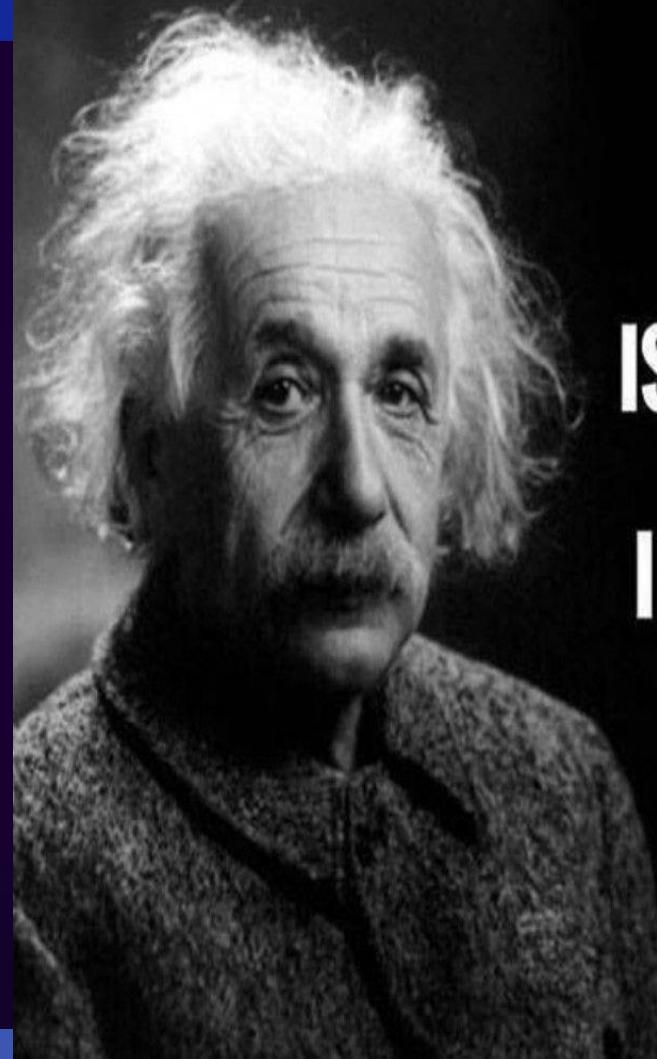


**YOUR POSITIVE
ACTION COMBINED
WITH POSITIVE
THINKING RESULTS IN
SUCCESS.**

DRIV ENERGY
your energy



AmbitionCircle

**"I CAN"
IS 100 TIMES
MORE
IMPORTANT
THAN
"I.Q"**

INSTAGRAM | AMBITIONCIRCLE

→ MODULE NO:-5

Real Time Big Data Models

SEMESTER-VII-COMPUTER

Fr.C.Rodrigues Institute of Technology,Vashi

→ Topics to be Discussed

5.1.1 A model for Recommendation systems

5.1.2 Content Based Recommendations

5.1.3 Collaborative Filtering

5.2.1 Case Study :Product Recommendation

5.3.1 Social Networks as Graphs,

5.3.2 Clustering of Social-Network Graphs

5.3.3 Direct Discovery of Communities in a social graph.

Recommendations



Products, movies,
music, news items, ...

amazon.com.



StumbleUpon

del.icio.us



m o v i e l e n s
helping you find the right movies

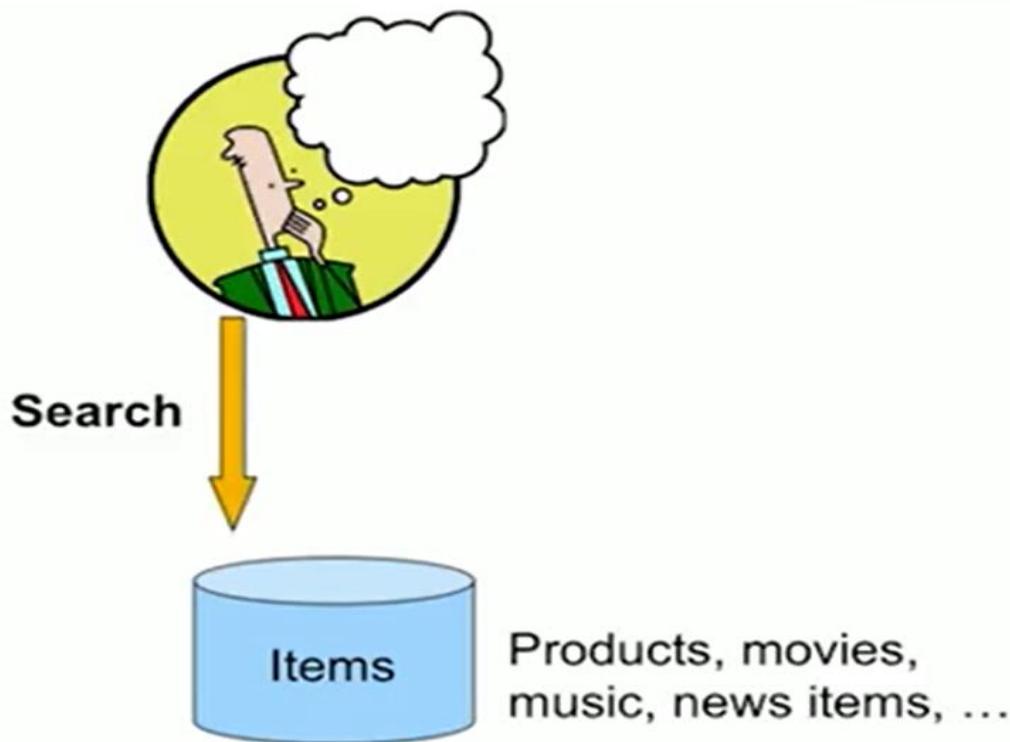
last.fm™
the social music revolution

Google™
News

You Tube

XBOX
LIVE

Recommendations



amazon.com.



StumbleUpon

del.icio.us



movielens
helping you find the *right* movies

last.fm™
the social music revolution

Google™
News

You Tube

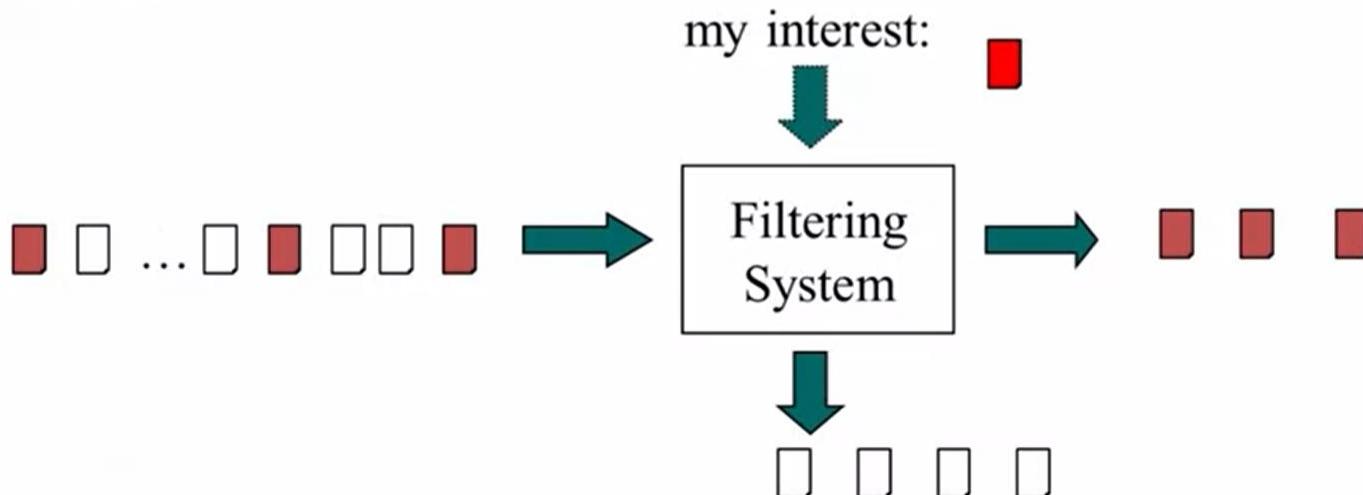
XBOX
LIVE

Recommendations



Recommender ≈ Filtering System

- Stable & long term interest, dynamic info source
- System must make a delivery decision immediately as a document “arrives”



A model for Recommendation systems

What Is Recommendation System?

- A recommendation system is a **subclass of Information filtering Systems** that seeks to **predict the rating or the preference a user might give to an item.**
- In simple words, it is an algorithm that **suggests relevant items to users.**
- There is an **extensive class of Web applications** that involve captures the patterns of peoples behavior in **predicting user responses** to options. Such a facility is called a recommendation system.
- Eg: In the case of Netflix **which movie to watch,**
- In the case of e-commerce **which product to buy**, or
- In the case of kindle **which book to read**, etc.

Recommendation systems

One of the most surprising part about Recommender Systems is that they suggest us things / advice every other day, without even realizing that'. There are many examples. Facebook, YouTube, LinkedIn are among the most popular ones. Let us see how they use **recommender systems**. You'll be amazed!

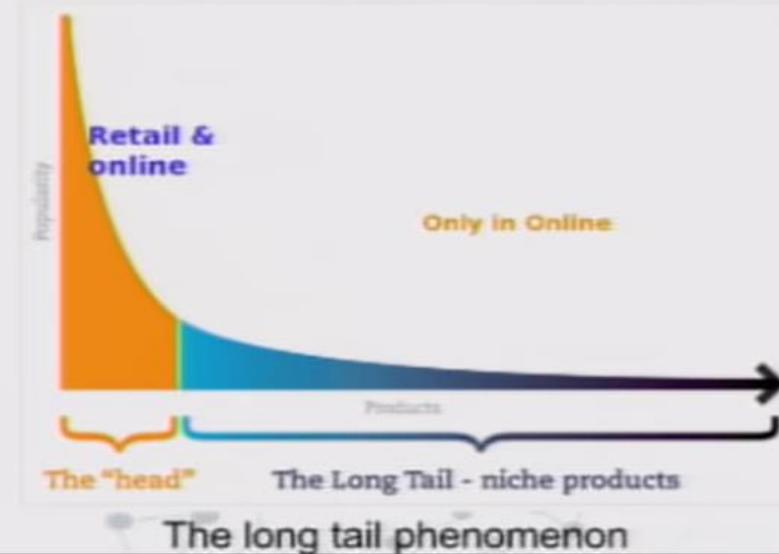
Facebook: Suggests us to make more friends using 'People You May Know' section



Similarly LinkedIn suggests you to connect with people you may know and YouTube suggests you relevant videos based on your previous browsing history. All of these are recommender systems in action.

While most of the people are aware of these features, only a few know that the algorithms used behind these features are known as 'Recommender Systems'.

.. Why Recommender Systems?



From Scarcity to Abundance

- Shelf space is a scarce commodity for traditional retailers
 - Also: TV networks, movie theaters,...
- The web enables near-zero-cost dissemination of information about products
 - From scarcity to abundance
 - Gives rise to the “Long Tail” phenomenon

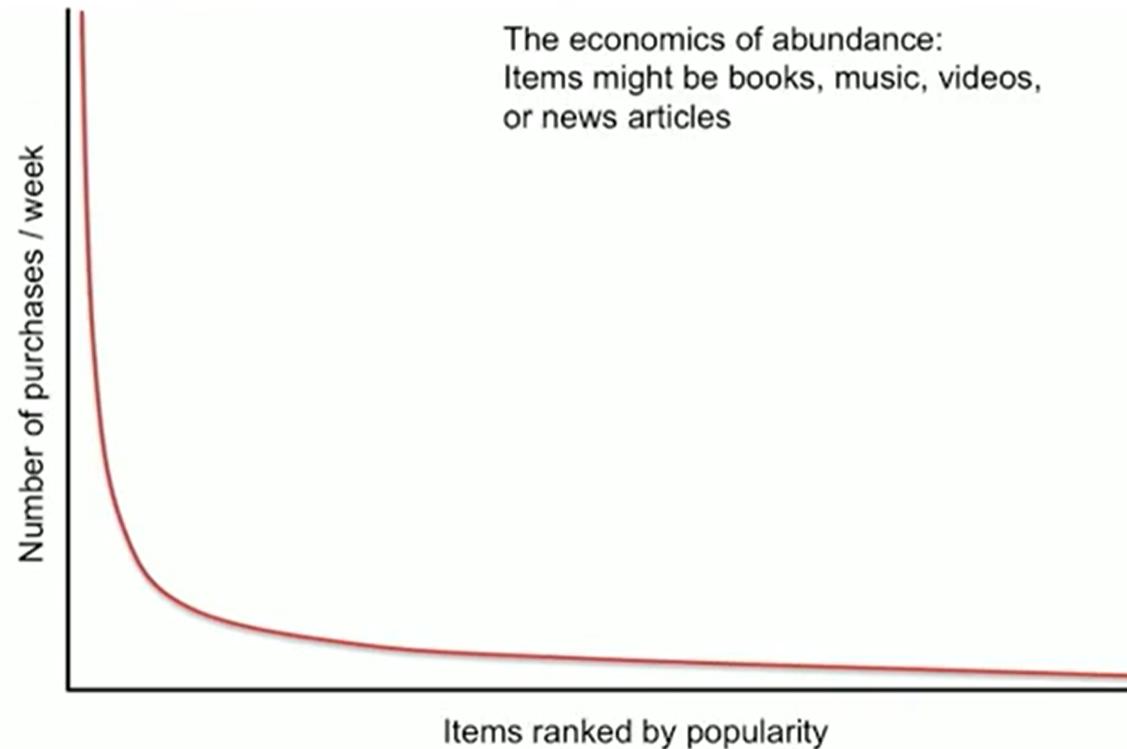
The Long Tail (1)

Number of purchases / week

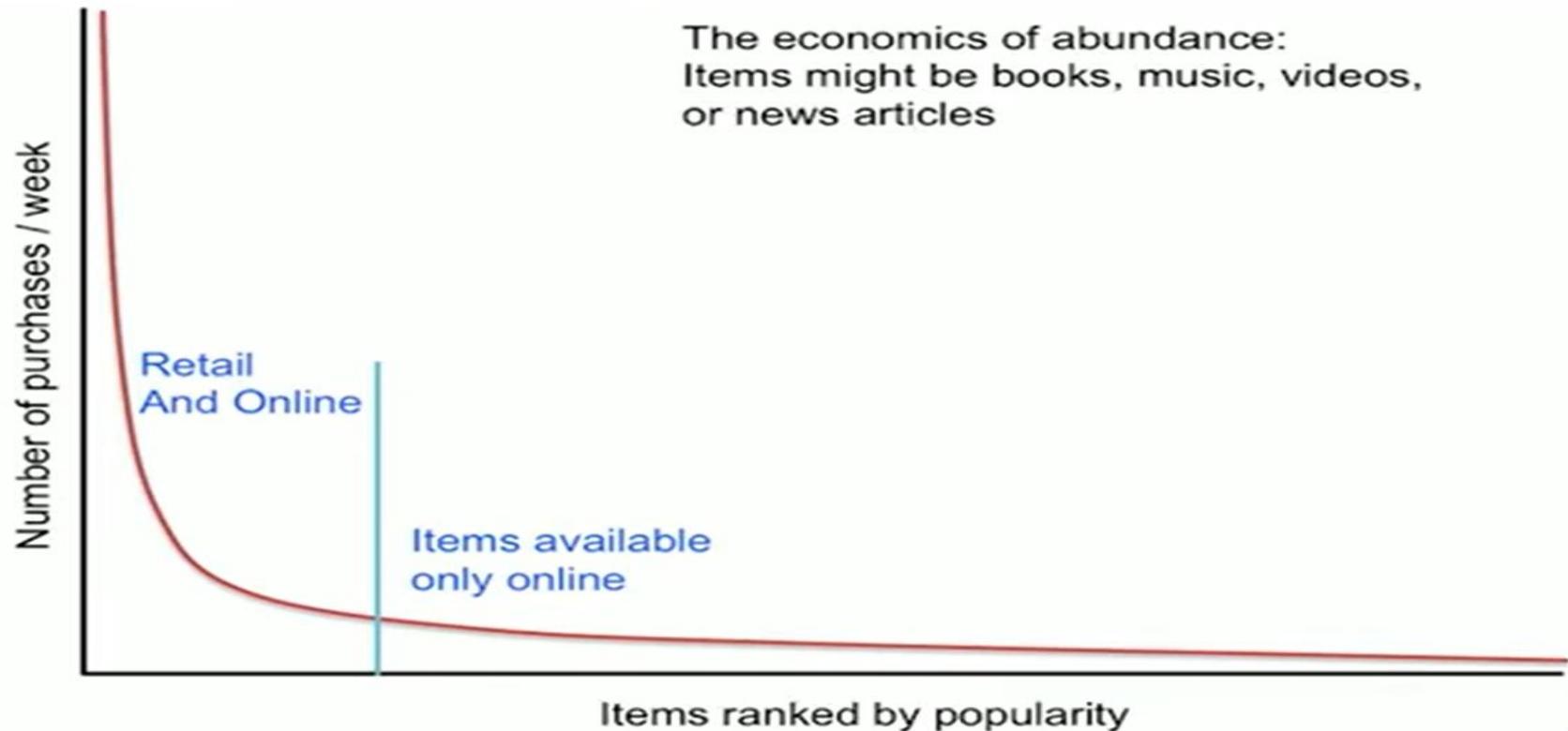
The economics of abundance:
Items might be books, music, videos,
or news articles

Items ranked by popularity

The Long Tail (1)



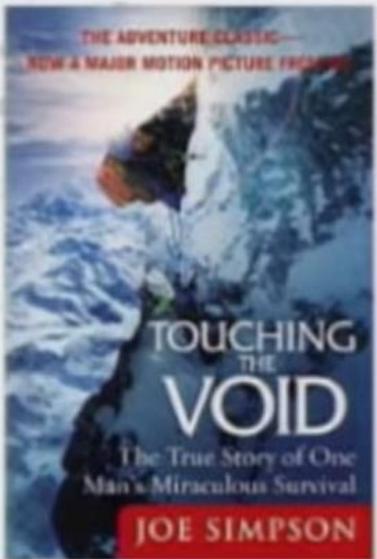
The Long Tail (1)



The Long Tail (2)

- More choice necessitates better filters
 - Recommendation engines
 - How **Into Thin Air** made **Touching the Void** a bestseller (<http://www.wired.com/wired/archive/12.10/tail.html>)
- Examples
 - Books, movies, music, news articles
 - People (friend recommendations on Facebook, LinkedIn, and Twitter)

Power of Recommendations: A Success story



Published in 1988



Published in 1996

"In 1988, a British mountain climber named Joe Simpson wrote a book called *Touching the Void*, a harrowing account of near death in the Peruvian Andes. It got good reviews, only a modest success, it was soon forgotten. Then, a decade later, a strange thing happened. Jon Krakauer wrote *Into Thin Air*, another book about a mountain-climbing tragedy, which became a publishing sensation. Suddenly, *Touching the Void* started to sell again." ... The Long Tail by Chris Anderson

Movie/TV show Recommendations

The screenshot shows the Netflix homepage with a light gray background featuring a network of circular nodes. At the top, there's a navigation bar with the Netflix logo, a 'Browse' dropdown, 'Kids' content, a 'DVD NEW EPISODES' section with a 'BLACK' thumbnail, a search bar, and a user profile for 'Divya'. Below the navigation is a 'Trending Now' section with four thumbnails:

- THE WEST WING**: Shows two men in suits.
- psych**: Shows a man in a green shirt with a thought bubble containing a cartoon pineapple.
- BABY COBRA**: Shows a woman in a suit standing next to a seal.
- MIDSOMER MURDERS**: Shows a silhouette of a town at night.

Friend Recommendations

facebook



You have a friend suggestion. What's this?

Friend Suggestions

Your friend suggestions are generated when one of your friends select you as someone who knows someone else on Facebook. If you add your suggested friends as friends, a normal friend request will be sent. If you do not, no one will be notified that you ignored a suggestion.

Okay

Job Recommendations

LinkedIn



10 jobs that match your preferences

sponsored

Front End Engineer & Growth Maker



Gliffy

San Francisco Bay Area

3 days ago



now

Principal Data Scientist



Move, Inc

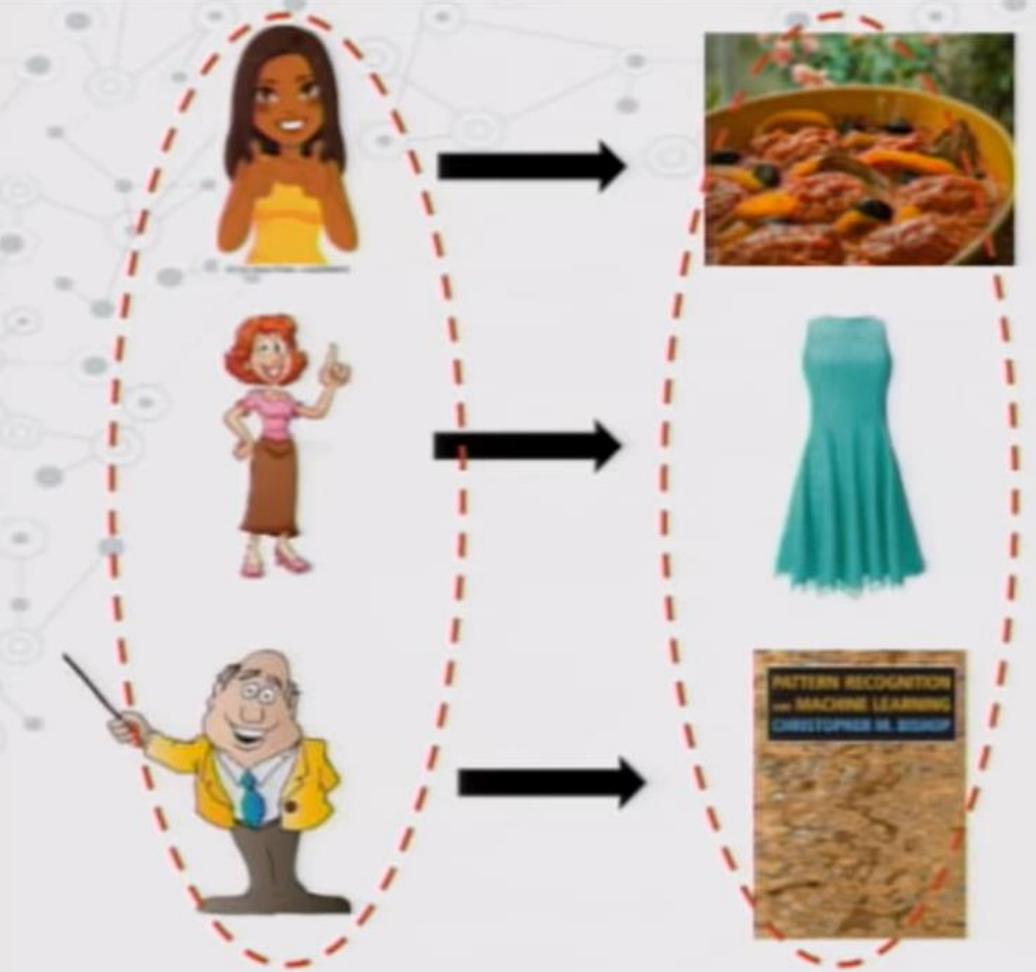
San Francisco Bay Area

3 hours ago



now

NCG - Research Scientist - Data Analytics



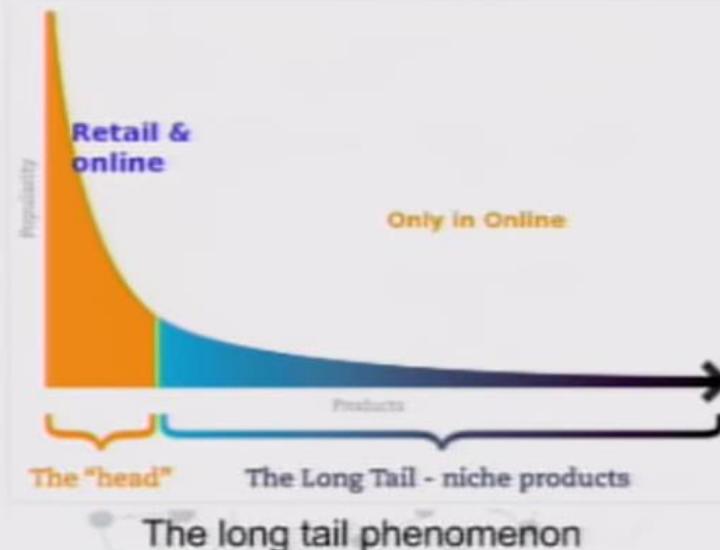
Users

Matching Items

A Naive understanding of Recommender Systems

1. Why Recommender Systems?

Goal of a Recommender System:
Identify products most relevant to
the user (Eg. Top n offers).



Types of Recommendations

- Editorial and hand curated
 - List of favorites
 - Lists of “essential” items
- Simple aggregates
 - Top 10, Most Popular, Recent Uploads
- Tailored to individual users
 - Amazon, Netflix, Pandora ...
 - Our focus here

Quiz

What are users and matching items the following cases:

-  a.) LinkedIn
-  b.) Facebook
-  c.) Amazon
-  d.) Netflix

Quiz

What are users and matching items the following cases:

- a.) LinkedIn (Users: members, Items: jobs)
- b.) Facebook (Users: members, Items: members)
- c.) Amazon (Users: members, Items: products, e.g., books)
- d.) Netflix (Users: members, Items: movies, TV shows)

How do Recommendation Engines work?

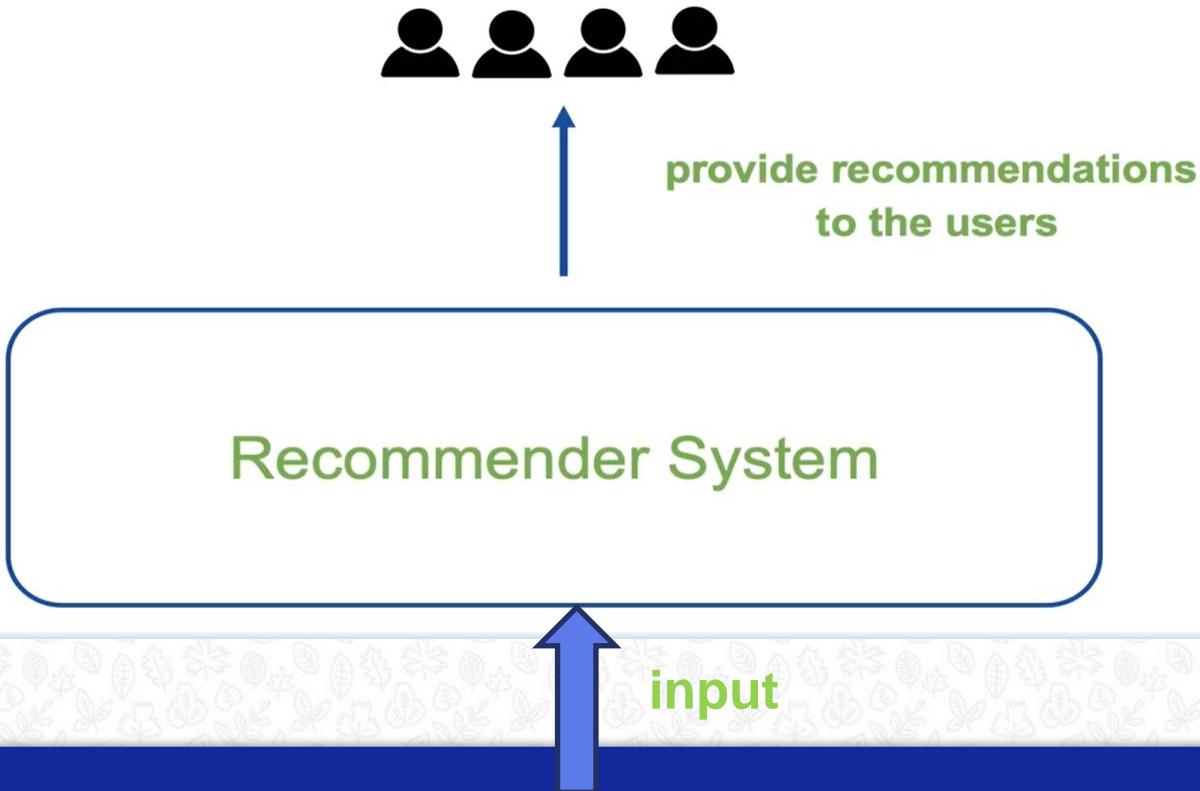


Typically, a recommendation engine processes data through the below four phases-

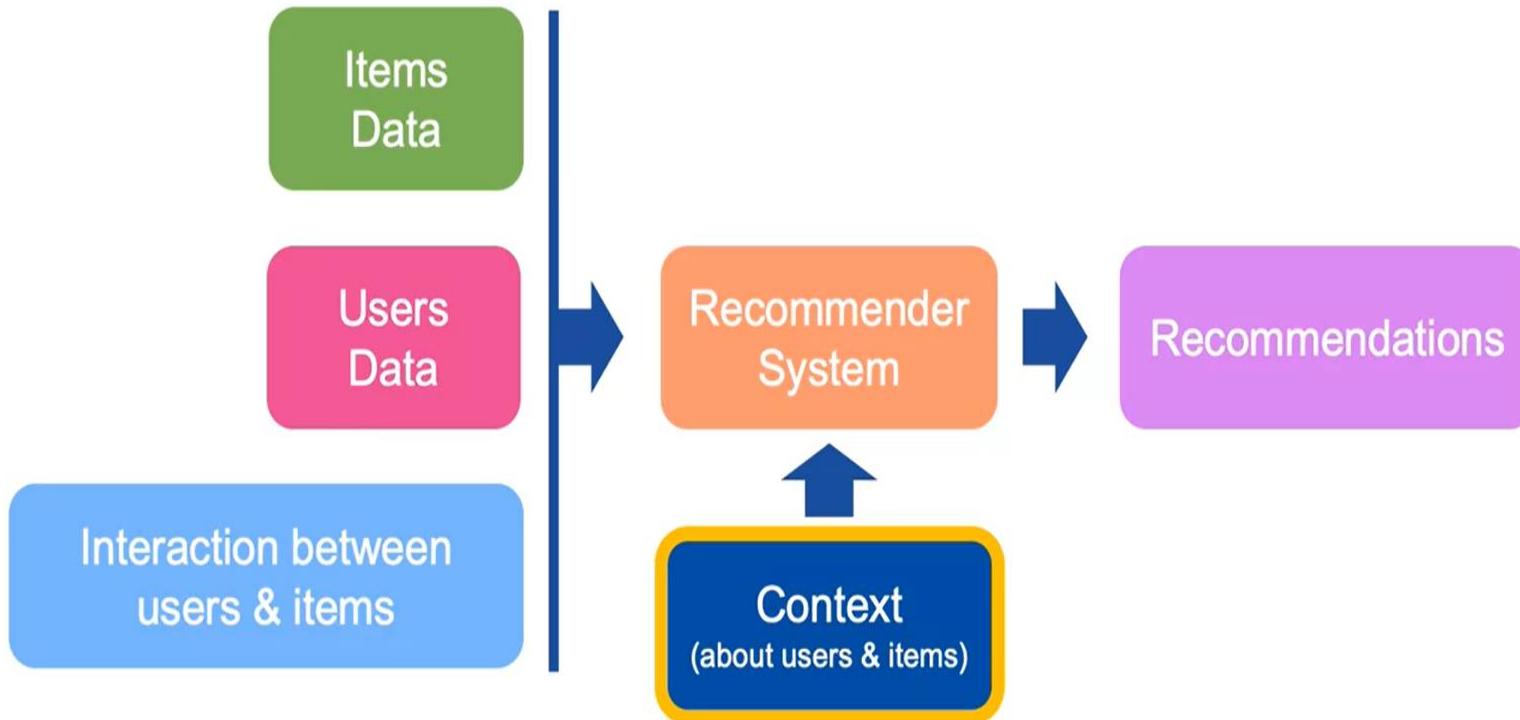
- 1. Collection**
- 2. Storing**
- 3. Analyzing**
- 4. Filtering**



Recommender Systems



Input Data



Solution 0: Popularity based Recommender System

Recommend items viewed/purchased by most people

Recommendations: Ranked list of items by their purchase count

Google

News

U.S. edition ▾

Top Stories

Most Popular

Republican Convention Day 3: Trump Makes an Entrance

New York Times | 3 hours ago

Day three of the Republican National Convention will feature expressions of support from several of the men who Donald J. Trump vanquished in the primaries.

Trump makes an entrance at the RNC

Today

1 day

1 week

1 month

The GOP's new convention theme: 'Lock her up!'

Washington Post | 3 hours ago

CLEVELAND -- The refrain of the Republican convention hasn't been "Make America Great Again." It's been "Lock her up!"

KOHL'S

JUMPING BEANS®
Coral Color Denim Jean for Girls
\$24.99
Denim & Jeanwear Collection

BATH TOWELS AND BATH RUGS
\$10-\$12.99
Bath & Laundry

See more recommendations

POPULAR PRODUCTS

Black \$14.99
Women's Essential Tee
4.5 stars (100+ reviews)

Black \$14.99
Women's Essential Tee
4.5 stars (100+ reviews)

White \$12.99
Women's Essential Tee
4.5 stars (100+ reviews)

Blue \$14.99
Men's Big Logo Microfleece
Tee
4.5 stars (100+ reviews)

Blue \$14.99
Women's Essential Tee
4.5 stars (100+ reviews)

Quiz

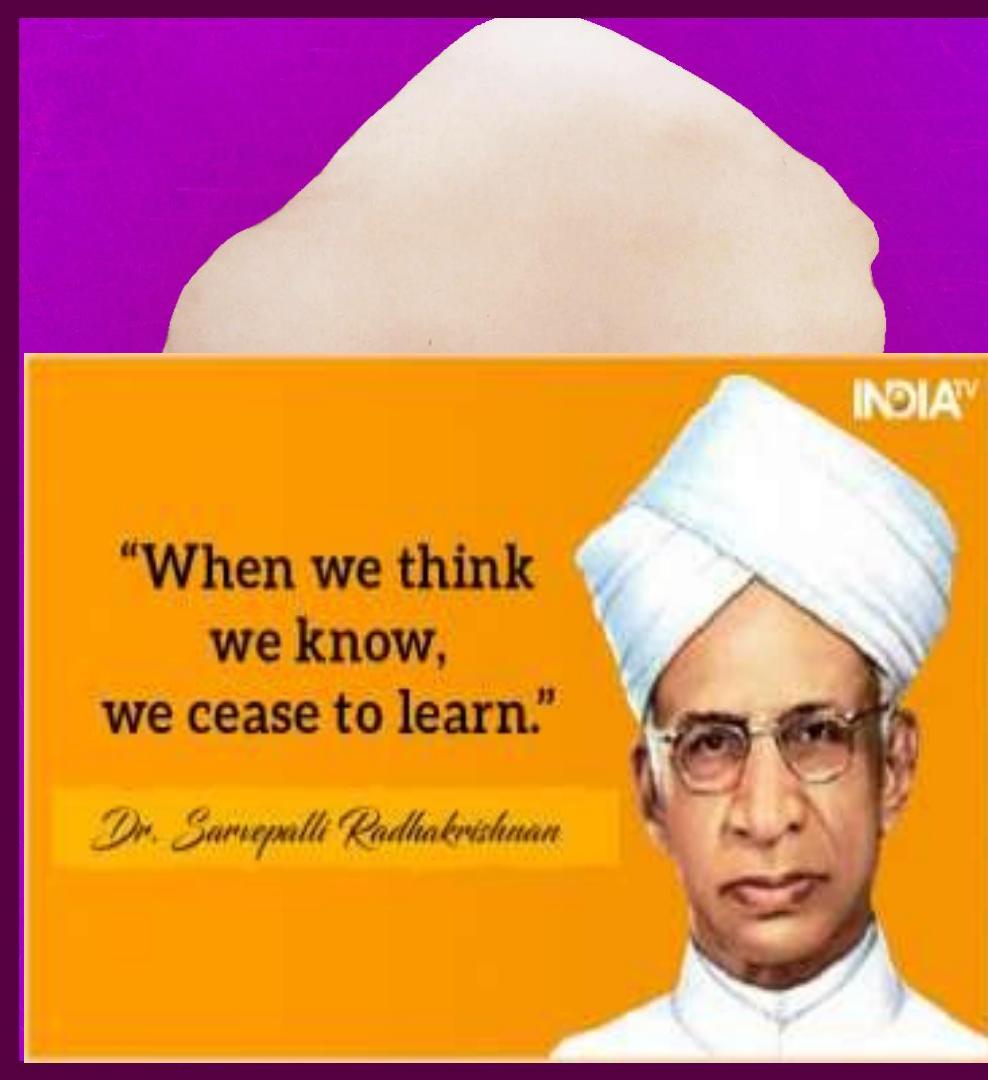
Which of the following is true of a popularity based recommender system?

Can generate Personalized Recommendations?	
Can use Context (Eg. time of day)?	
Can use User Features?	
Can use Item Features?	
Can use Purchase History?	
Is it Scalable?	

Quiz

Which of the following is true of a popularity based recommender system?

Can generate Personalized Recommendations?	✗
Can use Context (Eg. time of day)?	✓
Can use User Features?	✓
Can use Item Features?	✓
Can use Purchase History?	✓
Is it Scalable?	✓

A portrait of Dr. Sarvepalli Radhakrishnan, an Indian philosopher and statesman, wearing a white turban and glasses. He is positioned on the left side of the image, with a large white cloud shape behind him.

"When we think
we know,
we cease to learn."

Dr. Sarvepalli Radhakrishnan

“Knowledge gives us power, love gives us the fullness.”

-Dr Sarvepalli Radhakrishnan

→ Topics to be Discussed

5.1.1 A model for Recommendation systems

5.1.2 Content Based Recommendations

5.1.3 Collaborative Filtering

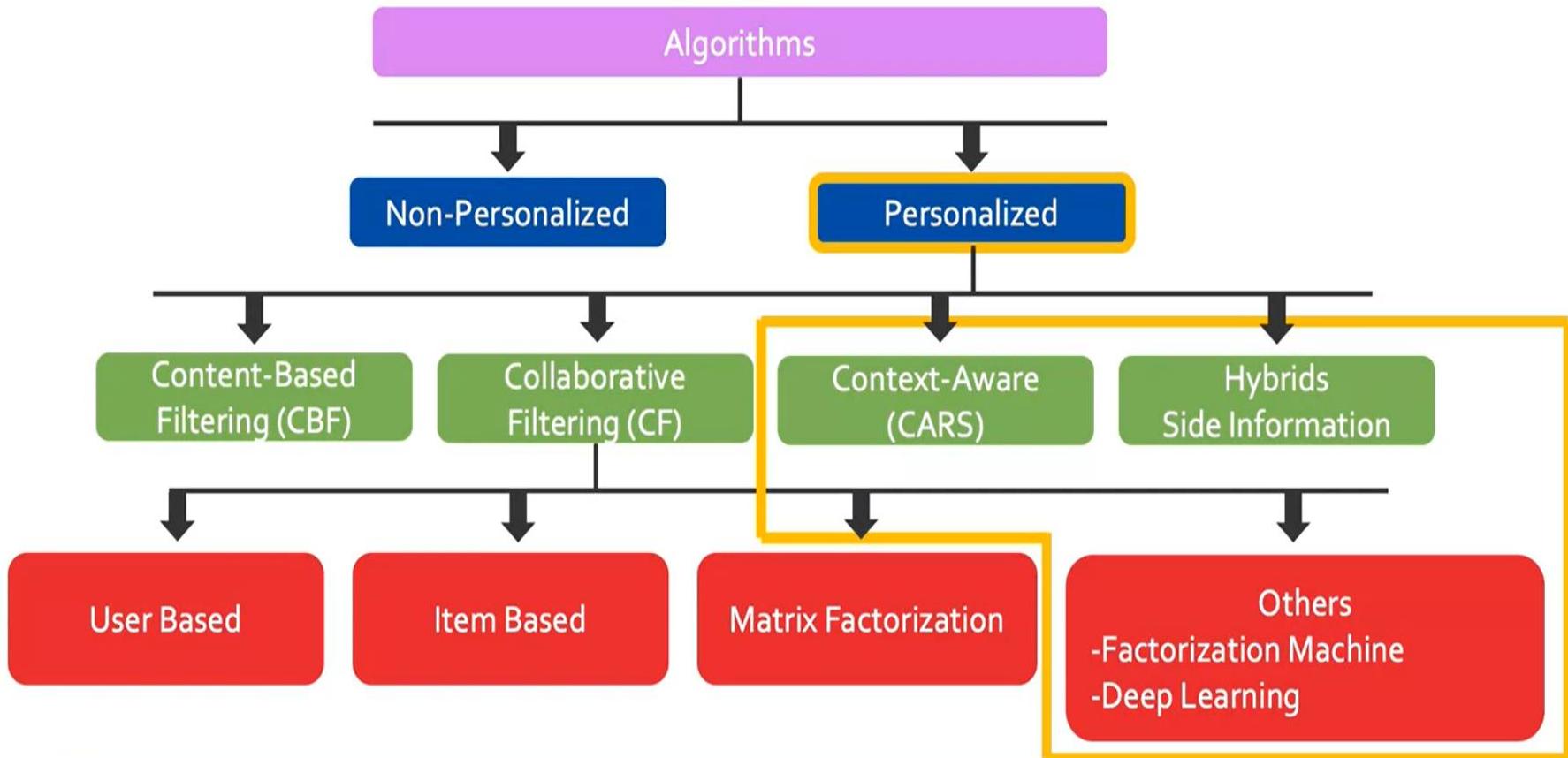
5.2.1 Case Study :Product Recommendation

5.3.1 Social Networks as Graphs,

5.3.2 Clustering of Social-Network Graphs

5.3.3 Direct Discovery of Communities in a social graph.

Taxonomy of Recommender Systems



Formal Model

- C = set of **Customers**
- S = set of **Items**
- **Utility function** $u: C \times S \rightarrow R$
 - R = set of ratings
 - R is a totally ordered set
 - e.g., 0-5 stars, real number in $[0,1]$

Utility Matrix

	Avatar	LOTR	Matrix	Pirates
Alice	1		0.2	
Bob		0.5		0.3
Carol	0.2		1	
David				0.4

Key Problems

(1) Gathering “known” ratings for matrix

- How to collect the data in the utility matrix

(2) Extrapolate unknown ratings from the known ones

- Mainly interested in high unknown ratings
 - We are not interested in knowing what you don't like but what you like

(3) Evaluating extrapolation methods

- How to measure success/performance of recommendation methods

(1) Gathering Ratings

- **Explicit**

- Ask people to rate items
- Doesn't scale: only a small fraction of users leave ratings and reviews

- **Implicit**

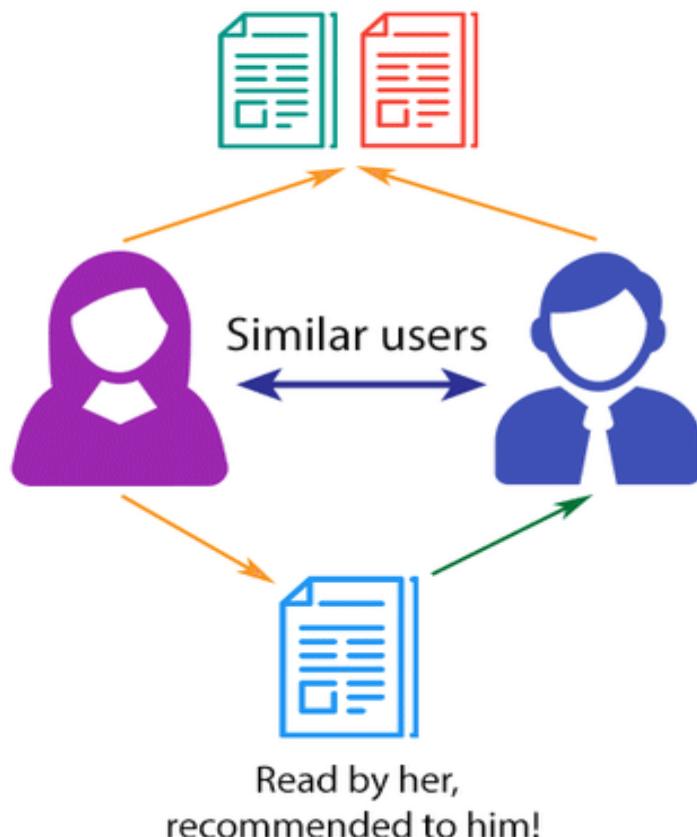
- Learn ratings from user actions
 - E.g., purchase implies high rating
- What about low ratings?

