Recommender Systems

Instructor: Ekpe Okorafor

1. Accenture – Big Data Academy
2. Computer Science African University of Science & Technology

Objectives

Objectives

- What is the difference between content based and collaborative filtering
- recommender systems
- Which limitations recommender systems frequently encounter
- How collaborative filtering can identify similar users and items
- How Tanimoto and Euclidean distance similarity metrics work

Outline

- What is a recommender system?
- Types of collaborative filtering
- Limitations of recommender systems
- Fundamental concepts
- Essential points
- Conclusion
- Hands-On Exercise: Implementing a Basic Recommender

Outline

- What is a recommender system?
- Types of collaborative filtering
- Limitations of recommender systems
- Fundamental concepts
- Essential points
- Conclusion
- Hands-On Exercise: Implementing a Basic Recommender

What is a Recommender System?

- **Recommenders are a type of filter**
- **They help users find relevant items within a huge selection**
  - How do you find an interesting movie among 95,000 choices?
  - They help you find things you didn't know to look for
- **Recommenders use preferences to predict preferences**
  - Input is feedback about likes and/or dislikes
  - Output is a list of suggested items based on feedback received
- **Two main types of recommenders**
  - Content-based
  - Collaborative filtering

Content-Based Recommenders

- **Content based recommenders consider an item's attributes**
  - These attributes describe the item
- **Examples of item attributes**
  - Movies: actor, director, screenwriter, producer, and location
  - Music: songwriter, style, musicians, vocalist, meter, and tempo
  - Books: author, publisher, subject, illustrations, and page count
- **A user's taste defines values and weights for each attribute**
  - These are supplied as input to the recommender

Content-Based Recommenders (Cont'd)

- **Content based recommenders are domain specific**
  - Because attributes don't transcend item types
- **Examples of content based recommendations**
  - You like 1977's science fiction films starring Mark Hamill, try *Star Wars*
  - You like rock from the 1980's, try *Beat It*

Collaborative Filtering

- **Collaborative filtering is an inherently social system**
  - It recommends items based on preferences of similar users
- **It’s similar to how you get recommendations from friends**
  - Query those people who share your interests
  - They’ll know movies you haven’t seen and would probably like
    - And you’ll be able to recommend some to them
- **This approach is not domain-specific**
  - System doesn’t “know” anything about the items it recommends
  - The same algorithm can used to recommend any type of product
- **We’ll discuss collaborative filtering in detail during this chapter**

Hybrid Recommenders

- **Content-based and collaborative filtering are two approaches**
- **Each has advantages and limitations**
  - We'll discuss these in a moment
- **It's also possible to combine these approaches**
  - For example, predict rating using content-based approach
  - Then predict rating using collaborative filtering
  - Finally, average these values to create a hybrid prediction
- **Research demonstrates that this can offer better results than using either system on its own**
  - Neflix and other companies use hybrid recommenders

