5.1 INTRODUCTION

5.1.1 What is Social Media Mining?

Social media bridges the gap between the real and virtual worlds, allowing us to analyze human interactions and community formations using computational techniques. This field involves integrating social theories with computational methods to study how individuals (referred to as **social atoms**) interact and how **communities (social molecules)** are formed.

Social media mining is the process of analyzing data from platforms like **Facebook, Twitter, Instagram, and YouTube** to find useful information. Businesses, researchers, and governments use this data to understand trends, opinions, and human behavior.

👉 **Example 1: Understanding Customer Opinions**
Imagine a company like **Nike** wants to know what people think about its new shoes. Instead of manually reading thousands of comments, social media mining can be used to **analyze tweets and reviews**. It can automatically detect:

- Positive comments (e.g., "I love the new Nike Air Max! So comfortable! 😍")
- Negative comments (e.g., "The quality of this shoe is terrible! 😡")
- Common complaints (e.g., "Too expensive!" or "Sizes are not accurate.")

By studying this data, Nike can improve its products and marketing strategies.

Key Aspects of Social Media Mining:

1. **Handling User-Generated Content:**
    - Social media contains massive amounts of unstructured data (text, images, videos, etc.).
    - Unlike traditional structured data, this content requires specialized techniques for analysis.
2. **Interdisciplinary Nature:**
    - Social media mining combines knowledge from various disciplines:
        - **Computer Science:** Algorithms and computational tools.
        - **Data Mining & Machine Learning:** Extracting patterns and making predictions.
        - **Social Network Analysis & Network Science:** Studying relationships and structures in networks.
        - **Sociology & Ethnography:** Understanding human behavior and social structures.
        - **Statistics & Mathematics:** Quantitative analysis and modeling.

3. **Applications of Social Media Mining:**
    - Sentiment analysis (understanding opinions on social media).
    - Trend prediction (predicting viral topics).
    - Fake news detection.
    - Personalized recommendations (suggesting content to users).
    - **Sentiment Analysis** – Understanding people's emotions in posts and comments.
        🔹 **Example:** A restaurant analyzes **Google reviews** to see if customers are happy or unhappy. If many customers complain about bad service, the restaurant can take action.
    - **Trend Detection** – Finding what is currently popular on social media.
        🔹 **Example:** Twitter uses **hashtags (#)** to track trending topics. If many people tweet **#NewiPhone**, it means the iPhone launch is trending.
    - **Fake News Detection** – Identifying false or misleading information.
        🔹 **Example:** Facebook uses AI to detect and remove **fake news articles** about elections or health issues (e.g., "Drinking lemon juice cures COVID-19" ❌).
    - **Influencer Identification** – Finding important people who can promote brands.
        🔹 **Example:** A fashion company looks at **Instagram data** to find influencers with many followers who talk about fashion. They might find that **@FashionQueen** has **1 million followers**, making her a great choice for brand promotion.
    - **Recommendation Systems** – Suggesting products or content based on user behavior.
        🔹 **Example:** Netflix analyzes what movies you watch and suggests similar movies (e.g., "Because you watched **Avengers**, you might like **Iron Man**").

Role of Data Scientists in Social Media Mining:

- Social media mining has created a new type of **data scientist** who:
    - Understands both social theories and computational techniques.
    - Specializes in analyzing complex social media data.
    - Bridges the gap between theoretical knowledge and practical insights.

5.2.2 New Challenges for Mining Social Media Data

Social media mining is still an emerging field, and there are more challenges than ready-made solutions. Some major challenges include:

1. Big Data Paradox

- Social media generates **huge** amounts of data.
- However, when zooming in on individual users (e.g., making personalized recommendations), **data for each person is limited**.
- There is **too much** social media data, but sometimes not enough specific data for individual users.
- 🔹 **Example:** Amazon wants to recommend products, but if a new user has never shopped before, there is little data about them
- Solution: Use data aggregation techniques to combine information from multiple sources.

2. Obtaining Sufficient Samples

- Social media platforms offer **limited** data access through APIs.
- A major challenge is determining **whether the collected data represents the full population**.
- If data samples are biased, the analysis may not be accurate.
- Solution: Develop strategies to ensure collected data is diverse and representative.
- Social media platforms limit access to data.
- 🔹 **Example:** A researcher studying **fake news on Facebook** can only collect a small amount of posts daily due to Facebook's restrictions.