(2) Extrapolating Utilities

- Key problem: matrix U is sparse
 - Most people have not rated most items
 - Cold start:
 - New items have no ratings
 - New users have no history

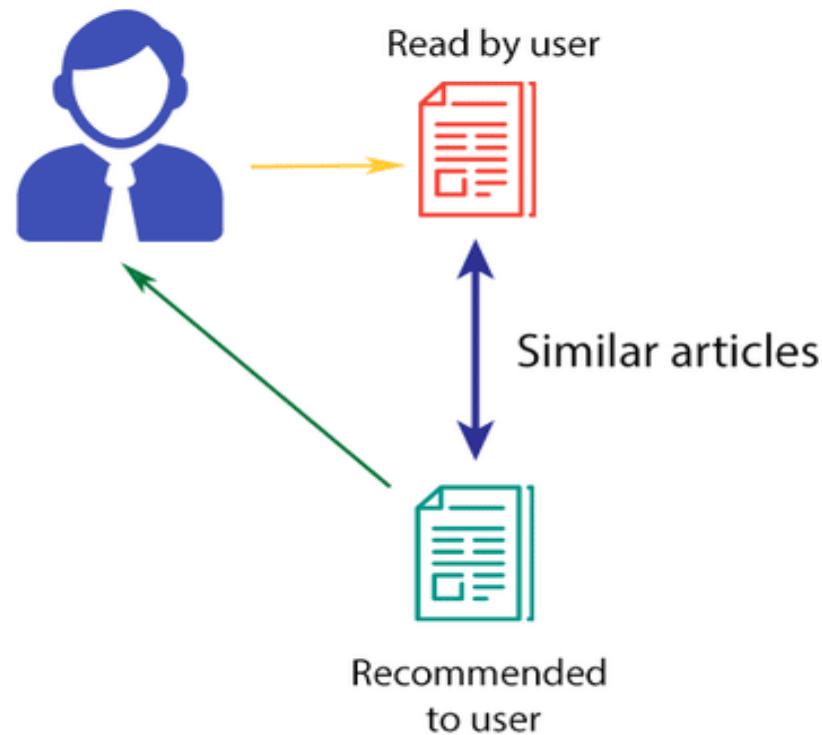
COLLABORATIVE FILTERING

Read by both users



CONTENT-BASED FILTERING

Read by user



A model for Recommendation systems

Recommendation systems use a number of different technologies.

We can classify these systems into two broad groups.

- Content-based systems examine properties of the items recommended. For instance, if a Netflix user has watched many cowboy movies, then recommend a movie classified in the database as having the “cowboy” genre.
- Collaborative filtering systems recommend items based on similarity measures between users and/or items. The items recommended to a user are those preferred by similar users.

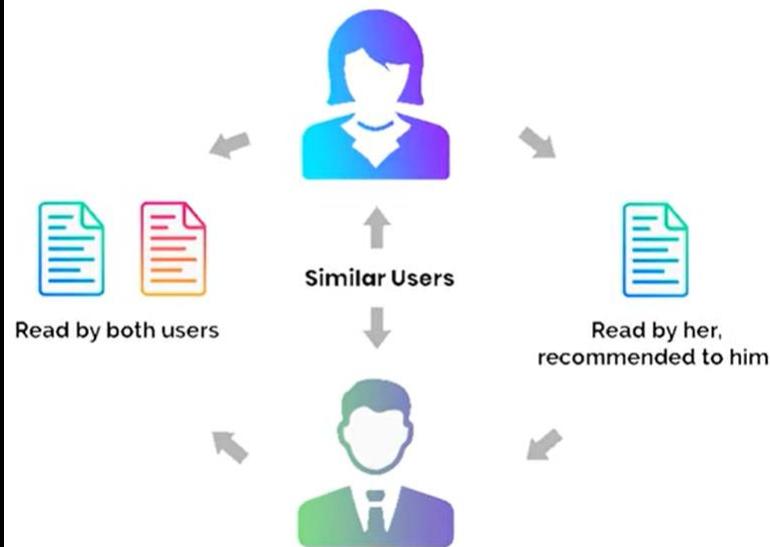
There are mainly three types of recommendation engines –

1. Collaborative Filtering

It is based on collecting & analyzing information & predicting what they will like based on the similarity with other users.



Collaborative filtering



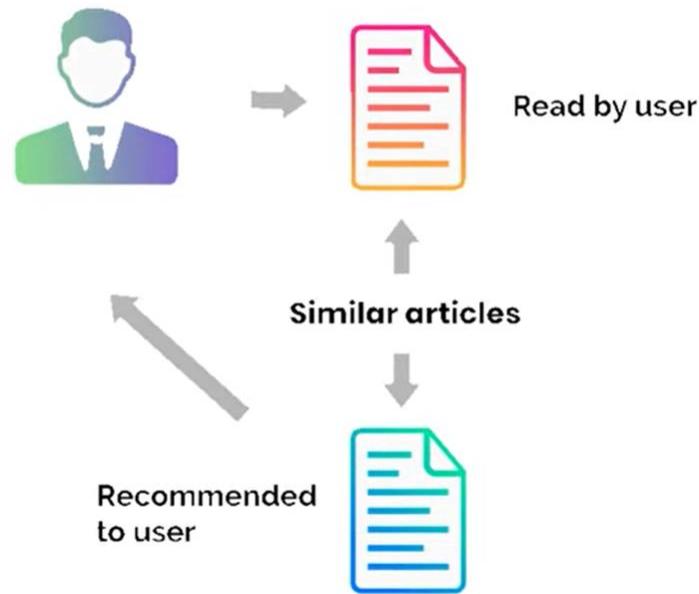
2. Content-Based Filtering

They are mainly based on the description of an item & a profile of the user's preferred choices.

For example, if a user likes to watch movies like *Mission Impossible*, then the recommender system recommends movies of the action genre or movies of Tom Cruise.

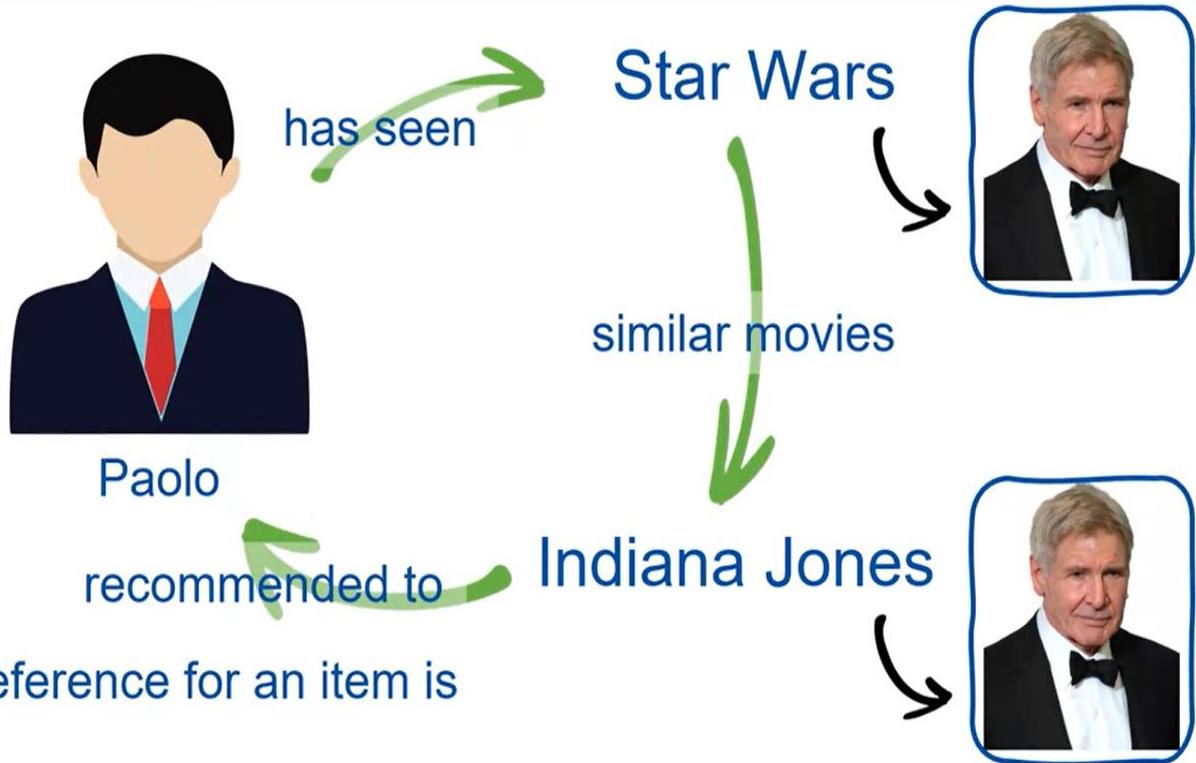
MISSION: IMPOSSIBLE

Content-based filtering



Content Based Filtering

- compare items based on their attributes



A user that expressed a preference for an item is likely to like similar items

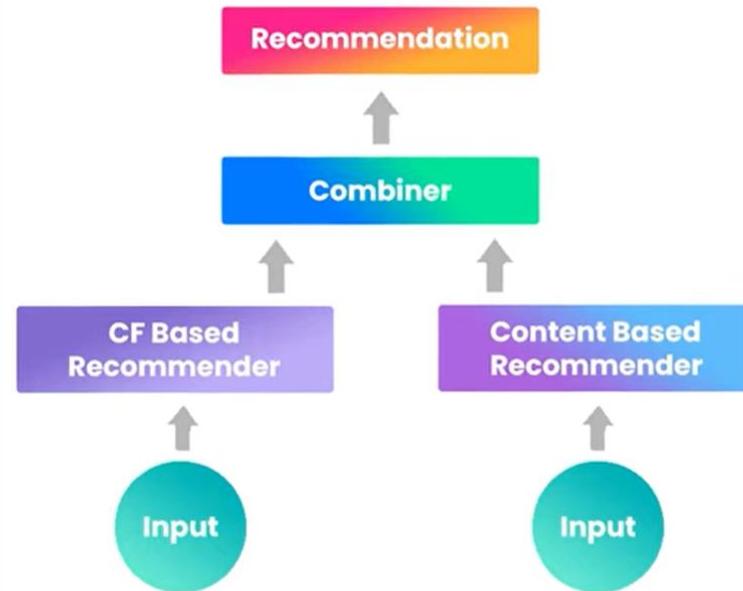
3. Hybrid Recommendation Systems

They are essentially the combination of diverse rating & sorting algorithms.

Netflix is an excellent **example** of a hybrid recommendation system



Hybrid Recommendations



Content-based Recommendations

Main idea: Recommend items to customer x similar to previous items rated highly by x

Examples:

- Movies
 - Same actor(s), director, genre, ...
- Websites, blogs, news
 - Articles with “similar” content
- People
 - Recommend people with many common friends

Plan of Action



likes
→

A yellow arrow pointing from the thought bubble towards the 'Item profiles' box.

Item profiles



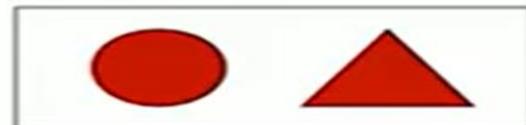
Plan of Action



likes



Item profiles



build



Red
Circles
Triangles

User profile

Plan of Action



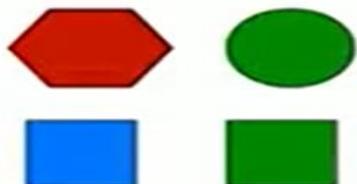
likes



Item profiles



build



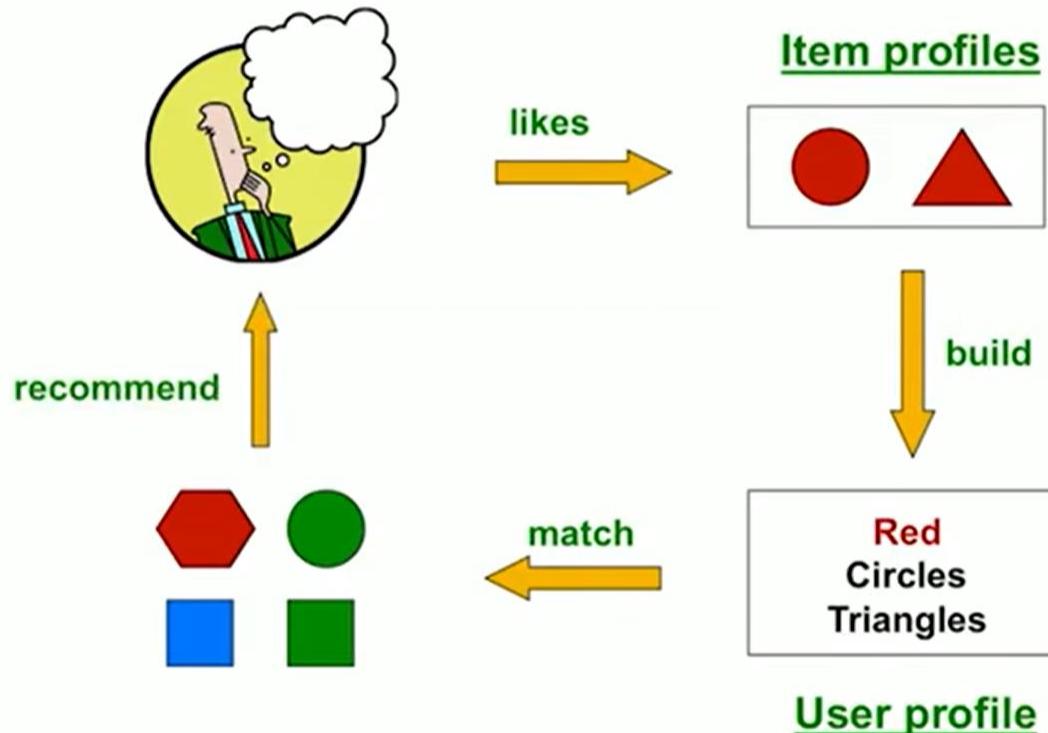
match



**Red
Circles
Triangles**

User profile

Plan of Action



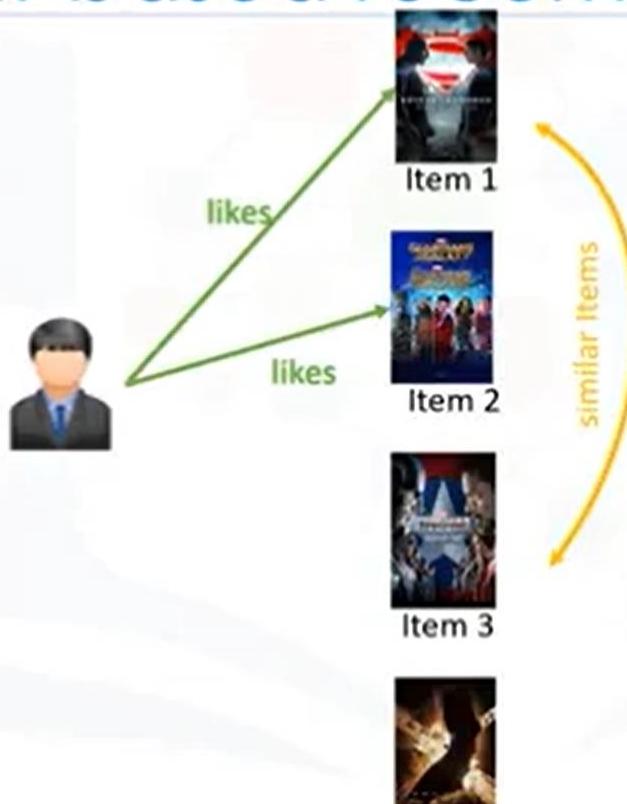
Content-based recommender systems



Content-based recommender systems



Content-based recommender systems



Content-based recommender systems



Item Profiles

- For each item, create an **item profile**
- Profile is a set of features
 - **Movies:** author, title, actor, director,...
 - **Images, videos:** metadata and tags
 - **People:** Set of friends

Item Profiles

- For each item, create an **item profile**
- Profile is a set of features
 - **Movies:** author, title, actor, director,...
 - **Images, videos:** metadata and tags
 - **People:** Set of friends
- Convenient to think of the item profile as a vector
 - One entry per feature (e.g., each actor, director,...)
 - Vector might be boolean or real-valued

Text features

- Profile = set of “important” words in item (document)
- How to pick important words?
 - Usual heuristic from text mining is **TF-IDF** (Term frequency * Inverse Doc Frequency)

User Profiles

- User has rated items with profiles i_1, \dots, i_n
- Simple: (weighted) average of rated item profiles
- Variant: Normalize weights using average rating of user
- More sophisticated aggregations possible

Content-based recommender systems

Batman v Superman



(Adventure, Super Hero)

Guardians of the Galaxy



(Comedy, Adventure, Super Hero, Sci-Fi)

Captain America: Civil War



(Comedy, Super Hero)



Hitchhiker's guide to the galaxy



(Comedy, Adventure, Sci-Fi)

Batman begins



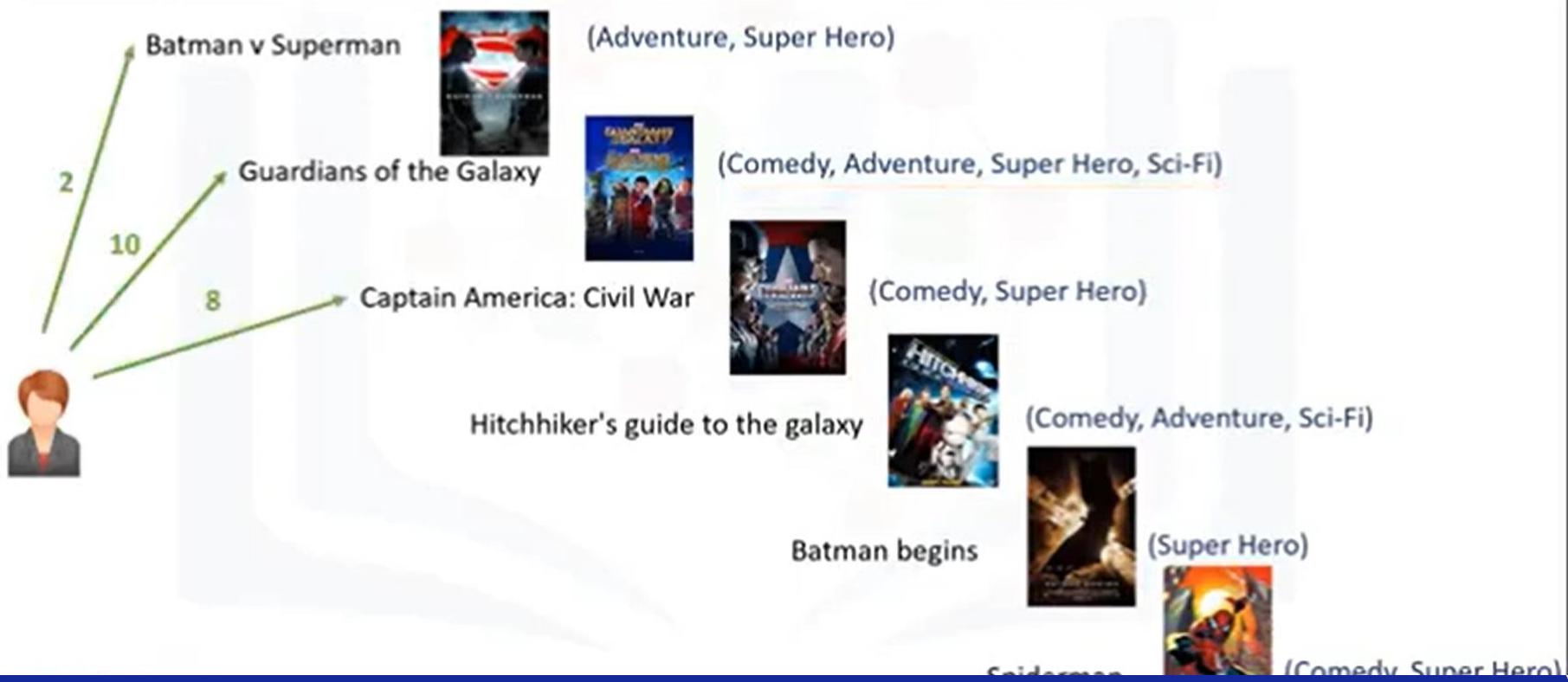
(Super Hero)

Spiderman



(Comedy, Super Hero)

Content-based recommender systems



Content-based recommender systems



Weighing the genres

	
	2
	10
	8

Input User Ratings

Weighing the genres

	2
	10
	8

Input User Ratings

	Comedy	Adventure	Super Hero	Sci-Fi
	0	1	1	0
	1	1	1	1
	1	0	1	0

Weighing the genres

	2
	10
	8

Input User Ratings

X

	Comedy	Adventure	Super Hero	Sci-Fi
	0	1	1	0
	1	1	1	1
	1	0	1	0

Weighing the genres

Weighted Genre Matrix



		Comedy	Adventure	Super Hero	Sci-Fi
2	X	0	1	1	0
10		1	1	1	1
8		1	0	1	0

Input User Ratings

Movies Matrix

=

	Comedy	Adventure	Super Hero	Sci-Fi
	0	2	2	0
	10	10	10	10
	8	0	8	0

Weighing the genres

2	
10	
8	

Input User Ratings

	Comedy	Adventure	Super Hero	Sci-Fi
	0	1	1	0
	1	1	1	1
	1	0	1	0

Movies Matrix

X

=

	Comedy	Adventure	Super Hero	Sci-Fi
	0	2	2	0
	10	10	10	10
	8	0	8	0

Weighted Genre Matrix

User profile

	Comedy	Adventure	Super Hero	Sci-Fi
	18	12	20	10

Weighing the genres



	2	X	0 1 1 0
	10		1 1 1 1
	8		1 0 1 0
Input User Ratings		Movies Matrix	

Weighted Genre Matrix

	Comedy	Adventure	Super Hero	Sci-Fi
	0	2	2	0
	10	10	10	10
	8	0	8	0

User profile

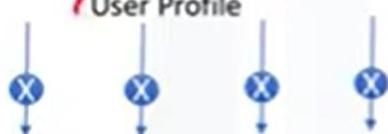
	Comedy	Adventure	Super Hero	Sci-Fi
	0.3	0.2	0.33	0.16

Candidate movies for recommendation

	Comedy	Adventure	Super Hero	Sci-Fi
	1	1	0	1
	0	0	1	0
	1	0	1	0

Finding the recommendation

	Comedy	Adventure	Super Hero	Sci-Fi
User Profile	0.3	0.2	0.33	0.16



	Comedy	Adventure	Super Hero	Sci-Fi
	1	1	0	1
	0	0	1	0
	1	0	1	0

Movies Matrix

Finding the recommendation

	Comedy	Adventure	Super Hero	Sci-Fi
User Profile	0.3	0.2	0.33	0.16



	Comedy	Adventure	Super Hero	Sci-Fi
	1	1	0	1
	0	0	1	0
	1	0	1	0

Movies Matrix

$$\begin{matrix} & \begin{matrix} \text{Comedy} & \text{Adventure} & \text{Super Hero} & \text{Sci-Fi} \end{matrix} \\ \begin{matrix} \text{Movie 1} \\ \text{Movie 2} \\ \text{Movie 3} \end{matrix} & = \begin{matrix} \text{Movie 1} & \begin{matrix} 0.3 & 0.2 & 0 & 0.16 \end{matrix} \\ \text{Movie 2} & \begin{matrix} 0 & 0 & 0.33 & 0 \end{matrix} \\ \text{Movie 3} & \begin{matrix} 0.3 & 0 & 0.33 & 0 \end{matrix} \end{matrix} \end{matrix}$$

Weighted Movies Matrix

Finding the recommendation

	Comedy	Adventure	Super Hero	Sci-Fi
User Profile	0.3	0.2	0.33	0.16



	Comedy	Adventure	Super Hero	Sci-Fi
	1	1	0	1
	0	0	1	0
	1	0	1	0

$$\text{Movies Matrix} = \begin{array}{c} \text{Weighted Movies Matrix} \\ \Sigma \end{array} = \begin{array}{c} \text{Recommendation Matrix} \end{array}$$

The diagram illustrates the calculation of a recommendation matrix. It shows the multiplication of the Movies Matrix and the Weighted Movies Matrix, resulting in the Recommendation Matrix. The Weighted Movies Matrix is obtained by multiplying the Movies Matrix by the User Profile matrix. The final recommendation values are calculated by summing the weighted scores for each genre.

	Comedy	Adventure	Super Hero	Sci-Fi
	0.3	0.2	0	0.16
	0	0	0.33	0
	0.3	0	0.33	0

Weighted Average

0.66
0.33
0.63

Content-based recommender systems



Content-based recommender systems



Pros: Content-based Approach

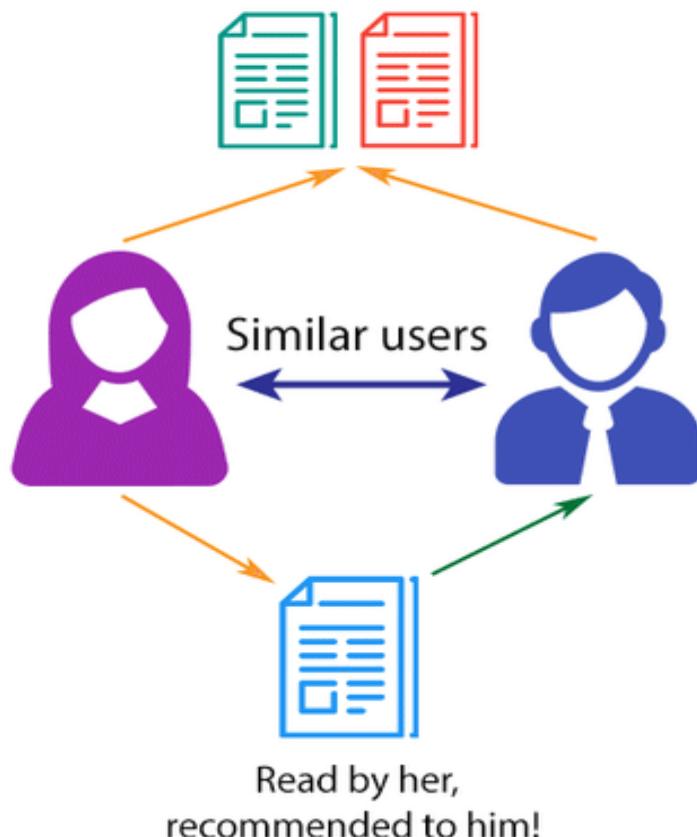
- No need for data on other users
- Able to recommend to users with unique tastes
- Able to recommend new & unpopular items
 - No first-rater problem
- Explanations for recommended items
 - Content features that caused an item to be recommended

Cons: Content-based Approach

- Finding the appropriate features is hard
 - E.g., images, movies, music
- Overspecialization
 - Never recommends items outside user's content profile
 - People might have multiple interests
 - Unable to exploit quality judgments of other users
- Cold-start problem for new users
 - How to build a user profile?

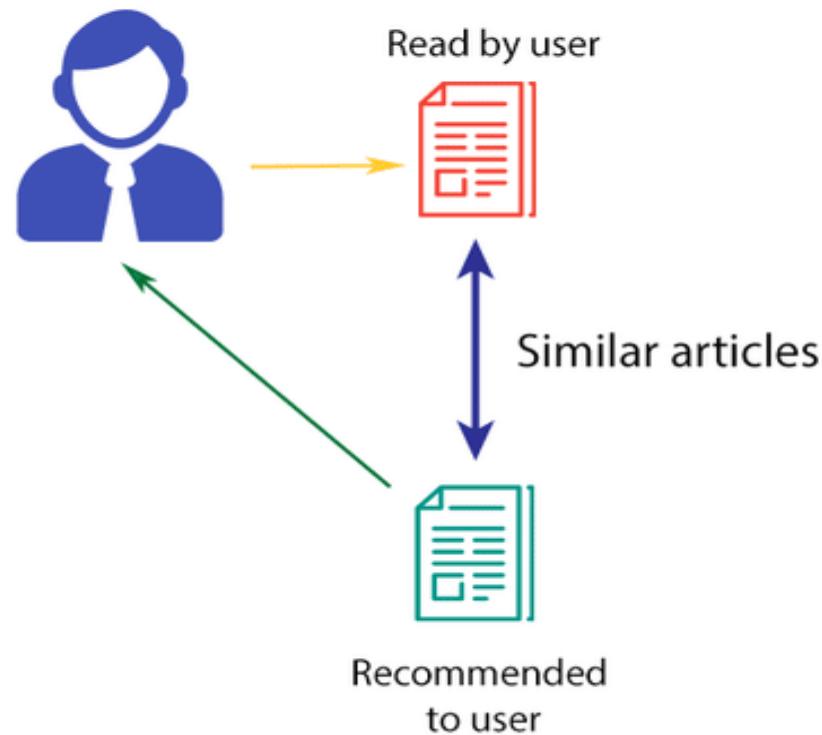
COLLABORATIVE FILTERING

Read by both users



CONTENT-BASED FILTERING

Read by user



→ Topics to be Discussed

5.1.1 A model for Recommendation systems

5.1.2 Content Based Recommendations

5.1.3 Collaborative Filtering

5.2.1 Case Study :Product Recommendation

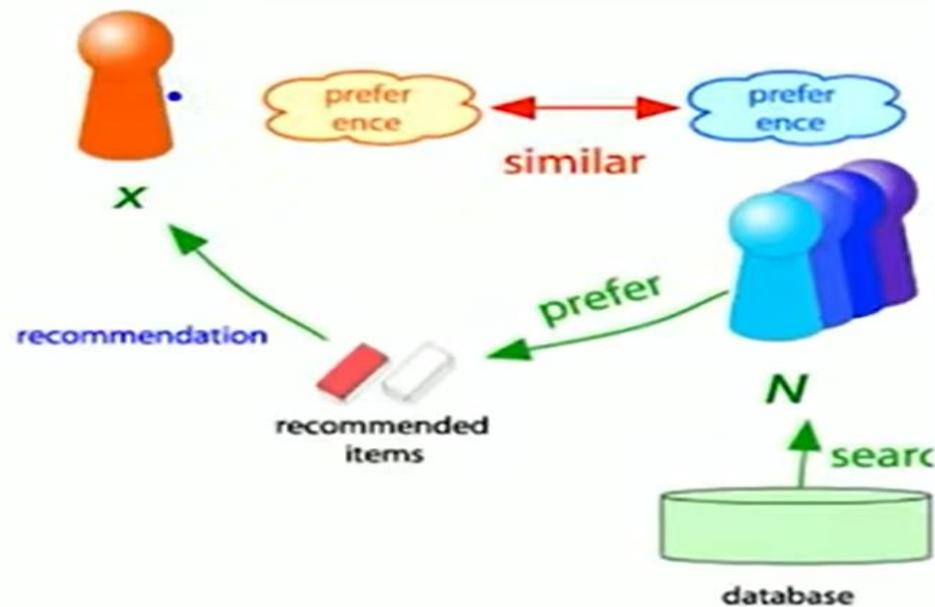
5.3.1 Social Networks as Graphs,

5.3.2 Clustering of Social-Network Graphs

5.3.3 Direct Discovery of Communities in a social graph.

Collaborative Filtering

- Consider user x
- Find set N of other users whose ratings are “similar” to x 's ratings
- Estimate x 's ratings based on ratings of users in N



Collaborative Filtering

What is Collaborative Filtering?

- **Collaborative Filtering** is a Machine Learning technique used to identify relationships between pieces of data.
- This technique is frequently used in recommender systems to identify similarities between user data and items.
- This means that if *Users A* and *B* both like *Product A*, and *User B* also likes *Product B*, then *Product B* could be recommended to *User A* by the system.

Collaborative filtering

- User-based collaborative filtering
- Item-based collaborative filtering



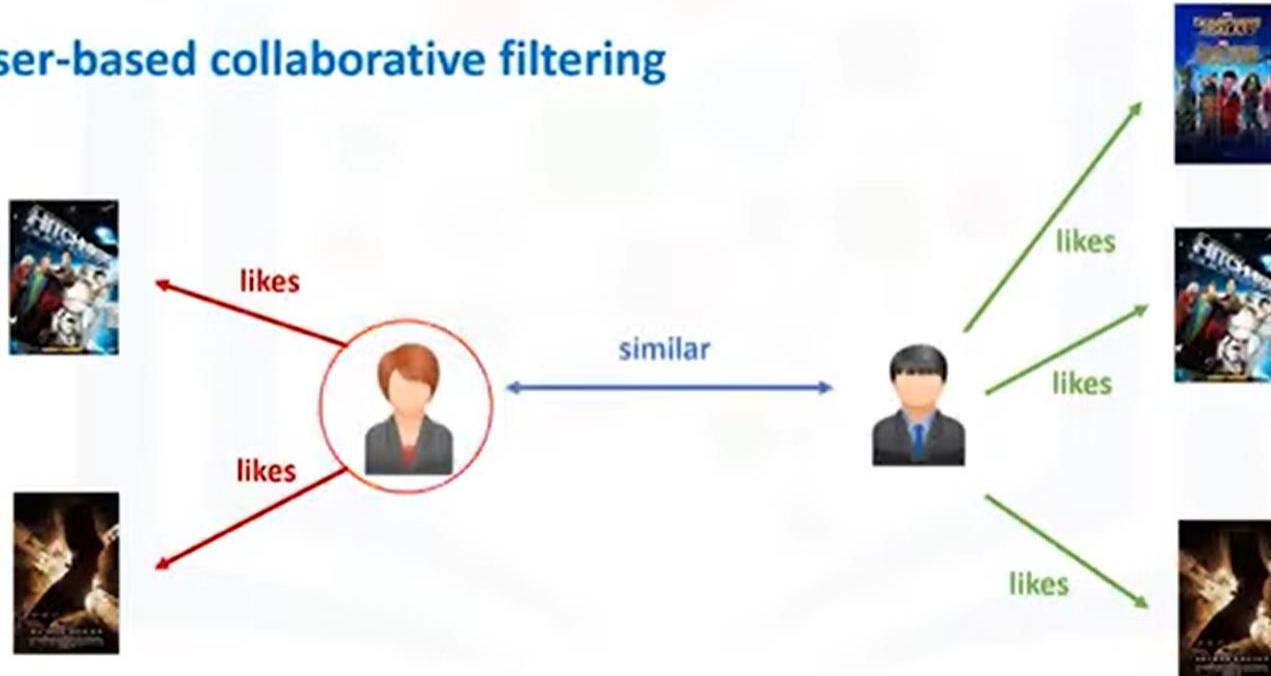
Collaborative filtering

- User-based collaborative filtering



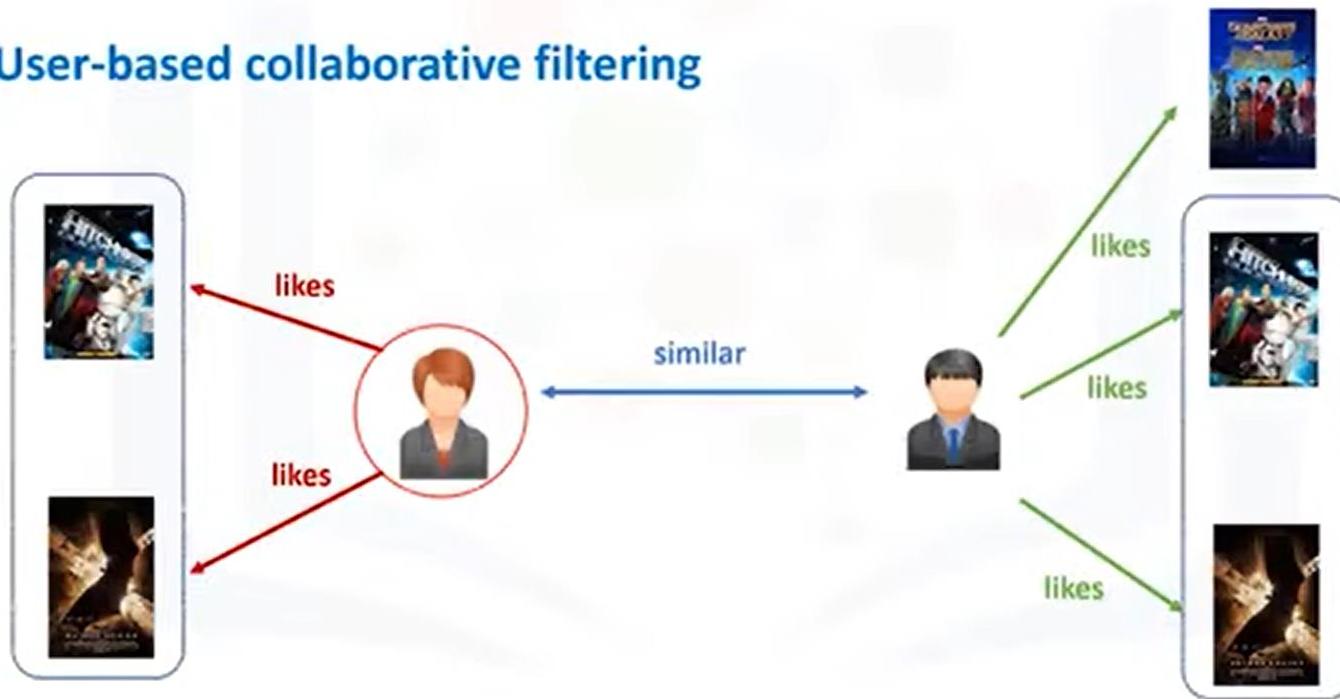
Collaborative filtering

- User-based collaborative filtering



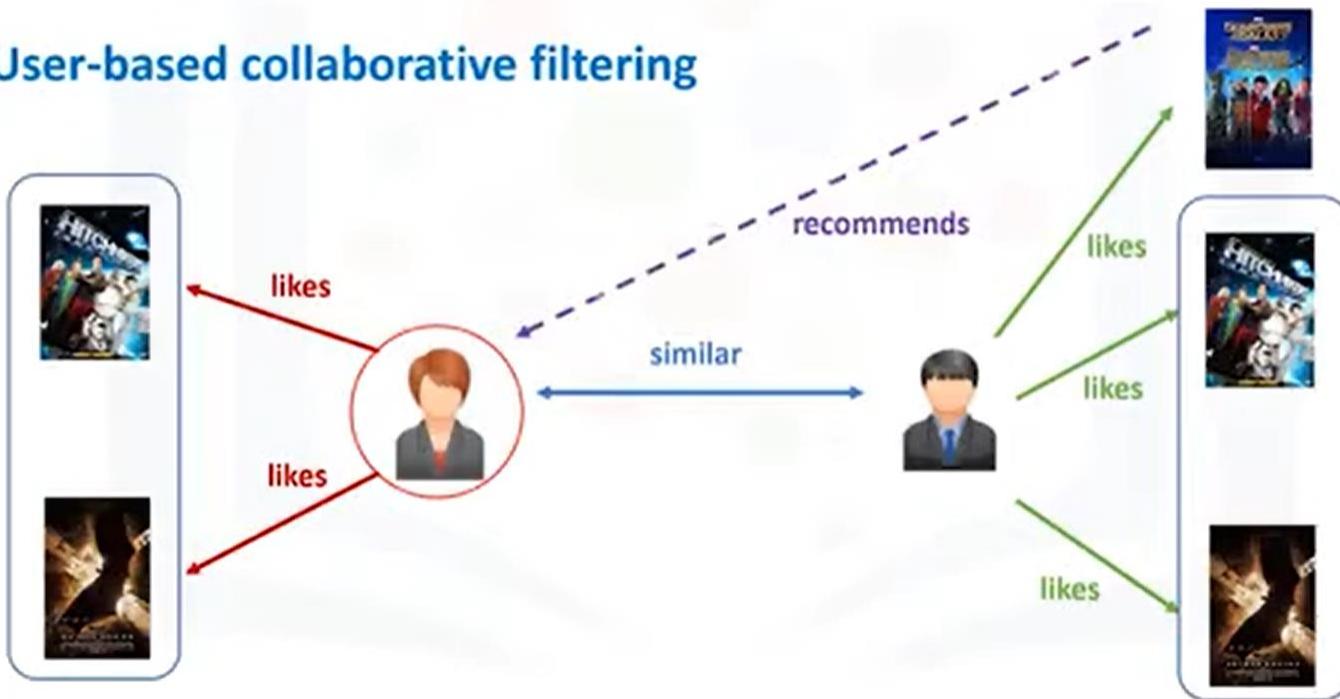
Collaborative filtering

- User-based collaborative filtering



Collaborative filtering

- User-based collaborative filtering



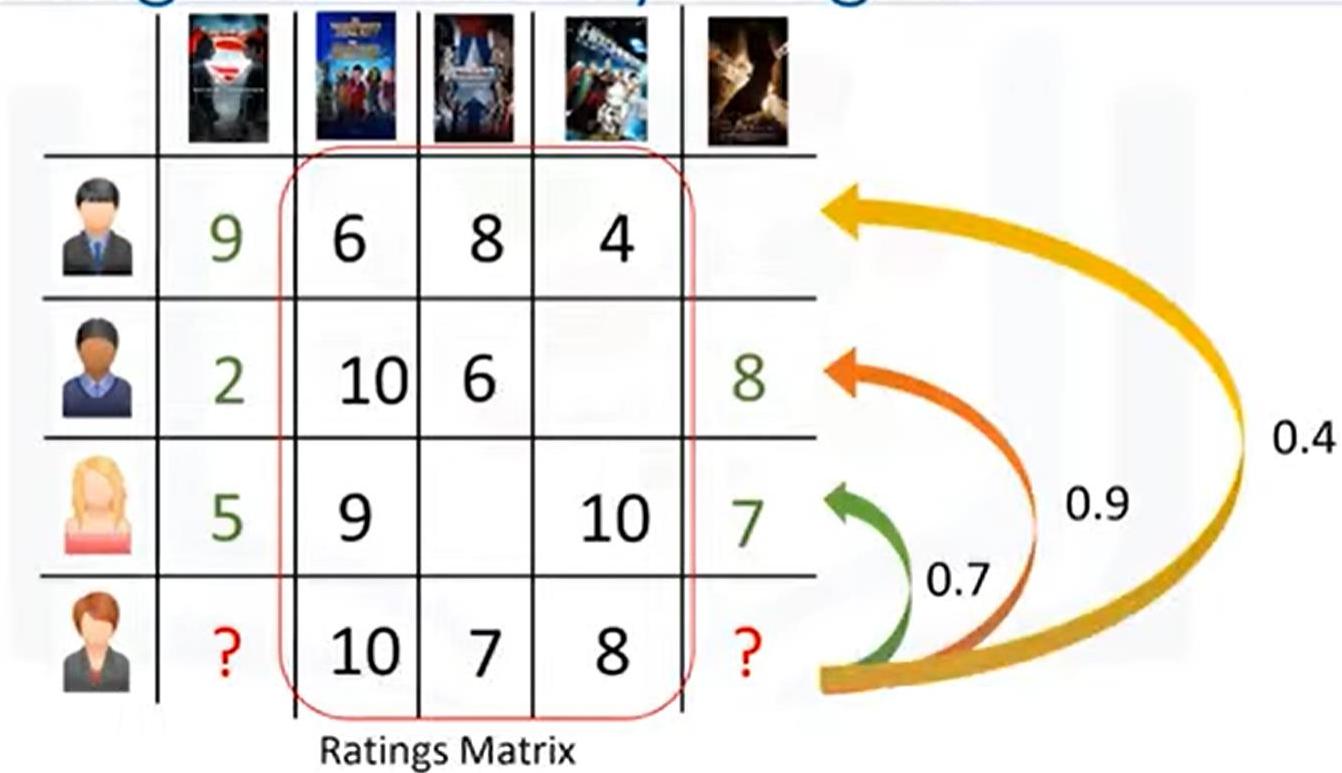
User ratings matrix

	9	6	8	4	
	2	10	6		8
	5	9		10	7
Active user		10	7	8	
Ratings Matrix					

Learning the similarity weights

	9	6	8	4	
	2	10	6		8
	5	9		10	7
	?	10	7	8	?
Ratings Matrix					

Learning the similarity weights



Creating the weighted ratings matrix

	9	
	2	8
	5	7

Ratings Matrix Subset

		Similarity Index
	x	0.4
	x	0.9
	x	0.7

Similarity Matrix

=

3.6	
1.8	7.2
3.5	4.9

Weighted Ratings Matrix

	Similarity Index	
	0.4	0.9
0.7	0.9	0.7
Similarity Matrix		
3.6	1.8	7.2
1.8	3.5	4.9
Weighted Ratings Matrix		

