SE ZG522: Big Data Systems

12 Machine Learning Recommender Systems

Dr. Kamlesh Tiwari
Assistant Professor, Department of CSIS,
BITS Pilani, Pilani Campus, Rajasthan-333031 INDIA

April 30, 2022 ONLINE (WILP @ BITS-Pilani Jan-May 2022)

http://ktiwari.in/bds

Introduction

Recommender Systems

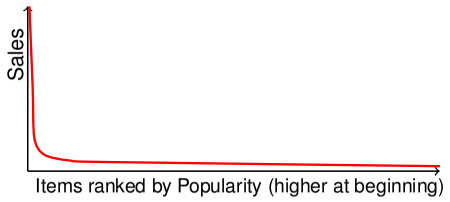
Software tools and techniques providing suggestions for items useful to a user

- What items to buy?
- What music to listen?
- What online news to read?

Recommendations are usually personalized, different users or groups receive diverse suggestions.

Plays an important role in highly rated Internet sites such as Amazon.com, YouTube, Netflix, Yahoo, Tripadvisor, and IMDb.

Sales vs Popularity



- Remember, too many choices are bad. (24,60,3) (6,40,30)

Importance [1]

Goal

- Increase number of items sold.
- Sell more diverse items.
- Increase the user satisfaction.
- Increase user fidelity.
- Better understand what the user wants.

Idea

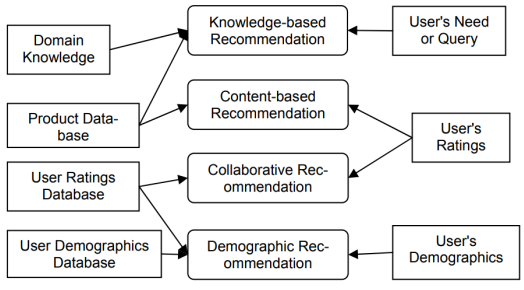
- Find Some/All Good Items
- Annotation in context
- Recommend a sequence/bundle
- Credible recommender
- Express self and help others

Recommendation Techniques

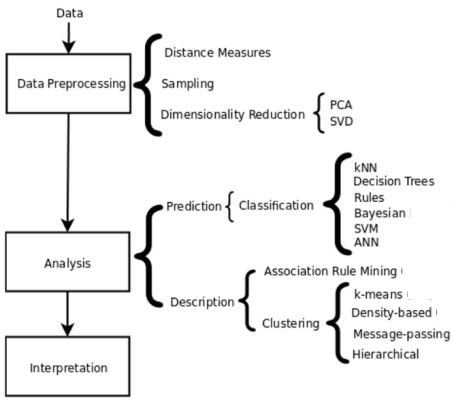
1. **Content-based:** recommend items similar to past liking. Similarity is based on features.
2. **Collaborative filtering:** recommend items that other users with similar tastes liked. Similarity is people-to-people correlation
3. **Hybrid:**

[1] Burke, R.: Hybrid web recommender systems. In: The Adaptive Web, pp. 377–408. Springer 2007

