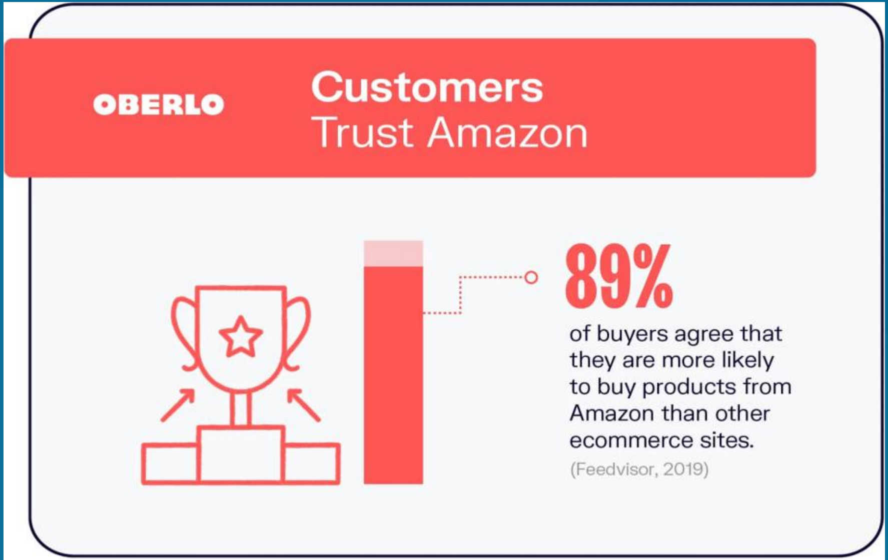
ECE271B Project:
Amazon Fine Foods Recommender to Customers

Group3: Haoming Zhang, Guoren Zhong

Problem Statement & Dataset Introduction

Motivations

- Amazon is the NO.1 most popular shopping App in the United States
- Over 89% customers trust Amazon
- The recommendation from Amazon to customer is important



Objective and Methodology

- Objective:
  - Predict the ratings of customers to products.
  - Decide which product to recommend.

- Methodology
  - Collaborative Filter
  - Latent factor model

Dataset

- Consists of reviews of fine foods from Amazon. (from Oct 1999 - Oct 2012)
  - 568,454 reviews
  - 256,059 users
  - 74,258 products
  - 260 users with > 50 reviews
  - 10 columns

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time	Summary	Text
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	1	5	1303862400	Good Quality Dog Food	I have bought several of the Vitality canned d...
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	0	1	1346976000	Not as Advertised	Product arrived labeled as Jumbo Salted Peanut...
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	1	4	1219017600	"Delight" says it all	This is a confection that has been around a fe...

J. McAuley and J. Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. WWW, 2013.

Dataset Manipulations

- Drop columns
  - Unrelated columns (Id, ProfileName, Time)
  - Text columns (Text, Summary)
  - Helpfulness columns (major missing values)
- Drop rows
  - Ignore users who have < 20 reviews

→ 68,015 rows, 3 columns

	ProductId	UserId	Score
0	B001GVISJM	A18ECVX2RJ7HUE	4
1	B001GVISJM	A2MUGFV2TDQ47K	5
2	B001GVISJM	A2A9X58G2GTBLP	5
3	B001EO5QW8	A2G7B7FKP2O2PU	5
4	B001EO5QW8	AQLL2R1PPR46X	5
5	B0059WXJKM	A25VFHVGI4CFTP	1
6	B001REEG6C	AY12DBB0U420B	5
7	B001GVISJW	A2YIO225BTKVPU	4
8	B001GVISJW	A1Z54EM24Y40LL	5
9	B001GVISJW	A281387UUS2IN5	3
10	B000ITVLE2	A3NID9D9WMIV01	5

J. McAuley and J. Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. WWW, 2013.

Rating Predictions

Problem Formulation

- Complete data is typically not available.
- Fill the blanks using the known data.

	Product 1	Product 2	Product 3	Product 4
User A	5.0		4.0	3.4
User B	3.0	3.6	3.5	
User C		4.6		4.8
User D	5.0		3.5	

Set up and Baseline

- Randomly divide the dataset into:
  - 80% training.
  - 20% testing.

- Baseline:
  - Prediction by average.
  - The MSE is around 1.37 on the test set.

Method 1: Collaborative Filter

Similarity-based Model

- Two steps:
  - Look for users similar to the active user.
  - Predict the rating using weighted ratings from the similar users.