Outline

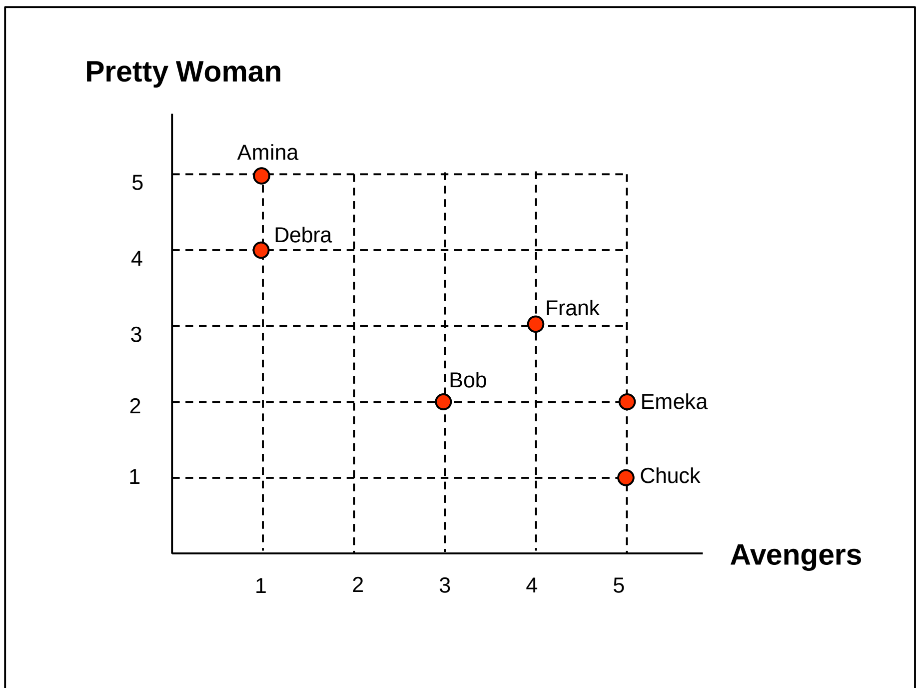
- What is a recommender system?
- Types of collaborative filtering
- Limitations of recommender systems
- Fundamental concepts
- Essential points
- Conclusion
- Hands-On Exercise: Implementing a Basic Recommender

Types of Collaborative Filtering

- **Collaborative filtering can be subdivided into two main types**
- **User-based: “What do users similar to you like?”**
  - For a given user, find other people who have similar tastes
  - Then, recommend items based on past behavior of those users
- **Item-based: “What is similar to other items you like?”**
  - Given items that a user likes, determine which items are similar
  - Make recommendations to the user based on those items

