An Introduction to Recommender Systems

Charu C. Aggarwal

Carlos Gonzalez

University of Oxford - Department of Economics

Machine Learning Reading Group

June 3rd, 2025

Outline

Outline

Recommender Systems

- **Repeated interaction** between a *platform* and a set of *users* can be exploited to learn users' preferences

- This knowledge can be leveraged in many different ways
  - In general, better understanding of user's preferences yields **higher profits**, i.e. price discrimination, market expansion (new products and new customers), etc.
  - But it can also be user **welfare improving**, i.e. increase in diversity, reduced search costs, shopping experience, etc.
  - **Q:** Is user welfare just a proxy for long term profit maximization?

- Today we focus on the dynamic of *item* recommendation

Families of Recommender Systems and Data Types

- User-item preference relations can be codified in two ways
  - **Knowledge-Based Systems:** Connections between users' types and item characteristics are hard-coded through a set of *rules*. The user is compelled to reveal their private type.
  - **Data driven:** Connections between users' types and items characteristics are learned/inferred automatically under regularity conditions
    - Similar users like the same items as in **user-based collaborative filtering**
    - Same user likes similar items as in **item-based collaborative filtering**
    - Same user likes the same characteristics in an item as in **content-based systems**
    - The learning process is grounded on explicit preferences like **ratings** or implicit preferences like **customer behavior**

Families of Recommender Systems and Data Types

- **Q:** Can similarity based recommendations lead to *echo chambers*?
- **Q:** What are the limits of what can and should be tracked?
- **Q:** How can Economics be leveraged to learn preferences from implicit behavior?

Assumptions, Goals and Uncertainty in RS

- The approach to learning is virtually **assumption free**
  - Often times, no underlying utility functions (in the Debreu and Von Neumann–Morgenstern sense), nor even rationality
  - Although, recent success when combining structural and reduced form approaches
- Preferences can be captured in cardinal terms (*matrix completion*) or ordinal terms (*rank identification*). Ordinal preferences are usually enough.
- There a two sources of uncertainty in preference learning
  - Statistical / small sample
  - **Intrinsic:** Preferences may be time varying, even "random" (taste for diversity)

What to recommend

- Given some preference learning what should you recommend? **Q:** What is the **objective function** to be maximized?
  - Trade-off between **exploitation** (best user-item match, *relevance*) and **exploration** (learn more about some users or some items)
  - Guide users' **preference discovery** process (*novelty* and *serendipity*)
  - **Others:** Cater for *diversity* (inherent randomness in preferences), *explainability*, etc. **Q:** Are agents more likely to comply as an exploration unit if recommendations are explained?
  - One more twist: Preferences are shaped by platform recommendations. **Q:** Any thougts?
  - **Q:** Should private firms optimize for socially desirable goals?

Outline

Data Types

- **Explicit** vs **Implicit** data
- **Two-sided** (bandit) vs **One-sided** (apple tasting)
- Discrete, Categorical, Binary and Unary
  - In unary settings, 1 means "yes" but 0 does not mean "no" like in buying decisions. Typical in implicit data settings
  - Different tools for matrix completion in discrete vs unary settings like the treatment of 0s

Outline

Families of Recommender Systems

- Classification is somewhat arbitrary, and focused on the spirit of the system, not necessarily the underlying models and data
- Three big **families**: Collaborative filtering (user/item **similarity**), content-based (item **characteristics**), knowledge-based (hard-coded **rules** with user inputs)
- Intermediate categories, hybrid methods and especial cases

Collaborative-Filtering

- **Heuristics**
  - *User-based CF:* Similar people like the same items
  - *Item-based CF:* The same user likes similar items
- They can take a non-parametric nearest-**neighbors** approach (*memory-based*) or a **modeling** approach (*model-based*). The latter are more successful, especially in sparse matrix contexts, the former are more explainable

Collaborative-Filtering

- CF as **generalized regression**. Let $(Y_{ij})_{n\times m-1}$ be a matrix of ratings, and Y_{im} the vector of ratings of movie m
- Classic regression. Assume $Y_{im} = f(Y_{ij})$ for some $f \in \mathcal{F}$. Pick some $\hat{f} \in \mathcal{F}$ such that $\hat{f}(Y_{ij}) \approx Y_{im}$. Finally, predict $\hat{Y}_{n+1,m} = \hat{f}(Y_{n+1,j})$
- In CF,
  - **Any covariate might be missing** (in both training and testing, not just Y_{im})
  - No distinction between **training and testing** sample (as Y_{im} are also missing for some i)
- *Transductive learning* (semi-supervised learning). Learn only what you need to learn, not necessary to learn structural rules

Content-Based Systems

- Conceptualized as an **intermediate category** between CF and KB
  - Items are perceived as (known) functions of **latent characteristics** or descriptions
  - When user likes an item, items with similar characteristics are recommended to the user (item-based CF)
  - Characteristics are usually hard-coded (KB)
- Easier to make predictions for new items. More difficult to make predictions for new users
- They also suffer from lack of diversity (echo chambers)

Knowledge-Based Systems

- What if users just say what they want? Wouldn't that be easier? **Q:** Connections to direct mechanisms in mechanism design?
- We could then map users' inputs to items via hard-coded rules or similarity functions
- Or we could map inputs to items' characteristics and then apply CB methods

Knowledge-Based Systems

- **Example:** User inputs “cheap near me”, system returns houses below £500 ordered by location from closest to furthest
- Very useful in settings with few ratings and highly heterogeneous products like the housing market
- Similar pros and cons that content-based
- **Utility-based systems:** Platform assumes a utility function on item attributes

Intermediate and Hybrid Systems

- **Demographic Systems:** Use attributes and map them to items in a KB way (i.e. "old customers do not like risky products") or use demographics as covariates in similarity functions to predict ratings *à la* CF
  - **Q:** What are the pros and cons of DS compared to user-input methods or agnostic data driven methods?
  - **Q:** Are data driven methods simply picking on demographics?
- **Hybrid Systems**
  - Interpolate across recommendations depending on user or context, i.e. demographic knowledge-based system for cold start and user-based CF after enough data has been collected
  - Combine predictions from different models, i.e. `avg`(CB rating, CF rating)

Outline

Evaluation of Recommender Systems

- Evaluation resembles those of classic ML (like OOS-MSE, ROC curves, etc.)
- But methods will depend on recommender's problem (prediction vs rank), and recommender's objective function (relevance, novelty, explainability, etc.)

Context-Aware Systems

- Adjust base models to **context** like time, location or social information
- Two approaches: Treat it as additional covariates (like demographic characteristics) in CF/CB settings or set rules as in KS
- They may worsen the **sparsity problem**. Dimension reduction methods
- (Social) networks are particularly interesting as there exist some structural connection between users or between items. These structural connections can be exploited by tuning benchmark algorithms or developing ad-hoc methods

Others

- **Q:** Influencer mediated recommendations
- **Q:** Trustworthy neighbors in CF
- **Q:** Attack-resistant systems
- **Q:** Privacy
- **Q:** User-identification