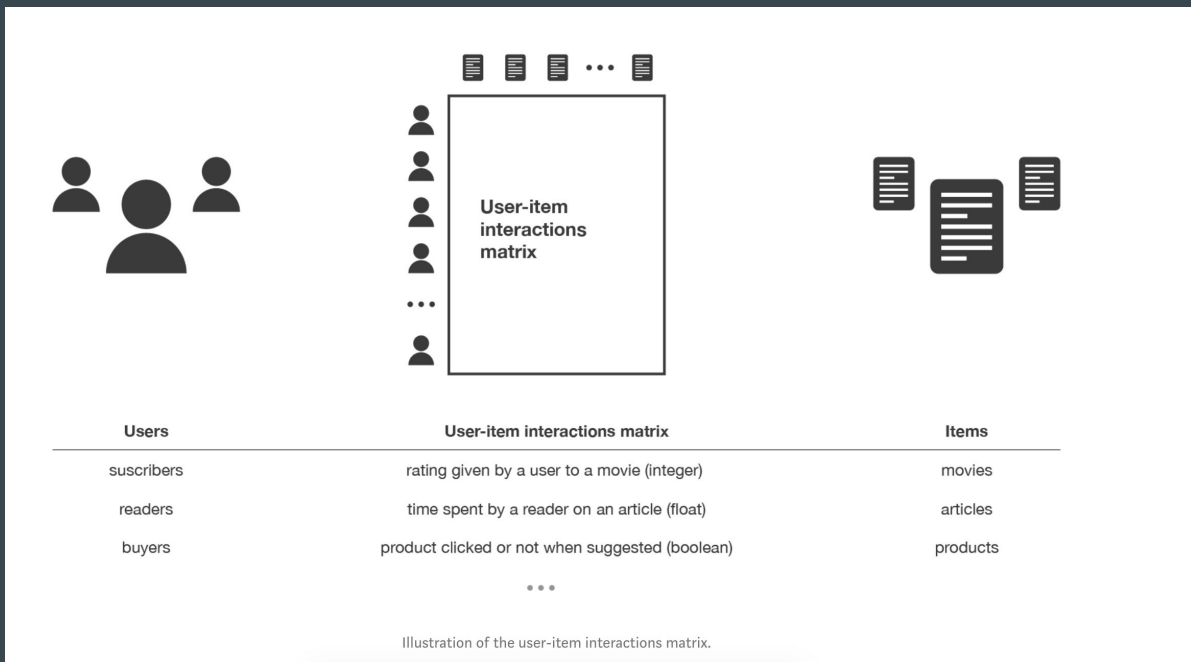
Data Sources

- User-item data (ratings)
- User data (surveys)
- Content data (text, product, script info)
- Knowledge data (user filters)

User-Item Data

User-Item Data

The most common way to build recommendation systems is via user-item interaction data.

Users	User-item interactions matrix	Items
suscribers	rating given by a user to a movie (integer)	movies
readers	time spent by a reader on an article (float)	articles
buyers	product clicked or not when suggested (boolean)	products

· · ·

Illustration of the user-item interactions matrix.

User-Item Data

This data might be:

- 0, 1 (whether or not a user has interacted with an item)
- A numeric value associated with how the user rates the item
- A numeric value associated with how much time a user spent with the item

Note: With numeric ratings, there tend to be many missing values in this matrix for all items a user has not interacted with

User-Item Data

Pros:

-   Often you can provide better recommendations using this method than constructing independent information about the users or items

Cons:

-   It requires a lot of user-item interactions to start building recommendations this way

User Data

User Data

For many recommendation systems, understanding your users is crucial to the survival of your business.

- At Nerdwallet, it is important that we don't recommend a Chase Preferred credit card to someone who already has this card.

- In dating apps, it is important we understand what our users are looking for in order to find a great match.

User Data

One of the most common ways to gain this information is to perform user research using surveys.

- The questions should be specific to how your product meets the customer needs.

User Data

Pros:

- This data can often help you provide highly personalized recommendations.

Cons:

- Depending on the situation, users may not feel comfortable sharing this information.

- It can be difficult to make some user feedback useful in recommendation systems.

Item (Content) Data

Item (Content) Data

There are some cases where introducing "content data" makes sense to improve your recommender. Unfortunately, this technique often introduces more noise than signal.

- Think of all the features you could make by featurizing every aspect of a movie.
    - The entire script
    - Actors and actresses (and characteristics of them)
    - Sound track
    - Scene set up information

Item (Content) Data

In the movie example, how much do you think each of these will assist in making better recommendations?

Item (Content) Data

Pros:

- Lots of creative solutions can be created based on content data.

Cons:

- Often the signal to noise ratio leans towards a lot of noise.
- It can take a lot of effort to featurize content in a meaningful way.
- Featurizing content data can be highly individualized to the product

Knowledge Data

Knowledge Data

- Knowledge data is gained by the user at the time in which the recommendation will be provided.

- If your product is a large purchase associated with specific user needs, this might be the only way to go!
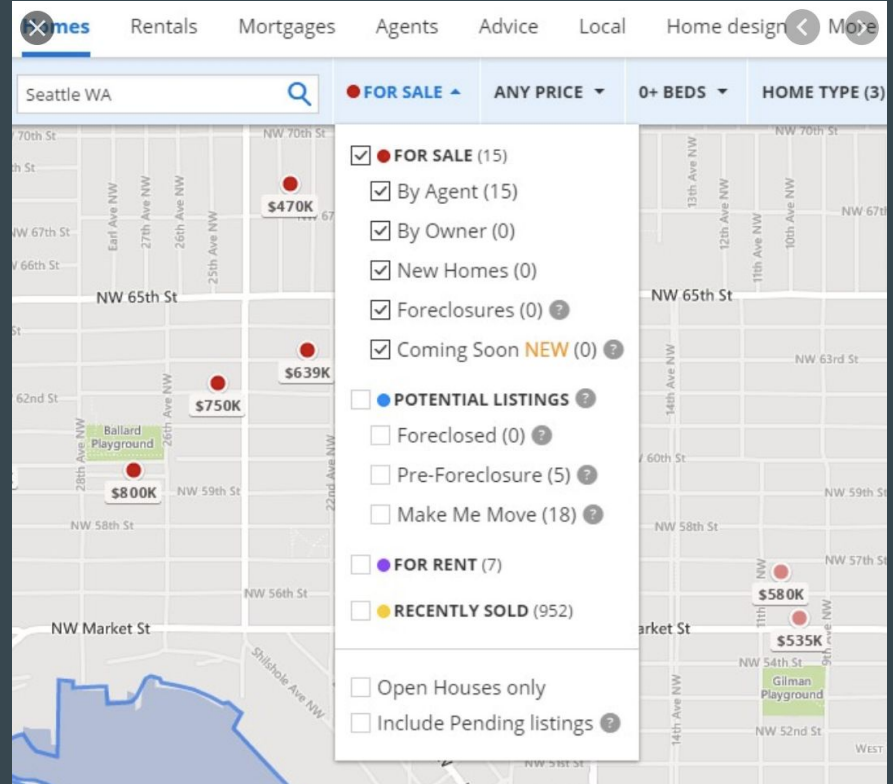
Recap

- The data used in recommendation systems is highly related to the type of recommender you want to build.

- The most common data collected for recommendation systems are user-item data.

- The use of user-item data means that you might be able to avoid the use of individual user or individual item data, because you can infer the relationship without these characteristics.