3. Noise Removal Fallacy

- Traditional data mining involves **removing noisy (irrelevant) data**.
- However, in social media mining:
    1. **Removing too much data may eliminate useful information** (worsening the Big Data Paradox).
    2. **Defining noise is difficult** because it depends on the context.
- Solution: Develop **smart filtering techniques** to distinguish useful data from noise.
- Social media contains spam, fake reviews, and bots.
- 🔹 **Example:** A company analyzing **Twitter data** might find many **spam tweets** (e.g., "Win $1,000 now! Click this link!"), which are not useful.

4. Evaluation Dilemma

- Traditional data mining techniques rely on **ground truth** (i.e., labeled data for training and testing models).
- In social media mining, **ground truth is often unavailable**.
- Without proper evaluation, **how can we validate our findings?**

- Solution: Use alternative evaluation methods, such as **cross-validation** and **crowdsourced labeling**.
- Unlike traditional research, social media data does not always have a "correct answer."
  - **Example:** If we analyze tweets to find out whether people like a political leader, how do we confirm if our findings are 100% accurate?
-

5.1.3 Types of social network graph

1. Null Graph

- **Definition:** A graph with **no nodes and no edges**.
- **Example:** Imagine a **social media platform with zero users**—there are no connections because no one exists.
- **Example:** A **new social media platform** with **zero users**.
- **Graph Representation:**
  (No nodes, No edges, just empty space)
-

---

2. Empty Graph (Edgeless Graph)

- **Definition:** A graph with **nodes but no edges**.
- **Example:** A classroom where students exist but **no one talks** to each other.
- **Example:** A classroom where students exist but **no one interacts**.
- **Graph Representation:**
- A B C D E (Just nodes, no edges)

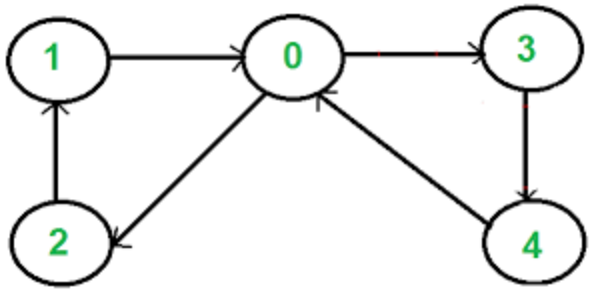
---

3. Directed, Undirected, and Mixed Graphs

- **Directed Graph:** Edges have **arrows** (one-way connections).
  - **Example: Twitter follows** (A follows B, but B may not follow A).

Directed Graph (one-way connections)

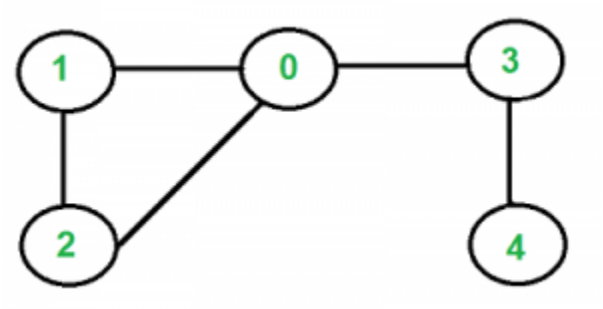
- **Example:** Twitter followers



- **Undirected Graph:** Edges have **no direction** (two-way connections).
    - **Example: Facebook friendships** (if A is a friend of B, then B is also a friend of A).

Undirected Graph (two-way connections)

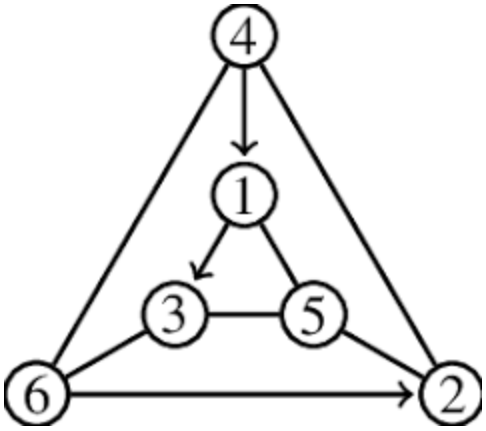
- **Example:** Facebook friendships



- **Mixed Graph:** Some edges are **directed**, and some are **undirected**.
    - **Example:** A **family tree** (parent-child relation is directed, but sibling relation is undirected).