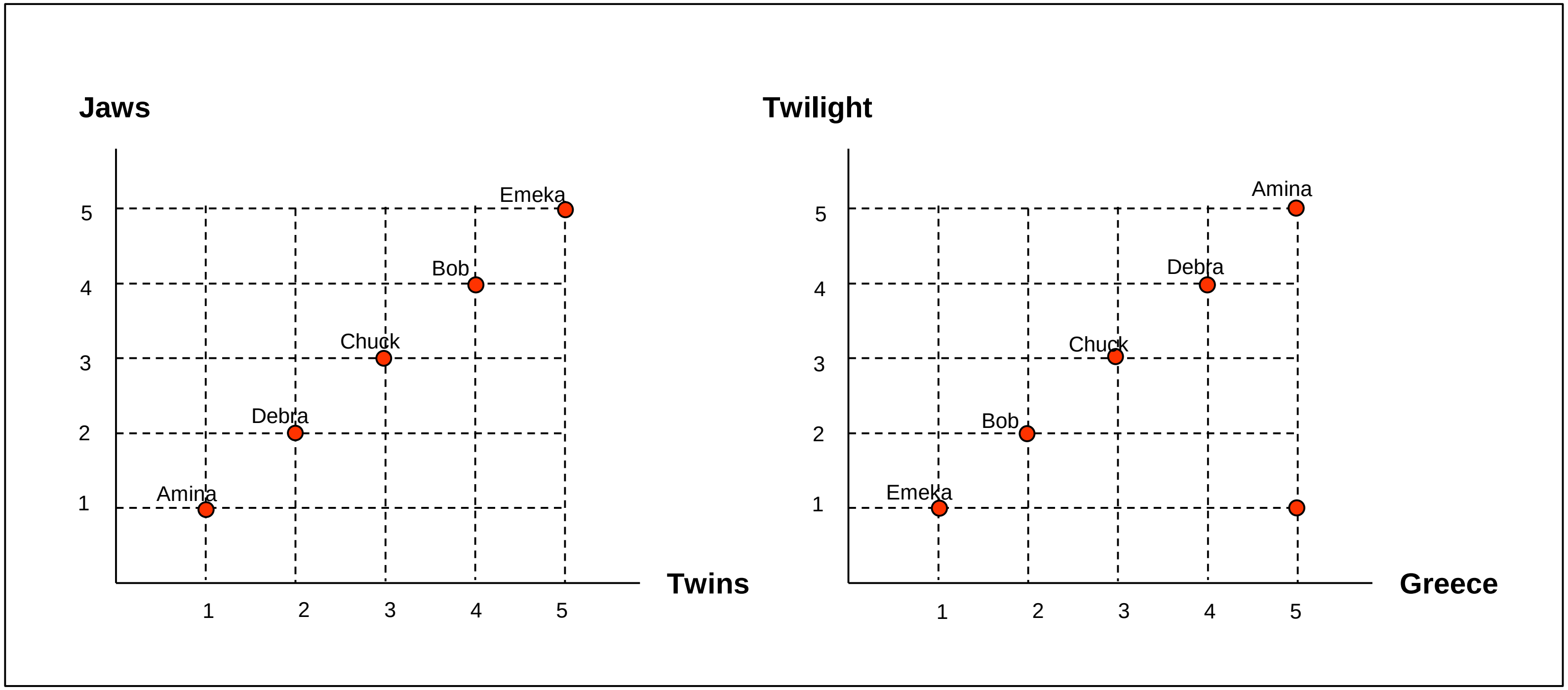
User-Based Collaborative Filtering

- **User-based collaborative filtering is social**
  - It takes a “people first” approach, based on common interests
- **In this example, Amina and Debra have similar tastes**
  - Each is likely to enjoy a movie that the other rated highly



Item-Based Collaborative Filtering

- **After examining more of these ratings, patterns emerge**
  – Strong correlations between movies suggest they are similar



Item-Based Collaborative Filtering (con’t)

- **The item-based approach was popularized by Amazon**
  - Given previous purchases, what would you be likely to buy?
- **Our example Movies could also use item-based filtering**
  - Suggest *Twins* after customer adds *Jaws* to the queue
- **Item-based CF usually scales better than user-based**
  - Successful companies have more users than products

Outline

- What is a recommender system?
- Types of collaborative filtering
- Limitations of recommender systems
- Fundamental concepts
- Essential points
- Conclusion
- Hands-On Exercise: Implementing a Basic Recommender

Limitations

- **The cold start problem is a limitation of collaborative filtering**
  - CF finds recommendations based on actions of similar users
  - So what do you do for a startup?
    - A new service has no users, similar or otherwise!
  - One workaround is to use content-based filtering at first
    - Eventually you'll have enough data for collaborative filtering
    - You can transition via a hybrid approach as you add users
- **Performance of sparse matrix operations**
  - Consider a dataset has 14 million customers and 100,000 movies
  - A matrix representation will have 1.4 trillion elements
    - Even active customers have only seen a few hundred movies
    - And they haven't rated all of these

Limitations (cont'd)

- **People aren't very good at rating things**
  - You may need to identify and correct for individual biases
  - Observe user behavior instead of asking for ratings
- **Individual tastes aren't always predictable**
  - One person may love *Halloween*, *Friday the 13th*, and *Saw*
  - Unlike similar users, this person may also love *Mary Poppins*
  - As always, using more input data will likely produce better results
- **A single account may correspond to multiple users**
  - Does the account holder like *Bambi*? Or is it her daughter?

Limitations (cont'd)

- **Item-based CF may predict previously satisfied needs**
  - The goal of item-based CF is to identify similar products
  - More helpful with pre-purchase suggestions than post-purchase
    - If I bought a toaster, ads for other toasters aren't helpful
    - But ads for bagels and jam might be helpful
  - Not an issue for some products (like movies or music)

Outline

- What is a recommender system?
- Types of collaborative filtering
- Limitations of recommender systems
- Fundamental concepts
- Essential points
- Conclusion
- Hands-On Exercise: Implementing a Basic Recommender

Input Data

- **The recommender accepts preference data as input**
  - These preferences represent what users like and dislike
  - Content-based recommenders also use attributes about an item
- **Input preferences can be collected in two ways**
  - Explicit: we ask users to rate items that they like or dislike
    - Neflix star ratings
    - TiVO “thumbs up” ratings
    - “How would you rank these items?”
  - Implicit: we observe user behavior to determine their preferences
    - Which movies does a customer watch?
    - Does customer move a movie up or down in the queue?
    - Does the customer finish the movie?

