Sefarim Recommender System

—

Benjamin Freund

Agenda

I. Problem Statement

II. Defining Terms and Scope

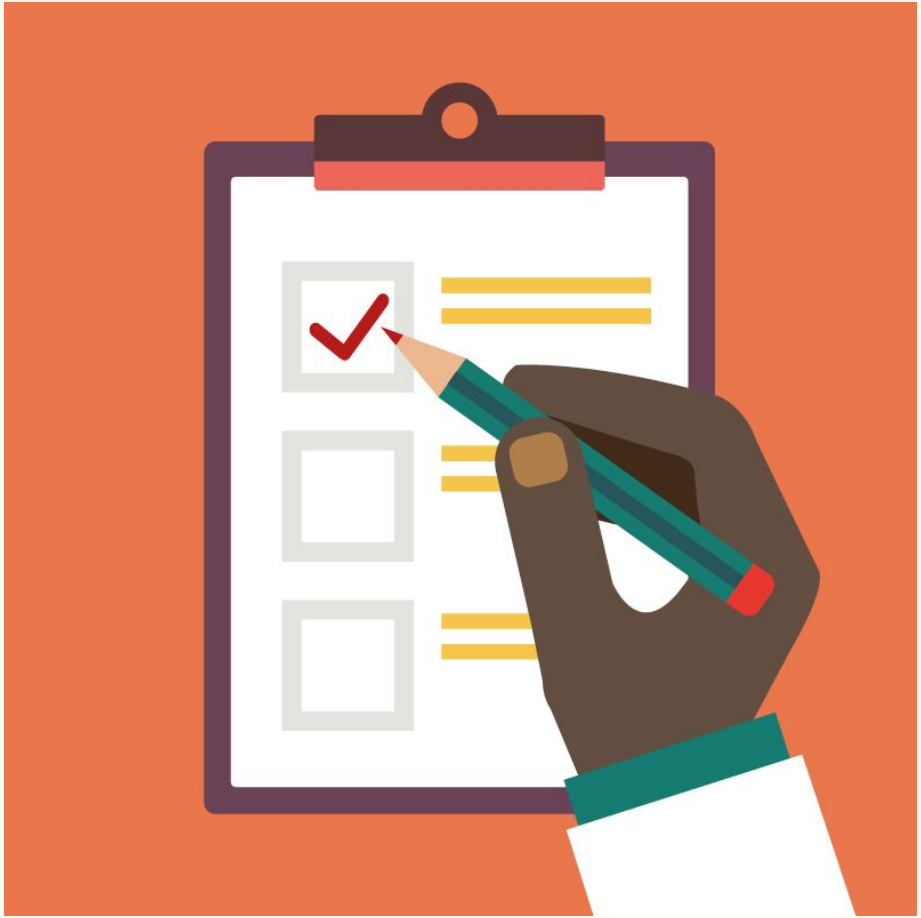
III. Data Collection

IV. Recommender Systems

V. Challenges and Limitations

VI. Lessons Learned

VII. Questions and Discussion



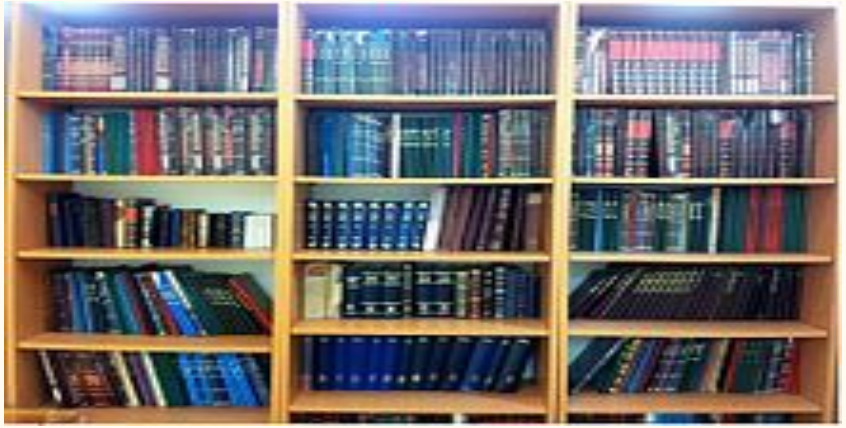
(Image Source)