Mixed Graph (some directed, some undirected)

- **Example:** A company network



---

4. Simple Graph vs. Multigraph

Simple Graph: Only **one edge** can exist between two nodes.

- **Example:** A **basic friendship network** where each connection represents a single friendship.

Simple Graph: One edge between two nodes

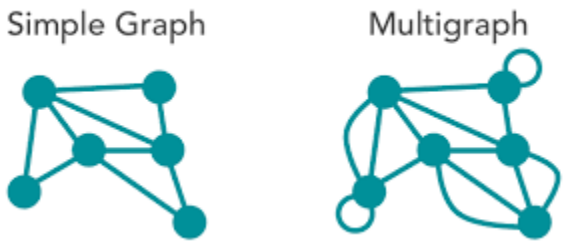
- **Example:** A basic social network

Multigraph: Multiple edges can exist between the same nodes.

- **Example:** Two people can be **friends, colleagues, and group members**, so multiple edges exist between them.

Multigraph: Multiple edges between two nodes

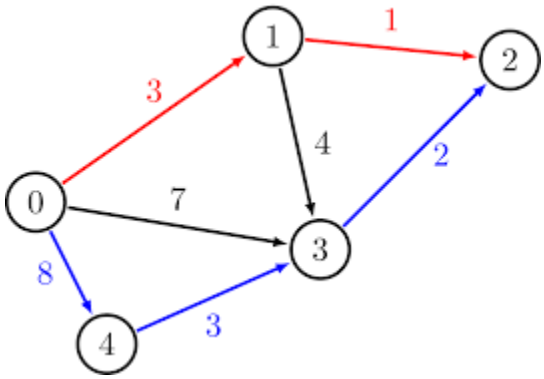
- **Example:** Two people can be **friends, colleagues, and family**



---

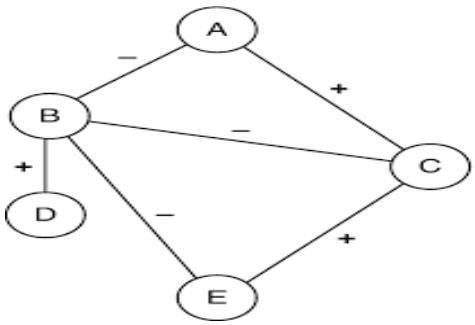
5. Weighted Graph

- **Definition:** Edges have **weights (values)** representing strength or cost.
- **Example: Google Maps** (nodes = cities, edges = roads, weight = distance).
- **Definition:** Each edge has a **weight (value)**.
- **Example: Google Maps** (weights = distances between cities).



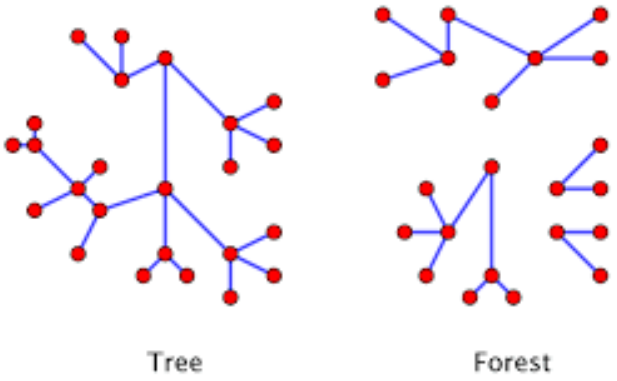
6. Signed Graph

- **Definition:** Edges have **positive (+) or negative (-) signs**.
- **Example:** A **social network with friends and enemies** (+ for friends, - for enemies).
- **Definition:** Each edge has a **positive (+) or negative (-) sign**.
- **Example: Friend-enemy relationships**



7. Trees and Forests

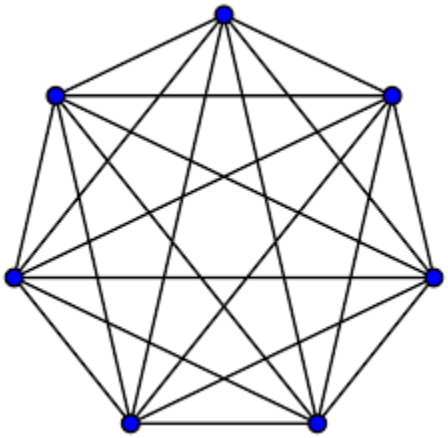
- **Tree:** A graph with **no cycles** (one unique path between any two nodes).
    - **Example:** A **company hierarchy** (CEO → Manager → Employee).
- **Forest:** A set of **multiple disconnected trees**.
    - **Example:** A **set of different family trees**.