Evaluating Input

- **How does collaborative filtering work?**
  - Create a matrix of users and items, populated with preferences
  - For a given user, identify other users with similar tastes
  - Find items new to this user, but rated highly by similar users

	Amina	Bob	Chuck	Debra	Emeka	Frank	Gina
Airplane	1	4			5		
Bambi	4			5		2	
Caddyshack		4	3		4		5
Dracula			5			4	
Eat Pray Love		2		5	1		1
Friday		4					5
Gunsmoke						4	5
Hang ‘Em High			5			4	5
Iron Man			3	1	4		5
Jane Eyre	5						
The Karate Kid	4		5	5	3		

Evaluating Input (cont'd)

- **Debra has preferences similar to Amina**

	Amina	Bob	Chuck	Debra	Emeka	Frank	Gina
Airplane	1	4			5		
Bambi	4			5		2	
Caddyshack		4	3		4		5
Dracula			5			4	
Eat Pray Love		2		5	1		1
Friday		4					5
Gunsmoke						4	5
Hang 'Em High			5			4	5
Iron Man			3	1	4		5
Jane Eyre	5						
The Karate Kid	4		5	5	3		

Evaluating Input (cont'd)

- **Based on this, we could recommend Eat Pray Love to Amina**

	Amina	Bob	Chuck	Debra	Emeka	Frank	Gina
Airplane	1	4			5		
Bambi	4			5		2	
Caddyshack		4	3		4		5
Dracula			5			4	
Eat Pray Love		2		5	1		1
Friday		4					5
Gunsmoke						4	5
Hang 'Em High			5			4	5
Iron Man			3	1	4		5
Jane Eyre	5						
The Karate Kid	4		5	5	3		

Evaluating Input (cont'd)

- **Similarly, we could recommend *Jane Eyre* to Debra**

	Amina	Bob	Chuck	Debra	Emeka	Frank	Gina
Airplane	1	4			5		
Bambi	4			5		2	
Caddyshack		4	3		4		5
Dracula			5			4	
Eat Pray Love		2		5	1		1
Friday		4					5
Gunsmoke						4	5
Hang 'Em High			5			4	5
Iron Man			3	1	4		5
Jane Eyre	5						
The Karate Kid	4		5	5	3		

Evaluating Input (cont'd)

- **More users mean stronger signals and better recommendations**
  - Whose preferences are similar to Bob?

	Amina	Bob	Chuck	Debra	Emeka	Frank	Gina
Airplane	1	4			5		
Bambi	4			5		2	
Caddyshack		4	3		4		5
Dracula			5			4	
Eat Pray Love		2		5	1		1
Friday		4					5
Gunsmoke						4	5
Hang 'Em High			5			4	5
Iron Man			3	1	4		5
Jane Eyre	5						
The Karate Kid	4		5	5	3		

Evaluating Input (cont'd)

- **Both Emeka and Gina's preferences are similar to Bob**
  - Ratings they share produce better recommendations for Bob

	Amina	Bob	Chuck	Debra	Emeka	Frank	Gina
Airplane	1	4			5		
Bambi	4			5		2	
Caddyshack		4	3		4		5
Dracula			5			4	
Eat Pray Love		2		5	1		1
Friday		4					5
Gunsmoke						4	5
Hang 'Em High			5			4	5
Iron Man			3	1	4		5
Jane Eyre	5						
The Karate Kid	4		5	5	3		

Evaluating Input (cont'd)