sum_similarityIndex

Σ

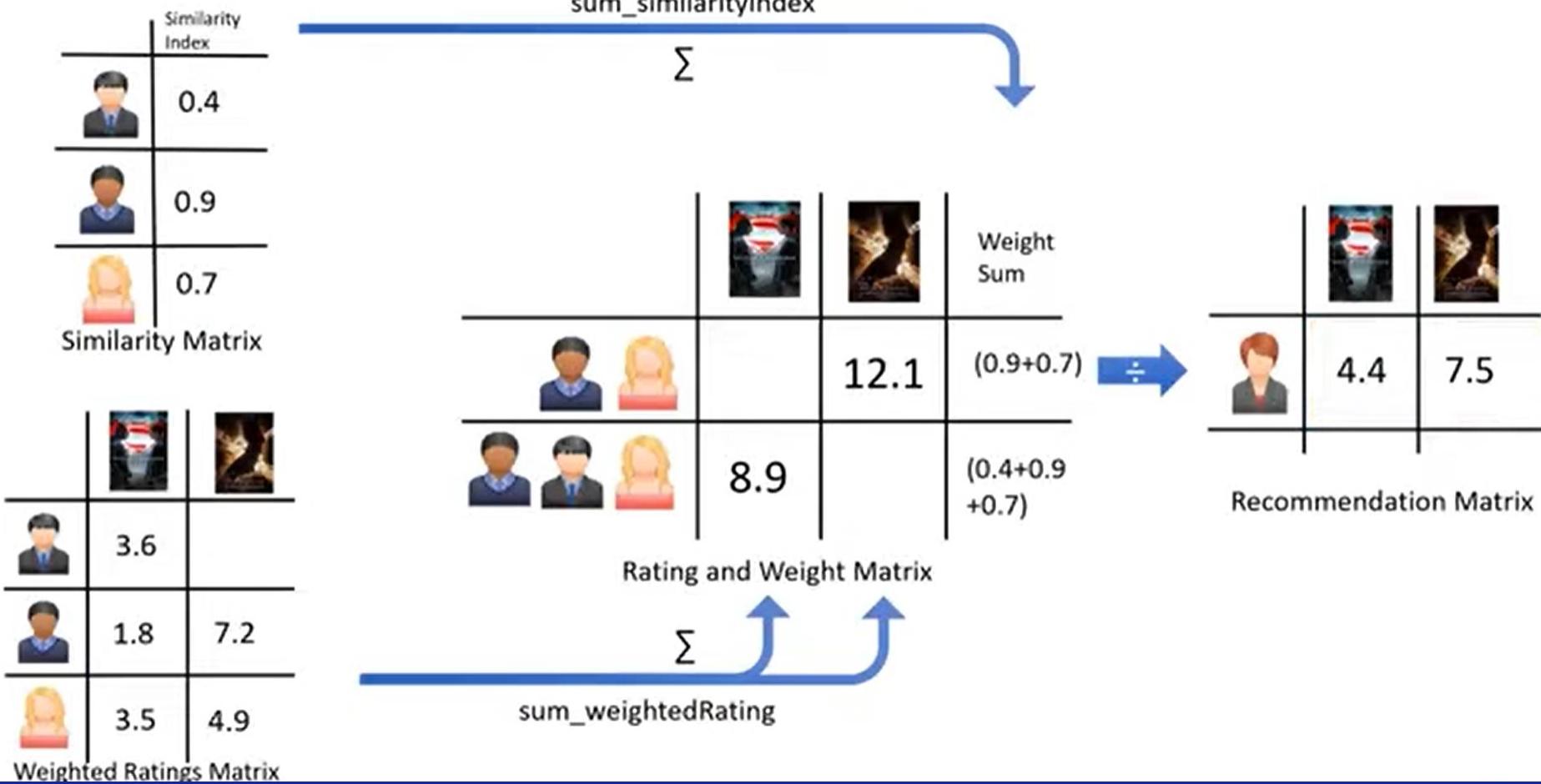


			Weight Sum
	12.1	(0.9+0.7)	
	8.9	(0.4+0.9+0.7)	
Rating and Weight Matrix			

Σ



sum_weightedRating



Similar Users (1)

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- Consider users x and y with rating vectors r_x and r_y
- We need a similarity metric $\text{sim}(x, y)$
- Capture intuition that $\text{sim}(A, B) > \text{sim}(A, C)$

Option 1: Jaccard Similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

Option 1: Jaccard Similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

$$\text{sim}(A, B) = |r_A \cap r_B| / |r_A \cup r_B|$$

Option 1: Jaccard Similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- $\text{sim}(A,B) = |r_A \cap r_B| / |r_A \cup r_B|$
- $\text{sim}(A,B) = 1/5; \text{sim}(A,C) = 2/4$
 - $\text{sim}(A,B) < \text{sim}(A,C)$

Option 1: Jaccard Similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- $\text{sim}(A,B) = |r_A \cap r_B| / |r_A \cup r_B|$
- $\text{sim}(A,B) = 1/5; \text{sim}(A,C) = 2/4$
 - $\text{sim}(A,B) < \text{sim}(A,C)$
- Problem: Ignores rating values!

Option 2: Cosine similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

Option 2: Cosine similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- $\text{sim}(A,B) = \cos(r_A, r_B)$
- $\text{sim}(A,B) = 0.38, \text{sim}(A,C) = 0.32$
 - $\text{sim}(A,B) < \text{sim}(A,C)$, but not by much

$$\text{Cos}(x, y) = x \cdot y / ||x|| * ||y||$$

Option 2: Cosine similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4	0	0	5	1	0	0
B	5	5	4	0	0	0	0
C				2	4	5	
D		3					3

- $\text{sim}(A,B) = \cos(r_A, r_B)$
- $\text{sim}(A,B) = 0.38$, $\text{sim}(A,C) = 0.32$
 - $\text{sim}(A,B) > \text{sim}(A,C)$, but not by much
- Problem: treats missing ratings as negative

Option 3: Centered cosine

- Normalize ratings by subtracting row mean

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

Option 3: Centered cosine

- Normalize ratings by subtracting row mean

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	2/3			5/3	-7/3		
B	1/3	1/3	-2/3				
C				-5/3	1/3	4/3	
D		0					0

Centered Cosine similarity (2)

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	2/3			5/3	-7/3		
B	1/3	1/3	-2/3				
C				-5/3	1/3	4/3	
D		0					0

Centered Cosine similarity (2)

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	2/3			5/3	-7/3		
B	1/3	1/3	-2/3				
C				-5/3	1/3	4/3	
D		0					0

- $\text{sim}(A,B) = \cos(r_A, r_B) = 0.09$; $\text{sim}(A,C) = -0.56$
 - $\text{sim}(A,B) > \text{sim}(A,C)$

Centered Cosine similarity (2)

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	2/3			5/3	-7/3		
B	1/3	1/3	-2/3				
C				-5/3	1/3	4/3	
D		0					0

- $\text{sim}(A,B) = \cos(r_A, r_B) = 0.09$; $\text{sim}(A,C) = -0.56$
 - $\text{sim}(A,B) > \text{sim}(A,C)$
- Captures intuition better
 - Missing ratings treated as “average”
 - Handles “tough raters” and “easy raters”

Centered Cosine similarity (2)

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	2/3			5/3	-7/3		
B	1/3	1/3	-2/3				
C				-5/3	1/3	4/3	
D		0					0

- $\text{sim}(A,B) = \cos(r_A, r_B) = 0.09$; $\text{sim}(A,C) = -0.56$
 - $\text{sim}(A,B) > \text{sim}(A,C)$
- Captures intuition better
 - Missing ratings treated as “average”
 - Handles “tough raters” and “easy raters”
- Also known as Pearson Correlation

Rating Predictions

- Let r_x be the vector of user x 's ratings
- Let N be the set of k users most similar to x who have also rated item i
- Prediction for user x and item i
- Option 1: $r_{xi} = 1/k \sum_{y \in N} r_{yi}$

Rating Predictions

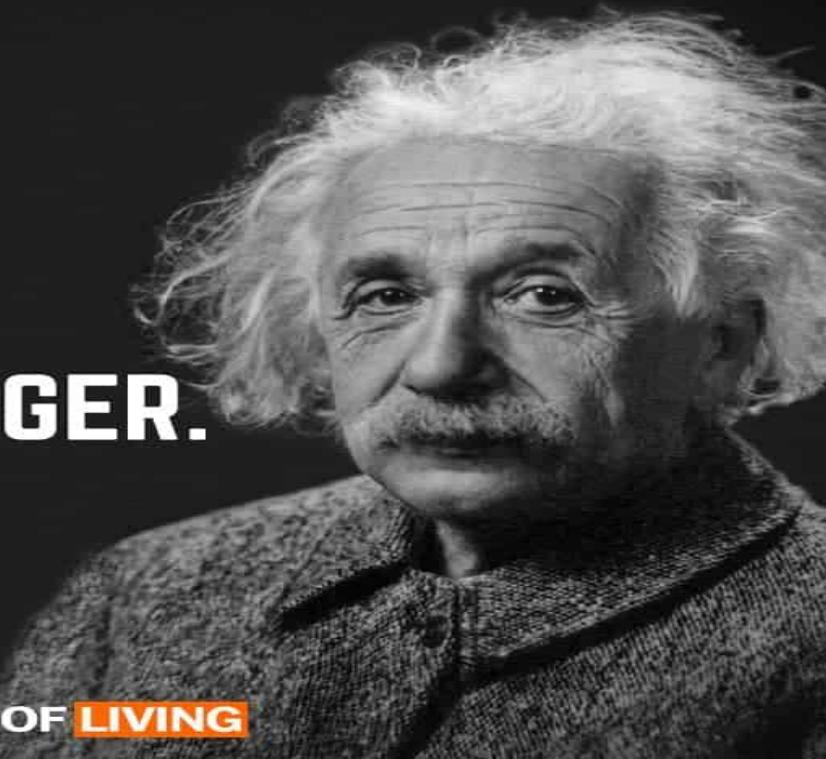
- Let r_x be the vector of user x 's ratings
 - Let N be the set of k users most similar to x who have also rated item i
 - Prediction for user x and item i
-
- Option 1: $r_{xi} = 1/k \sum_{y \in N} r_{yi}$
 - Option 2: $r_{xi} = \sum_{y \in N} s_{xy} r_{yi} / \sum_{y \in N} s_{xy}$

Rating Predictions

- Let r_x be the vector of user x 's ratings
- Let N be the set of k users most similar to x who have also rated item i
- Prediction for user x and item i
- Option 1: $r_{xi} = 1/k \sum_{y \in N} r_{yi}$
- Option 2: $r_{xi} = \sum_{y \in N} s_{xy} r_{yi} / \sum_{y \in N} s_{xy}$
where $s_{xy} = \text{sim}(x, y)$

**IT'S NOT THAT
I'M SO SMART.
IT'S JUST THAT
I STAY WITH
PROBLEMS LONGER.**

- ALBERT EINSTEIN



THE ART OF LIVING

→ Topics to be Discussed

5.1.1 A model for Recommendation systems

5.1.2 Content Based Recommendations

5.1.3 Collaborative Filtering

5.2.1 Case Study :Product Recommendation

5.3.1 Social Networks as Graphs,

5.3.2 Clustering of Social-Network Graphs

5.3.3 Direct Discovery of Communities in a social graph.

User based collaboration filtering

Name	Inshorts(I1)	HT(I2)	NYT(I3)	TOI(I4)	BBC(I5)
Alice	5	4	1	4	?
U1	3	1	2	3	3
U2	4	3	4	3	5
U3	3	3	1	5	4

User based collaboration filtering

Step 1: Calculating the similarity between Alice and all the other users

Name	Inshorts(I1)	HT(I2)	NYT(I3)	TOI(I4)	BBC(I5)	Avg
Alice	5	4	1	4	?	3.5
U1	3	1	2	3	3	2.25
U2	4	3	4	3	5	3.5
U3	3	3	1	5	4	3

User based collaboration filtering

$$r'_{ip} = r_{ip} - \bar{r}_i$$

Name	Inshorts(I1)	HT(I2)	NYT(I3)	TOI(I4)	BBC(I5)	Avg
Alice	5	4	1	4	?	3.5
U1	3	1	2	3	3	2.25
U2	4	3	4	3	5	3.5
U3	3	3	1	5	4	3

Name	Inshorts(I1)	HT(I2)	NYT(I3)	TOI(I4)
Alice	1.5	0.5	-2.5	0.5
U1	0.75	-1.25	-0.25	0.75
U2	0.5	-0.5	0.5	-0.5
U3	0	0	-2	2

User based collaboration filtering

Now, we calculate the similarity between Alice and all the other users.

$$Sim(Alice, U1) = \frac{((1.5*0.75)+(0.5*-1.25)+(-2.5*-0.25)+(.5*0.75))}{\sqrt{(1.5^2+0.5^2+2.5^2+0.5^2)}\sqrt{(0.75^2+1.25^2+0.25^2+0.75^2)}} = 0.301$$

$$Sim(Alice, U2) = \frac{((1.5*0.25)+(0.5*-0.5)+(-2.5*0.5)+(.5*-0.5))}{\sqrt{(1.5^2+0.5^2+2.5^2+0.5^2)}\sqrt{(0.5^2+0.5^2+0.5^2+0.5^2)}} = -0.33$$

$$Sim(Alice, U3) = \frac{((1.5*0)+(0.5*0)+(-2.5*-2)+(.5*2))}{\sqrt{(1.5^2+0.5^2+2.5^2+0.5^2)}\sqrt{(0^2+0^2+2^2+2^2)}} = 0.707$$

User based collaboration filtering

Step 2: Predicting the rating of the app not rated by Alice

$$r_{up} = \bar{r}_u + \frac{\sum_{i \in users} sim(u,i)*r_{ip}}{\sum_{i \in users} |sim(u,i)|}$$

$$r_{(Alice,I5)} = 3.5 + \frac{(0.301*0.75)+(-0.33*1.5)+(0.707*1)}{|0.301|+|-0.33|+|0.707|} = 3.83$$

Item-Item Collaborative Filtering

- So far: User-user collaborative filtering
- Another view: Item-item
 - For item i , find other similar items
 - Estimate rating for item i based on ratings for similar items
 - Can use same similarity metrics and prediction functions as in user-user model

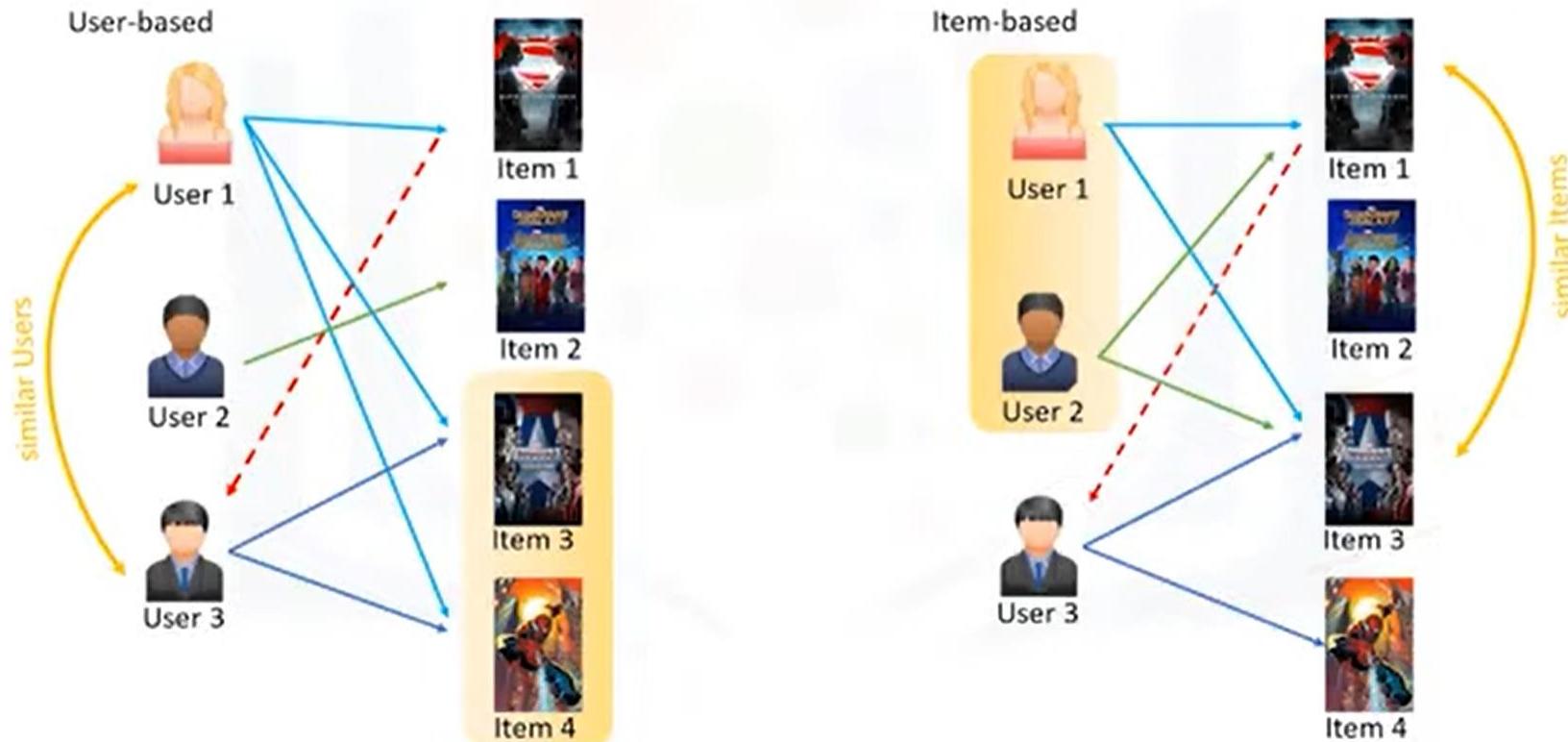
$$r_{xi} = \frac{\sum_{j \in N(i;x)} s_{ij} \cdot r_{xj}}{\sum_{j \in N(i;x)} s_{ij}}$$

s_{ij} ... similarity of items i and j

r_{xj} ... rating of user x on item j

$N(i;x)$... set items rated by x similar to i

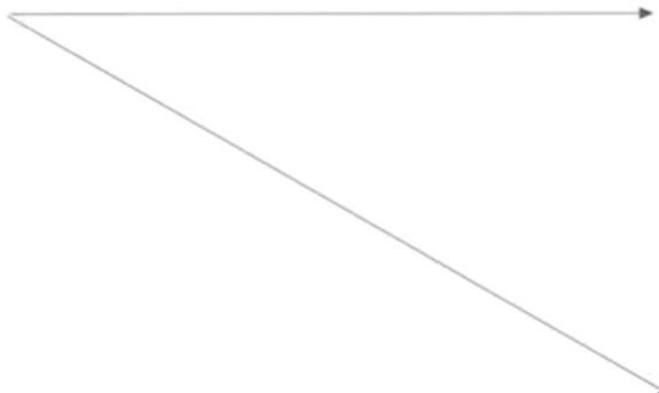
Collaborative filtering



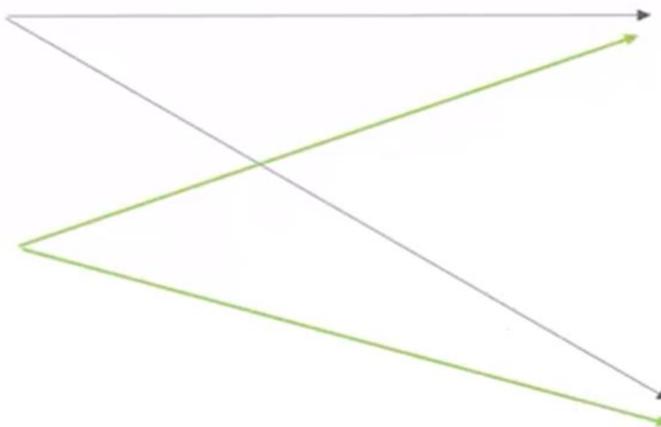
Item-Based Collaborative Filtering

- Find every pair of movies that were watched by the same person
- Measure the similarity of their ratings across all users who watched both
- Sort by movie, then by similarity strength
- (This is just one way to do it!)

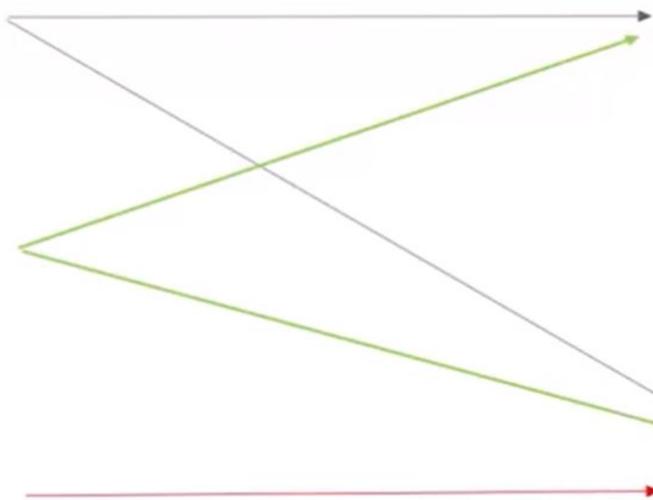
Item-Based Collaborative Filtering



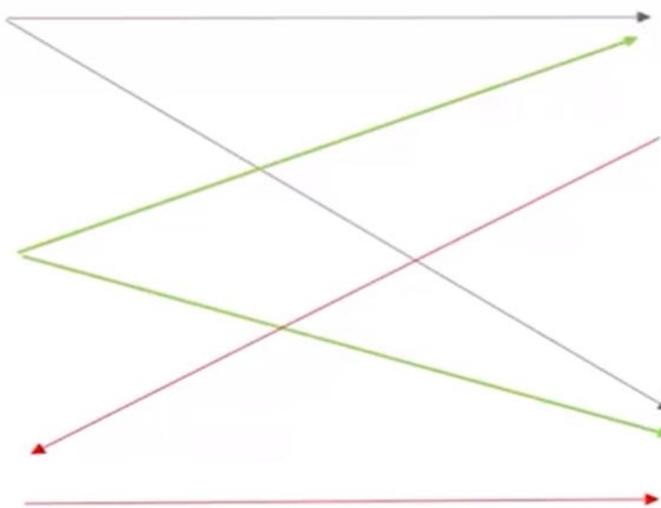
Item-Based Collaborative Filtering



Item-Based Collaborative Filtering



Item-Based Collaborative Filtering



Item based collaboration filtering

User/Item	Item_1	Item_2	Item_3
User_1	2	–	3
User_2	5	2	–
User_3	3	3	1
User_4	–	2	2



Item based collaboration filtering

User/Item	Item_1	Item_2	Item_3
User_1	2	-	3
User_2	5	2	-
User_3	3	3	1
User_4	-	2	2

Step 1: Finding similarities of all the item pairs.

Form the item pairs. For example in this example the item pairs are (Item_1, Item_2), (Item_1, Item_3), and (Item_2, Item_3). Select each item to pair one by one. After this, we find all the users who have rated for both the items in the item pair. Form a vector for each item and calculate the similarity between the two items using the cosine formula stated above.

Item based collaboration filtering

User/Item	Item_1	Item_2	Item_3
User_1	2	-	3
User_2	5	2	-
User_3	3	3	1
User_4	-	2	2

$$\text{Similarity}(I1, I2) = \frac{(5*2)+(3*3)}{\sqrt{5^2+3^2}\sqrt{2^2+3^2}} = 0.90$$

$$\text{Similarity}(I2, I3) = \frac{(3*1)+(2*2)}{\sqrt{3^2+2^2}\sqrt{1^2+2^2}} = 0.869$$

$$\text{Similarity}(I1, I3) = \frac{(2*3)+(3*1)}{\sqrt{2^2+3^2}\sqrt{3^2+1^2}} = 0.789$$

Step 2: Generating the missing ratings in the table

Now, in this step we calculate the ratings that are missing in the table.

User/Item	Item_1	Item_2	Item_3
User_1	2	-	3
User_2	5	2	-
User_3	3	3	1
User_4	-	2	2

Rating of Item_2 for User_1

$$r(U_1, I_2) = \frac{r(U_1, I_1)*s_{I_1 I_2} + r(U_1, I_3)*s_{I_3 I_2}}{s_{I_1 I_2} + s_{I_3 I_2}} = \frac{(2*0.9) + (3*0.869)}{(0.9 + 0.869)} = 2.49$$

$$\text{Similarity}(I_1, I_2) = 0.90$$

$$\text{Similarity}(I_2, I_3) = 0.869$$

$$\text{Similarity}(I_1, I_3) = 0.789$$

Step 2: Generating the missing ratings in the table

Now, in this step we calculate the ratings that are missing in the table.

User/Item	Item_1	Item_2	Item_3
User_1	2	-	3
User_2	5	2	-
User_3	3	3	1
User_4	-	2	2

Rating of Item_3 for User_2

$$r(U_2, I_3) = \frac{r(U_2, I_1)*s_{I_1 I_3} + r(U_2, I_2)*s_{I_2 I_3}}{s_{I_1 I_3} + s_{I_2 I_3}} = \frac{(5*0.789) + (2*0.869)}{(0.789 + 0.869)} = 3.43$$

$$\text{Similarity}(I_1, I_2) = 0.90$$

$$\text{Similarity}(I_2, I_3) = 0.869$$

$$\text{Similarity}(I_1, I_3) = 0.789$$

Step 2: Generating the missing ratings in the table

Now, in this step we calculate the ratings that are missing in the table.

User/Item	Item_1	Item_2	Item_3
User_1	2	-	3
User_2	5	2	-
User_3	3	3	1
User_4	-	2	2

Rating of Item_1 for User_4

$$r(U_4, I_1) = \frac{r(U_4, I_2)*s_{I_1 I_2} + r(U_4, I_3)*s_{I_1 I_3}}{s_{I_1 I_2} + s_{I_1 I_3}} = \frac{(2*0.9) + (2*0.789)}{(0.9 + 0.789)} = 2.0 -$$

$$\text{Similarity}(I_1, I_2) = 0.90$$

$$\text{Similarity}(I_2, I_3) = 0.869$$

$$\text{Similarity}(I_1, I_3) = 0.789$$

Item-Item CF ($|N|=2$)

	users											
	1	2	3	4	5	6	7	8	9	10	11	12
1	1		3			5			5		4	
2			5	4			4			2	1	3
3	2	4		1	2		3		4	3	5	
4		2	4		5			4			2	
5			4	3	4	2					2	5
6	1		3		3			2			4	

 - unknown rating  - rating between 1 to 5

Item-Item CF ($|N|=2$)

	users												
	1	2	3	4	5	6	7	8	9	10	11	12	$\text{sim}(1,m)$
movies	1	1		3		?	5			5		4	1.00
	2			5	4			4			2	1	3
	3	2	4		1	2		3		4	3	5	-0.18
	4		2	4		5			4			2	0.41
	5			4	3	4	2					2	-0.10
	6	1		3		3			2			4	-0.31

Here we use Pearson correlation as similarity:

- 1) Subtract mean rating m_i from each movie i

$$m_1 = (1+3+5+5+4)/5 = 3.6$$

row 1: [-2.6, 0, -0.6, 0, 0, 1.4, 0, 0, 1.4, 0, 0.4, 0]

- 2) Compute cosine similarities between rows

Item-Item CF ($|N|=2$)

	users											
	1	2	3	4	5	6	7	8	9	10	11	12
1	1		3		?	5			5		4	
2			5	4			4			2	1	3
3	2	4		1	2		3		4	3	5	
4		2	4		5			4			2	
5			4	3	4	2					2	5
6	1		3		3			2			4	

sim(1,m)

1.00

-0.18

0.41

-0.10

-0.31

0.59

Compute similarity weights:

$$s_{13}=0.41, s_{16}=0.59$$

Item-Item CF ($|N|=2$)

	users												
	1	2	3	4	5	6	7	8	9	10	11	12	
1	1		3		?	5			5		4		sim(1,m)
2			5	4			4			2	1	3	1.00
3	2	4		1	2		3		4	3	5		-0.18
4		2	4		5			4			2		0.41
5			4	3	4	2					2	5	-0.10
6	1		3		3			2			4		-0.31
													0.59

Predict by taking weighted average:

$$r_{15} = \underbrace{(0.41 * 2)}_{*} + \underbrace{(0.59 * 3)}_{*} / (0.41 + 0.59) = 2.6$$

Item-Item v. User-User

- In theory, user-user and item-item are dual approaches
- In practice, item-item outperforms user-user in many use cases
- Items are “simpler” than users
 - Items belong to a small set of “genres”, users have varied tastes
 - Item Similarity is more meaningful than User Similarity

Collaborative Filtering: Complexity

- Expensive step is finding k most similar users (or items): $O(|U|)$
 - $|U|$ = size of utility matrix
- Too expensive to do at runtime
 - Could pre-compute
 - Naïve pre-computation takes time $O(n \cdot |U|)$
 - Where n = number of users (items)
- **We already know how to do this!**
 - Near-neighbor search in high dimensions (**LSH**)
 - Clustering
 - Dimensionality reduction (coming soon!)

Pros/Cons of Collaborative Filtering

- + **Works for any kind of item**
 - No feature selection needed
- - **Cold Start:**
 - Need enough users in the system to find a match
- - **Sparsity:**
 - The user/ratings matrix is sparse
 - Hard to find users that have rated the same items
- - **First rater:**
 - Cannot recommend an unrated item
 - New items, Esoteric items
- - **Popularity bias:**
 - Tends to recommend popular items

Hybrid Methods

- **Add content-based methods to collaborative filtering**
 - Item profiles for new item problem
 - Demographics to deal with new user problem
- **Implement two or more different recommenders and combine predictions**
 - Perhaps using a linear model
 - Example: global baseline + collaborative filtering

“A LITTLE PROGRESS EACH
DAY ADDS UP TO BIG
RESULTS.”

SATYA NANI



work
^{HARD}
dream
^{BIG}
Never
^{GIVE UP}

→ Topics to be Discussed

- ✓ 5.1.1 A model for Recommendation systems
- ✓ 5.1.2 Content Based Recommendations
- ✓ 5.1.3 Collaborative Filtering

5.2.1 Case Study :Product Recommendation

- 5.3.1 Social Networks as Graphs,
- 5.3.2 Clustering of Social-Network Graphs
- 5.3.3 Direct Discovery of Communities in a social graph.

Case Study :Product Recommendation

- An increasing number of online companies are utilizing recommendation systems to **increase user interaction and enrich shopping potential.**
- Use cases of recommendation systems have been expanding rapidly across many aspects of **eCommerce and online media** over the last 4-5 years, and we expect this trend to continue.
- Recommendation systems (often called “recommendation engines”) have the **potential to change the way websites communicate with users** and to allow companies to **maximize their ROI based on the information they can gather on each customer’s preferences and purchases.**

Case Study :Product Recommendation

Potential Benefits of Recommendation Engines

1.“Improving with use” (retention): One of the core potential benefits of recommendation systems is their ability to continuously **calibrate to the preferences of the user**.

- This makes products that become more and more “sticky” in their **customer retention** as time goes on:
- You’re much less likely to switch to a Netflix competitor when Netflix has such a **wonderful sense of which movies and shows you might want to watch next** (i.e. they “know you so well”).
- Because most of Netflix’s revenues come from a fixed-rate recurring billing model subscription, the company’s biggest ROI “win” with **recommendation systems is retention**.

Case Study :Product Recommendation

2. Improving cart value:

- Items would quickly **be out-of-date or irrelevant for many customers.**
- By using various **means of “filtering”**, eCommerce giants can find opportune times to **suggest** (on their site, via email, or through other means) **new products that you’re likely to buy.**
- Amazon’s **quick delivery and emphasis on customer service** has earned them millions of customers.
- Recommendation engines play a **role not only in helping customers find more of what they need** (and see Amazon as an authority), **but these systems also improve cart value.**
- If Amazon doesn’t have to pay much more **for shipping** to send you two or three times as many products, their profit margins improve.

Case Study :Product Recommendation

3.Improved engagement and delight:

- Sometimes seeing an ROI doesn't involve explicitly asking for payment.
- Many companies use these systems to simply **encourage engagement and activity on their product or platform.**
- **YouTube has subscription options**, but the majority of the **firm's revenues are driven through advertisements** placed across its wide array of video properties.
- The company makes more money when users **come back time and time again.**
- YouTube doesn't optimize for short-term view length, as this might encourage pushy or flashy tactics that wouldn't genuinely delight users.

Case Study :Product Recommendation

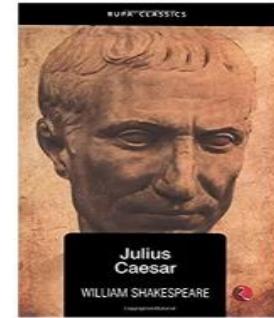
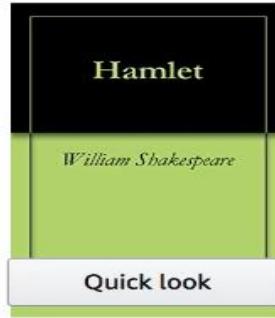
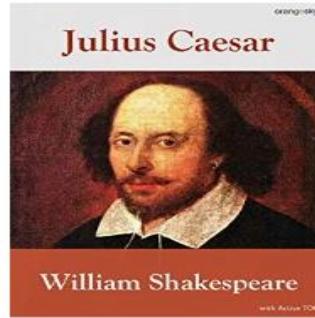
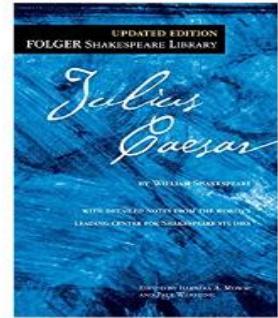
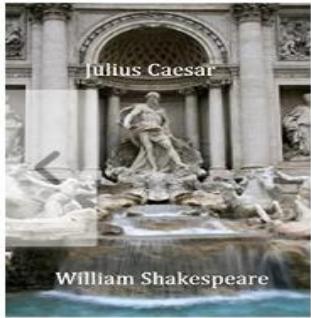
3.Improved engagement and delight:

- Instead, the service aims to encourage long-term use, because advertising views is the ROI that these systems serve at YouTube.
- **Facebook is another obvious example** of a similar application of recommendation engines

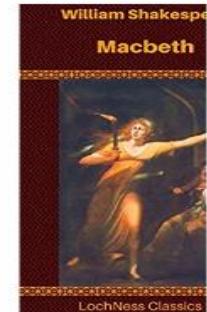
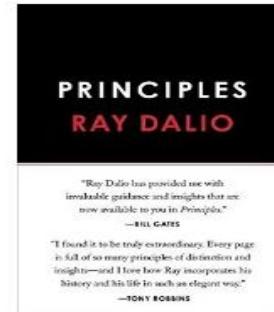
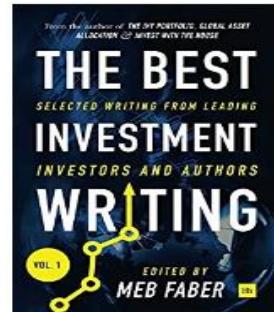
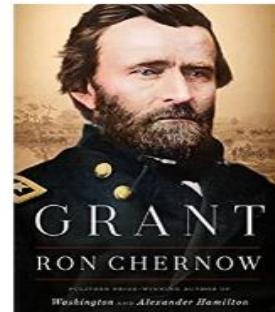
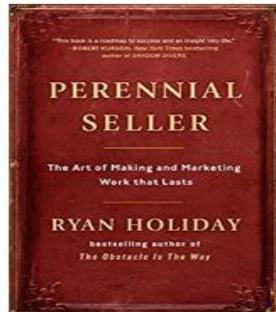
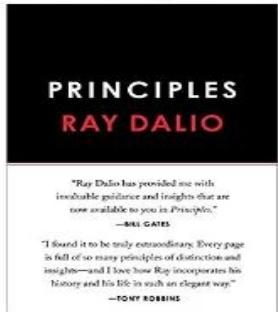
Real World Applications Today

1—Amazon

Inspired by your browsing history [See more](#)



New for you [See more](#)



1 – Amazon

Real World Applications Today

- Amazon has single-handedly put a spotlight on the retail value of AI, and recommendations are part of what put the company on the map. Amazon.com uses recommendations as a targeted marketing tool throughout its website.
- When a customer clicks on the “your recommendations” the link leads to another page where recommendations may be filtered even further by subject area, product types, and ratings of previous products and purchases.
- The customer **can even see why a particular product has been recommended**.
- “At Amazon.com, we use **recommendation algorithms to personalize the online store for each customer**. The store radically changes based on customer interests, showing programming titles to a software engineer and baby toys to a new mother,” explain Greg Linden, Brent Smith, and Jeremy York in their paper Amazon.com Recommendations: Item-to-Item Collaborative Filtering.
- In this instance, collaborative filtering doesn’t just match each user to similar customers. The item-to-item connects each user’s purchase to similar items and compiles a recommendation list from them. For example, if you’re enthusiastic about the latest technology, you may find your Amazon web page suggests the latest device and gadgets, if cooking is your thing, you’re sure to find plenty of recommendations for recipe books and cookware.
- According to McKinsey & Company, 35% of Amazon.com’s revenue is generated by its recommendation engine.