---

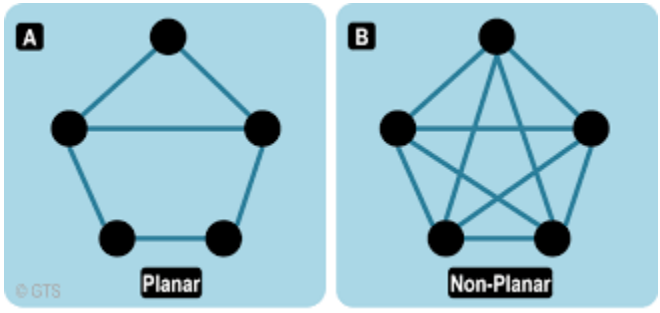
9. Complete Graph

- **Definition:** A graph where **every node is connected to every other node**.
- **Example:** A **small WhatsApp group** where **everyone knows everyone**.
- *All nodes are connected to all others.*



10. Planar vs. Non-Planar Graph

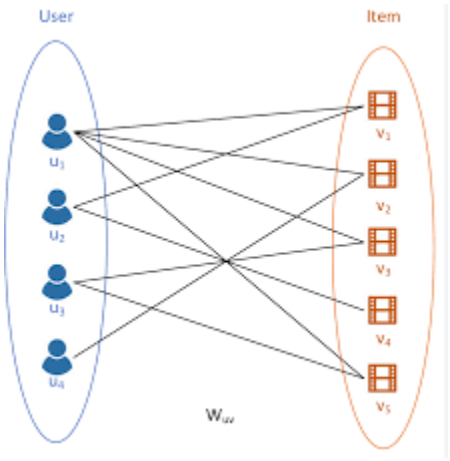
- **Planar Graph:** Can be drawn **without edges crossing**.
    - **Example:** A **basic road network** without flyovers.
- **Non-Planar Graph:** Some edges must **cross each other**.
    - **Example:** A **complex subway system with tunnels and overpasses**.



---

11. Bipartite Graph

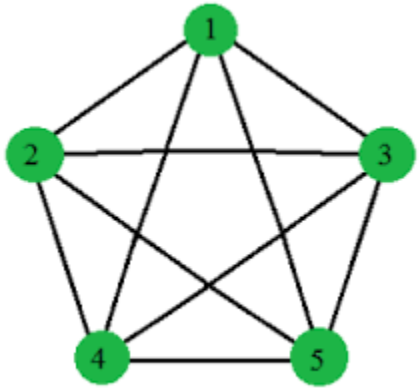
- **Definition:** Nodes are divided into **two sets**, and edges only **connect nodes from different sets**.
- **Example:** A **student-course network** (students on one side, courses on the other).
- Two groups, edges only between groups.



---

12. Regular Graph

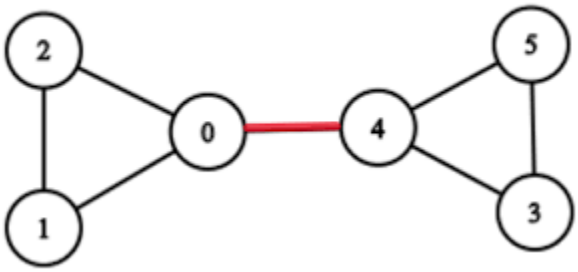
- **Definition:** Every node has the **same number of connections**.
- **Example:** A **sports tournament** where **every team plays the same number of matches**.
- *All nodes have the same number of connections.*



4 Regular

13. Bridges

- **Definition:** An edge whose removal **splits the graph into disconnected parts**.
- **Example:** A **single road connecting two islands**—if destroyed, they become disconnected.



5.2 Mining in Social Media

5.2.1 Influence in Social Media

Influence

Influence is defined as the ability to produce an effect without direct force or command. This section focuses on measuring and modeling influence in social media.

Influence in social media refers to the ability of a person, account, or entity to impact others' opinions, behaviors, or actions without direct control.

---

1 Measuring Influence

Influence can be measured using **prediction-based** or **observation-based** methods.