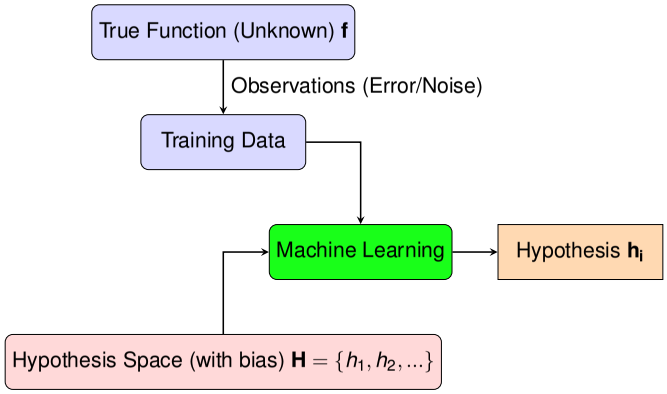
Techniques and their knowledge sources



Main steps and methods in a Data Mining problem



The Flow of ML



Similarity Measures

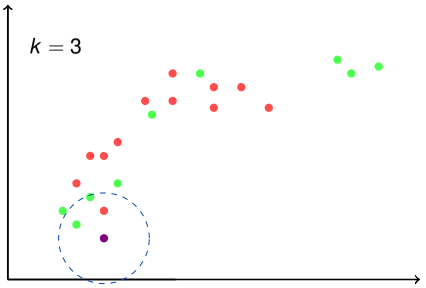
1. **Euclidean Distance:** $d(x,y) = \sqrt{\sum_{k=1}^{n}(x_k - y_k)^2}$
2. **Hamming Distance:**[2] $d(x,y) = \sum_{k=1}^{n} I_{x_k \neq y_k}$
3. **Minkowski (or L_p):**[3] $d(x,y) = \left(\sum_{k=1}^{n} |x_k - y_k|^r\right)^{1/r}$
4. **Mahalanobis:**[4] $d(x,y) = \sqrt{(x-y)\sigma^{-1}(x-y)^T}$
5. **Cosine:** $\cos(x,y) = \frac{x.y}{||x||.||y||}$
6. **Pearson Correlation:** $Pearson(x,y) = \frac{\sum(x,y)}{\sigma_x \times \sigma_y}$
   Also called centered cosine
7. **Jaccard:** $d(x,y) = \frac{x \cap y}{x \cup y}$

[2] I is an indicator random variable
[3] $r = 1$ city-block, Manhattan, taxicab, L1 norm. $r = \infty$ supremum, L_{max} or L_∞
[4] where σ is covariance matrix of the data

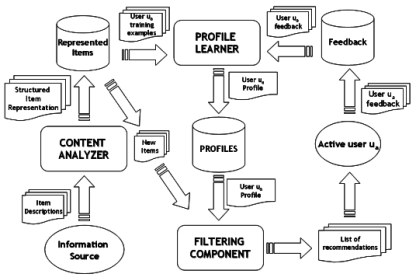
K Nearest Neighbor (KNN)

You are most likely as your friends. (it is bias)

- Search k other datum points (most difficult); do majority voting
- A lazy learner, non-parametric model



Architecture of a Content-based Recommender



Content-based systems need techniques for representing the items and producing the user profile. Strategies for comparing the user profile is also needed.

Example Movie rating

	M1	M2	M3	M4	M5	M6	M7
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- A&B seems similar but A&C not
- Is Jaccard[5] good? sim(A,B)=1/5, sim(A,C)=2/4 NO
- Is Cosine good? sim(A,B)=.38, Sim(A,C)=.32 YES marginally[6]
- Centered Cosine? sim(A,B)=0.09, sim(A,C)=-0.56 Good
  (missing values are average rating now)

[5] $|r_A \cap r_B| / |r_A \cup r_B|$
[6] Putting 0 at missing places is bad assumption

Rating Prediction (user-to-user collaborative filtering)

- Let rating of user x is r_x
- Let N is a set of k users near to x who have rated item i

How to obtain prediction of user x for item i

1. **Average:**
$$r_{x_i} = \frac{1}{k} \sum_{y \in N} r_{yi}$$
2. **Weighted Average:**
$$r_{x_i} = \frac{\sum_{y \in N} s_{xy} r_{yi}}{\sum_{y \in N} s_{xy}}$$

s_{xy} being similarity of user x and y

Item-to-Item Collaborative Filtering)

- Determine neighborhood of item i
- Items such that the user x have given a rating
- Obtain weighted score
- and normalize

$$r_{xi} = \frac{\sum_{j \in N(i;x)} s_{ij} . r_{xj}}{\sum_{j \in N(i;x)} s_{ij}}$$

In practice Item-to-Item outperforms user-to-user

Looks like items are much simpler to explain

Notable Issues with Recommender Systems

1. **Cold-start:**
   Where the system does not have historical data from new users. Difficult for the system to produce recommendations relevant to the users
2. **Sparsity:**
   Users rating data are rarely compared to the total number of items available, it is difficult to find other users who are rating similar to the same item
3. **Scalability:**
   Arises for large datasets and continues to increase over time

Conclusion

1. Recommender Systems are Software tools and techniques
2. Useful for increasing sales
3. Can be achieved using content similarity based approach
4. Or collaborative filtering based approach
5. Item-to-Item collaborative filtering outperforms user-to-user

Thank You!

Thank you very much for your attention!