- **We could recommend Gunsmoke, Karate Kid, or Iron Man to Bob**
  - Highest confidence about Iron Man, based on stronger signal

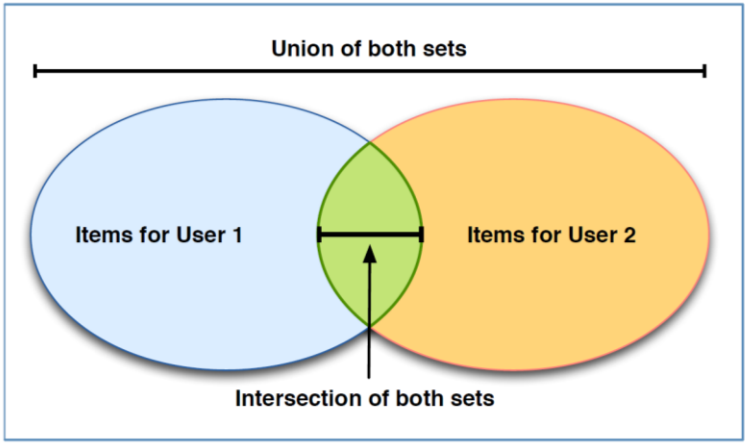
	Amina	Bob	Chuck	Debra	Emeka	Frank	Gina
Airplane	1	4			5		
Bambi	4			5		2	
Caddyshack		4	3		4		5
Dracula			5			4	
Eat Pray Love		2		5	1		1
Friday		4					5
Gunsmoke						4	5
Hang 'Em High			5			4	5
Iron Man			3	1	4		5
Jane Eyre	5						
The Karate Kid	4		5	5	3		

Basic Similarity Metrics

- **It’s easy for humans to see similarities between users**
  - But how can a computer find these similarities?
  - More importantly, how we can measure them?
- **There are many similarity metrics**
  - We’ll briefly cover two now, and discuss several in depth later
- **Choosing one involves several factors, including**
  - The type of preference data available
  - Performance at scale
- **They work by comparing vectors of data**
  - The elements could be users or items
  - You need to calculate metrics for every pair

Tanimoto Coefficient

- **Tanimoto coefficient is applicable when you have binary (boolean) data**
  - Did customer watch a given movie or not?
  - Did customer finish this movie or not?
- **Also known as the Jaccard coefficient, Tanimoto compares two sets**
  - Based on the ratio of union (all items) and intersection (common items)



Tanimoto Coefficient (cont'd)

- **The Tanimoto coefficient is easy to compute in R**

```
Tanimoto <- function(set_a, set_b){
    intersection <- set_a &(set_b)

    len_a <- len(set_a)
    len_b <- len(set_b)
    len_i <- len(intersection)

    return float(len_i) / (len_a + len_b - len_i)
    }
```

- **The value ranges between 0.0 and 1.0**
  - A value of 1.0 indicates both sets exactly match one another
  - Value moves towards 0.0 as number of common items decreases

Tanimoto Coefficient (cont'd)

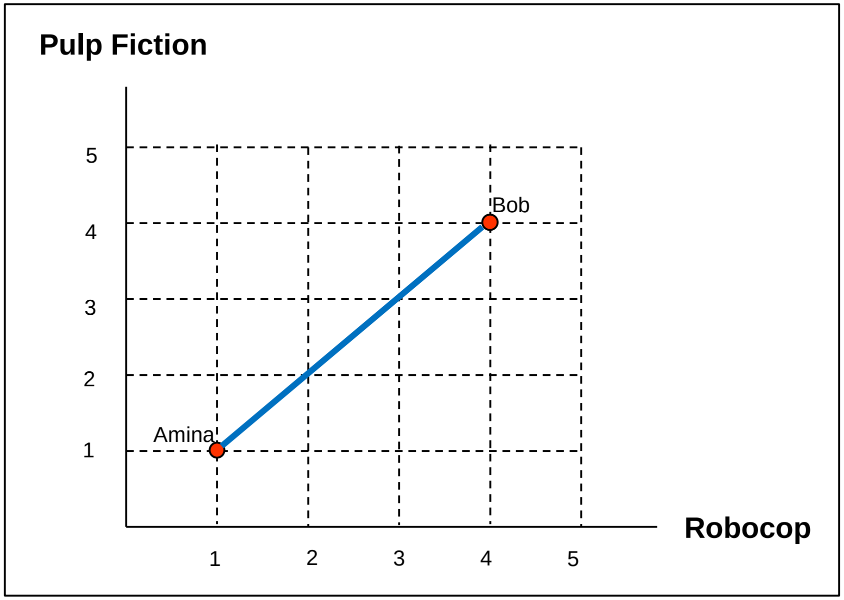
- **Consider the following input**
  - An 'X' in the matrix below indicates customer watched the movie