2 – Netflix

Real World Applications Today

- According to a paper written by Netflix executives Carlos A. Gomez-Uribe and Neil Hunt, the video streaming service's AI recommendation system saves the company around \$1 billion each year.
- This allows them to **invest more money on new content which viewers will continue to view**, giving them a good ROI.
- "We have discovered through the years that there is tremendous value to our subscribers in incorporating recommendations to personalize as much of Netflix as possible," say by Xavier Amatriain and Justin Basilico (Personalization Science and Engineering) in their Netflix Tech Blog.
- Netflix uses **RS personalized diversity to generate Top Ten recommendations** for user households, so that it can offer videos that each member of the household may be interested in. The company also focuses on awareness and promoting trust to help develop its personalized approach.
- Netflix implements these strategies by explaining why it makes video recommendation and encouraging members to give feedback, so no opportunities to personalize are missed.
- According to McKinsey, **75 percent of what users watch on Netflix come from product recommendations**.

3 – Spotify

Case Study :Product Recommendation

- Possibly one of Spotify's most innovative **uses of AI and recommendation systems** is their popular Discover Weekly playlist. Known as Release Radar, this algorithmically powered tool updates **personal playlists on a weekly basis** so that users won't miss newly released music by artists they like.
- "With the huge amount of new music released every week, it can be difficult to keep up with the latest tracks," Spotify's Matt Ogle, who leads the development of Discover Weekly, said in a statement. "With Release Radar, we wanted to create the simplest way for you to find all the newly released music that matters the most to you, in one playlist."
- Discover Weekly works by looking at the **2 billion plus playlist** created by users, each based on music fans' individual tastes. Spotify then collates this information with the company's own playlists and fills in the blanks by comparing a user's listening habits to those of users with similar tastes. The approach also uses **collaborative filtering in combination with deep learning to detect patterns within huge amount of data to improve weekly selections**.
- The new recommendation system has helped Spotify increase its number of monthly users from 75 million to 100 million at a time, in spite of competition from rival streaming service Apple Music.

5 – YouTube

A model for Recommendation systems

- Fortunately for us, YouTube has a series of “Help” videos that answer basic user questions about YouTube and the technology that supports it.
- The YouTube online video community uses RS to create personalized recommendations so users can quickly and easily find videos that are relevant to their interests.
- Because of the value of keeping users engaged, YouTube strives to keep the recommendations updated on a regular basis, to reflect each user’s activity on the site and to simultaneously highlight the wide range of available content.
- The RS is driven by the Google Brain deep learning artificial intelligence project and is comprised of two neural networks. The first collects and collates information on users’ watch history and uses collaborative filtering to select hundreds of videos. This process, known as candidate generation, uses feedback from users to train the model. The second neural network ranks the selected videos in order to make recommendations to users.

According to YouTube after implementation of the RS for more than a year, it has been successful in terms of their stated goals, with recommendations accounting for around 60 percent of video clicks from the homepage.

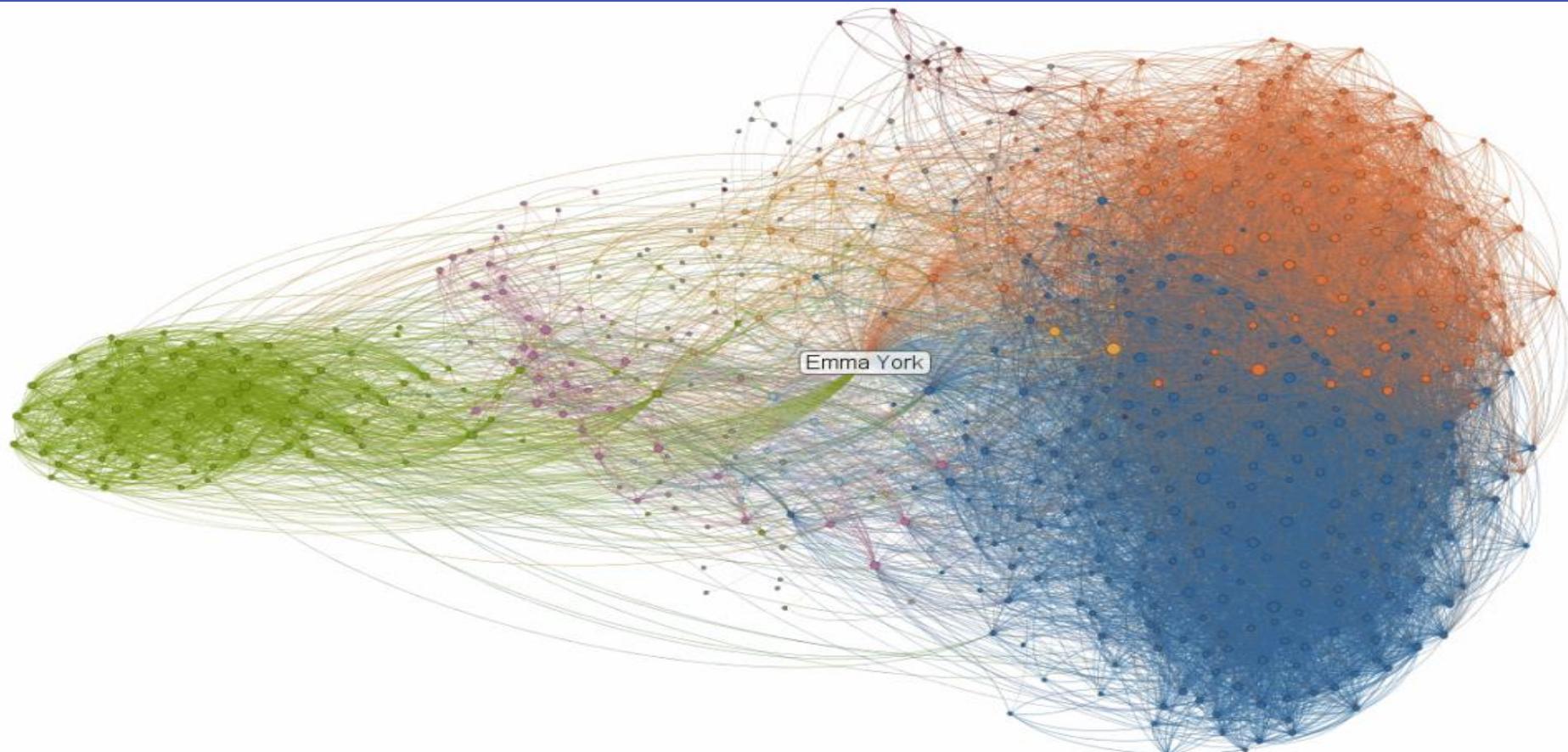
→ Topics to be Discussed

- ✓ 5.1.1 A model for Recommendation systems
- ✓ 5.1.2 Content Based Recommendations
- ✓ 5.1.3 Collaborative Filtering

- ✓ 5.2.1 Case Study :Product Recommendation

- 5.3.1 Social Networks as Graphs,
- 5.3.2 Clustering of Social-Network Graphs
- 5.3.3 Direct Discovery of Communities in a social graph.

Social Graph



Social networks

- What is a social network?
 - A collection of entities (nodes of the graph)
 - Typically people, but can be other entities
 - At least one relationship between the entities of the network
 - Typically represented by edges between the nodes
 - For example: friends
 - Sometimes Boolean: two people may be friends, or not
 - May have a degree
 - Discrete degree: friends, family, acquaintances, or none
 - A real number: the fraction of the average day two people spend talking to each other



Disclaimer: The brand logos are used for academic purpose only

What is a graph?

$G = (V, E)$

- V represents the set of vertices (nodes)
- E represents the set of edges (links)
- Both vertices and edges may contain additional information
- Different types of graphs:
 - Directed vs. undirected edges
 - Presence or absence of cycles
- Graphs are everywhere:
 - Hyperlink structure of the Web
 - Highway system
 - Social networks

Some graph problems

- **Finding shortest paths**
 - Routing Internet traffic and UPS trucks
- **Finding minimum spanning trees**
 - Telco laying down fiber
- **Finding Max Flow**
 - Airline scheduling
- **Identify “special” nodes and communities**
 - Breaking up terrorist cells, spread of avian flu
- **Bipartite matching**
 - Tinder
- **PageRank**



Representing graphs

Two common representations:

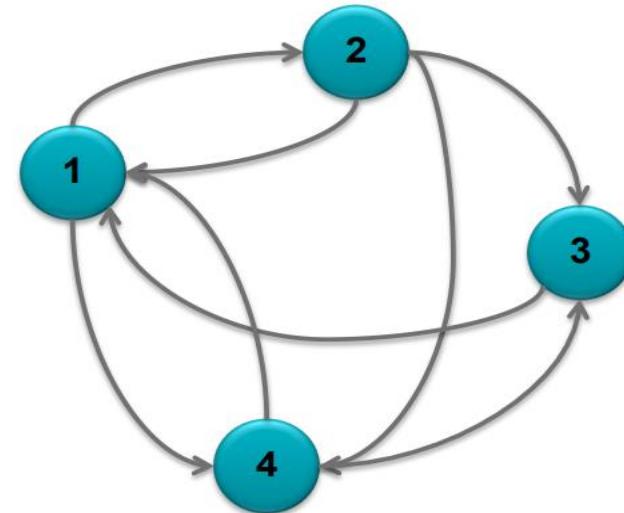
- Adjacency matrix
- Adjacency list

Adjacency matrices

Represent a graph as an $n \times n$ square matrix M

- $n = |\mathcal{V}|$
- $M_{ij} = 1$ means a link from node i to j

	1	2	3	4
1	0	1	0	1
2	1	0	1	1
3	1	0	0	0
4	1	0	1	0



Adjacency list

Take adjacency matrices... and throw away all the zeros

	1	2	3	4
1	0	1	0	1
2	1	0	1	1
3	1	0	0	0
4	1	0	1	0



- 1: 2, 4
- 2: 1, 3, 4
- 3: 1
- 4: 1, 3

Social Networks as Graphs

What is a Social Network?

When we think of a social network, we think of Facebook, Twitter, Google+, or another website that is called a “social network,” and indeed this kind of network is representative of the broader class of networks called “social.”

The essential characteristics of a social network are:

- a) There is a **collection of entities that participate in the network**. Typically, these entities are people, but they could be something else entirely.

Social Networks as Graphs

What is a Social Network?

- a) There is **at least one relationship between entities of the network**. On Facebook or its ilk, **this relationship is called friends**. Sometimes the relationship is all-or-nothing; two people are either friends or they are not. However, in other examples of social networks, the **relationship has a degree**. This degree could be discrete; e.g., friends, family, acquaintances, or none as in Google+. It could be a real number; an example would be the fraction of the average day that two people spend talking to each other.

Social Networks as Graphs

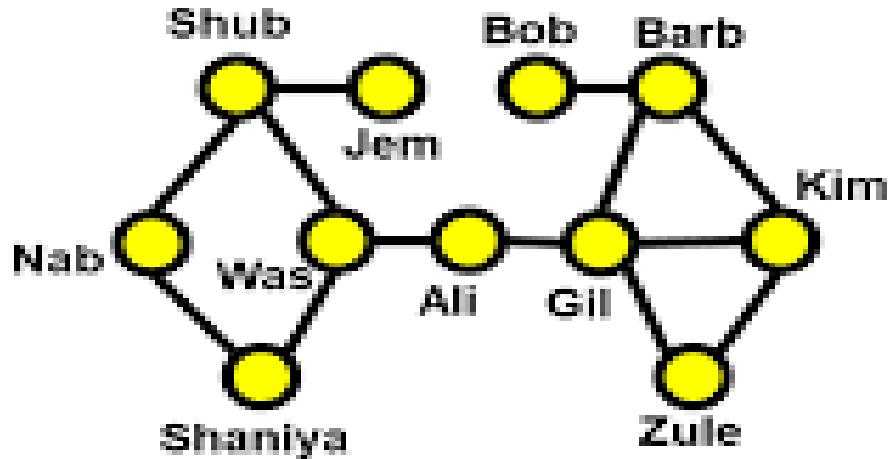
What is a Social Network?

- a) There is an assumption of non randomness or locality. This condition is the hardest to formalize, but the intuition is that **relationships tend to cluster**.

That is, if entity A is related to both B and C, then there is a higher probability than average that B and C are related

Social Networks as Graphs

Social Networks as Graphs



(a) Original social network data about users

Social Networks as Graphs

Social Networks as Graphs

- Example of a tiny social network.
- The entities are the nodes A through G.
- The relationship, which we might think of as “**friends**,” is represented by the edges.
- For instance, B is friends with A, C, and D.

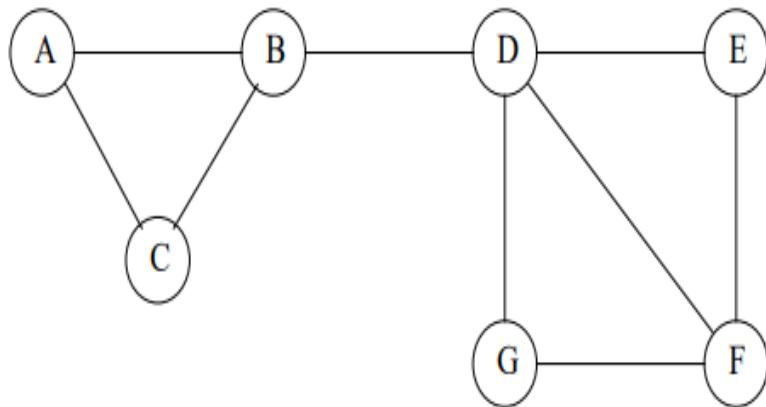
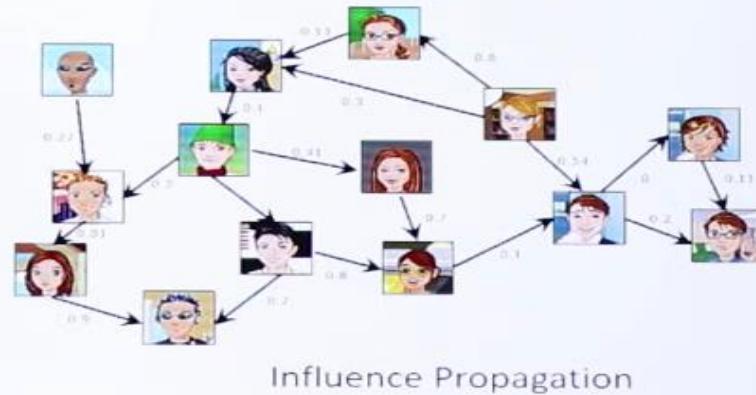
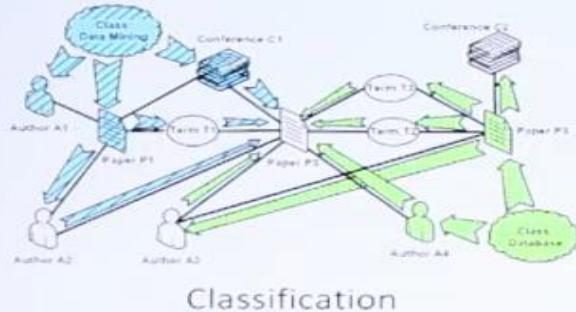


Figure 10.1: Example of a small social network

Social Networks analysis

Social Network Analytics



Social Networks Analysis

- Social Network Analysis (SNA) is the process of exploring or examining the social structure by using graph theory.
- It is used for measuring and analyzing the structural properties of the network.
- It helps to measure relationships and flows between groups, organizations, and other connected entities.

Social Networks Analysis

Basically, there are two types of social networks:

- Ego network Analysis
- Complete network Analysis

SNA usually refers to varied information and knowledge entities, but most actual studies focus on human (node) and relational (tie) analysis. The tie value is social capital.

Social Networks Analysis

- Studying the **complete social network**, including all ties in a defined population.
- Studying **egocentric components**, including all ties and personal communities, which involves studying relationship between the focal points in the network and the social ties they make in their communities.

Social Networks Analysis

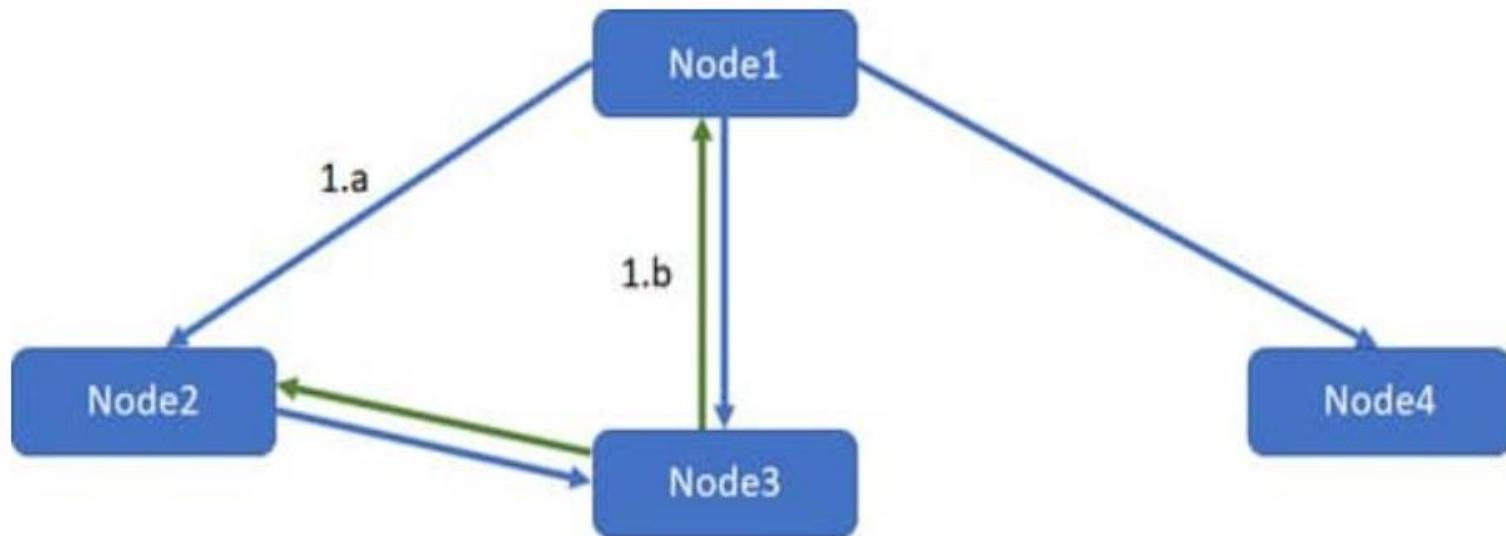


Figure 1

Social Networks Analysis

1.a Directed Edge:

- The nodes connected by this **edge are ordered**, that is, the connection between the nodes is **one way**.
- For example, Twitter, Instagram are predominantly directed edge networks.
- You can follow someone without them following you back.

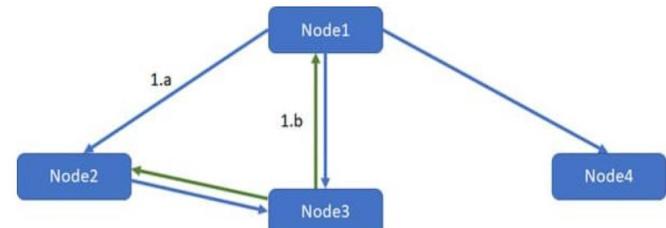
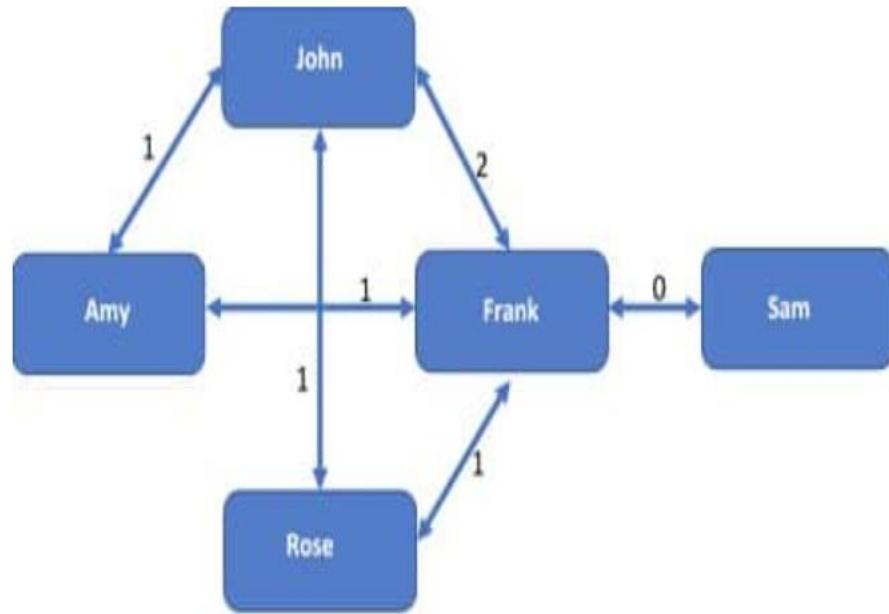


Figure 1

Social Networks Analysis

1.b Undirected Edge:

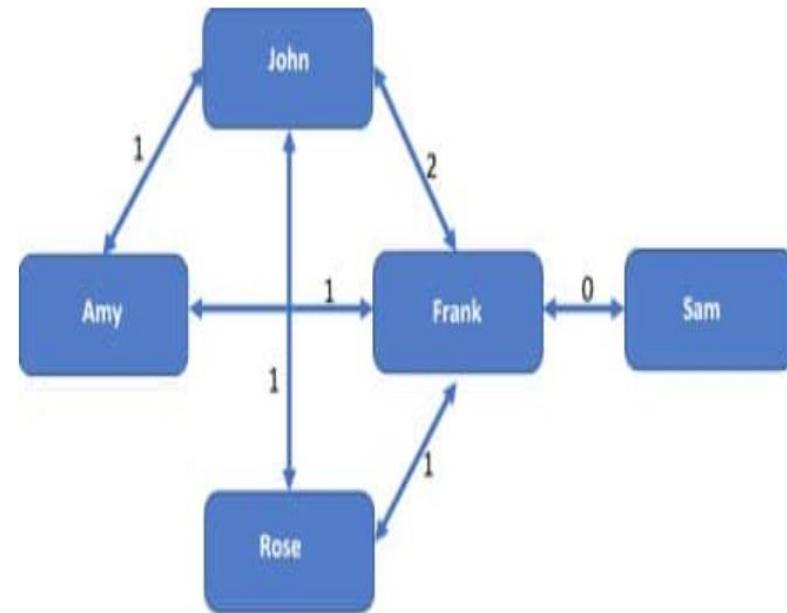
- The relationship between the nodes connected by this edge is mutual, i.e., the connection is applicable both ways.
- E.g., Befriending a person on Facebook, LinkedIn automatically creates a two-way connection.



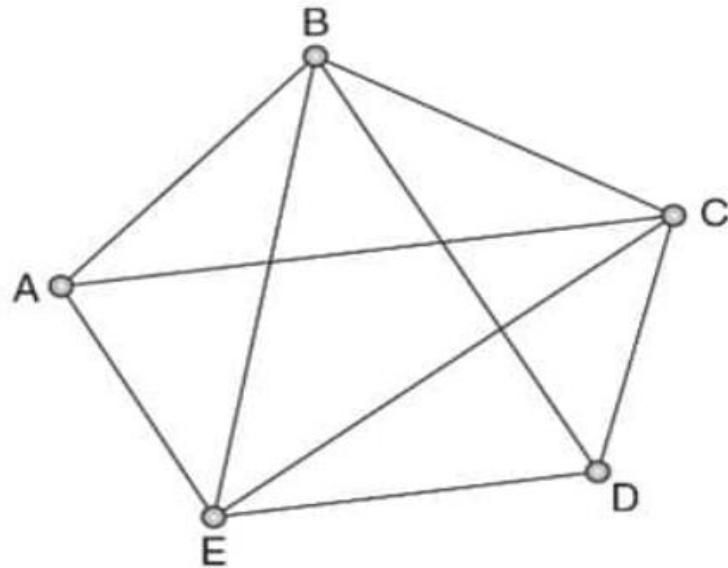
Social Networks Analysis

2. Weight:

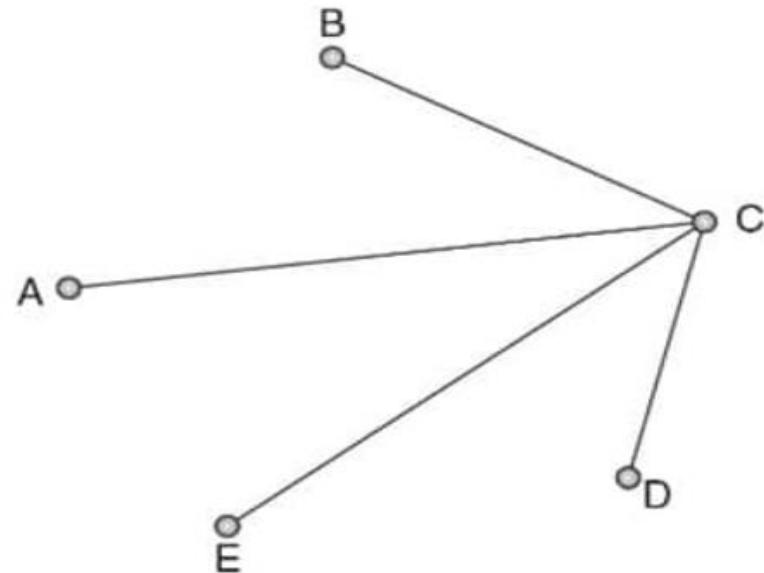
- In a weighted network, an edge carries a label (weight) between the nodes.
- Different applications can have their own definition of weight.
- In social media analysis, a weight can define the **number of mutual connections** between the nodes connected by that edge.



Social Networks Analysis



(a)
High-density five-point network



(b)
Low-density five-point network

Social Networks Analysis

3. **Density:** The relation between the number of existing connections in a network and all possible connections in the network is calculated as follows:

$$\text{Density} = \frac{\text{Actual Connections}}{\text{Potential Connections}}$$

where,

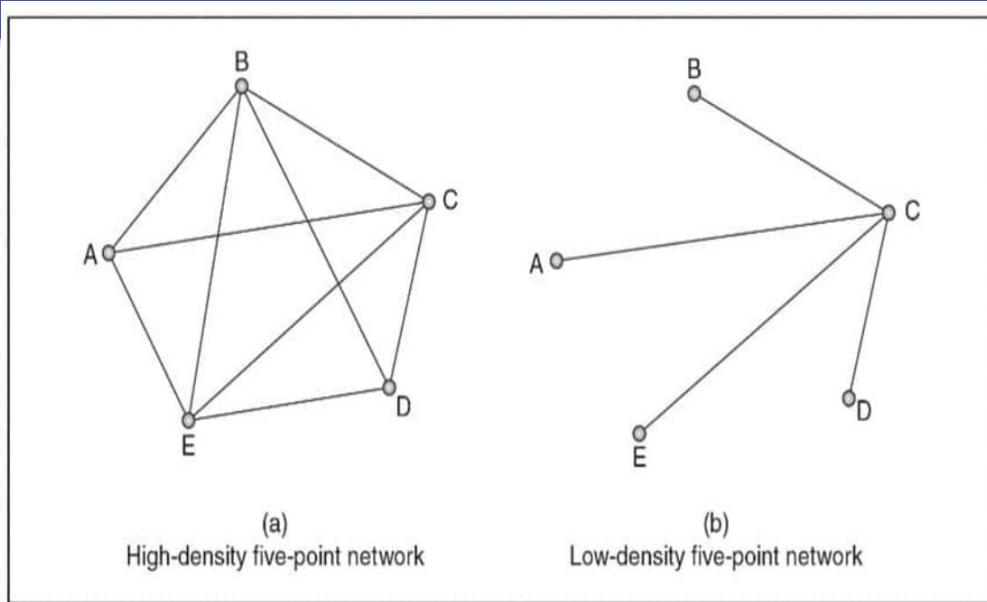
$$\text{Potential Connections} = \frac{n*(n-1)}{2}$$

and n = number of nodes in the network

Social Networks Analysis

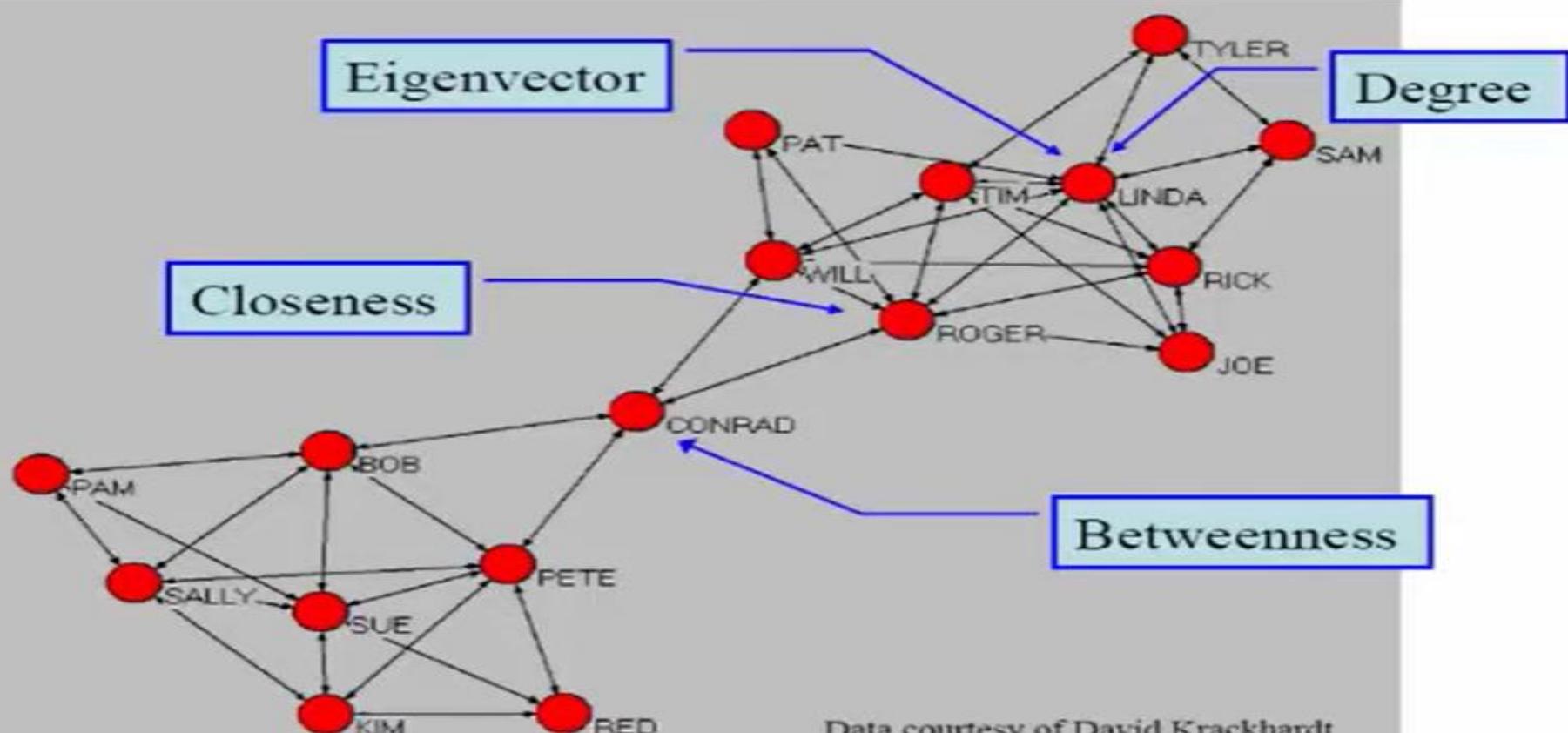
In Figure, we have a five-point/node network.

- The total possible connections in this network are 10.
- Figure **a** has nine edges; its density is 90%. Hence it is a high-density network.
- Whereas Figure **b** has only four edges, it has a low density of 40%.



Centrality Measures:

Social Networks Analysis



Data courtesy of David Krackhardt

Network Measures of Centrality

- **Degree**: the number of edges connected to a node.
- **Betweenness**: extent to which a particular node lies on the shortest path between other nodes.
- **Closeness**: the average of the shortest distances to all other nodes in the graph.
- **Eigenvector**: a measure of the extent to which a node is connected to influential other nodes. i.e. Google's PageRank uses this measure.

What do the Measures Tell Me?

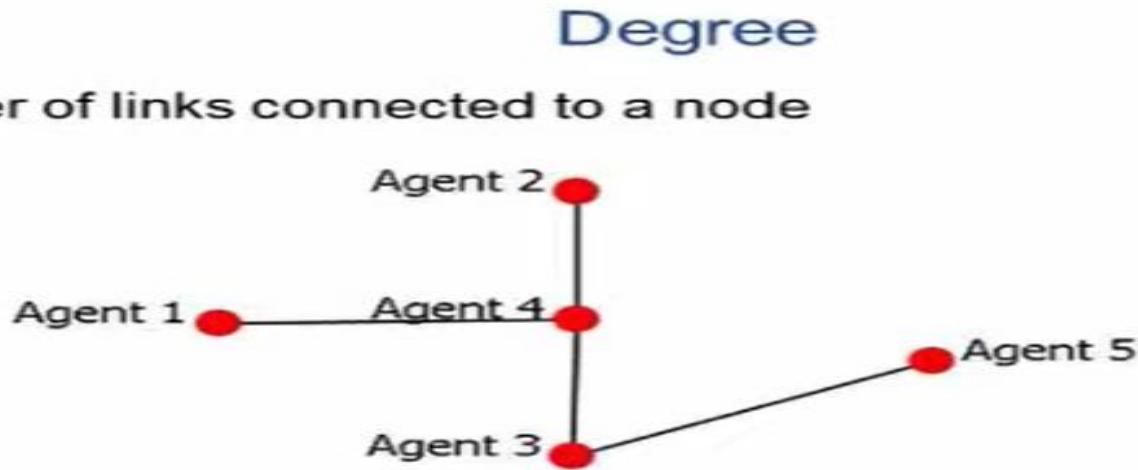
- **Degree**: exposure to the network, opportunity to directly influence.
- **Betweenness**: informal power; gate keeping; brokering; controls flow of info; liaison between sub-comp.
- **Closeness**: estimates time to hear info; indirect influence; point of rapid diffusion.
- **Eigenvector**: connected to influential nodes of high degree; “not what you know but who you know.”

Centrality Measures

Degree Walkthrough

Definition

Degree: the number of links connected to a node



What do the Measures Tell Me?

Degree: exposure to the network, opportunity to directly influence. Possible target for collection of information on the network.

Social Networks Analysis

Centrality Measures:

Calculating Degree

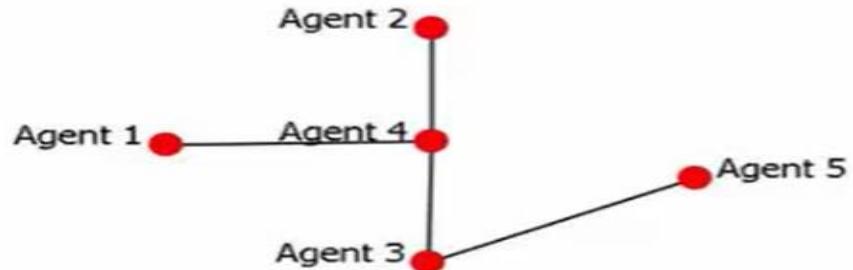


$$C_{Di} = \frac{\sum_{j=1}^n a_{ij}}{n - 1}$$

Social Networks Analysis

Centrality Measures:

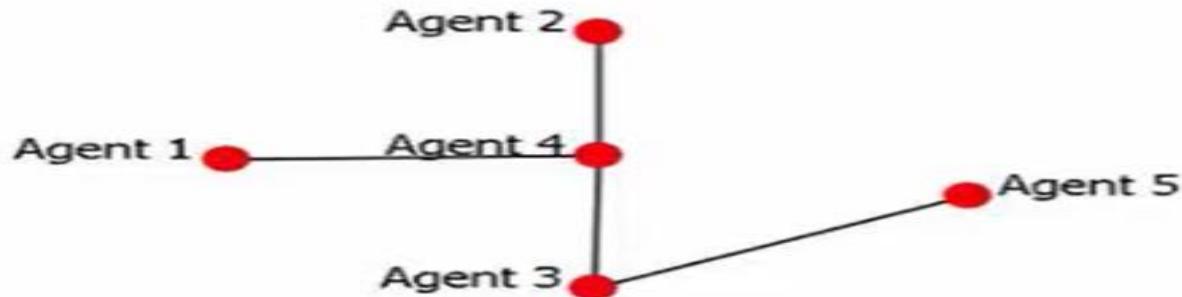
Calculating Degree



$$C_{Di} = \frac{\sum_{j=1}^n a_{ij}}{n - 1}$$

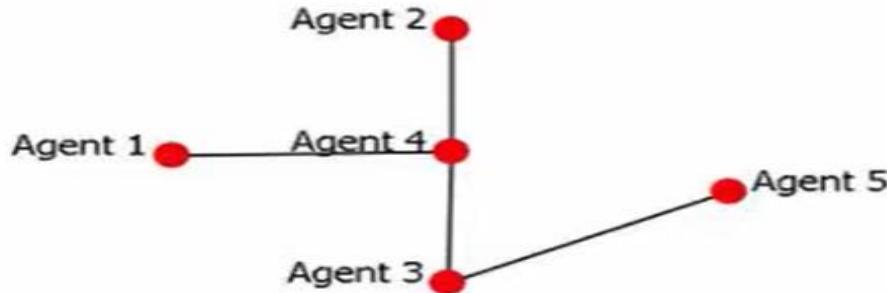
A	A	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5
Agent 1	0	0	0	1	0	
Agent 2	0	0	0	1	0	
Agent 3	0	0	0	1	1	
Agent 4	1	1	1	0	0	
Agent 5	0	0	1	0	0	

Calculating Degree



A	A	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Sum
Agent 1	0	0	0	1	0	1	1
Agent 2	0	0	0	1	0	0	1
Agent 3	0	0	0	1	1	1	2
Agent 4	1	1	1	0	0	0	3
Agent 5	0	0	1	0	0	0	1

Calculating Degree

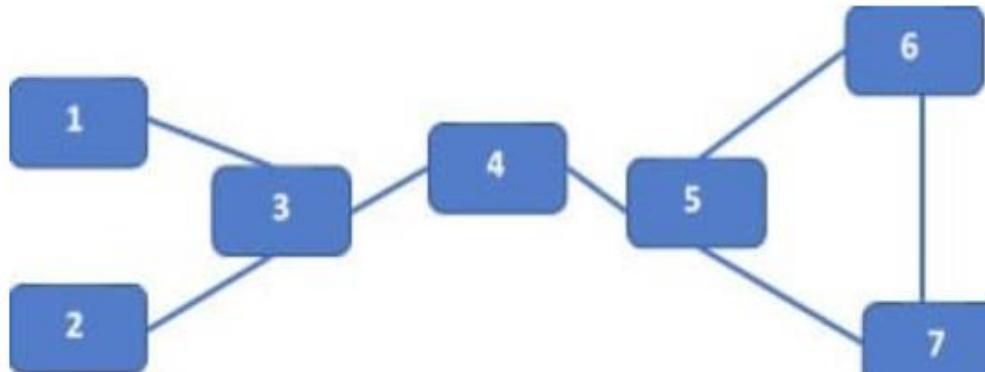


A	A	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Sum	/	(N-1)	=	Degree
Agent 1		0	0	1	0		1		4		1/4
Agent 2		0	0	1	0		1		4		1/4
Agent 3		0	0	1	1		2		4		2/4
Agent 4		1	1	1	0		3		4		3/4
Agent 5		0	0	1	0		1		4		1/4

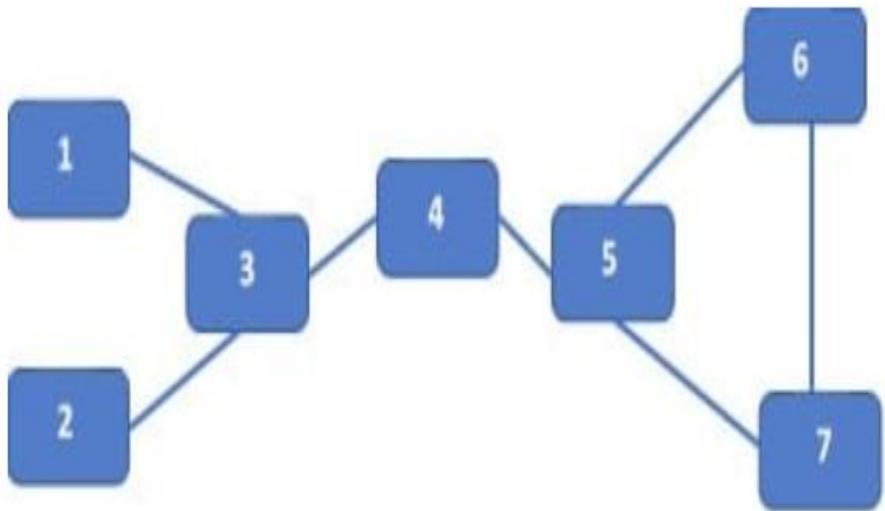
Social Networks Analysis

a) **Degree Centrality:** Measures the **number of direct ties** to a node; this will indicate the most connected node in the group.