I. Problem Statement

- Many book recommender systems exist
- Looking at Jewish books in particular
- Recommender systems like this exist, but are simple and could be misleading
    - Example: Artscroll has a "People Also Purchased" recommender. Could be misleading if one customer bought two unrelated books for a one-time purchase

II. Defining Terms and Scope

- Recsys: Short for “Recommender Systems”
- Sefarim: Hebrew for Jewish books (singular: sefer)
  - Judaism places a strong emphasis on the giving over of tradition through teaching and writing
- Initially chose to focus only on sefarim written by Achronim
  - Achronim: “the leading rabbis and poskim (Jewish legal decisors) living from roughly the 16th century to the present” (Wikipedia)



(Image Source)

III. Data Collection

- Initial Google Form: “Rate these sefarim” and “Favorite sefarim” questions
- Reached out to Jewish bookstores to no avail
- Sefarim Sale dataset
  - # of users: 335
  - # of genres: 26
    - Estimated # of books: 100,000
  - Limitations:
    - Genres, not books
    - Implicit ratings (counts), not explicit ratings

IV. Recommender Systems

- Collaborative Filtering, User Based (Towards Data Science example)
- Collaborative Filtering, Item Based (Towards Data Science example)
- Content Based (Data Camp example)

- Final Recommender:
  - Compares an item based recsys to a user based one
    - If user based is better, then uses the Collaborative Filtering, User Based recsys
    - If item based is better, then randomly selects between Collaborative Filtering, Item Based recsys and Content Based recsys

IV. Recommender Systems (cont.)

- Algorithm: KNN
  - “kNN is a machine learning algorithm to find clusters of similar users based on common book ratings, and make predictions using the average rating of top-k nearest neighbors” (Towards Data Science)
- Distance: Cosine similarity
  - “Cosine similarity is a metric used to measure how similar the documents are irrespective of their size. Mathematically, it measures the cosine of the angle between two vectors projected in a multi-dimensional space” (Machine Learning Plus)

IV. Recommender Systems (cont.)

RMSE penalizes outliers more, as it grows exponentially. MAE is better for data without outliers.

Formula

$$\mathrm{RMSD} = \sqrt{\frac{\sum_{i=1}^{N} (x_i - \hat{x}_i)^2}{N}}$$

RMSD = root-mean-square deviation
i = variable i
N = number of non-missing data points
x_i = actual observations time series
$\hat{x}_i$ = estimated time series

Formula

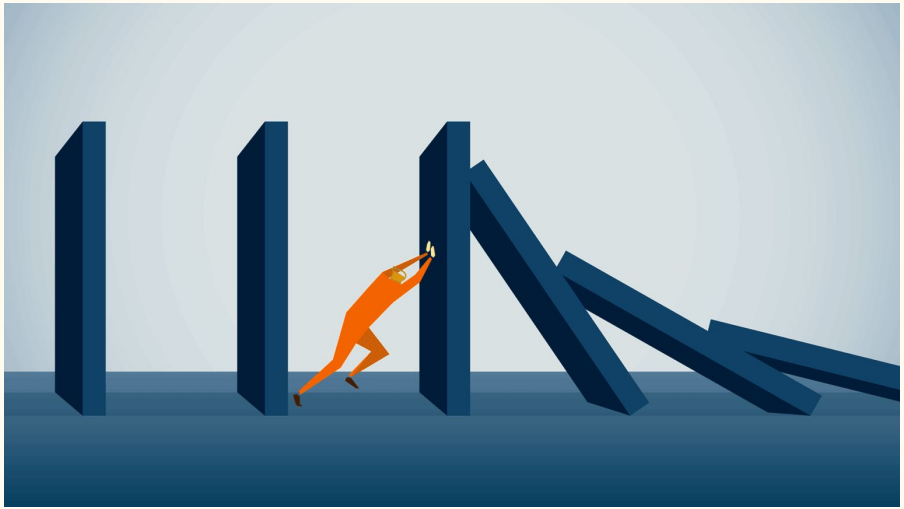
$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error
y_i = prediction
x_i = true value
n = total number of data points

Formula images sourced from Google

V. Challenges and Limitations

- Limitations of the dataset
- Final recommender function compares built in recsyses
- Unfamiliar topic



(Image Source)

VI. Lessons Learned

- Fundamentals first!
- Gather as much domain knowledge as possible
- Details are important, especially when coding
- No recsys is perfect, can always be improved upon

Thanks for listening!
Any questions?