1. **Prediction-Based Measures**
    - Influence is predicted based on an individual's attributes or position in a network.
    - Centrality measures like **PageRank** and **degree centrality** (e.g., in-degree on Twitter) are used.

This method predicts influence based on a person's position in the network.

🔹 **Example: Twitter Followers and PageRank**

- If a person has **1 million followers on Twitter**, they are likely to be influential.
- Google's **PageRank algorithm** ranks web pages based on their importance in the network.
- In social media, similar algorithms measure influence based on how well-connected a user is.

👉 **Example:**
Consider **Elon Musk** on Twitter. He has millions of followers, so his tweets automatically gain attention, even before they spread widely.

2. **Observation-Based Measures**
    - Influence is measured based on actual effects in different contexts:
        1. **Role models** (e.g., celebrities, teachers) – Audience size indicates influence.

           Celebrities, teachers, or leaders are influential based on the number of people following them.

A YouTube influencer like MrBeast has a massive audience, making him influential.

👉 Example:

If MrBeast promotes a new brand of chocolate, his fans might buy it just because he recommends it.

2. **Information spread** – Measured by cascade size or population affected.

   Influence can be measured by how widely information spreads.

- A tweet that gets **100,000 retweets** shows significant influence.

  👉 **Example:**
  During the **Black Lives Matter movement**, hashtags like #BLM gained massive traction. A single influential tweet led to **millions of shares**, showing the power of influence.

3. **Participation impact** – When an action increases the value of an item (e.g., buying a product like a fax machine).

   When more people adopt something, it becomes more valuable (e.g., social media platforms).

   A product like **WhatsApp** became popular because everyone started using it.

   👉 **Example:**
   If **Elon Musk tweets about a new cryptocurrency**, and thousands of people start buying it, its value increases because of participation.

Case Studies in Social Media Influence Measurement

1. **Blogosphere Influence**
   - Influential bloggers are identified using factors like:
     1. **Recognition** (number of in-links)
     2. **Activity generation** (comments received)
     3. **Novelty** (fewer citations mean more originality)

        4. **Eloquence** (longer blog posts suggest higher quality)
    - Influence is modeled using an **i-graph** that tracks influence flow through links.
2. **Twitter Influence**
    - Common influence measures:
        1. **In-degree (followers count)** – Indicates audience size.
        2. **Mentions (@username)** – Shows engagement in conversations.
        3. **Retweets** – Measures content virality.
    - **Spearman's rank correlation** shows that the number of followers does not strongly correlate with mentions or retweets.

---

2 Modeling Influence

Influence models help explain how individuals affect each other in social media. There are **explicit** and **implicit** network models.

1. **Modeling Influence in Explicit Networks (Linear Threshold Model - LTM)**
    - Nodes in a network adopt behaviors when a threshold of their neighbors have already done so.
    - The LTM algorithm simulates influence spread, with thresholds assigned randomly between [0,1].
    - The model ensures that once thresholds are fixed, the influence spread is deterministic.

      This model assumes a person adopts an idea when a certain number of their friends already have.

      👉 **Example:**
      Imagine a **new fitness app** launches.

        i. At first, only a few people use it.
        ii. If **5 out of 10 friends** start using it, you might also start using it.
        iii. The adoption threshold determines when a person is influenced.

2. **Modeling Influence in Implicit Networks (Linear Influence Model - LIM)**
    - Influence spreads in a network, but the source of influence is unknown.
    - Influence function I(u,t) estimates how many people are influenced at time t.
    - Can be modeled using **parametric** (e.g., power-law distribution) or **non-parametric** (e.g., LIM) approaches.
    - Nodes represent individuals, and edges represent influence relationships.
    - Edge weights are probabilities that an active node influences its neighbors.

- The LIM model refines influence estimation by using matrix calculations to predict influence values over time.