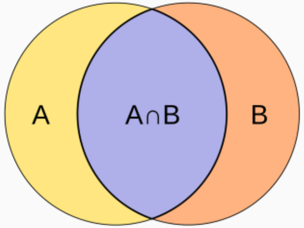
$$r(u,i) = \frac{1}{Z}\sum_{v\in U_i\setminus\{u\}} r_{v,i}\cdot \mathrm{sim}(u,v)$$

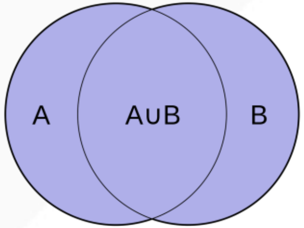
- $r_{u,j}$: rating of user u to product j
- $\mathrm{sim}(u,v)$: similarity between user u and v
- $Z = \sum_{v\in U_i\setminus\{u\}} \mathrm{sim}(u,v)$: normalization constant

Similarity Rules

- Jaccard similarity

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A - B|}$$





- Cosine similarity

$$C(x,y) = \cos(\vec{x},\vec{y}) = \frac{\vec{x}\cdot\vec{y}}{\|\vec{x}\|\|\vec{y}\|} = \frac{\sum_{i\in I_{xy}} r_{x,i} r_{y,i}}{\sqrt{\sum_{i\in I_{xy}} r_{x,i}^2}\sqrt{\sum_{i\in I_{xy}} r_{y,i}^2}}$$

Results

- Similarity-based Collaborative filtering: MSE = 0.742 on test set.
- Much better than the baseline.

Method 2: Latent Factor Model

Matrix Factorization

Unknown features

	Product 1	Product 2	Product 3	Product 4
Factor X	1.0	0.4	0.5	0.2
Factor Y	0.0	0.6	0.5	0.8

	Factor X	Factor Y
User A	5	3
User B	3	4
User C	4	5
User D	5	2

	Product 1	Product 2	Product 3	Product 4
User A	5.0	**3.8**	4.0	3.4
User B	3.0	3.6	3.5	**3.8**
User C	**4.0**	4.6	**4.5**	4.8
User D	5.0	**3.2**	3.5	**2.6**

Latent Factor Model & Optimization

- Latent Factor Model

$$f(u,i) = \alpha + \beta_u + \beta_i$$

α: global average

β_u: how much does the user tend to rate above mean

β_i: does this item tend to receive high ratings

- The optimization problem

$$\arg\min_{\alpha,\beta} \underbrace{\sum\nolimits_{u,i} (\alpha + \beta_u + \beta_i - R_{u,i})^2}_{\text{error}} + \underbrace{\lambda \left[\sum\nolimits_u \beta_u^2 + i \sum \beta_i^2 \right]}_{\text{regularization}}$$

Update Parameters

- The Update Rules:

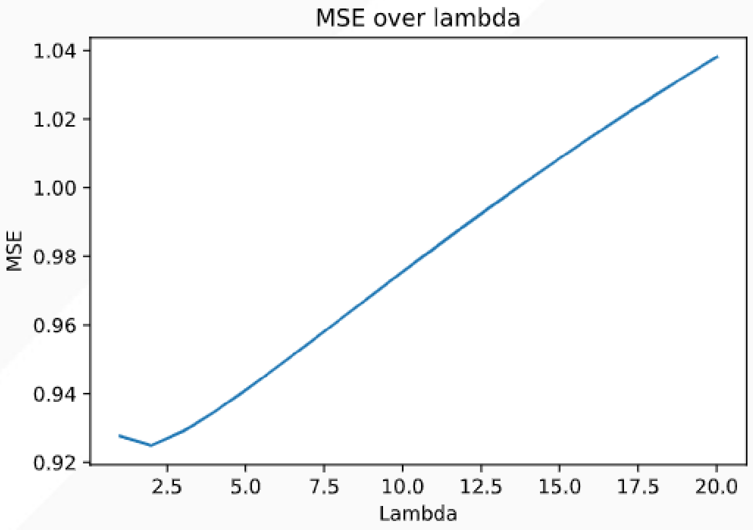
$$\alpha = \frac{\sum_{u,i} R_{u,i} - (\beta_u + \beta_i)}{N_{\text{train}}}$$

$$\beta_u = \frac{\sum_{i \in I_u} R_{u,i} - (\alpha + \beta_i)}{\lambda + |I_u|}$$

$$\beta_i = \frac{\sum_{u \in U_i} R_{u,i} - (\alpha + \beta_u)}{\lambda + |U_i|}$$

Results

- Latent Factor Model: MSE = 0.92 when λ is 2
- Slightly worse than the first method



Conclusion

Final results

- Similarity-based Collaborative filtering: MSE = 0.742 on test set.
- Latent Factor Model: MSE = 0.92 when λ is 2

Future work

- Apply different models.
- Tune or add parameters.
- Generate recommendations for each user.

Reference

https://www.oberlo.com/blog/amazon-statistics

https://www.kaggle.com/snap/amazon-fine-food-reviews

https://cseweb.ucsd.edu/classes/fa20/cse258-a/slides/recommendation_clean.pdf

J. McAuley and J. Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. WWW, 2013.

UC San Diego