	Amina	Frank	Gina
Airplane		X	X
Bambi	X	X	
Caddyshack		X	X
Eat Pray Love	X		
Gunsmoke		X	X
Hang 'Em High		X	X

- **Frank and Gina share similar taste (value = 0.8)**
- **But Alice and Gina don't (value = 0.0)**

Euclidean Distance

- **Euclidean distance is a measure of similarity for numeric data**
  - “How many stars did the customer give this movie?”
  - “How many times did the customer watch this movie?”
- **Effectively the same as plotting it and measuring with a ruler**



Euclidean Distance (con't)

- Euclidean distance is also easy to calculate in R
  - Simple calculation based on parallel elements from each list

```
euclidean <- function(set_a, set_b) {
    sqrt(sum((set_a - set_b) ^ 2))

    library(foreach)
    foreach(i = 1:nrow(set_a), .combine = c)
        %do% euclidean(set_a[i,],set_b[i,])
    }
```

- A lower number indicates a stronger similarity
  - Though this is often inverted to provide a value in the 0.0 – 1.0 range

Euclidean Distance (cont'd)

- **Consider the following input**
  - Each element in the matrix below is the user's rating of a movie

	Amina	Frank	Gina
Airplane	1	4	5
Bambi	4	2	1
Caddyshack	2	4	5
Eat Pray Love	5	1	1
Gunsmoke	1	5	5
Hang 'Em High	1	4	5

- **Frank and Gina's preferences are close (distance of 2.0)**
  - Alice and Gina's preferences aren't (distance of 9.05)

Recommender Output

- **Quick recap of how a user-based recommender works**
  - Takes preference data as input
  - It finds similar users based on similarity metrics
- **What does a recommender produce as output?**
  - A list of items along with the predicted ratings for each
- **What do we do with this output?**
  - Remove items known to be of little value
  - Sort remaining items in descending order of predicted rating
  - Present this to the user in the application

Outline

- What is a recommender system?
- Types of collaborative filtering
- Limitations of recommender systems
- Fundamental concepts
- Essential points
- Conclusion
- Hands-On Exercise: Implementing a Basic Recommender

Essential Points

- **Recommenders are filtering systems**
- **Content-based recommenders consider item attributes**
- **Collaborative filters consider actions of other users**
- **Preferences can be collected implicitly or explicitly**
- **Similarity metrics are chosen, in part, based on data type**

Outline

- What is a recommender system?
- Types of collaborative filtering
- Limitations of recommender systems
- Fundamental concepts
- Essential points
- Conclusion
- Hands-On Exercise: Implementing a Basic Recommender

Conclusion

In this session you have learned

- **What is the difference between content-based and collaborative filtering recommender systems**
- **Which limitations recommender systems frequently encounter**
- **How collaborative filtering can identify similar users and items**
- **How Tanimoto and Euclidean distance similarity metrics work**

Outline

- What is a recommender system?
- Types of collaborative filtering
- Limitations of recommender systems
- Fundamental concepts
- Essential points
- Conclusion
- Hands-On Exercise: Implementing a Basic Recommender