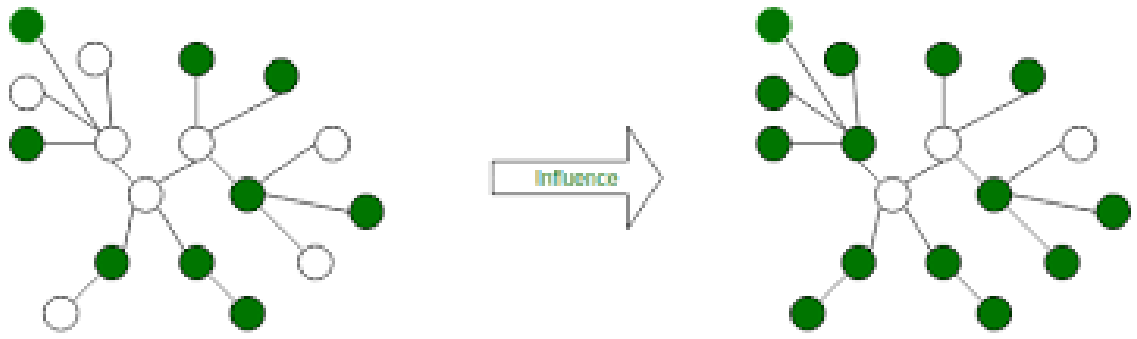
  Edge Weights as Probabilities: LIM focuses on influence probabilities rather than cumulative influence thresholds.

  Independent Influence Attempts: Each active node attempts to influence its neighbors at every step, even if it failed previously.

Sometimes, influence spreads without knowing the exact source.

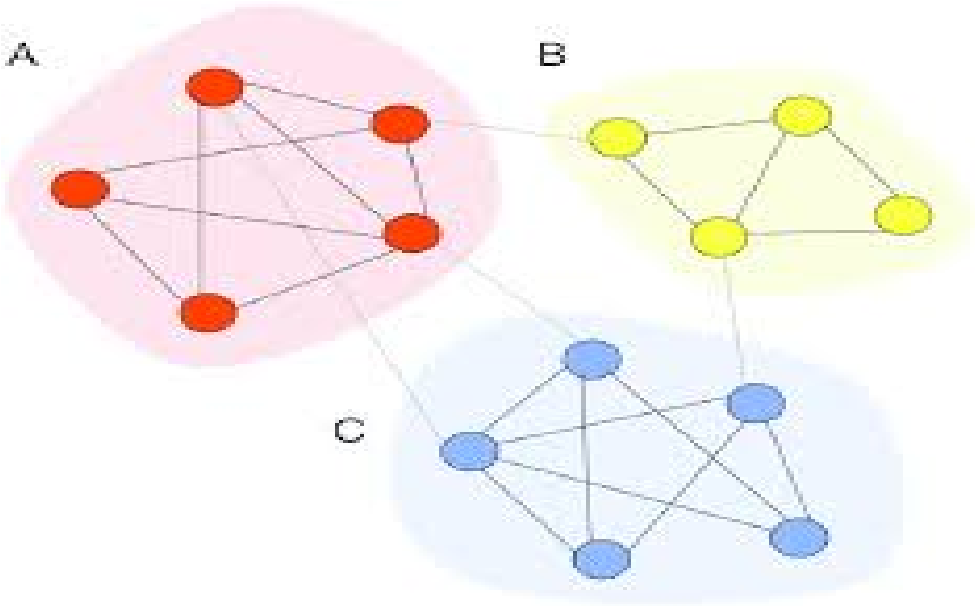
👉 **Example:**
A **song goes viral on TikTok**, but no one knows who started it.

- The influence spreads automatically as more people use the song in their videos.
- The model calculates how many people will be influenced over time.



5.2.1 Homophily

- **Definition**: Homophily refers to the tendency of similar individuals to form connections.
    - Commonly observed in social networks, as similar people often connect (e.g., same interests, beliefs, or demographics).
    - Unlike influence, where one individual impacts another, homophily is mutual and based on similarity.
- **Real-World Example**:
    - On social media, users with shared hobbies (e.g., gaming or cooking) are more likely to follow each other.



---

1 Measuring Homophily

- Homophily is measured by observing changes in the **assortativity** of a network over time.
- **Assortativity**: The degree to which nodes with similar attributes are connected.

2 Modeling Homophily

Homophily can be modeled using a variation of the **Independent Cascade Model**.