Let's consider the network in *Figure*. The degree centrality score of a network is the sum of edges connected to that node. For Node 1, the degree centrality is 1, and for Nodes 3 and 5, the score is 3. The **standardized score is calculated by dividing the score by $(n-1)$** , where n is the number of nodes in the network.



Social Networks Analysis

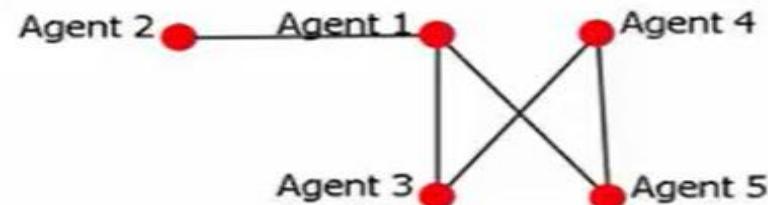


Node	Score	Standardized Score
1	1	1/6
2	1	1/6
3	3	3/6 = ½
4	2	2/6 = 1/3
5	3	3/6 = ½
6	2	2/6 = 1/3
7	2	2/6 = 1/3

We can see that nodes 3 and 5 have a high degree centrality of 0.5, i.e., they are the most well-connected nodes in the network.

Practical Exercise 2

You Calculate Degree



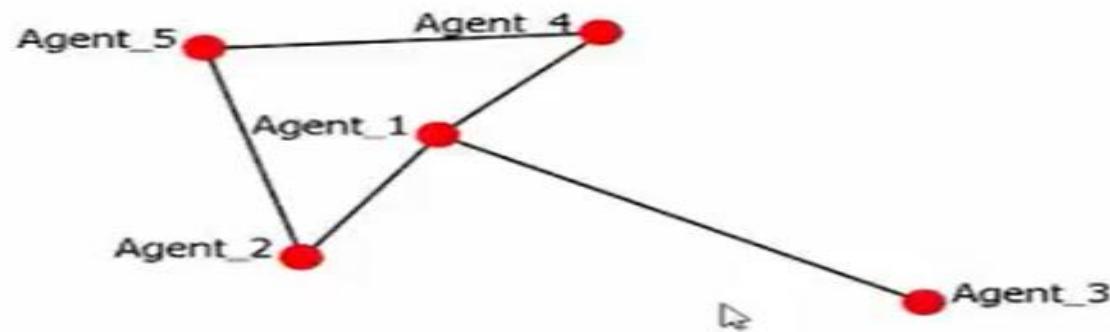
A	A	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Sum	/	(N-1)	=	Degree
Agent 1			1	1	0	1	3		4		3/4
Agent 2		1		0	0	0	1		4		1/4
Agent 3		1	0		1	0	2		4		2/4
Agent 4		0	0	1		1	2		4		2/4
Agent 5		1	0	0	1		2		4		2/4

Centrality Measures

Closeness

Definition

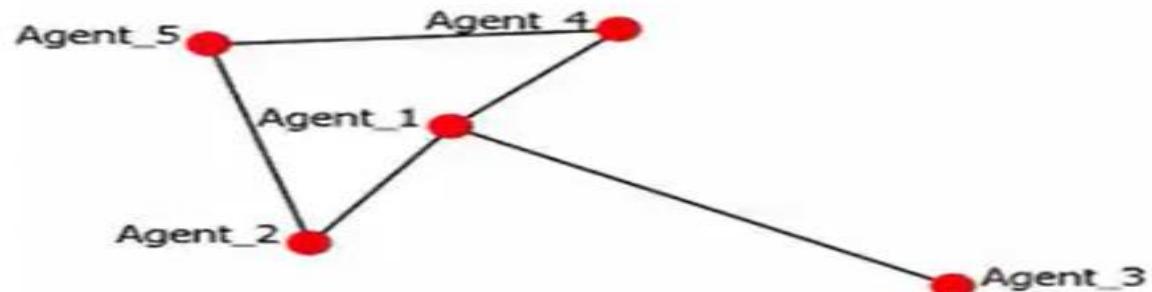
Closeness: the average of the shortest distance to all other nodes in the graph.



What do the Measures Tell Me?

Closeness: estimates time to hear info;
indirect influence; point of rapid diffusion.
Possible target for collection.

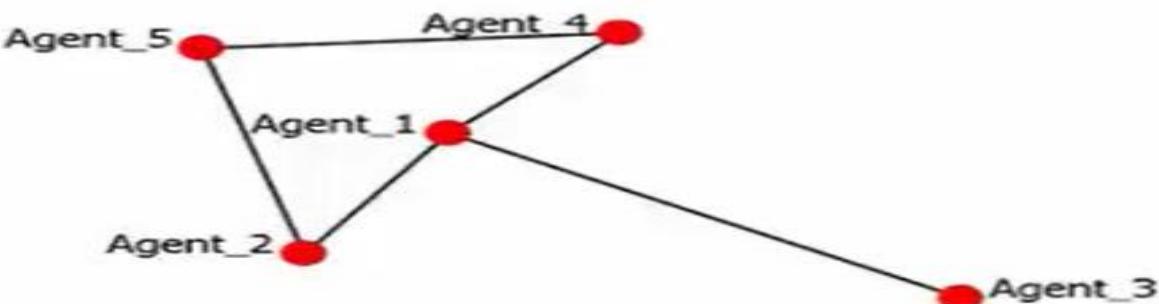
Calculating Closeness



	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5
Agent 1	█				
Agent 2		█			
Agent 3			█		
Agent 4				█	
Agent 5					█

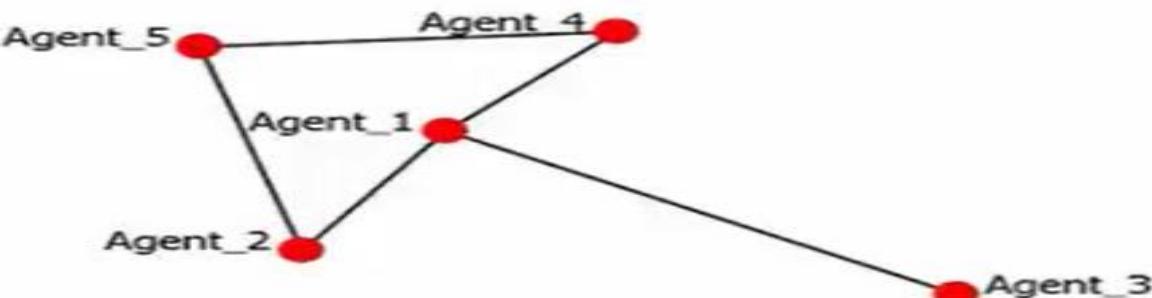


Calculating Closeness



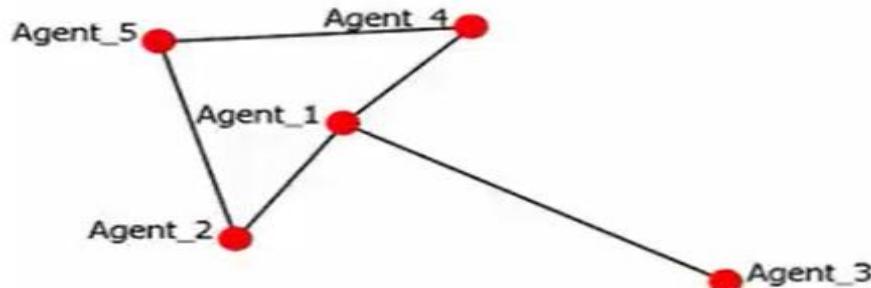
	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5
Agent 1	1	1	1	2	
Agent 2	1	2	2	1	
Agent 3	1	2	2	3	
Agent 4	1	2	2	1	
Agent 5	2	1	3	1	

Calculating Closeness



	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	SUM
Agent 1		1	1	1	2	5
Agent 2	1		2	2	1	6
Agent 3	1	2		2	3	8
Agent 4	1	2	2		1	6
Agent 5	2	1	3	1		7

Calculating Closeness

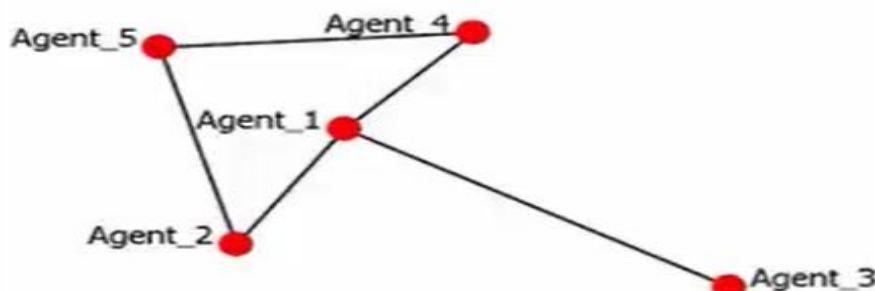


	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5
Agent 1	1	1	1	2	
Agent 2	1	2	2	2	1
Agent 3	1	2	2	2	3
Agent 4	1	2	2	2	1
Agent 5	2	1	3	1	2

SUM	/	(n-1)
5	/	4
6	/	4
8	/	4
6	/	4
7	/	4

Closeness

Calculating Closeness

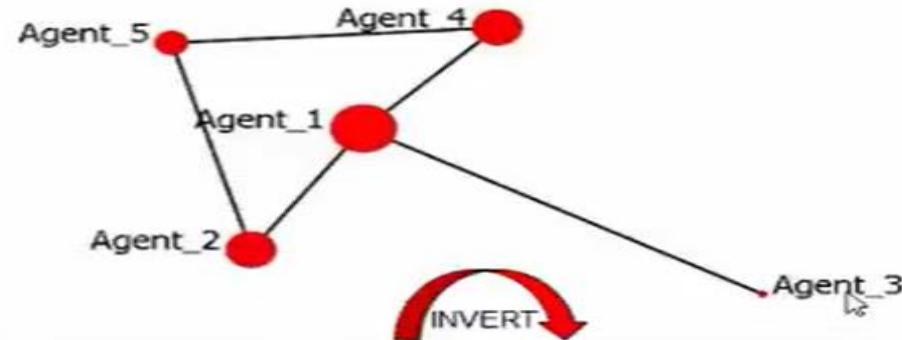
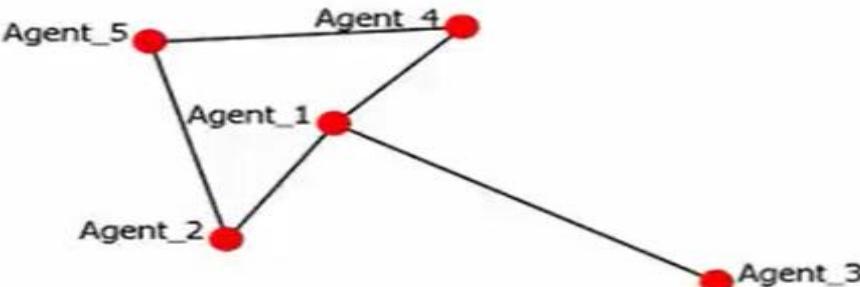


	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5
Agent 1		1	1	1	2
Agent 2	1		2	2	1
Agent 3	1	2		2	3
Agent 4	1	2	2		1
Agent 5	2	1	3	1	

INVERT ↘

SUM	/	(n-1)	Closeness
5	/	4	4/5
6	/	4	4/6
8	/	4	4/8
6	/	4	4/6
7	/	4	4/7

Calculating Closeness



	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5
Agent 1	1	1	1	2	
Agent 2	1	2	2	1	
Agent 3	1	2	2	1	3
Agent 4	1	2	2	1	
Agent 5	2	1	3	1	

SUM	/	(n-1)
5	/	4
6	/	4
8	/	4
6	/	4
7	/	4

Closeness
4/5
4/6
4/8
4/6
4/7

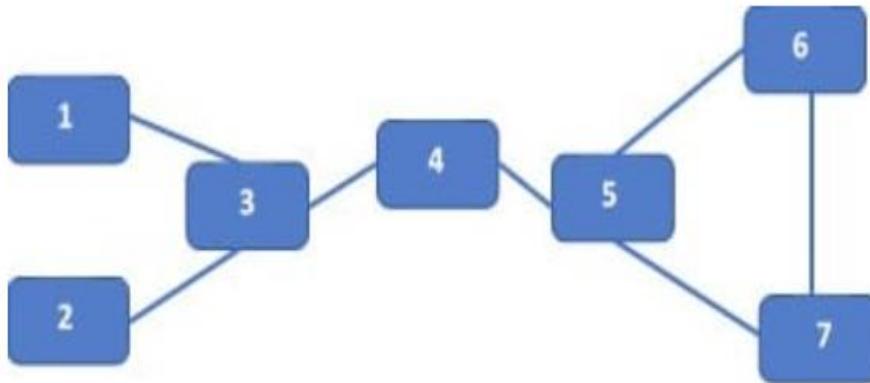
Social Networks Analysis

b) Closeness Centrality:

- Closeness measures **how close a node is to the rest of the network.**
- It is the **ability of the node to reach the other nodes** in the network.
- It is calculated as the inverse of the sum of the distance between a node and other nodes in the network.

Social Networks Analysis

Let us take node 1 from *Figure*; the sum of distances from node 1 to all other nodes is 16.



Destination Nodes ->	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Total Distance
Distance from Node 1 to Destination Node	2	1	2	3	4	4	16

Social Networks Analysis

Hence the Closeness score for node 1 will be $1/16$. The standardized score is calculated by multiplying the score by $(n-1)$.

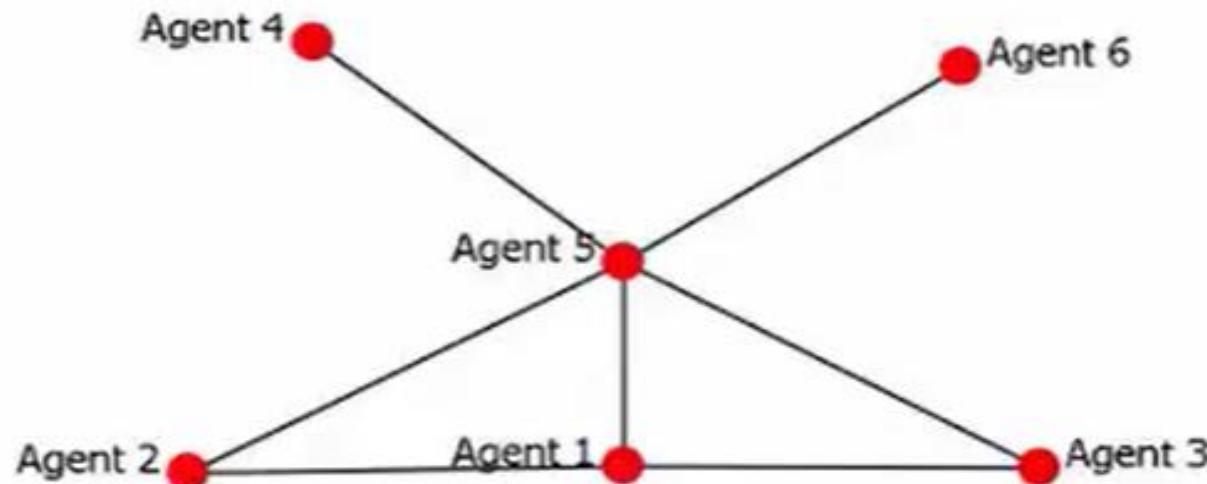
Node	Score	Standardized Score
1	$1/16$	$6/16 = 3/8$
2	$1/16$	$6/16 = 3/8$
3	$1/11$	$6/11$
4	$1/10$	$6/10 = 3/5$
5	$1/11$	$6/11$
6	$1/15$	$6/15 = 2/5$
7	$1/15$	$6/15 = 2/5$

We can conclude that node 4 is the closest/central node in the network with the highest closeness score of 0.6.

Practical Exercise 4

You Calculate Closeness

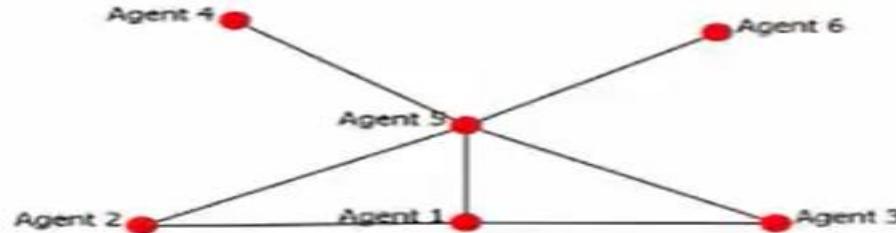
Adjacency Matrix:



Centrality Measures:

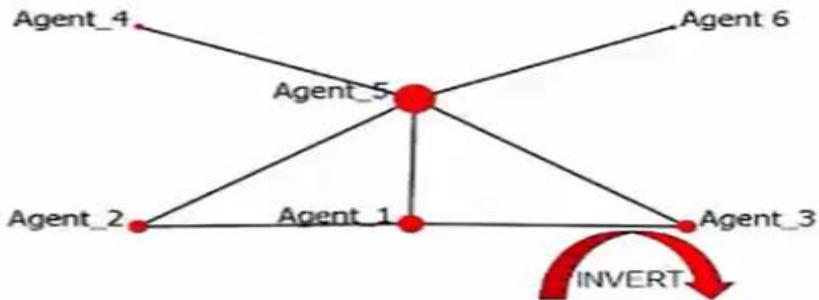
Social Networks Analysis

You Calculate Closeness



	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Agent 6
Agent 1		1	1	2	1	2
Agent 2	1		2	2	1	2
Agent 3	1	2		2	1	2
Agent 4	2	2	2		1	2
Agent 5	1	1	1	1		1
Agent 6	2	2	2	2	1	

You Calculate Closeness



	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Agent 6
Agent 1		1	1	2	1	2
Agent 2	1		2	2	1	2
Agent 3	1	2		2	1	2
Agent 4	2	2	2		1	2
Agent 5	1	1	1	1		1
Agent 6	2	2	2	2	1	

SUM	/	(n-1)	=	Closeness
7	/	5		5/7
8	/	5		5/8
8	/	5		5/8
9	/	5		5/9
5	/	5		5/5
9	/	5		5/9

Social Networks Analysis

Centrality Measures

Betweenness

Definition

Betweenness: extent to which a particular node lies on the shortest path between other nodes.



What do the Measures Tell Me?

Betweenness: informal power; gate keeping; brokering; controls flow of info; liaison between sub-comp. Possible target for disruption of network.

Social Networks Analysis

Calculating Betweenness

$$nC_2 = \frac{n(n - 1)}{2}$$

$$\frac{5(5-1)}{2} = \frac{5(4)}{2} = \frac{20}{2} = 10$$

paths possible



Agent 1

Agent 2

Agent 3

Agent 4

Agent 5

From	To
1	1
1	2
1	3
1	4
1	5

Social Networks Analysis

Calculating Betweenness

$$nC_2 = \frac{n(n - 1)}{2}$$

$$\frac{5(5-1)}{2} = \frac{5(4)}{2} = \frac{20}{2} = 10$$

paths possible



Agent 1

From	To
1	1
1	2
1	3
1	4
1	5

Agent 2

From	To
2	1
2	2

Agent 3

From	To
2	3
2	4

Agent 4

From	To
2	5

Agent 5

From	To

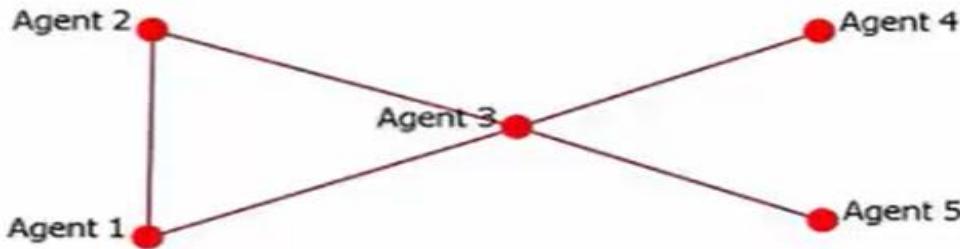
Social Networks Analysis

Calculating Betweenness

$$nC_2 = \frac{n(n - 1)}{2}$$

$$\frac{5(5-1)}{2} = \frac{5(4)}{2} = \frac{20}{2} = 10$$

paths possible



Agent 1

From	To
1	1
1	2
1	3
1	4
1	5

Agent 2

From	To
2	1
2	2

Agent 3

From	To
3	1
3	2
3	3
3	4
3	5

Agent 4

From	To
4	1
4	2
4	3
4	4
4	5

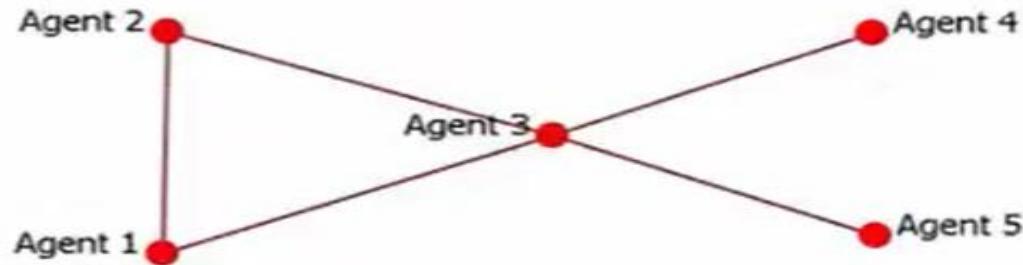
Agent 5

From	To
5	1
5	2
5	3
5	4
5	5

Social Networks Analysis

Calculating Betweenness

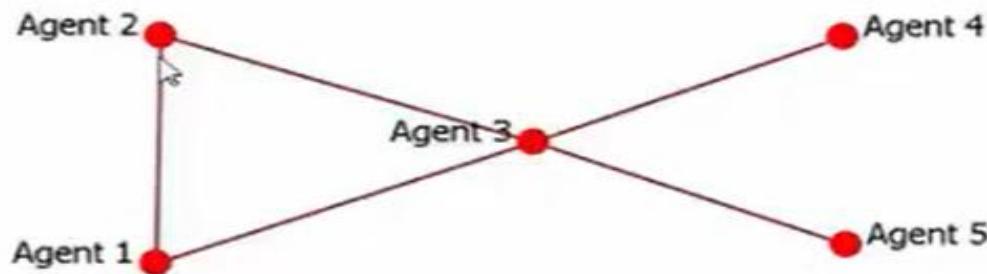
From	To	Geodesic
1	2	
1	3	
1	4	
1	5	
2	3	
2	4	
2	5	
3	4	
3	5	
4	5	



Social Networks Analysis

Calculating Betweenness

From	To	Geodesic
1	2	(1,2)
1	3	(1,3)
1	4	(1,3,4)
1	5	(1,3,5)
2	3	(2,3)
2	4	(2,3,4)
2	5	(2,3,5)
3	4	(3,4)
3	5	(3,5)
4	5	(4,3,5)



Social Networks Analysis

Calculating Betweenness

From	To
1	2
1	3
1	4
1	5
2	3
2	4
2	5
3	4
3	5
4	5

Geodesic
(1,2)
(1,3)
(1,3,4)
(1,3,5)
(2,3)
(2,3,4)
(2,3,5)
(3,4)
(3,5)
(4,3,5)

Social Networks Analysis

Calculating Betweenness

From	To	Geodesic
1	2	(1,2)
1	3	(1,3)
1	4	(1,3,4)
1	5	(1,3,5)
2	3	(2,3)
2	4	(2,3,4)
2	5	(2,3,5)
3	4	(3,4)
3	5	(3,5)
4	5	(4,3,5)

Social Networks Analysis

Calculating Betweenness

From	To
1	2
1	3
1	4
1	5
2	3
2	4
2	5
3	4
3	5
4	5

Geodesic
(1,2)
(1,3)
(1,3,4)
(1,3,5)
(2,3)
(2,3,4)
(2,3,5)
(3,4)
(3,5)
(4,3,5)

1	2	3	4	5
0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	0	1	0	0
0	0	0	0	0
0	0	1	0	0
0	0	1	0	0
0	0	0	0	0
0	0	0	0	0
0	0	1	0	0

Social Networks Analysis

Calculating Betweenness

From	To	Geodesic
1	2	(1,2)
1	3	(1,3)
1	4	(1,3,4)
1	5	(1,3,5)
2	3	(2,3)
2	4	(2,3,4)
2	5	(2,3,5)
3	4	(3,4)
3	5	(3,5)
4	5	(4,3,5)

1	2	3	4	5	
0	0	0	0	0	
0	0	0	0	0	
0	0	1	0	0	
0	0	1	0	0	
0	0	0	0	0	
0	0	1	0	0	
0	0	1	0	0	
0	0	0	0	0	
0	0	0	0	0	
0	0	1	0	0	
0	0	0	0	0	
0	0	5	0	0	Sum

Social Networks Analysis

Calculating Betweenness

$$\frac{(n - 1)(n - 2)}{2} = \text{Denominator}$$



0	0	5	0	0	Numerator

Social Networks Analysis

Calculating Betweenness

$$\frac{(n - 1)(n - 2)}{2} = \text{Denominator}$$

$$\frac{(5 - 1)(5 - 2)}{2} = \frac{(4)(3)}{2} = 6$$



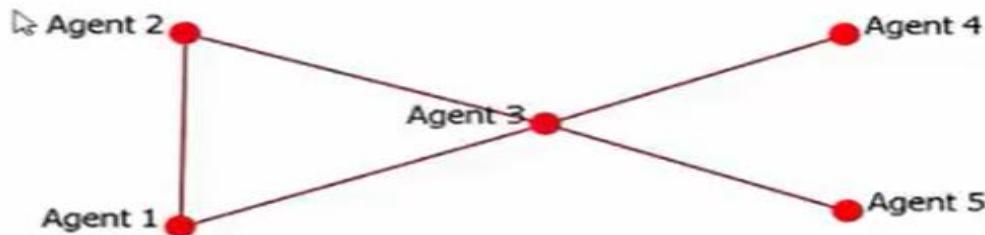
0	0	5	0	0	Numerator
6	6	6	6	6	Denominator

Social Networks Analysis

Calculating Betweenness

$$\frac{(n - 1)(n - 2)}{2} = \text{Denominator}$$

$$\frac{(5 - 1)(5 - 2)}{2} = \frac{(4)(3)}{2} = 6$$



0	0	5	0	0	Numerator
6	6	6	6	6	Denominator

0/6	0/6	5/6	0/6	0/6	Betweenness
-----	-----	-----	-----	-----	-------------

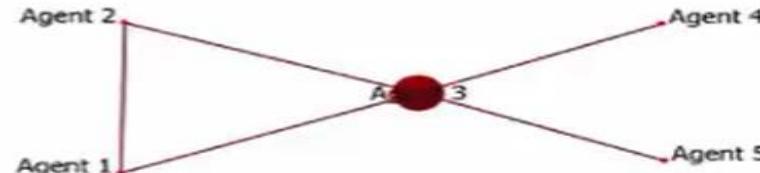
Social Networks Analysis

Calculating Betweenness

$$\frac{(n - 1)(n - 2)}{2} = \text{Denominator}$$

$$\frac{(5 - 1)(5 - 2)}{2} = \frac{(4)(3)}{2} = 6$$

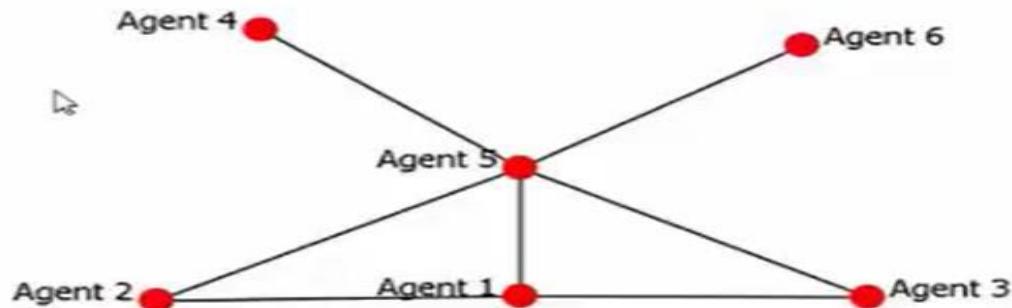
0	0	5	0	0	Numerator
6	6	6	6	6	Denominator
0/6	0/6	5/6	0/6	0/6	Betweenness



Social Networks Analysis

You Calculate Degree/Betweenness

From	To	Path
1	2	
1	3	
1	4	
1	5	
1	6	
2	3	
2	4	
2	5	
2	6	
3	4	
3	5	
3	6	
4	5	
4	6	
5	6	



Social Networks Analysis

You Calculate Degree/Betweenness

From	To	Path
1	2	(1,2)
1	3	(1,3)
1	4	(1,5,4)
1	5	(1,5)
1	6	(1,5,6)
2	3	(2,1,3)(2,5,3)
2	4	(2,5,4)
2	5	(2,5)
2	6	(2,5,6)
3	4	(3,5,4)
3	5	(3,5)
3	6	(3,5,6)
4	5	(4,5)
4	6	(4,5,6)
5	6	(5,6)

Social Networks Analysis

You Calculate Degree/Betweenness

1	2	3	4	5	6	
.5	0	0	0	7.5	0	SUM
10	10	10	10	10	10	Denominator

$$\frac{(n - 1)(n - 2)}{2}$$

$$\frac{(6 - 1)(6 - 2)}{2} = \frac{20}{2} = 10$$

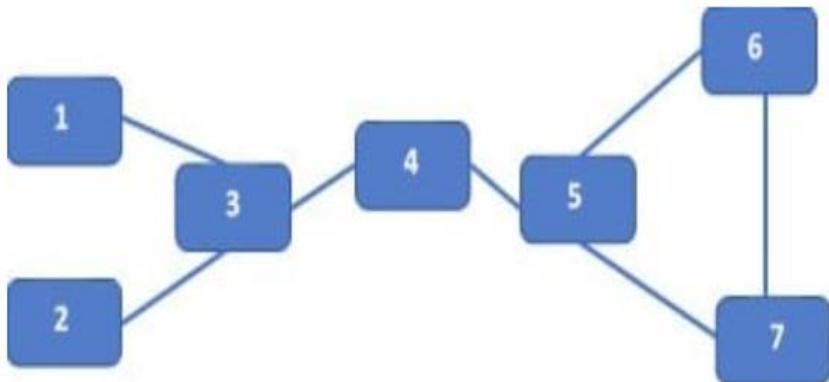
.05	0	0	0	.75	0	Betweenness
-----	---	---	---	-----	---	-------------



Social Networks Analysis

c) Betweenness Centrality:

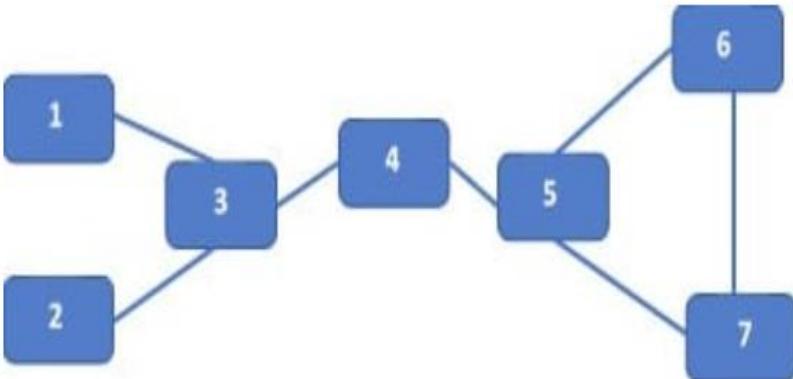
- It is a measure of how often a node appears in the shortest path connecting two other nodes.
- Let us take node 5 in *Figure*. Node 5 occurs in 9 shortest paths between a pair of nodes (as shown in *Table*).



Node pairs	Path value of Node 5
1,5	$\frac{1}{2}$
1,6	$\frac{1}{3}$
1,7	$\frac{1}{3}$
2,5	$\frac{1}{2}$
2,6	$\frac{1}{3}$
2,7	$\frac{1}{3}$
3,5	1
3,6	$\frac{1}{2}$
3,7	$\frac{1}{2}$
Total Score for Node 5	13/3

Social Networks Analysis

- If node 5 is the only node in the path, then the **path value is 1**.
- If it is one of the ‘n’ nodes in the shortest path, then the **path value is $1/n$** .
- The sum of path values for node 5 for **all nine pairs of nodes** is its **betweenness score**.
- These values are then standardized by dividing the score by $(n-1)*(n-2)/2$



Node	Score	Standardized Score
1	0	0
2	0	0
3	16/3	16/45
4	13/3	13/45
5	13/3	13/45
6	0	0
7	0	0

Social Networks Analysis

Nodes with high betweenness centrality are critical in controlling and maintaining flow in the network; hence these are critical nodes in the network

→ Topics to be Discussed

- ✓ 5.1.1 A model for Recommendation systems
- ✓ 5.1.2 Content Based Recommendations
- ✓ 5.1.3 Collaborative Filtering

- ✓ 5.2.1 Case Study :Product Recommendation

- ✓ 5.3.1 Social Networks as Graphs,
5.3.2 Clustering of Social-Network Graphs
5.3.3 Direct Discovery of Communities in a social graph.

Clustering Social Networks

- Social networks have gained popularity recently with the advent of sites such as MySpace, Friendster, Facebook, etc. The number of users participating in these networks is large, e.g., a hundred million in MySpace, and growing.
- These networks are a rich source of data as users populate their sites with personal information.
- A fundamental problem related to these networks is the **discovery of clusters or communities**. Intuitively, a cluster is a collection of individuals with dense friendship patterns internally and sparse friendships externally.

What is Clustering?

Applications of Clustering



Customer
Segmentation

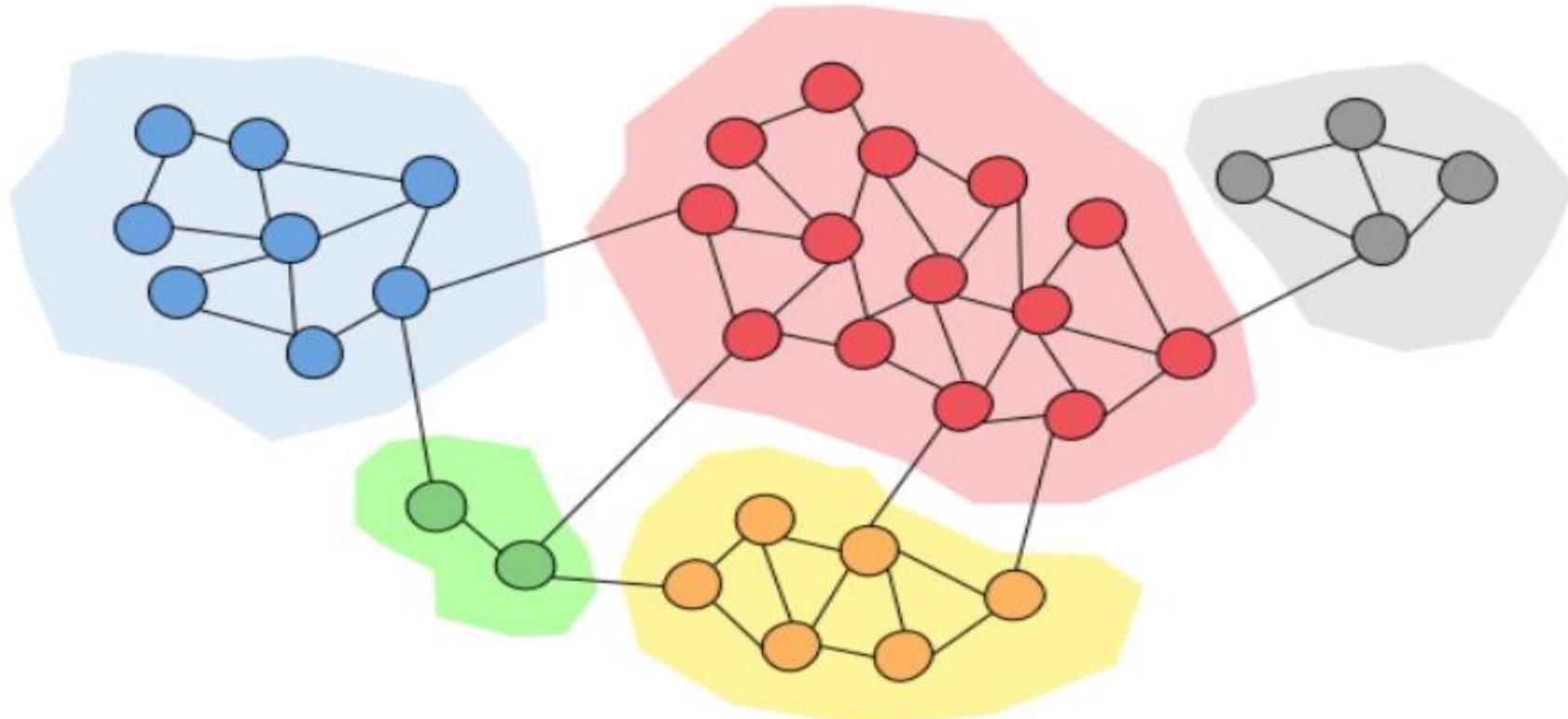


Social Network
Analysis



City Planning

Clustering Social Networks



Clustering Social Networks

Why Community Detection?

When analyzing different networks, it may be important to discover communities inside them. Community detection techniques are useful for social media algorithms to discover people with common interests and keep them tightly connected.

Community detection can be used in machine learning to detect groups with similar properties and extract groups for various reasons. For example, this technique can be used to discover manipulative groups inside a social network or a stock market.

Clustering of Social-Network Graphs

Clustering of Social-Network Graphs:

Clustering of the graph is considered as a way to identify communities.

Clustering of graphs involves following steps:

1. Distance Measures for Social-Network Graphs
2. Applying Standard Clustering Methods
3. Betweenness: The Girvan-Newman Algorithm:
4. Direct Detection of Community : Clique Percolation Method

Clustering of Social-Network Graphs

Distance Measures for Social-Network Graphs

If we were to apply standard clustering techniques to a social-network graph, **our first step would be to define a distance measure.** When the edges of the graph have labels, these labels might be usable as a distance measure, depending on what they represented. But when the edges are unlabeled, as in a “friends” graph, there is not much we can do to define a suitable distance. Our first instinct is to assume that nodes are close if they have an edge between them and distant if not. Thus, we could say that the distance $d(x, y)$ is 0, if there is an edge (x, y) and 1 if there is no such edge. We could use any other two values, such as 1 and ∞ , as long as the distance is closer when there is an edge.

Clustering of Social-Network Graphs

Applying Standard Clustering Methods

There are two general approaches to clustering:
hierarchical (agglomerative) and point-assignment.

- Hierarchical clustering of a social-network graph starts by combining some two nodes that are connected by an edge. Successively, edges that are not between two nodes of the same cluster would be chosen randomly to combine the clusters to which their two nodes belong. The choices would be random, because all distances represented by an edge are the same.

Clustering of Social-Network Graphs

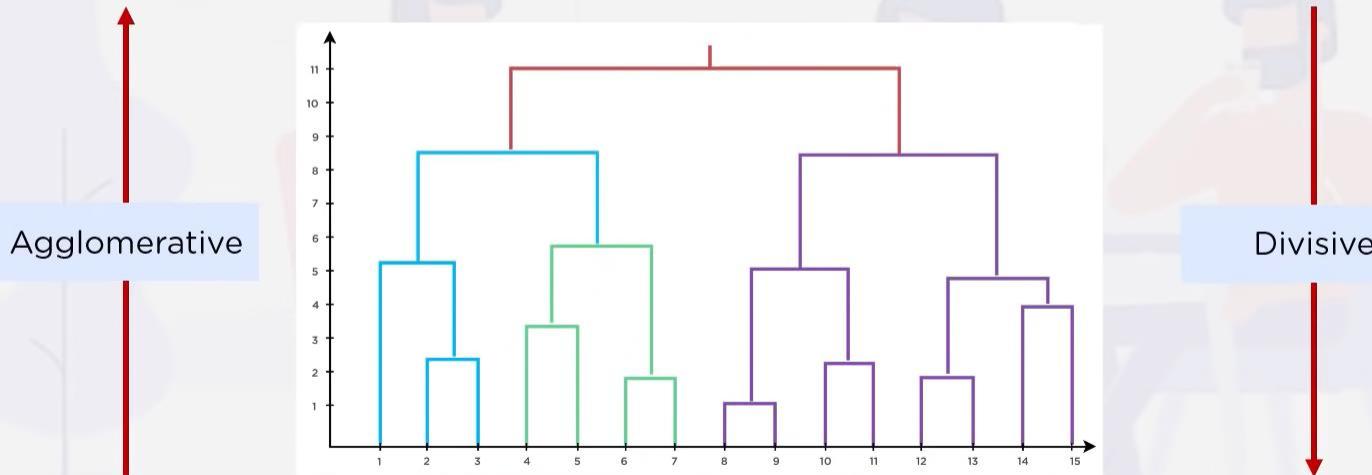
Applying Standard Clustering Methods

There are two general approaches to clustering:
hierarchical (agglomerative) and point-assignment.

- Now, consider a point-assignment approach to clustering social networks. Again, the fact that all edges are at the same distance will introduce a number of random factors that will lead to some nodes being assigned to the wrong cluster.

Types of Hierarchical Clustering

Divisive Clustering is known as top-down approach



Clustering of Social-Network Graphs

3. Betweenness:

Since there are problems with standard clustering methods, several **specialized clustering techniques have been developed to find communities in social networks**. The simplest one is based on finding the edges that are least likely to be inside the community.

Define the betweenness of an edge (a, b) to be the number of pairs of nodes x and y such that the edge (a, b) lies on the shortest path between x and y . To be more precise, since there can be several shortest paths between x and y , edge (a, b) is credited with the fraction of those shortest paths that include the edge (a, b) . As in golf, a high score is bad. It suggests that the edge (a, b) runs between two different communities; that is, a and b do not belong to the same community.

Clustering of Social-Network Graphs

4. The Girvan-Newman Algorithm:

In order to exploit the betweenness of edges, we need to calculate the **number of shortest paths going through each edge**. We shall describe a method called the Girvan-Newman (GN) Algorithm, which visits each node X once and computes the number of shortest paths from X to each of the other nodes that go through each of the edges. The algorithm begins by performing a breadth-first search (BFS) of the graph, starting at the node X . Note that the level of each node in the BFS presentation is the length of the shortest path from X to that node. Thus, the edges that go between nodes at the same level can never be part of a shortest path from X .

Edges between levels are called DAG edges (“DAG” stands for directed, acyclic graph). Each DAG edge will be part of at least one shortest path from root X . If there is a DAG edge (Y, Z) , where Y is at the level above Z (i.e., closer to the root), then we shall call Y a parent of Z and Z a child of Y , although parents are not necessarily unique in a DAG as they would be in a tree.

Clustering of Social-Network Graphs

5. Using betweenness to find communities:

The betweenness scores for the edges of a graph behave something like a distance measure on the nodes of the graph. It is not exactly a distance measure, because it is not defined for pairs of nodes that are unconnected by an edge, and might not satisfy the triangle inequality even when defined. However, we can cluster by taking the edges in **order of increasing betweenness and add them to the graph one at a time**. At each step, the connected components of the graph form some clusters. The higher the betweenness we allow, the more edges we get, and the larger the clusters become.

More commonly, this idea is expressed as a process of edge removal. Start with the graph and all its edges; then remove edges with the highest betweenness, until the graph has broken into a suitable number of connected components.

Clustering of Social-Network Graphs

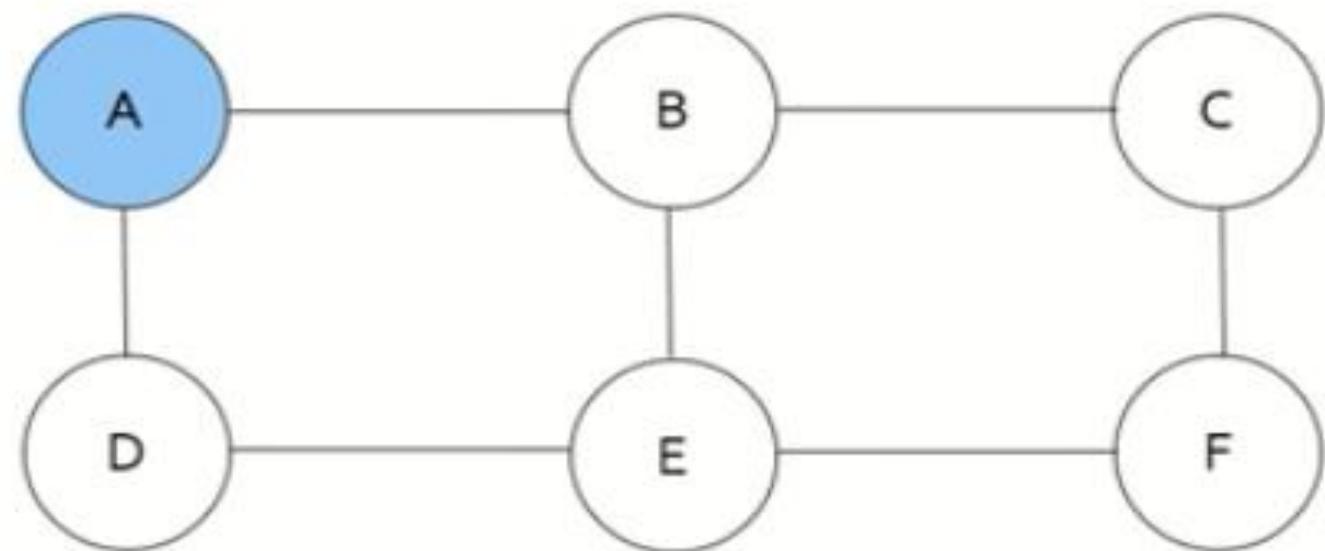
Girvan-Newman Iteration:

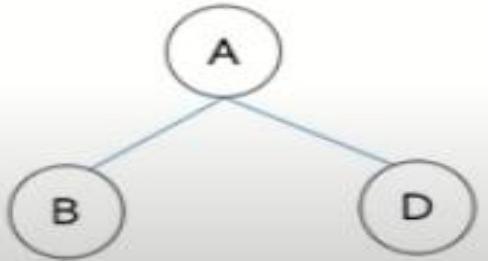
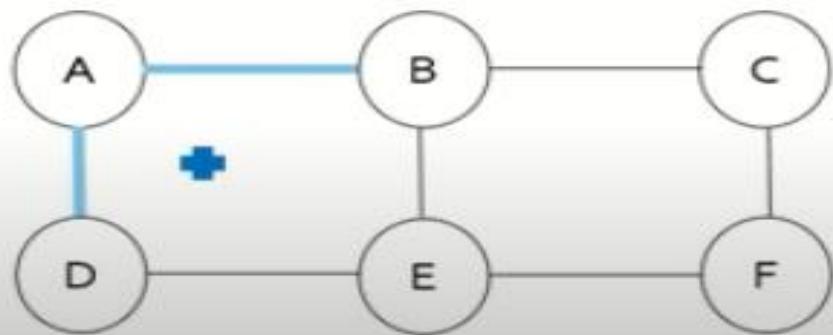
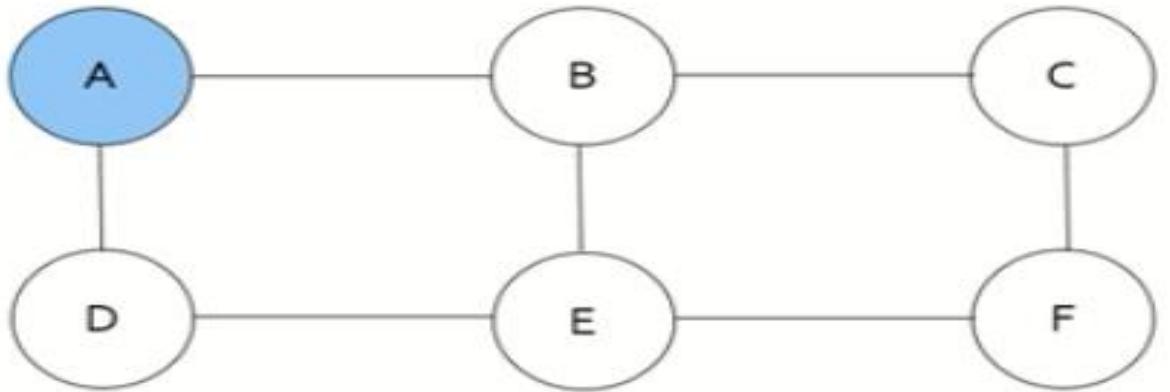
First we need to compute the edge betweenness of every edge in the graph

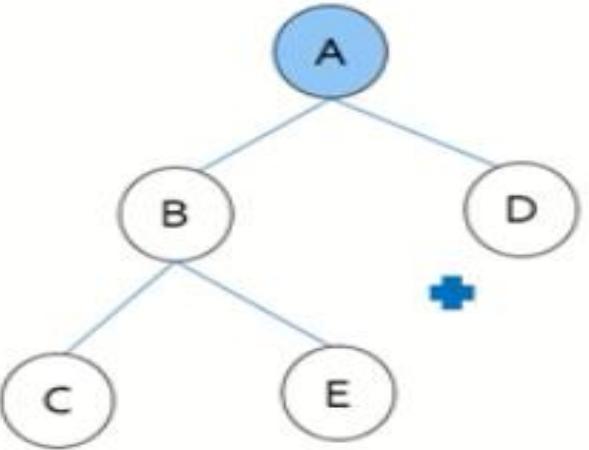
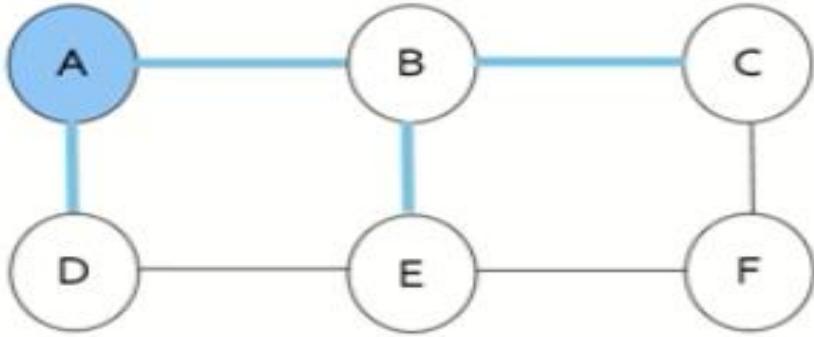
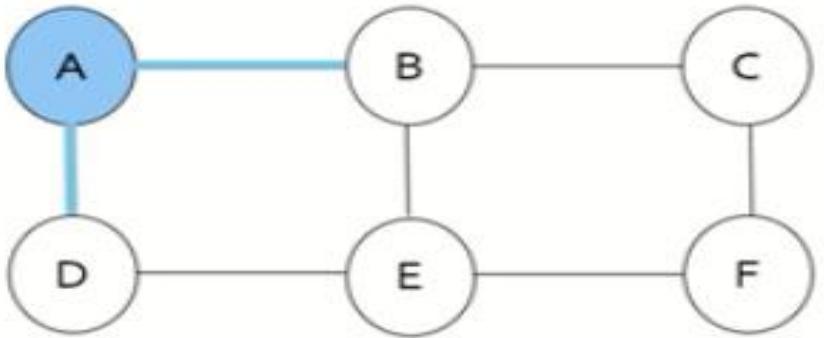
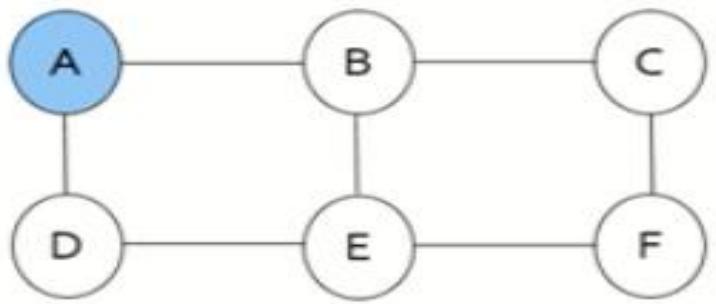
1. Select a node X , and perform BFS to find number of shortest path from the node X to each node, and assign the numbers as score to each node.
2. Starting from the leaf nodes, we calculate the credit of edge by $(1 + (\text{sum of the edge credits to the node})) * (\text{score of destination node} / \text{score of starting node})$
3. Compute the edge credits of all edges in the graph G , and repeat from step 1. until all of the nodes are selected
4. Sum up all of the edge credit we compute in step 2 and divide by 2, and the result is the edge betweenness of edges.

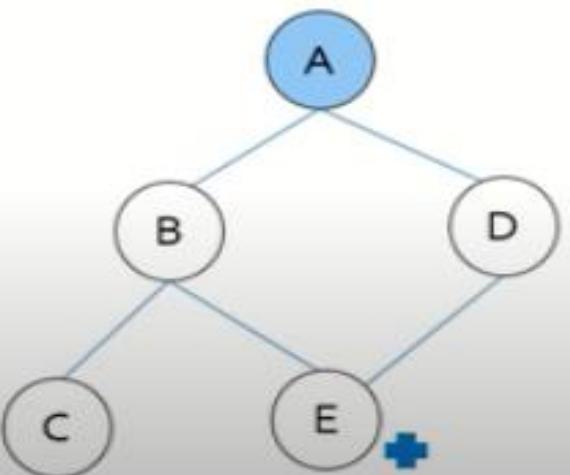
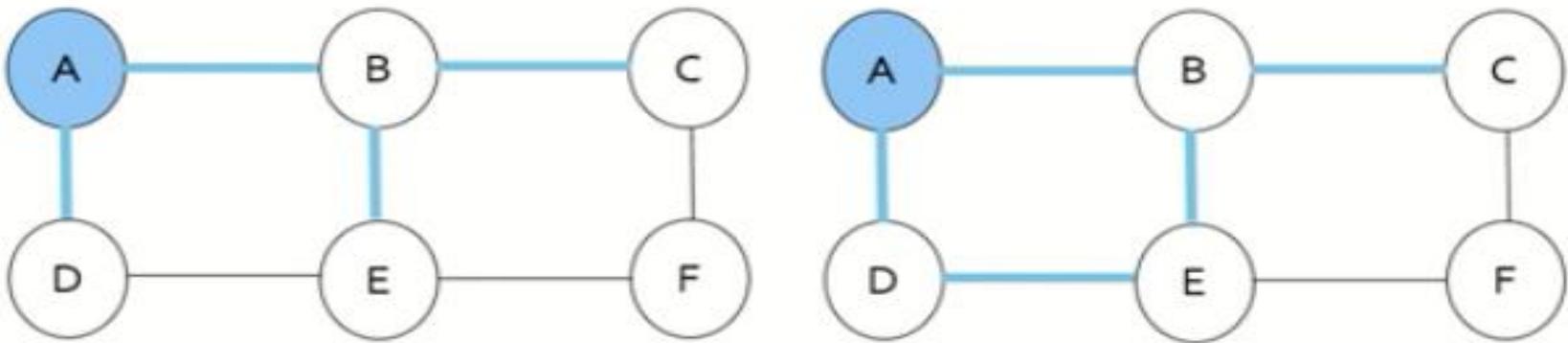
- Repeat until no edges are left:
 - ✓ Calculate edge betweenness for every edge in the graph.
 - ✓ Remove the edge with highest edge betweenness.
 - ✓ Calculate edge betweenness for remaining edges.
 - ✓ Connected components are communities

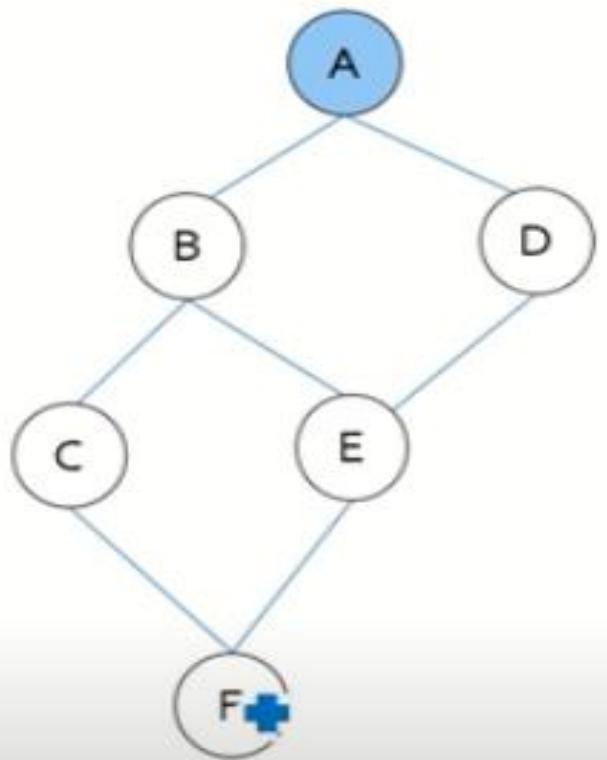
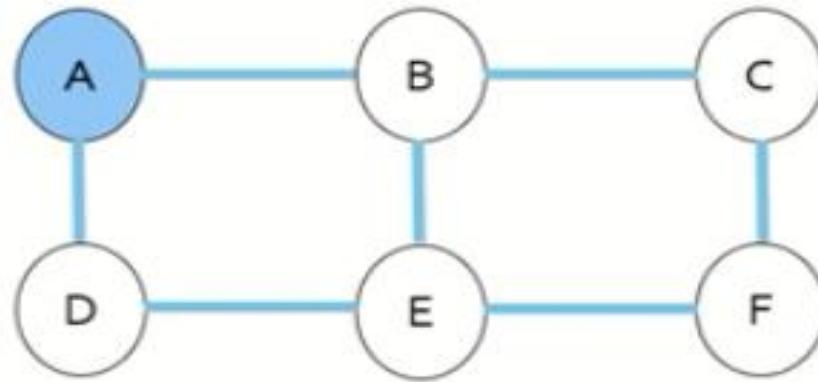
Example

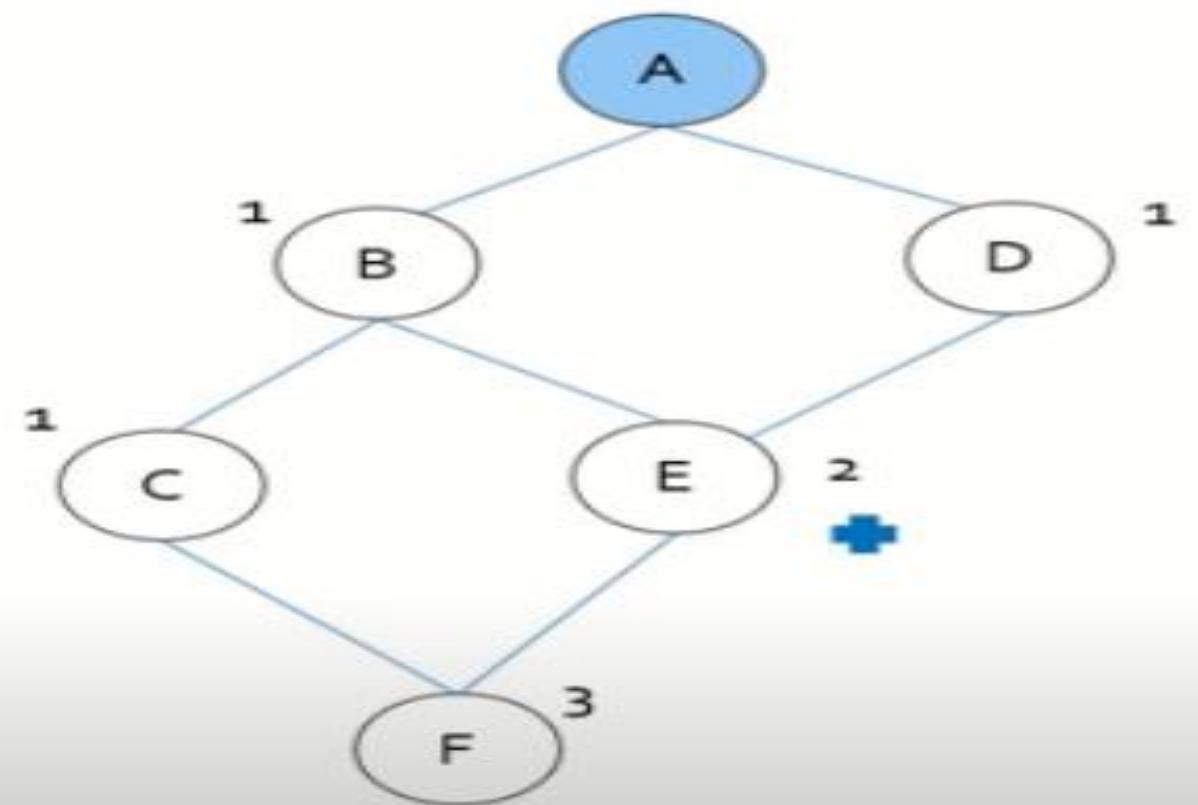


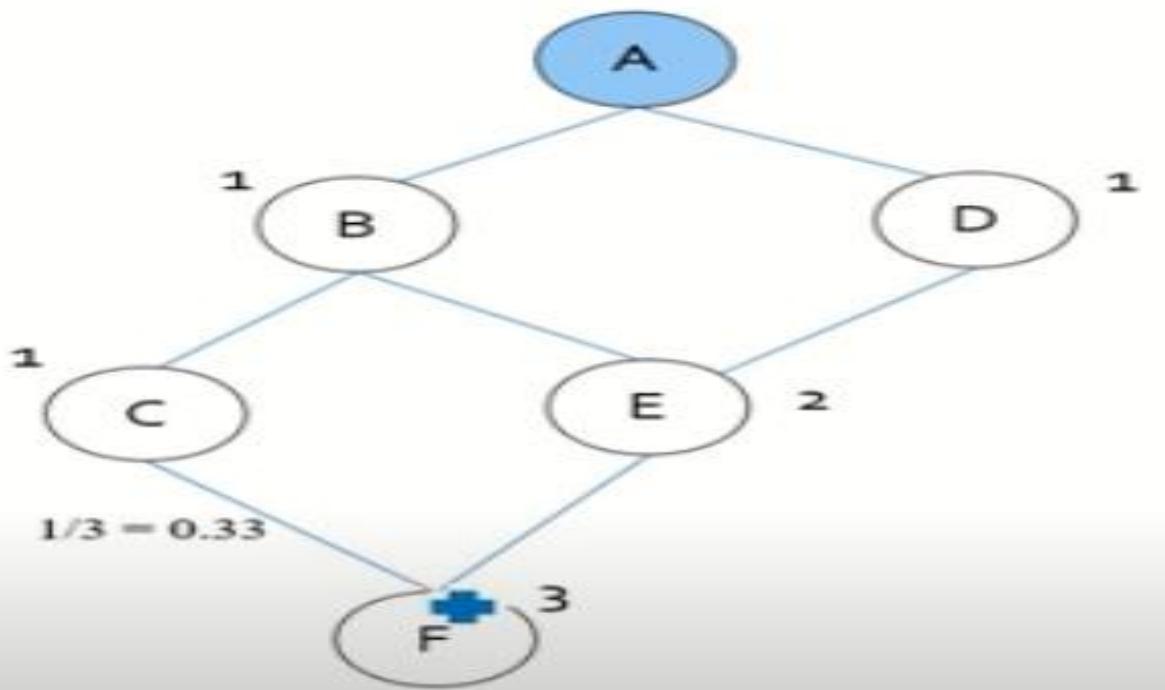


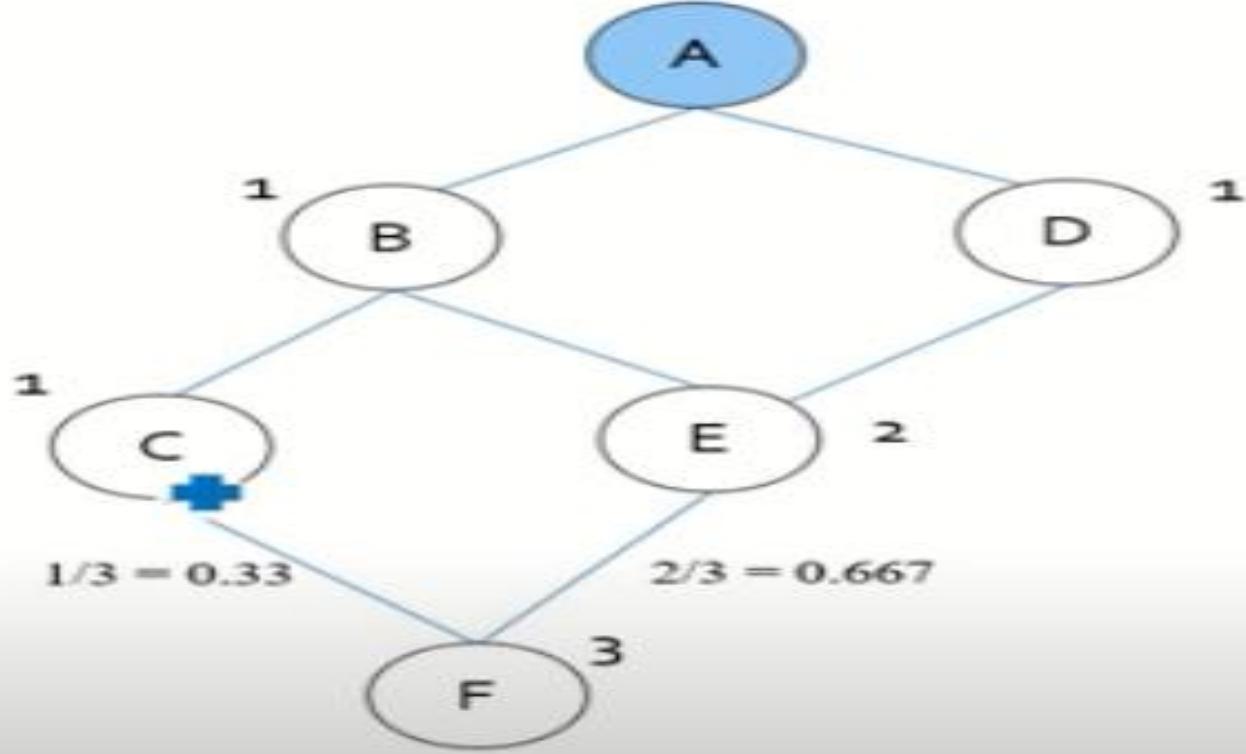


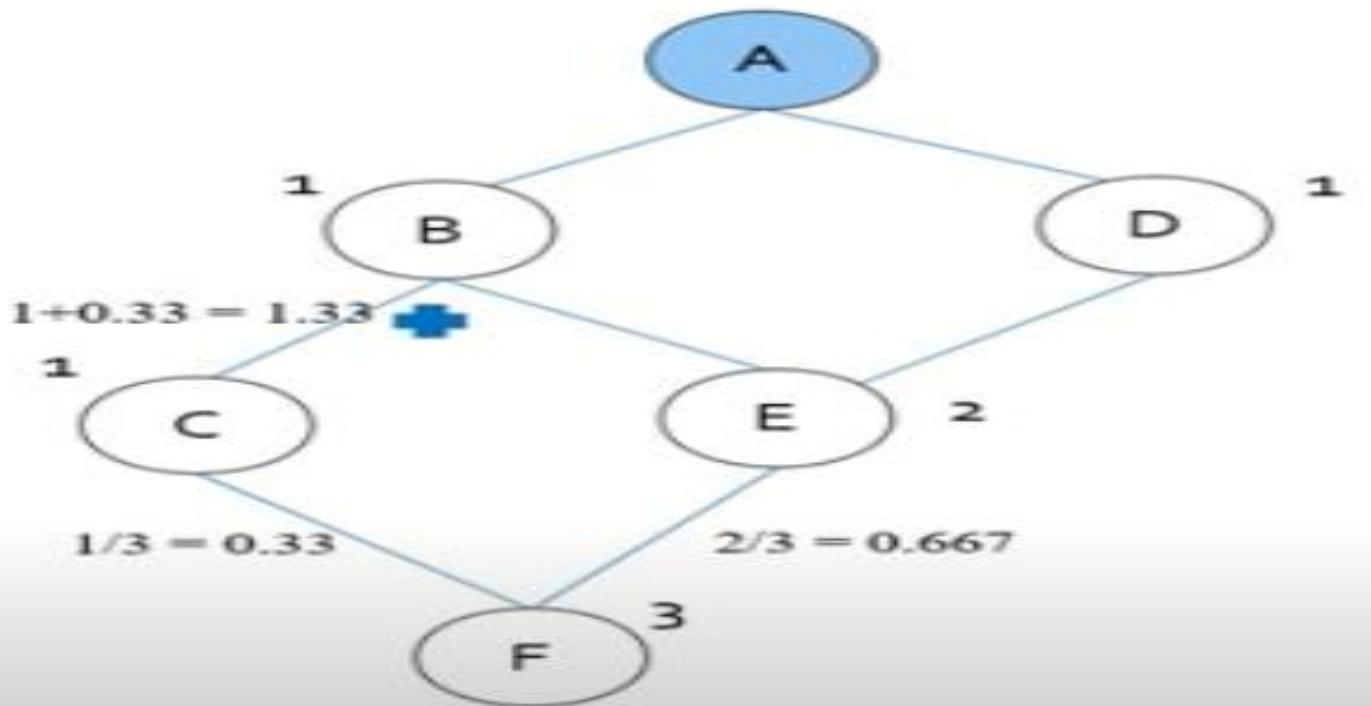


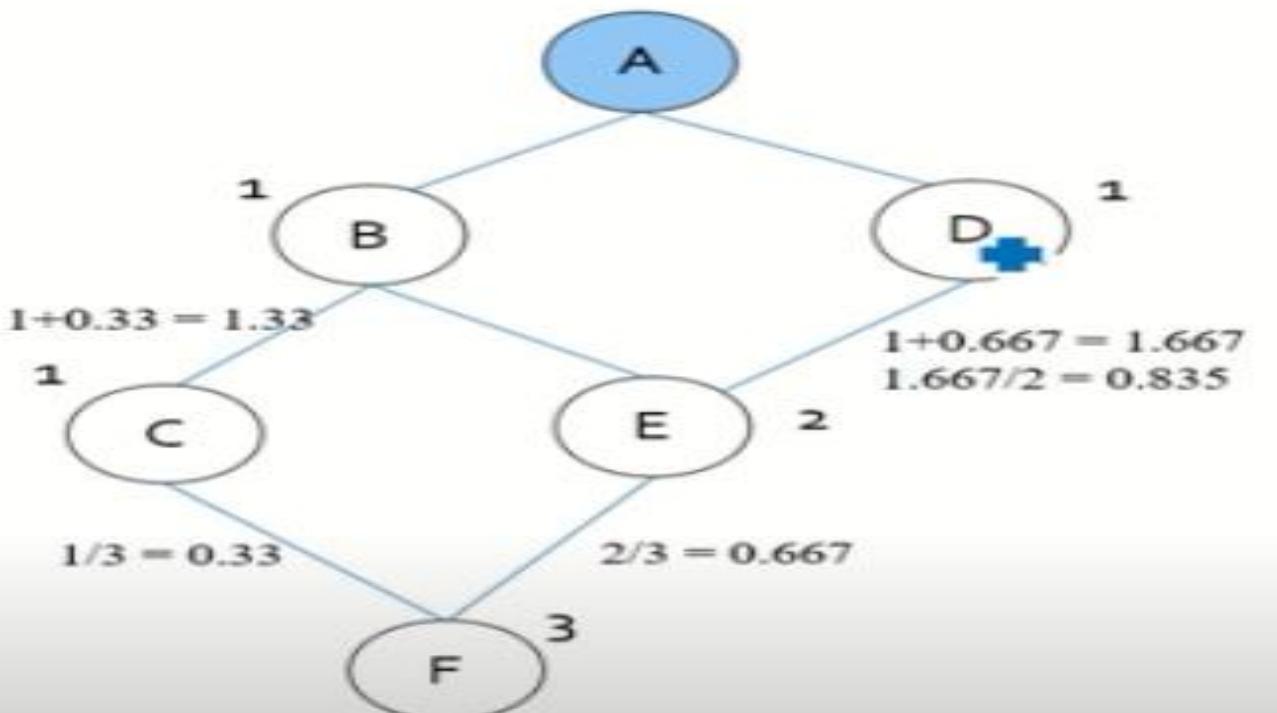


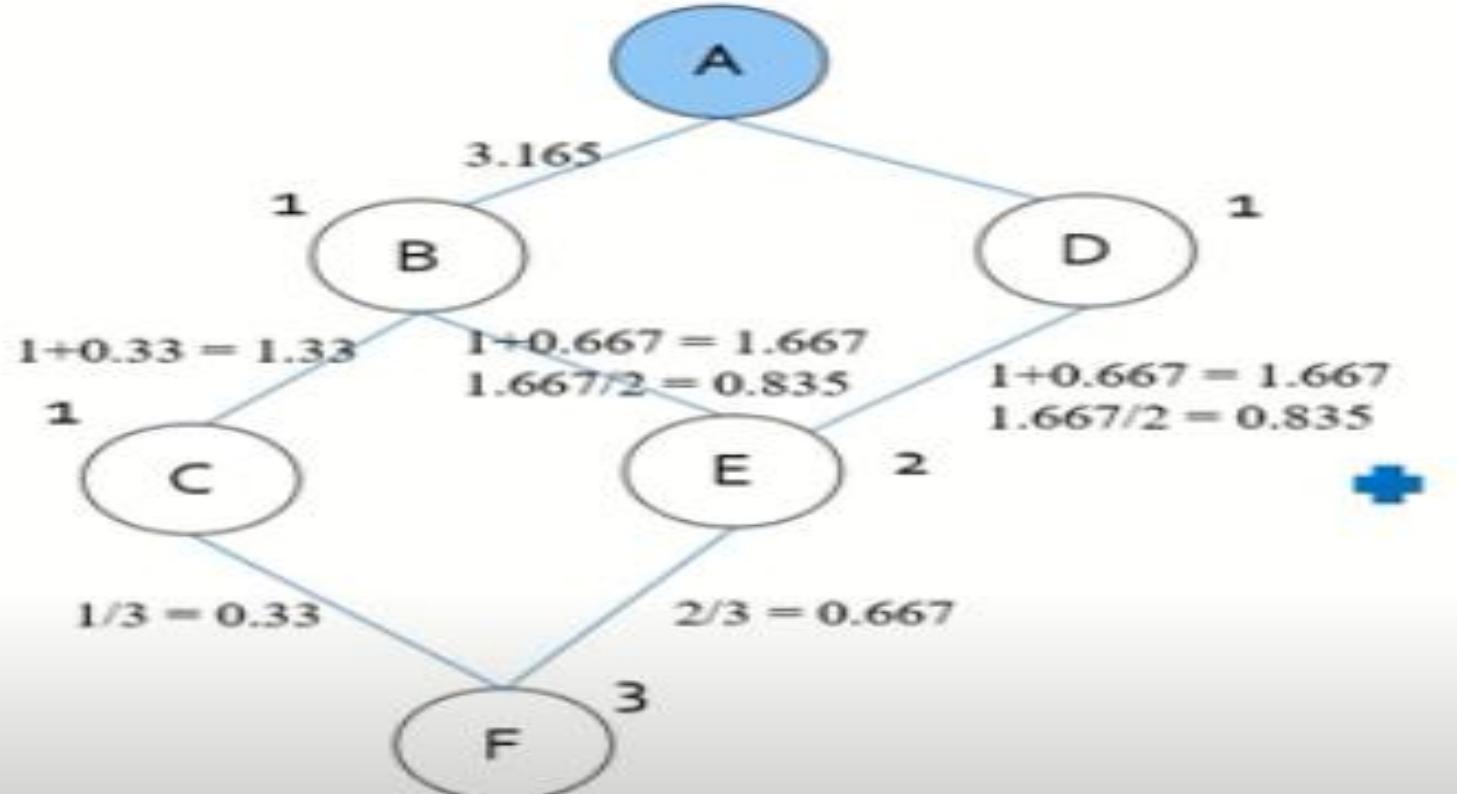


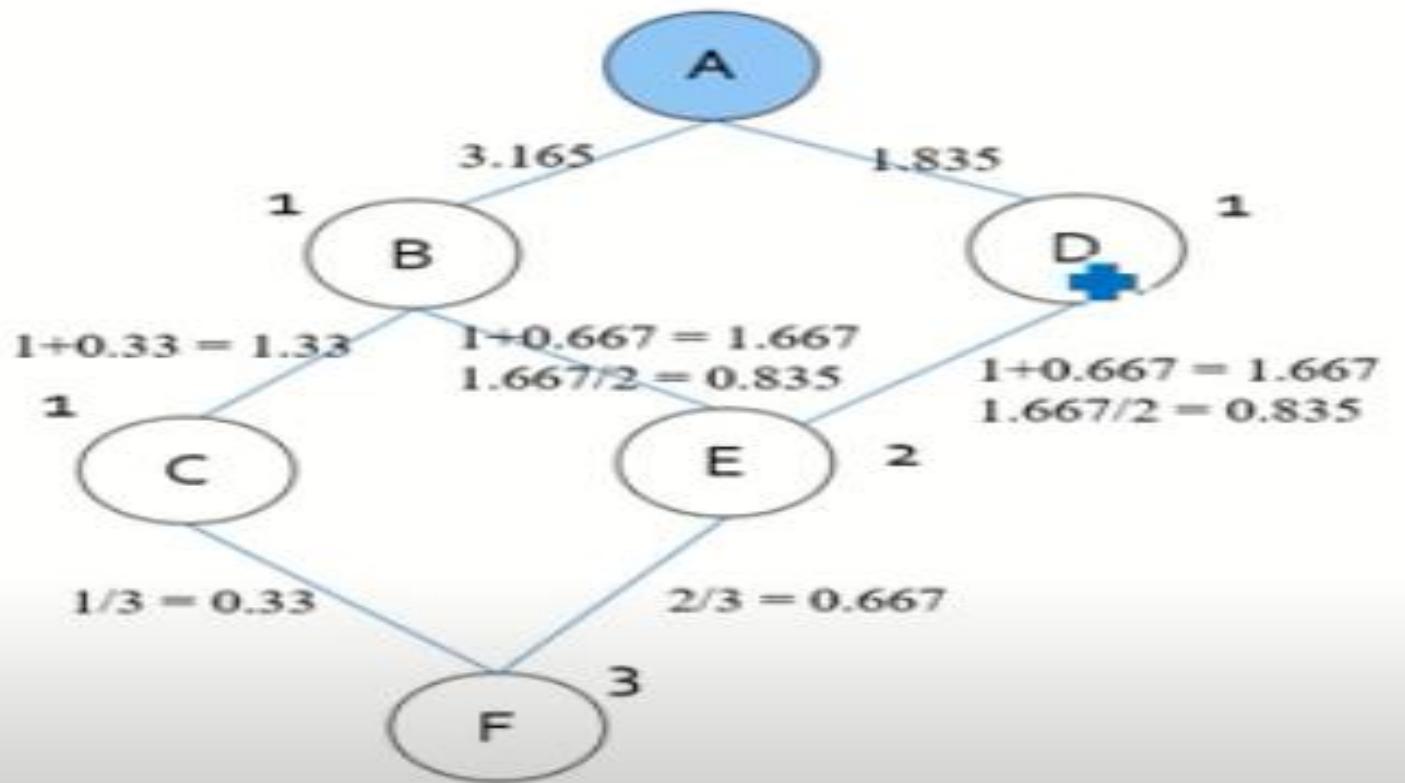




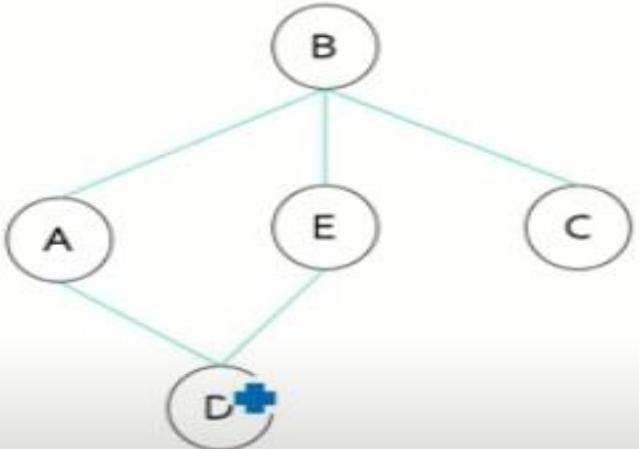
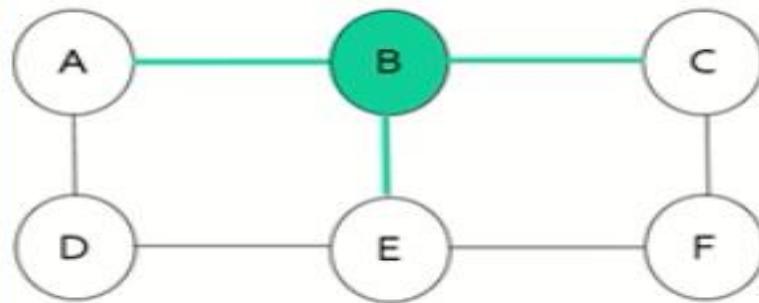


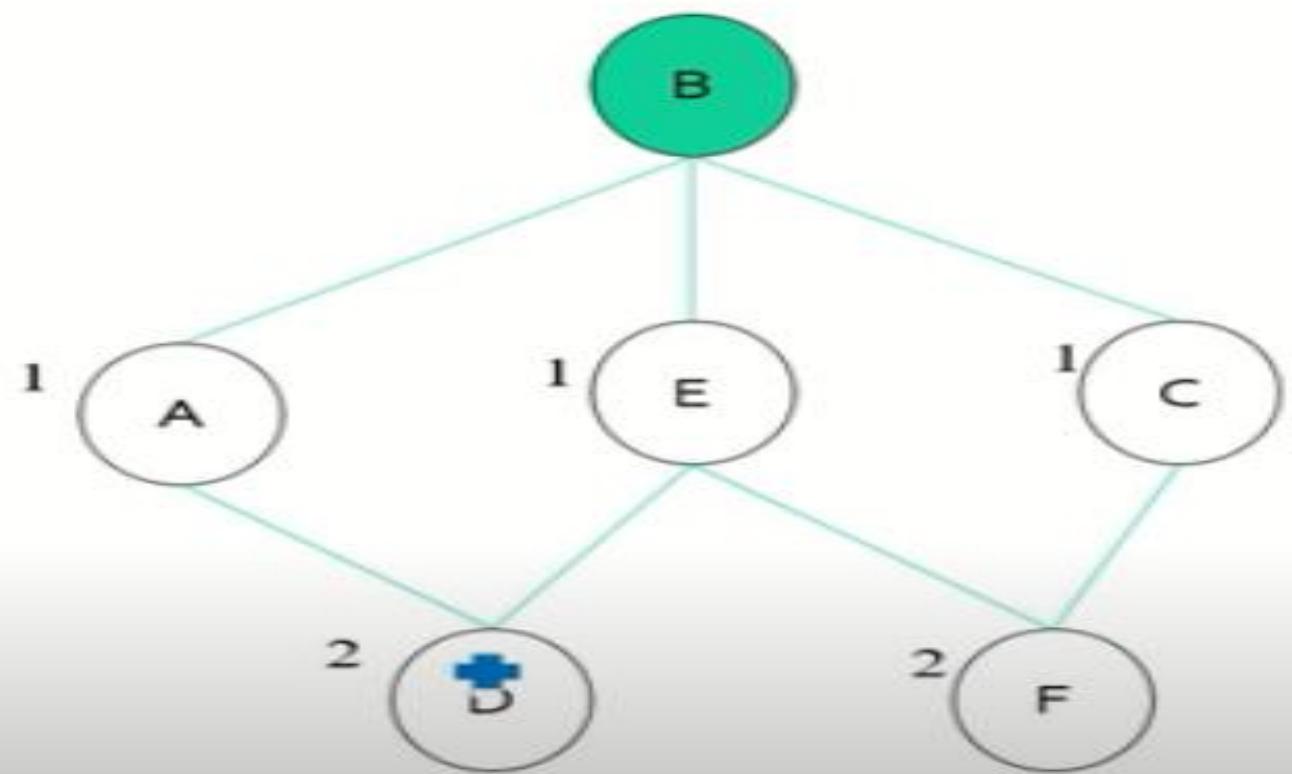


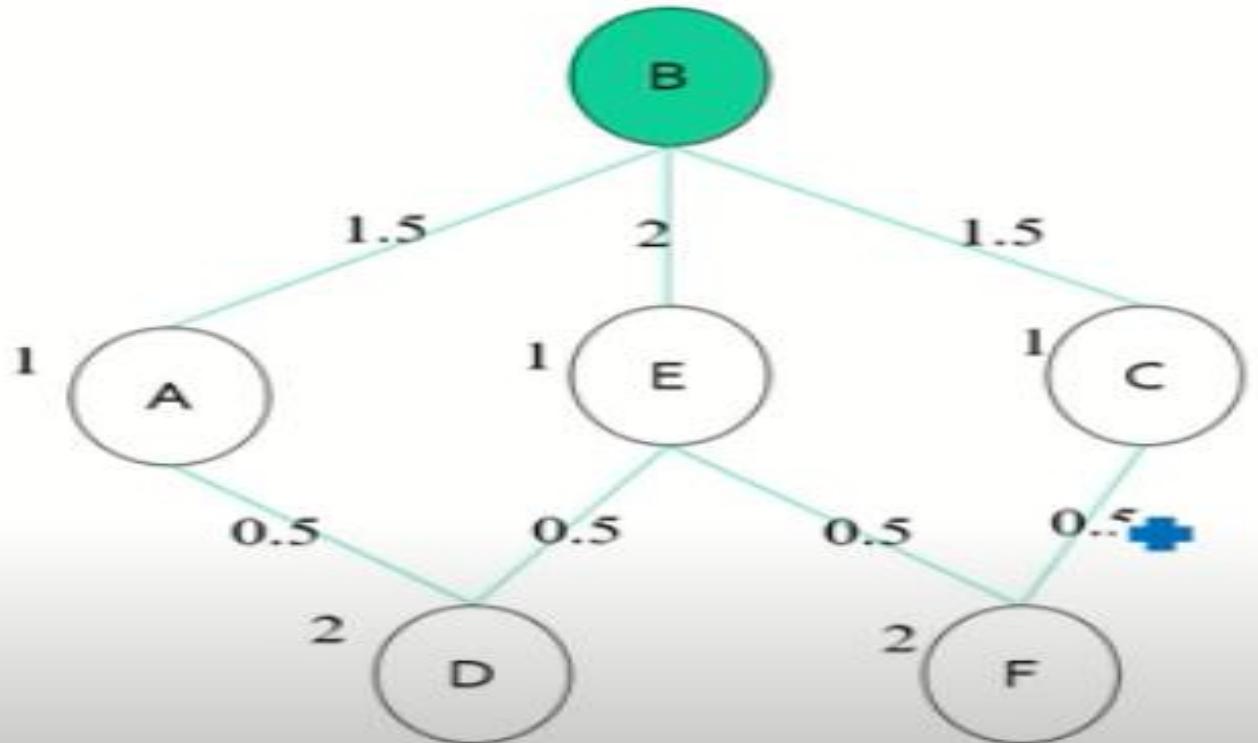




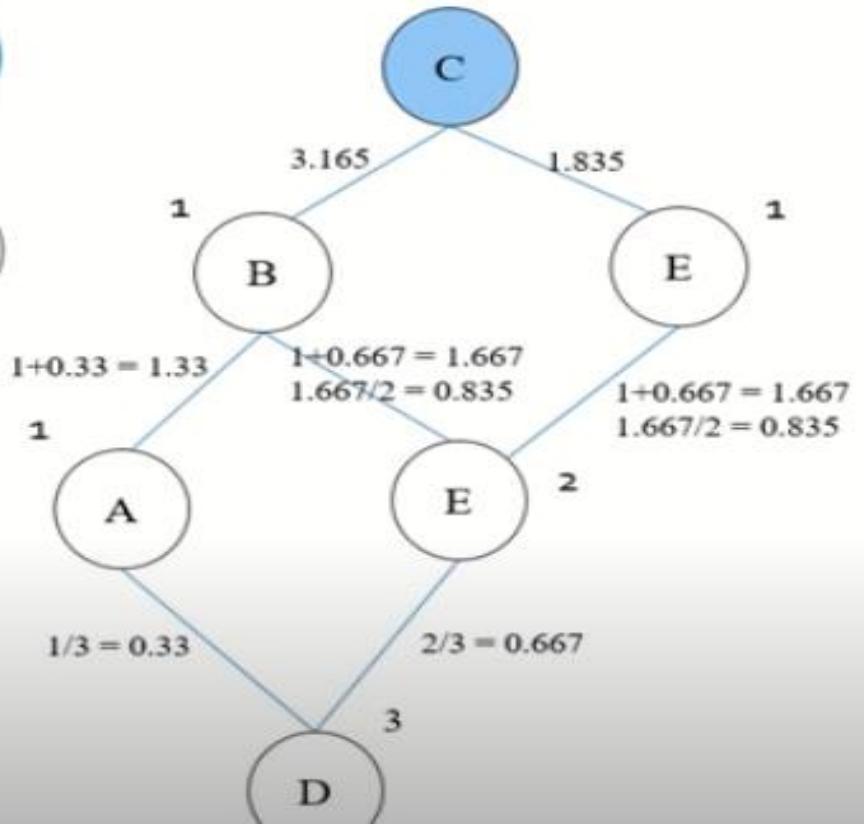
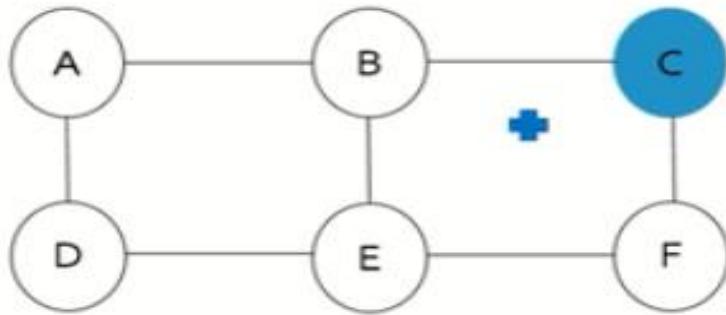
Node B



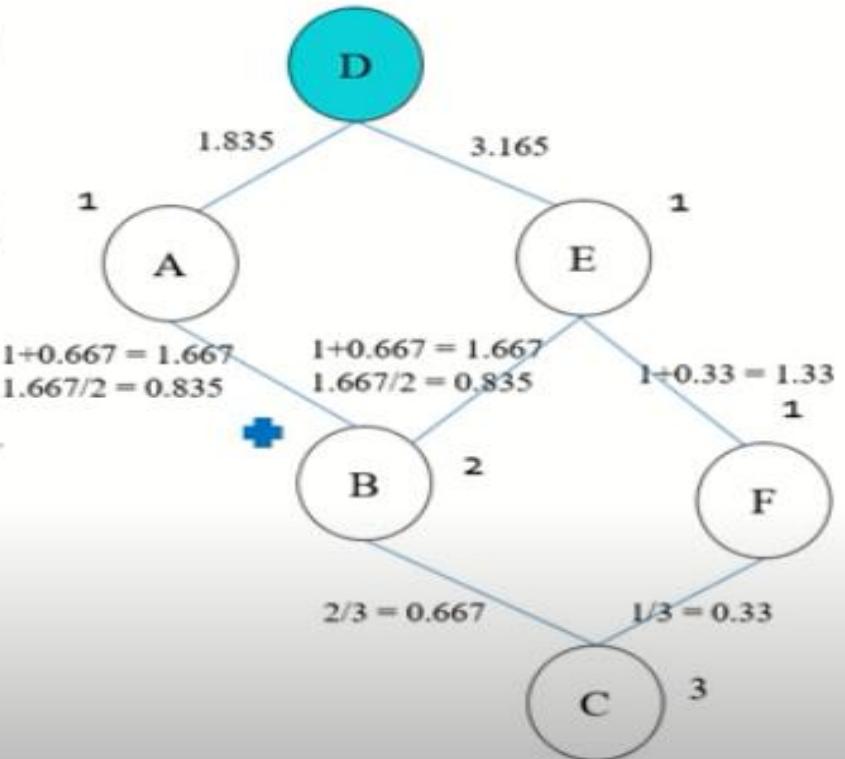
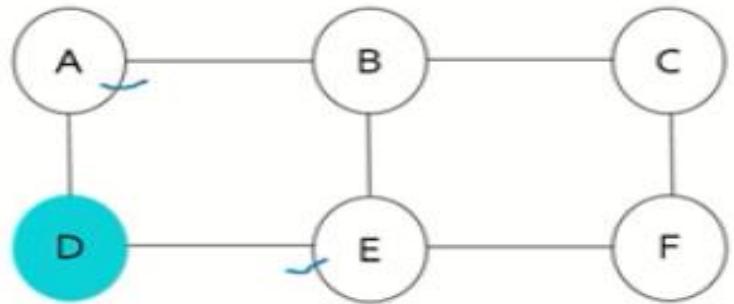




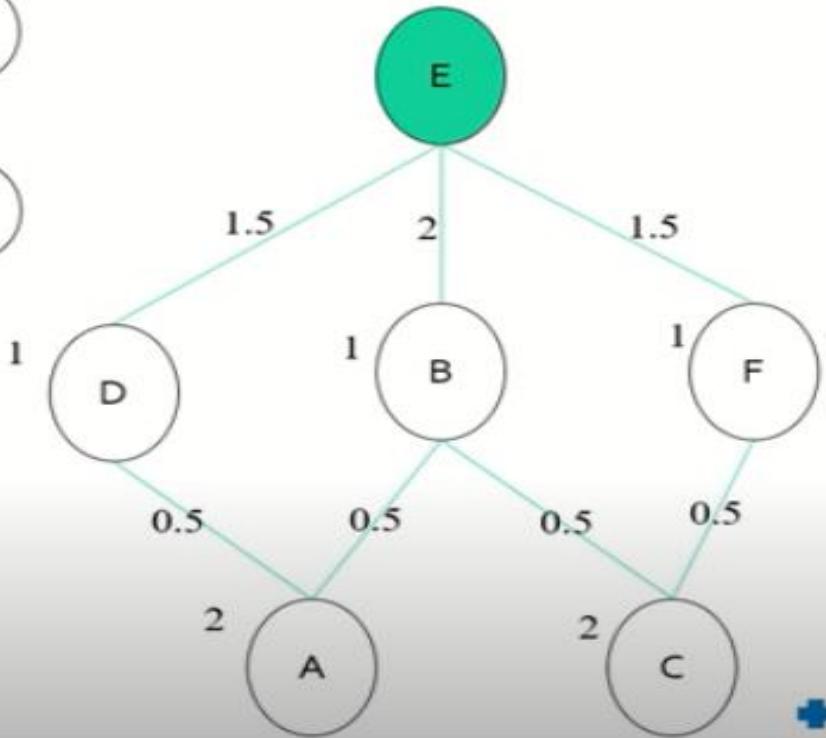
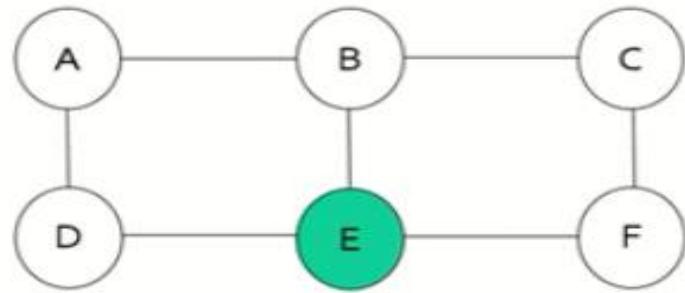
Node C



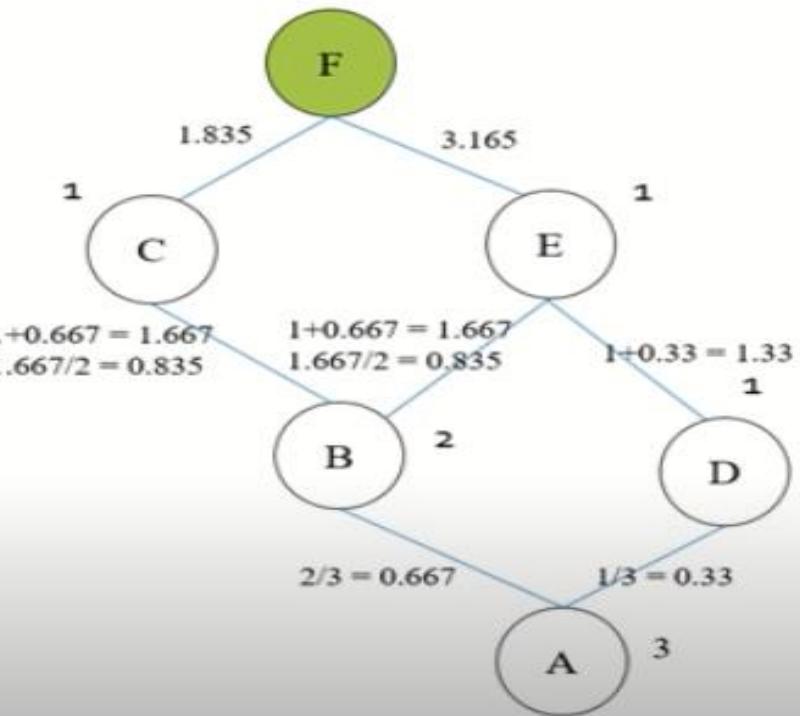
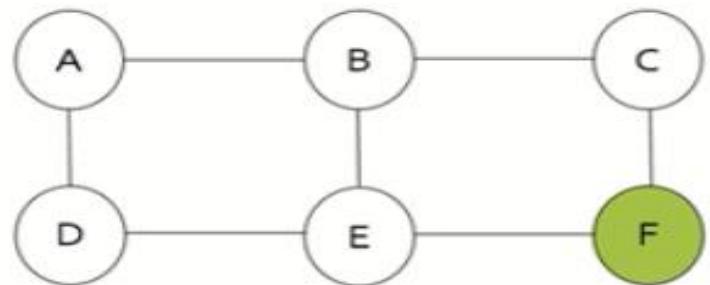
Node D



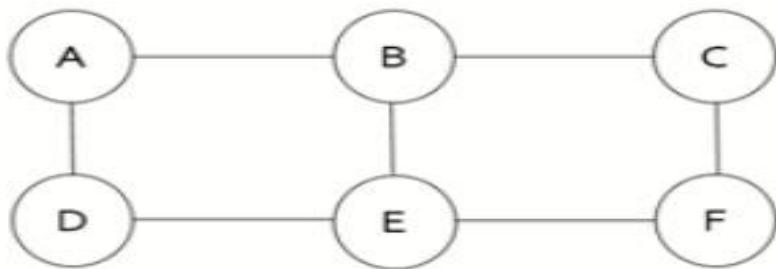
Node E



Node F

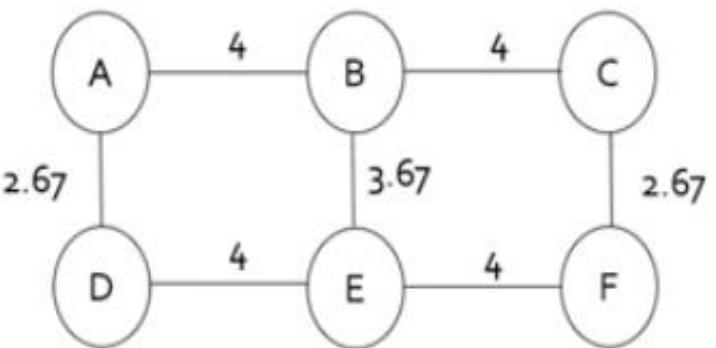
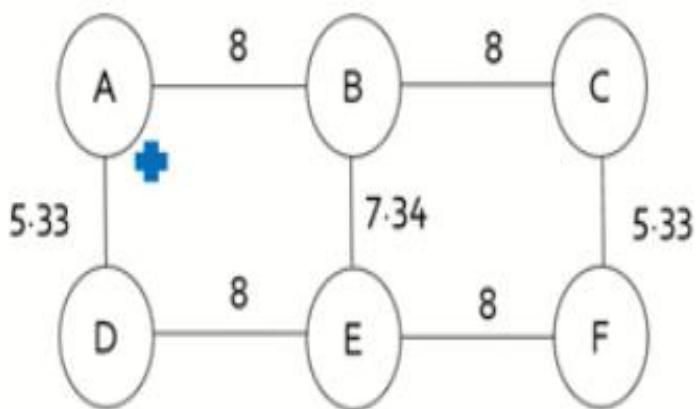


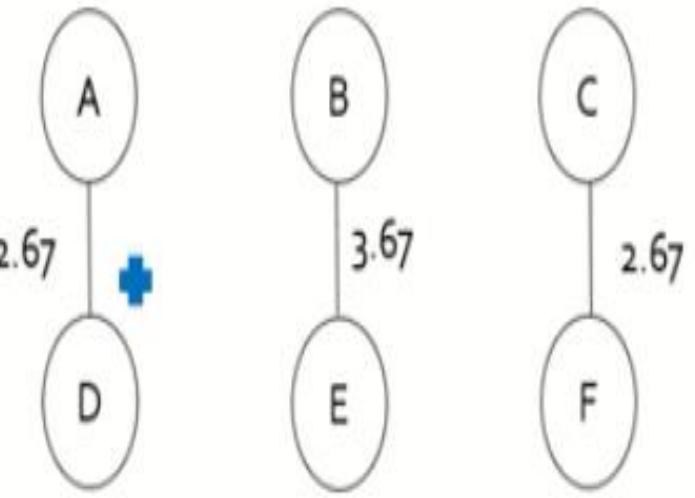
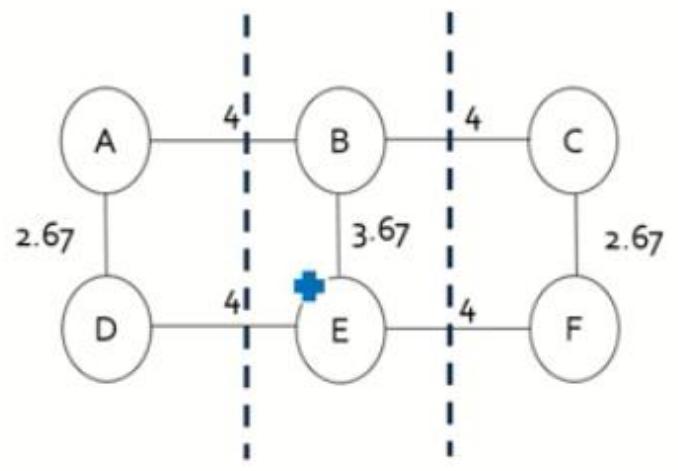
Cumulative value for Each Edge



EDGES	EDGE BETWEENNESS
AB	$3.165+1.5+1.33+0.835+0.5+0.667 = 8$
AD	$1.835+0.5+0.33+1.835+0.5+0.33 = 5.33$
BC	$3.165+1.5+1.33+0.835+0.5+0.667 = 8$
BE	$0.835+2+0.835+0.835+2+0.835 = 7.34$
CF	$1.835+0.5+0.33+1.835+0.5+0.33 = 5.33$
DE	$3.165+1.5+1.33+0.835+0.5+0.667 = 8$
EF	$3.165+1.5+1.33+0.835+0.5+0.667 = 8$

Since its undirected graph divide edge weight by 2





Betweenness Centrality

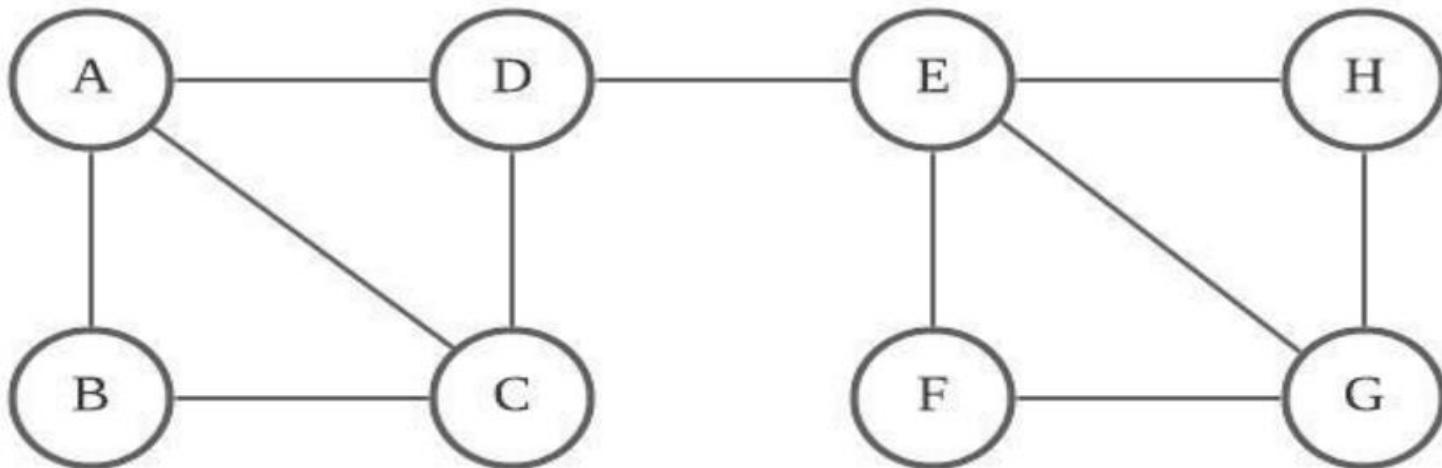
- The vertex betweenness centrality $BC(v)$ of a vertex v is defined as follow

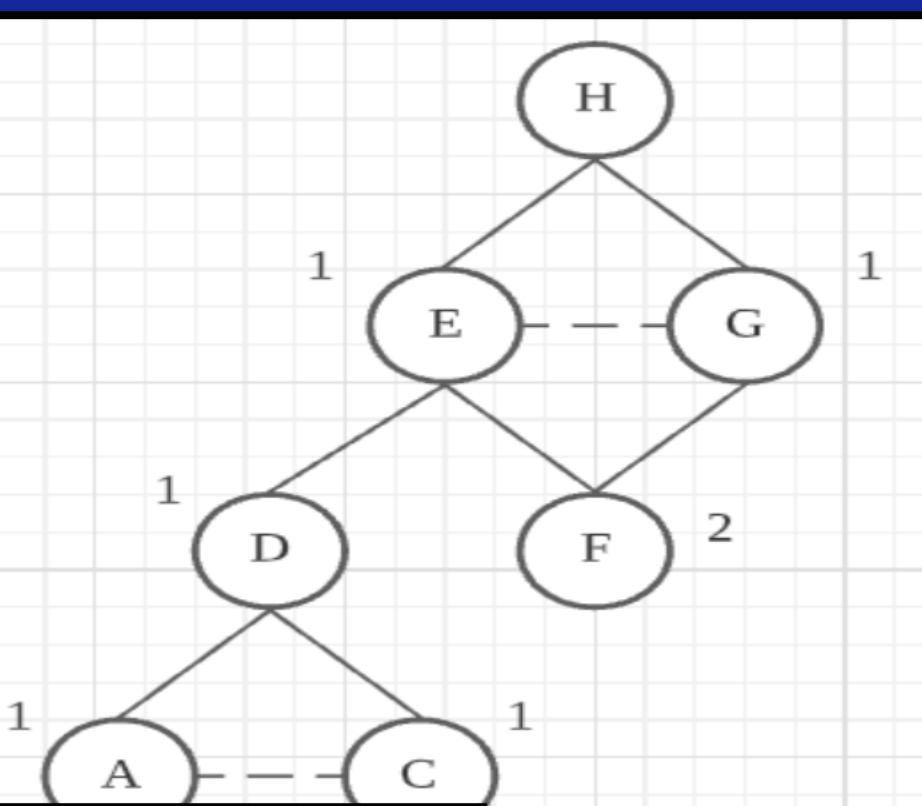
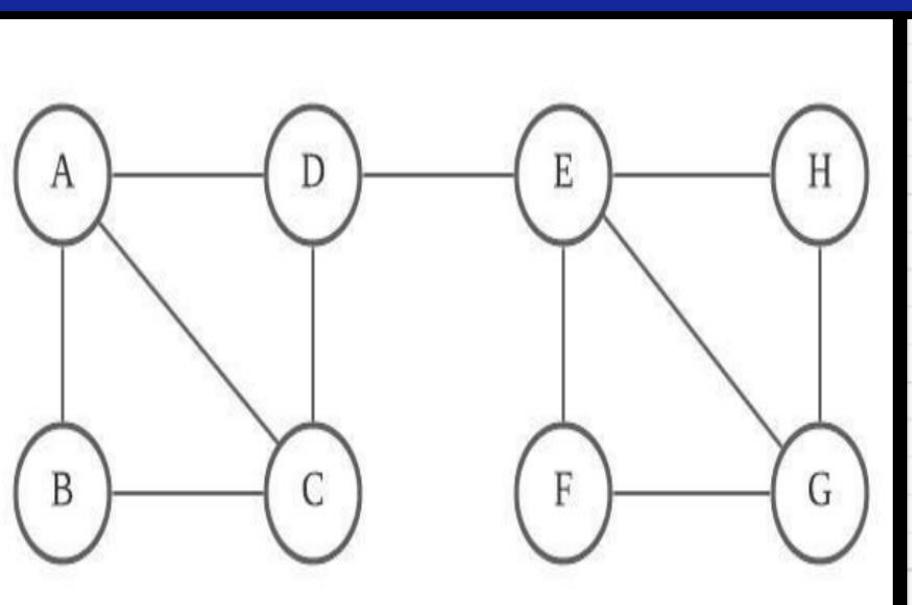
$$BC(v) = \sum_{u,w \in V} \left(\frac{\sigma_{uw}(v)}{\sigma_{uw}} \right)$$

σ_{uw} = Total number of shortest paths between node u and w.

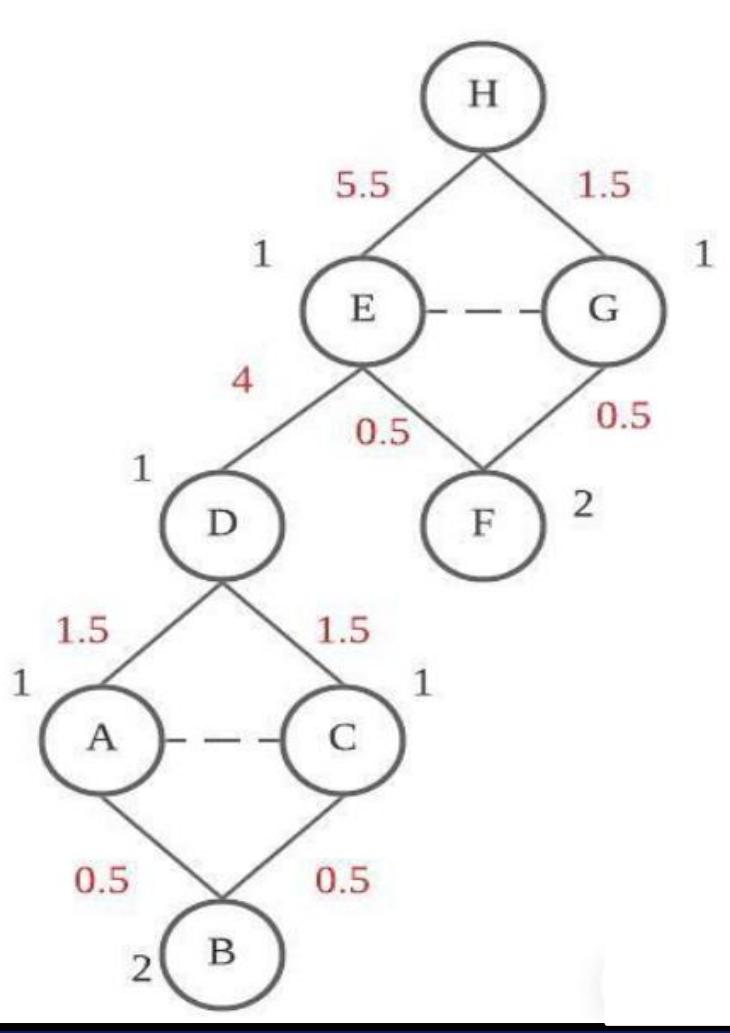
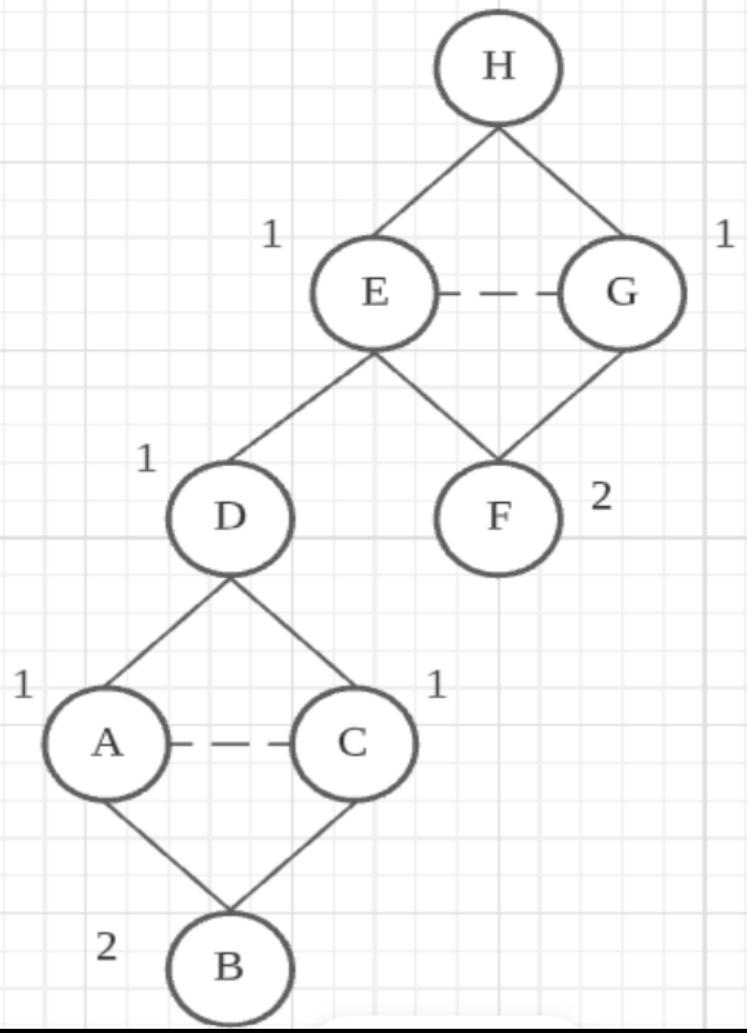
$\sigma_{uw}(v)$ = Total number of shortest paths between node u and w that pass through v.

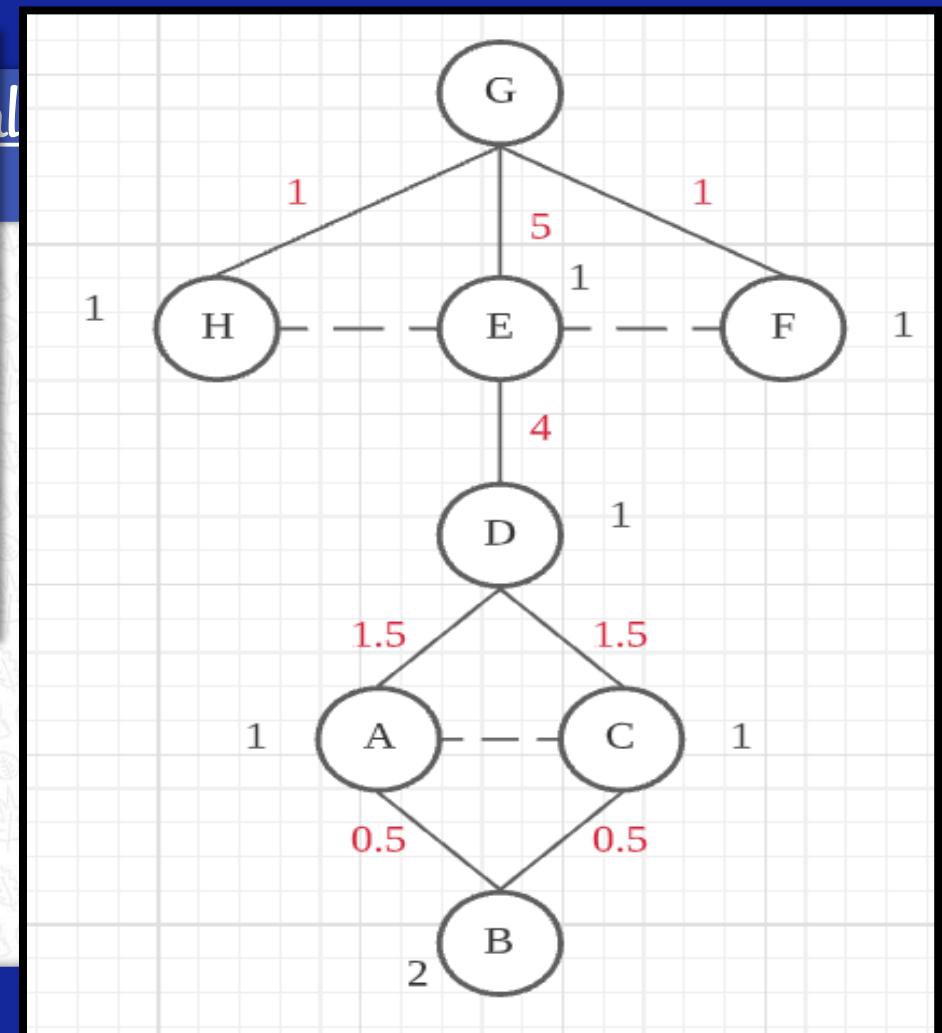
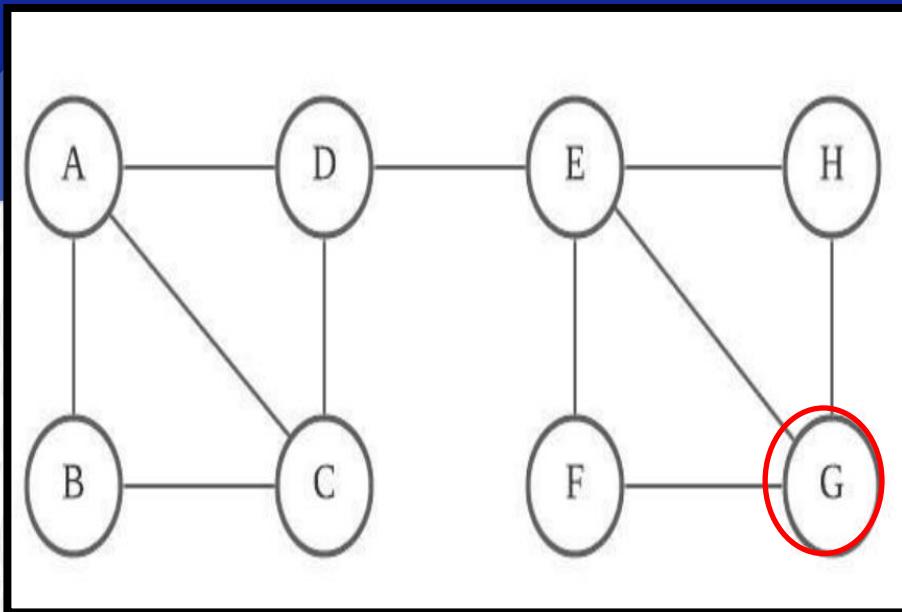
Clustering of Social-Network Graphs





$$EdgeCredit = (1 + \sum IncomingEdgeCredit) * \frac{ScoreOfDestination}{ScoreOfStart}$$





Edge Betweenness Centrality

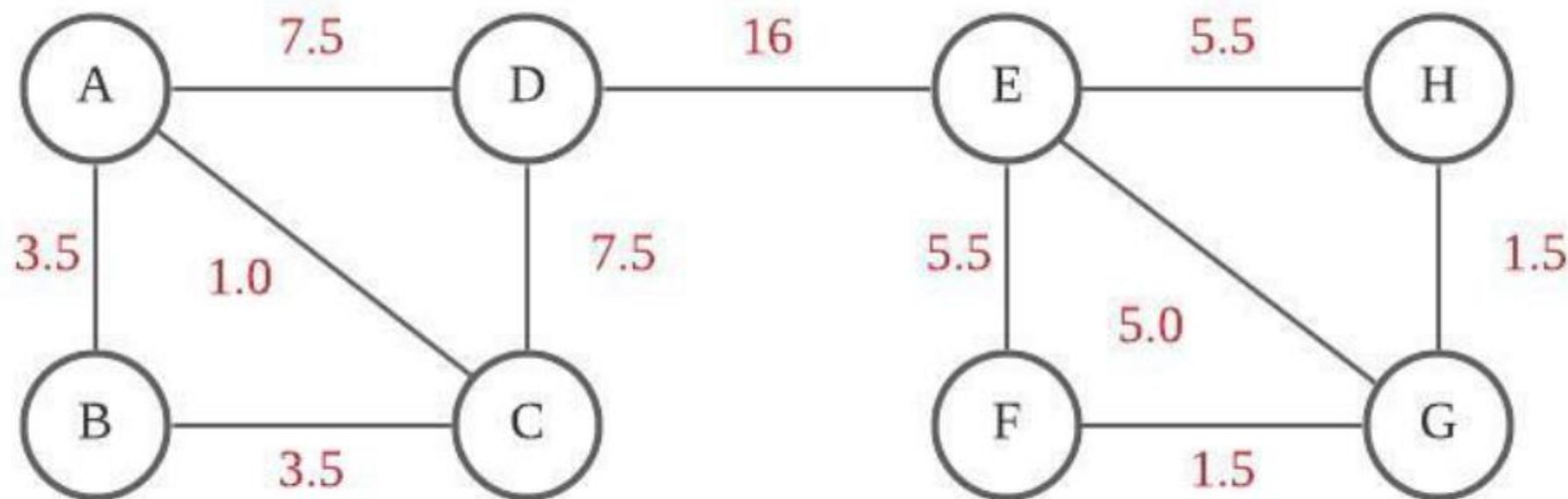
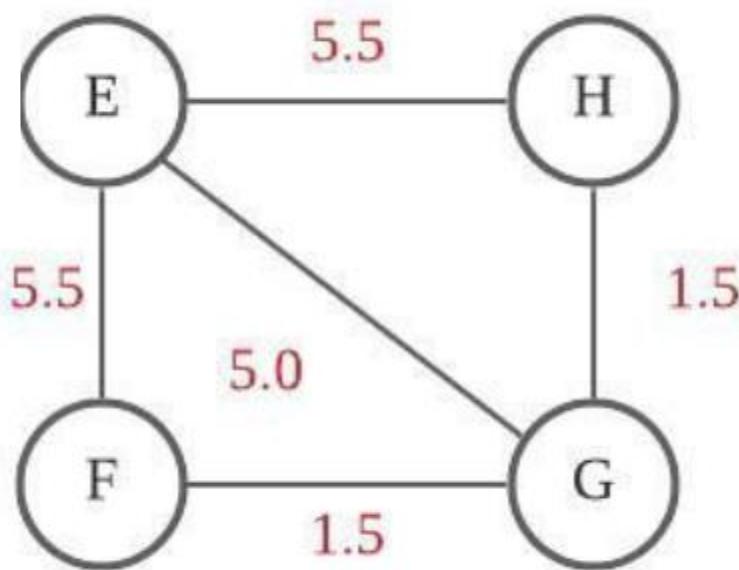
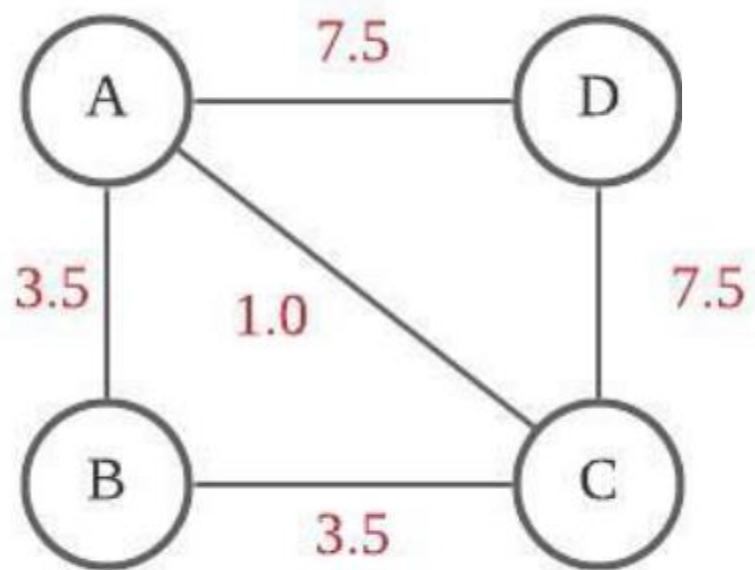


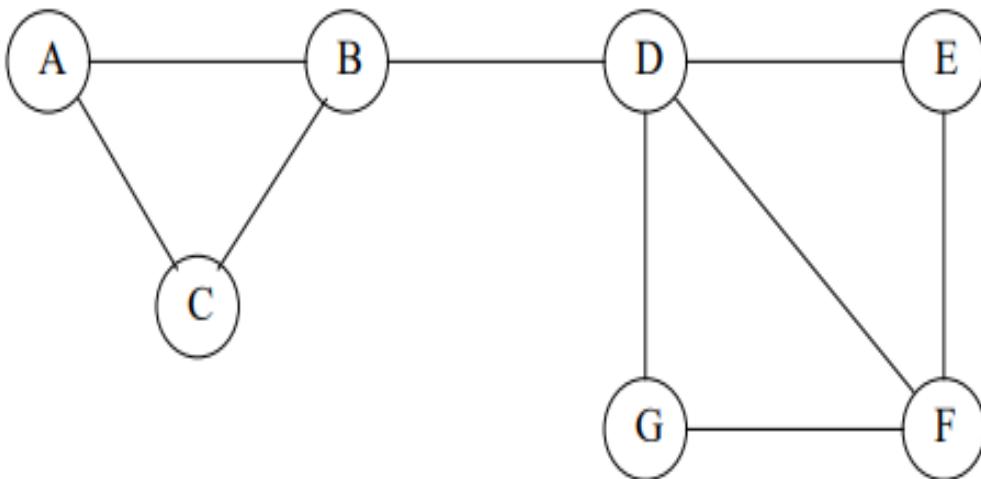
Figure 6. Edge Betweenness Centrality

Shortest Spanning Tree



Clustering of Social-Network Graphs

Example 2



Clustering of Social-Network Graphs

Example 2

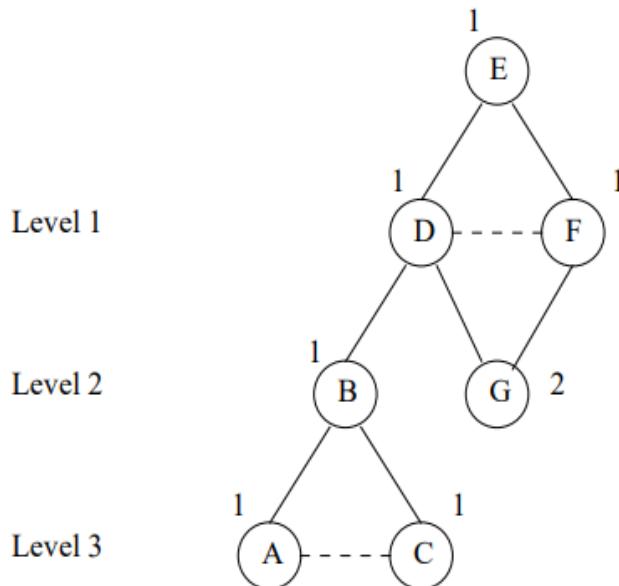
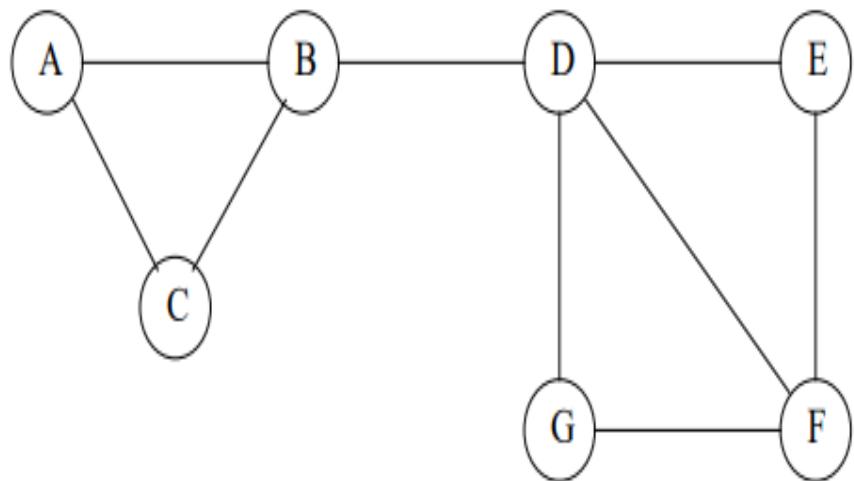


Figure 10.4: Step 1 of the Girvan-Newman Algorithm

Clustering of Social-Network Graphs

Example 2

Level 1

Level 2

Level 3

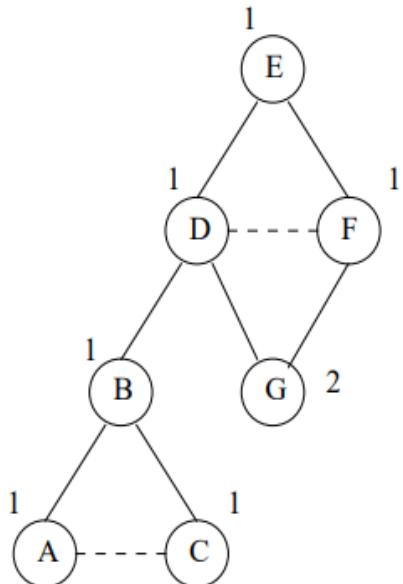
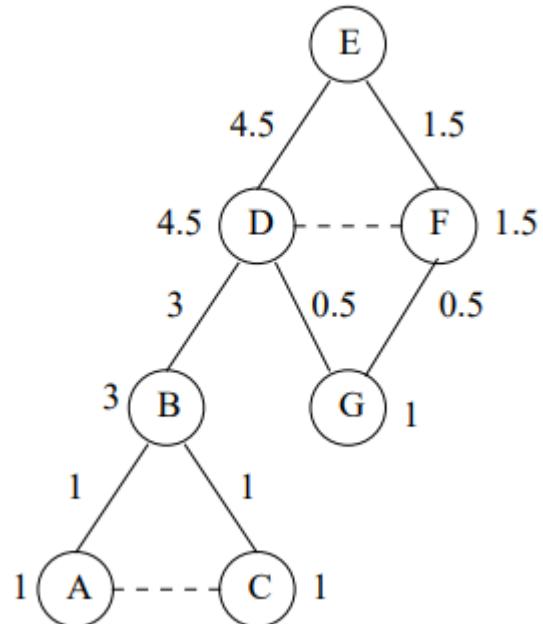


Figure 10.4: Step 1 of the Girvan-Newman Algorithm



Clustering of Social-Network Graphs

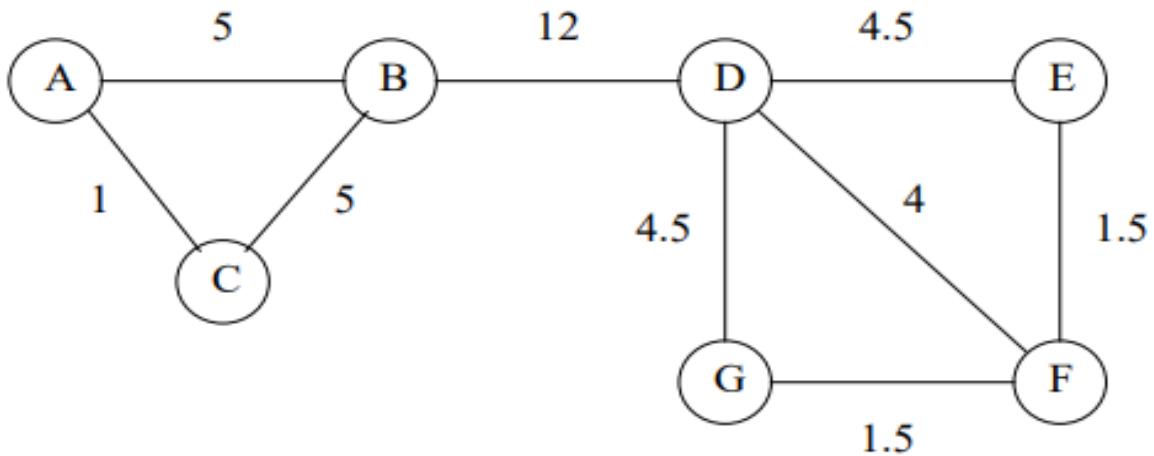


Figure 10.7: Betweenness scores for the graph of Fig. 10.1

Clustering of Social-Network Graphs

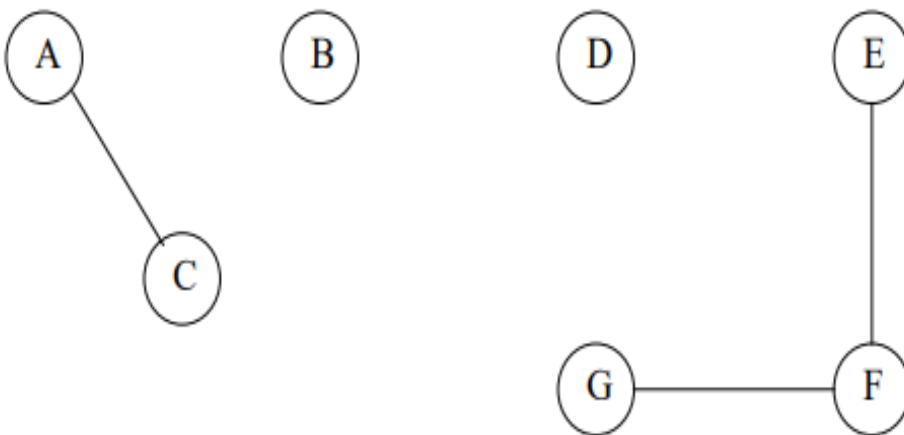


Figure 10.8: All the edges with betweenness 4 or more have been removed

→ Topics to be Discussed

- ✓ 5.1.1 A model for Recommendation systems
- ✓ 5.1.2 Content Based Recommendations
- ✓ 5.1.3 Collaborative Filtering

- ✓ 5.2.1 Case Study :Product Recommendation

- ✓ 5.3.1 Social Networks as Graphs,
- ✓ 5.3.2 Clustering of Social-Network Graphs
- 5.3.3 Direct Discovery of Communities in a social graph.

Direct Discovery of Communities in a social graph

In this section, we shall see a technique for discovering communities directly by looking for subsets of the nodes that have a relatively large number of edges among them.

Direct Discovery of Communities in a Social Graph

Direct Discovery of Communities in a Social Graph

- We can classify communities in a social network as a group of entities which are closely knit and can belong strictly to a single community or can belong to more than one community.
- Single community – Disjoint
- More than one – Overlapping

Direct Discovery of Communities in a Social Graph

- Overlapping Community –is more natural
- How to identity community ??
- Method for community Detection :
 - Clique Percolation Method (CPM)

Clique Percolation Method (CPM)

What is CPM?

- Method to find overlapping communities
- Based on concept:
 - internal edges of community likely to form cliques
 - Intercommunity edges unlikely to form cliques

Why to find social groups and communities?

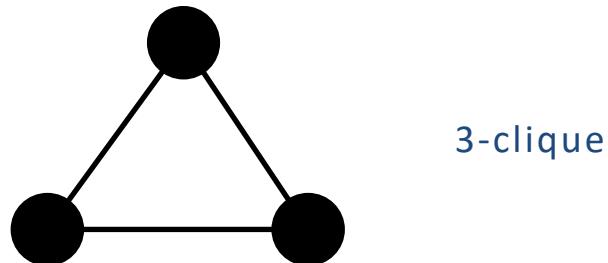
- behavior analysis
- location-based interaction analysis
- recommender systems development
- link prediction
- customer interaction and analysis & marketing
- media use
- Security
- Social studies

Clique

- Clique: Complete graph
- k-clique: Complete graph with k vertices

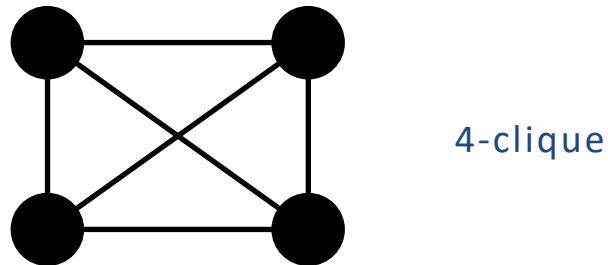
Clique

- Clique: Complete graph
- k -clique: Complete graph with k vertices



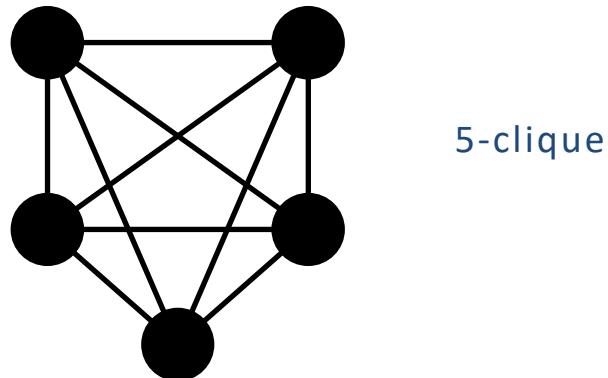
Clique

- Clique: Complete graph
- k -clique: Complete graph with k vertices



Clique

- Clique: Complete graph
- k -clique: Complete graph with k vertices



k-Clique Communities

- **Adjacent k-cliques**

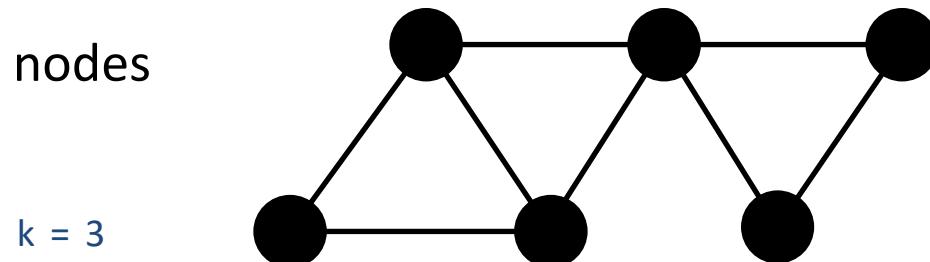
Two k-cliques are adjacent when they share **k-1**

nodes

k-Clique Communities

- **Adjacent k-cliques**

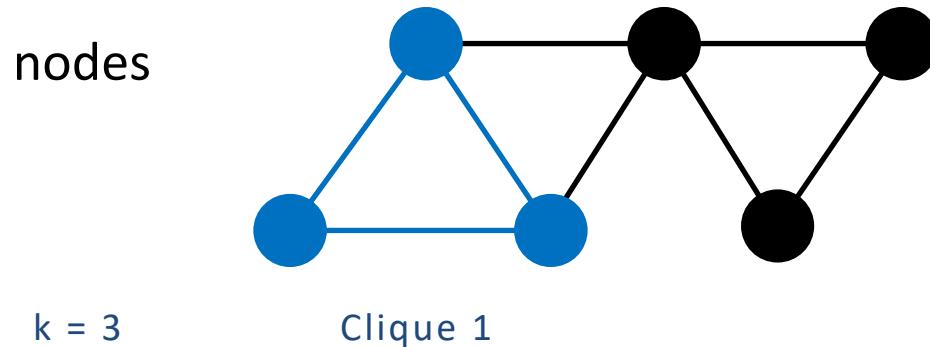
Two k-cliques are adjacent when they share k-1



k -Clique Communities

- **Adjacent k -cliques**

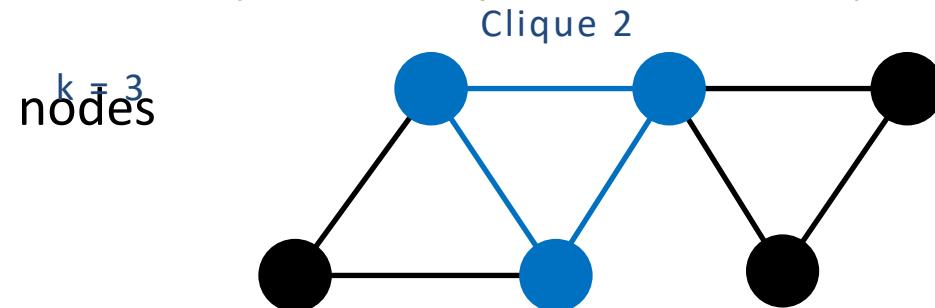
Two k -cliques are adjacent when they share **$k-1$**



k-Clique Communities

- Adjacent k-cliques

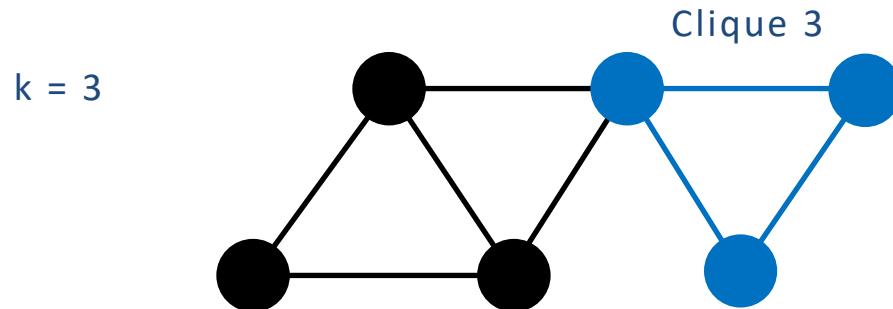
Two k-cliques are adjacent when they share k-1



k -Clique Communities

- **Adjacent k -cliques**

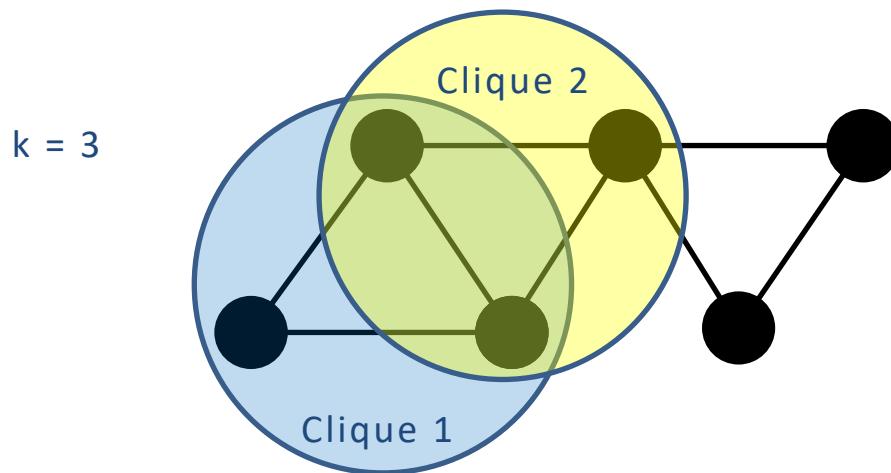
Two k -cliques are adjacent when they share $k-1$ nodes



k -Clique Communities

- **Adjacent k -cliques**

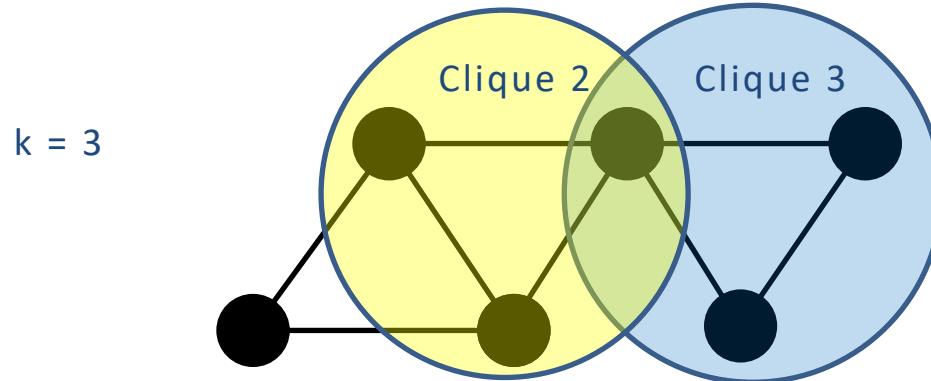
Two k -cliques are adjacent when they share $k-1$ nodes



k -Clique Communities

- **Adjacent k -cliques**

Two k -cliques are adjacent when they share $k-1$ nodes



k-Clique Communities

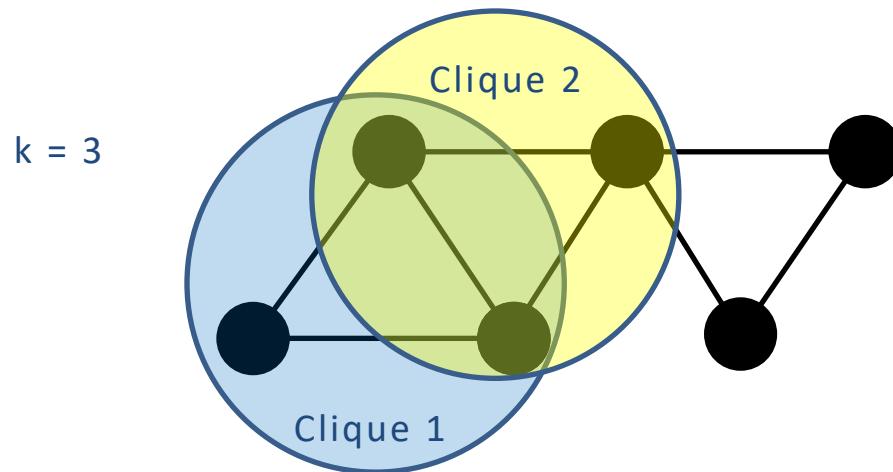
- **k-clique community**

Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques

k-Clique Communities

- **k-clique community**

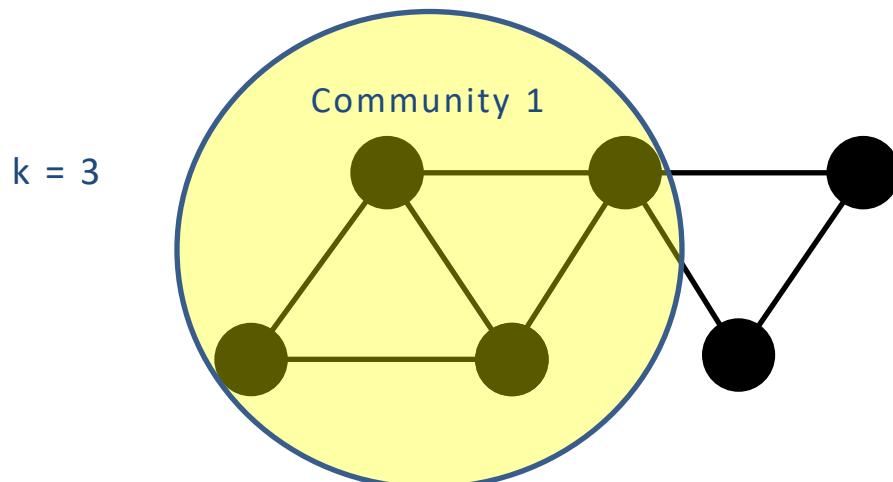
Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques



k-Clique Communities

- **k-clique community**

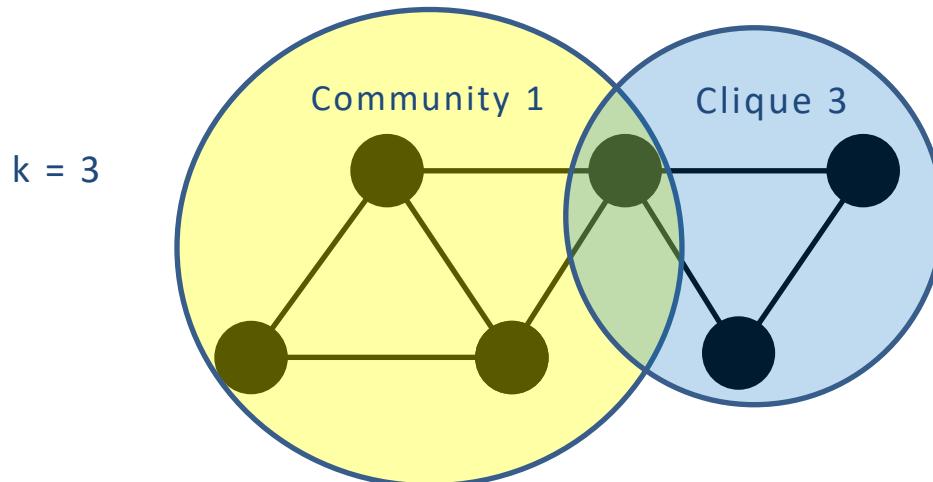
Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques



k-Clique Communities

- **k-clique community**

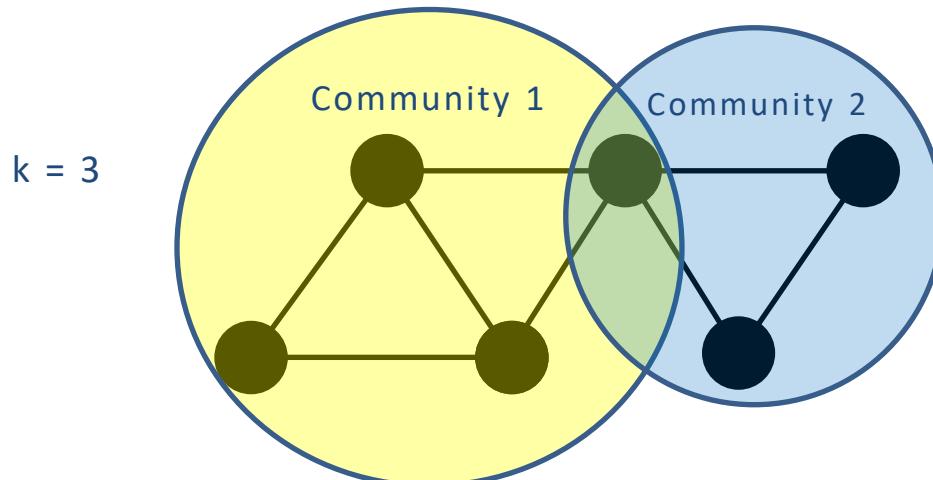
Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques



k-Clique Communities

- **k-clique community**

Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques



CPM Algorithm

- **Input** :- The social graph G , representing a network and a clique size k .
- **Output** : Set of discovered Communities C
- **Step1** : All k -clique present in G are extracted
- **Step 2:** A new graph , the clique graph , G_c formed where each node represented an identified clique and two vertices in G_c are connected by an edge ,if they have $k-1$ common vertices.
- **Step 3:** Connected components in G_c are identified
- **Step 4:** Each connected component in G_c represents a community.
- **Step 5:** Set C be the set of communities formed for G .

CPM Algorithm in short

- Locate maximal cliques
 - Largest possible clique size can be determined from degrees of vertices
 - Starting from this size, find all cliques, then reduce size by 1 and repeat
- Convert from cliques to k-clique communities

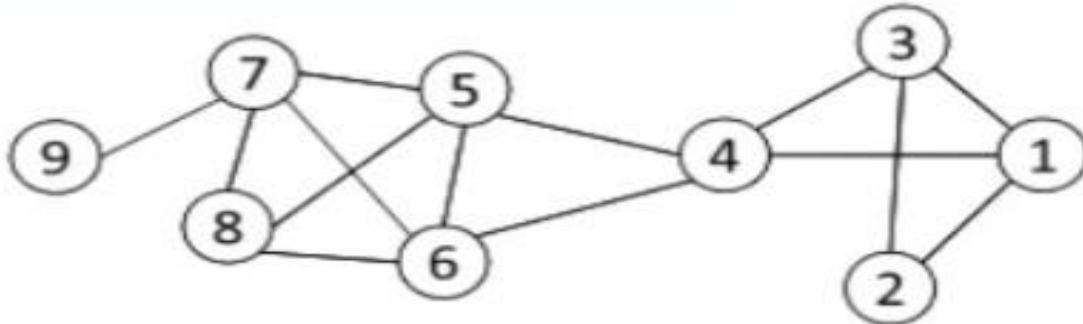
CPM

– Procedure

1. Find out all cliques of size k in a given network
2. Construct a clique graph. Two cliques are adjacent if they share $k-1$ nodes
3. Each connected components in the clique graph form a community

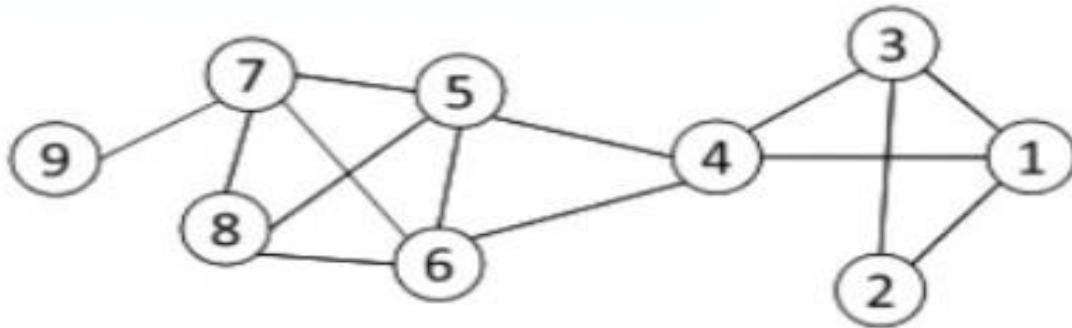
Example: Clique Percolation Method

Step 1: Find all Cliques of size 3



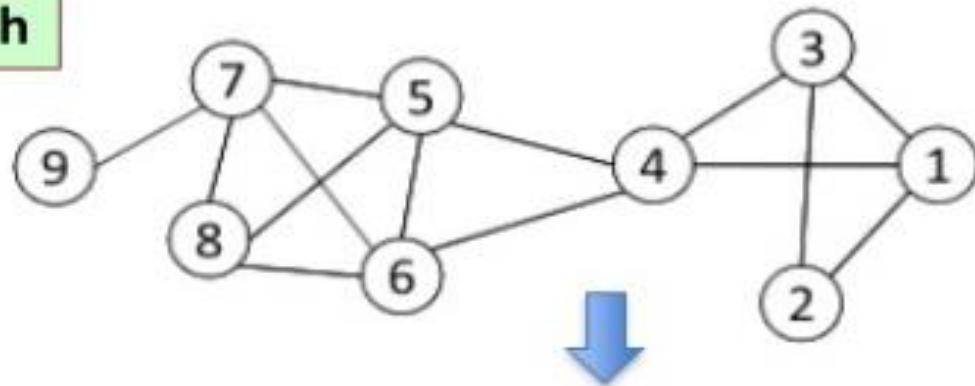
Example: Clique Percolation Method

Step 1: Find all Cliques of size 3



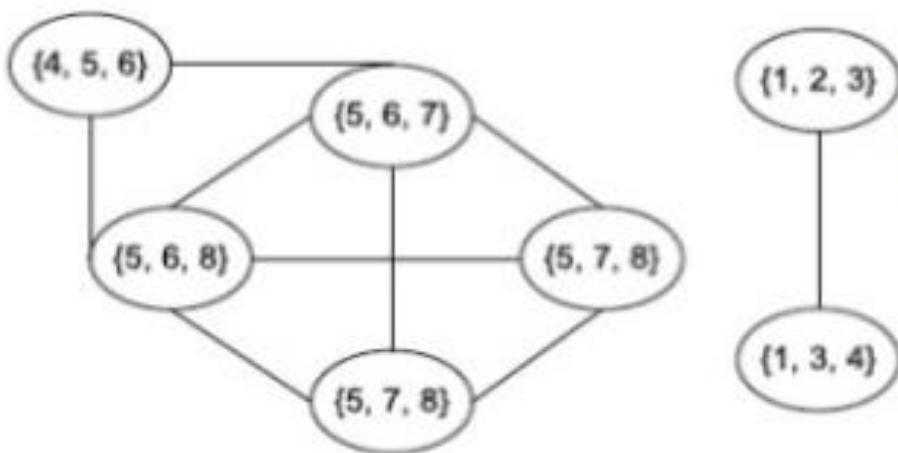
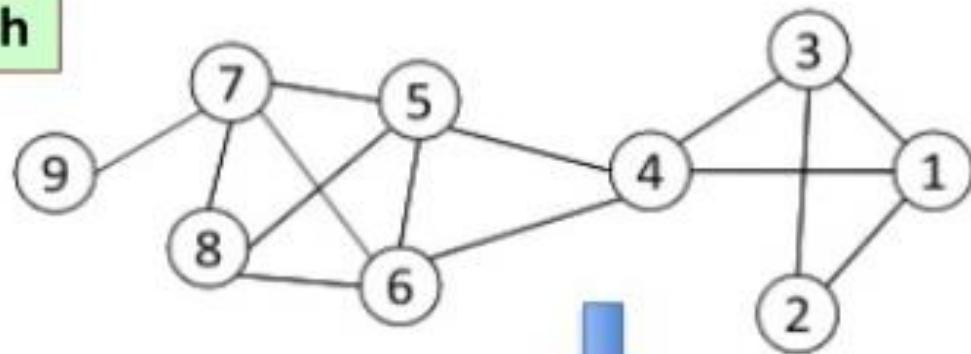
{1, 2, 3}, {1, 3, 4}, {4, 5, 6}, {5, 6, 7}, {5, 6, 8}, {5, 7, 8},
{6, 7, 8}

Step 2: Construct Clique Graph



{1, 2, 3}, {1, 3, 4}, {4, 5, 6},
{5, 6, 7}, {5, 6, 8}, {5, 7, 8},
{6, 7, 8}

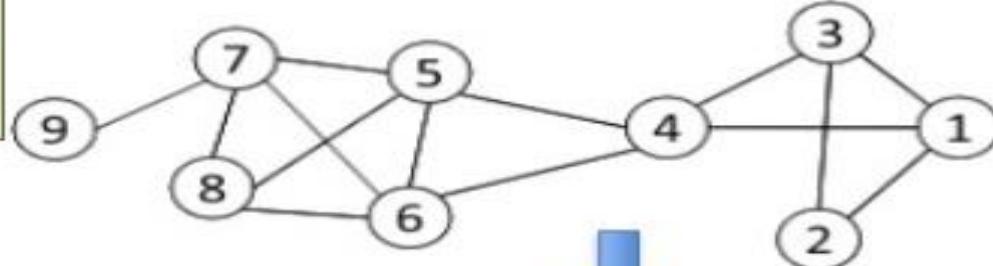
Step 2: Construct Clique Graph



$\{1, 2, 3\}, \{1, 3, 4\}, \{4, 5, 6\},$
 $\{5, 6, 7\}, \{5, 6, 8\}, \{5, 7, 8\},$
 $\{6, 7, 8\}$

Step 3: Finding Communities

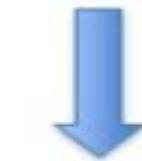
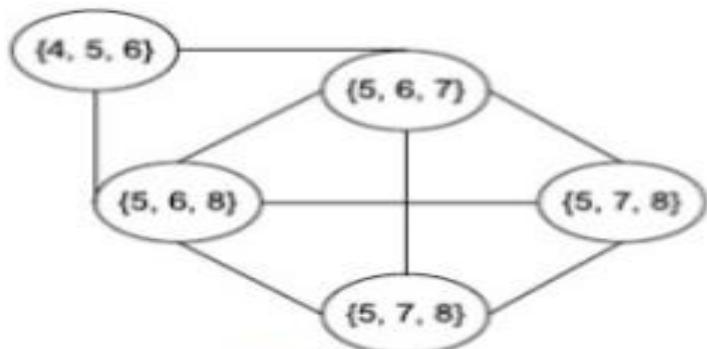
Two cliques are adjacent if they share $k-1$ nodes (i.e. $k-1=2$)



{1, 2, 3}

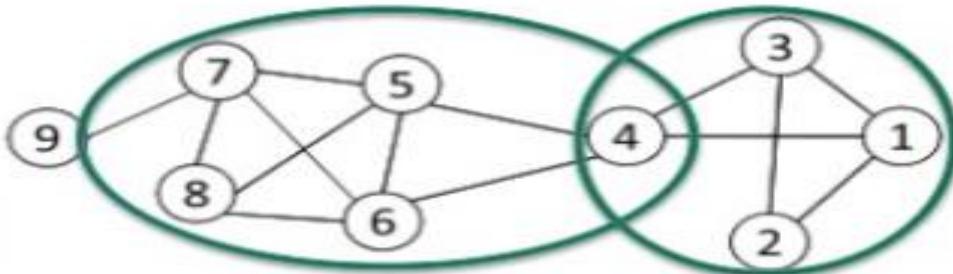
{1, 3, 4}

{1, 2, 3}, {1, 3, 4}, {4, 5, 6},
{5, 6, 7}, {5, 6, 8}, {5, 7, 8},
{6, 7, 8}



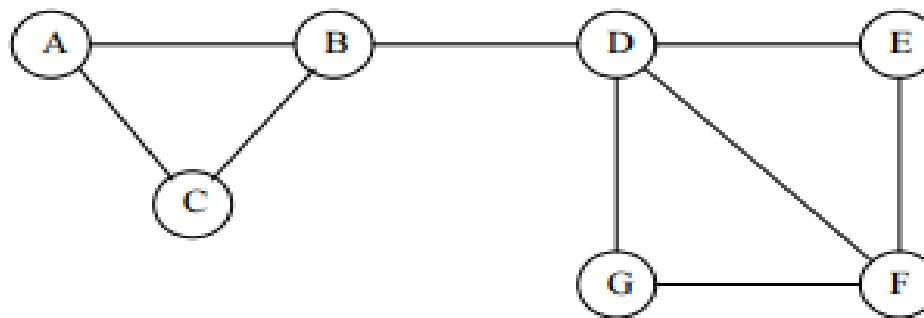
Communities:

{1, 2, 3, 4}
{4, 5, 6, 7, 8}



Example 1

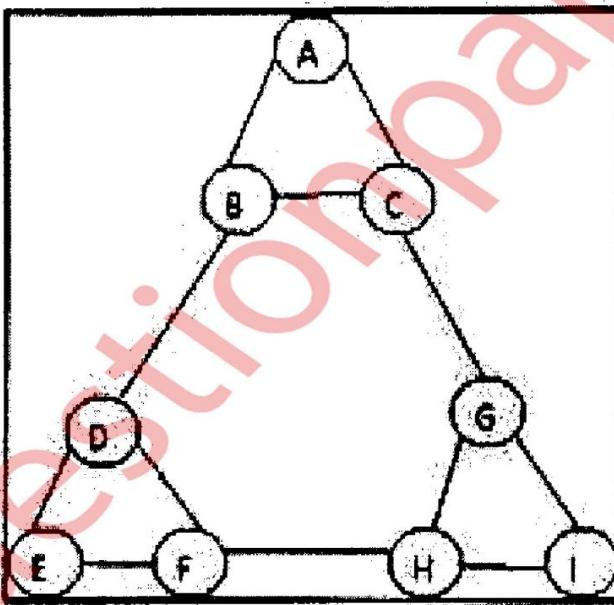
- Find all cliques for the given graph using CPM showing all steps.



Exercise1: MU COMP May2016

(b) For the graph given below use Clique percolation and find all communities

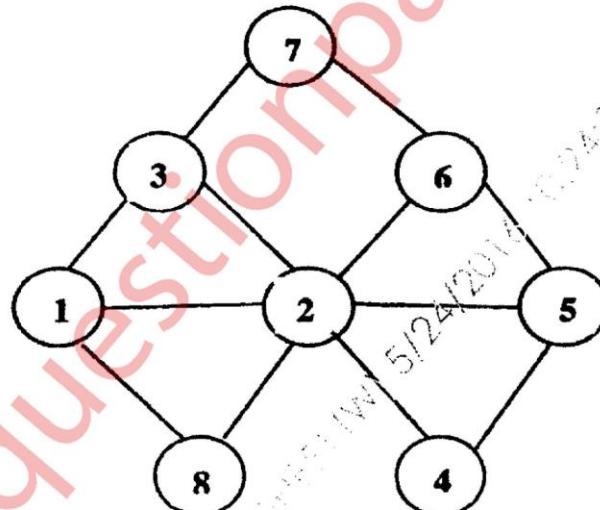
10



MUPD16443 Sept Paper
MAY

Exercise 2: MU IT May2016

- b) For following graph, show how the clique percolation method (CPM) find cliques. 10
· Explain with steps?



CPM-A Quick Glance

- The **clique percolation method** is a popular approach for analyzing the overlapping community structure of networks.

Applications

1. CPM had been used to detect communities from the studies of cancer metastasis through various social networks
2. Document clustering

→ Topics to be Discussed

- ✓ 5.1.1 A model for Recommendation systems
- ✓ 5.1.2 Content Based Recommendations
- ✓ 5.1.3 Collaborative Filtering

- ✓ 5.2.1 Case Study :Product Recommendation

- ✓ 5.3.1 Social Networks as Graphs,
- ✓ 5.3.2 Clustering of Social-Network Graphs
- ✓ 5.3.3 Direct Discovery of Communities in a social graph.



To be good, and to
do good, is all
we have to do.