Difference Between Influence and Homophily

While both **influence** and **homophily** shape connections and behaviors in social networks, they differ fundamentally in their nature, direction, and effects. Here's a comparison:

Aspect	Influence	Homophily
Definition	Influence is the process where one individual affects the behavior, opinions, or decisions of another.	Homophily is the tendency of similar individuals to connect based on shared attributes.
Direction	Influence flows **directionally** from one person (the influencer) to another.	Homophily is **mutual**; both individuals connect due to shared similarities.
Nature	Active: It involves a change in behavior or opinion.	Passive: It involves forming connections based on pre-existing similarity.
Driving Factor	Influence arises from **persuasion, authority, or expertise** of one individual.	Homophily arises from **inherent similarities** like demographics, interests, or beliefs.
Formation of Links	Links form when one individual influences another to connect or adopt behaviors.	Links form because individuals already share common attributes.
Example	A social media influencer persuades their followers to buy a product or adopt a habit.	Gamers on a social network connect with each other because they share an interest in gaming.
Observable Effect	Can lead to **behavioral or opinion change**.	Leads to **assortativity** in the network (formation of clusters based on similarity).
Social Network Outcome	Can create **information cascades** or viral effects as influence spreads through the network.	Creates **segmentation or clustering** where similar individuals form dense groups.
Timeframe	Influence can be instantaneous or gradual.	Homophily is a gradual process based on accumulating connections.

Examples

Influence:

- **Scenario**: A fitness influencer on Instagram promotes a new workout routine.
    - Result: Followers start adopting the same routine based on the influencer's recommendation.

Homophily:

- **Scenario**: On Facebook, two individuals who like fitness and healthy eating connect.
    - Result: Their connection is based on shared interests, without one influencing the other.

5.2.2 Behavior Analytics

Behavior analytics in social media mining involves analyzing users' activities, interactions, and preferences to understand patterns, predict behavior, and derive actionable insights. This field leverages data from social media platforms to uncover trends in user behavior and decision-making.

Key Objectives

1. **Understand User Behavior**:
    - Analyze how users interact with content (e.g., likes, shares, comments).
    - Identify patterns in content consumption and preferences.
2. **Predict Future Behavior**:
    - Forecast user responses to campaigns or content.
    - Predict churn rates, virality, or trends.
3. **Enhance Personalization**:
    - Tailor recommendations based on user profiles and past activity.
4. **Identify Influencers**:
    - Detect users with high influence and engagement potential.
5. **Detect Anomalies**:
    - Spot unusual behavior (e.g., fake accounts or bots).

Key Techniques in Behavior Analytics

1. **Network Analysis**:
    - Understand how users are connected and interact (e.g., graph-based analysis for influence and clustering).
    - Analyze social graphs for centrality, clusters, and communities.
2. **Sentiment Analysis**:
    - Analyze text data (posts, comments) to understand user emotions and opinions.
3. **Trend Analysis**:
    - Identify trending topics and hashtags by analyzing user-generated content over time.
4. **Clickstream Analysis**:
    - Track the sequence of pages or content users engage with on social platforms.
5. **Engagement Metrics**:
    - Measure likes, shares, retweets, and comments to gauge user engagement.
6. **Behavioral Segmentation**:
    - Segment users into groups based on similar behaviors (e.g., active users vs. passive observers).

7. **Topic Modeling**:
    - Use NLP techniques to extract dominant topics in discussions or posts.

Applications in Social Media Mining

1. Marketing and Advertising:

- **Personalized Campaigns**: Deliver tailored ads based on user interests and past behavior.
- **Audience Insights**: Identify target groups for specific products or services.
- **Content Optimization**: Create content that resonates with specific audience segments.

2. Recommender Systems:

- Suggest content (e.g., videos, articles, or products) based on user preferences.
- Example: Netflix and YouTube recommendations.

3. Influence and Virality:

- Identify key influencers in a network and analyze their impact.
- Study how content goes viral and what triggers information cascades.

4. Customer Experience:

- Monitor feedback to improve services or products.
- Resolve customer grievances by analyzing complaints or negative sentiments.

5. Fake News and Misinformation:

- Detect patterns of misinformation spread using user behavior and engagement data.
- Identify bots or fake accounts promoting false information.

6. Social Network Monitoring:

- Track behavior in groups or communities to identify leaders or influencers.
- Monitor growth or decline in engagement within communities.

7. Event Prediction:

- Predict outcomes like elections, stock market trends, or public events based on social media discussions.

Challenges in Behavior Analytics

1. **Data Privacy**:
    - Balancing insights with respect for user privacy and data protection regulations.
2. **Data Overload**:
    - Managing and analyzing large-scale, unstructured data efficiently.
3. **Anonymity and Fake Profiles**:
    - Dealing with inaccurate or misleading data from anonymous or fake accounts.
4. **Real-Time Processing**:
    - Providing actionable insights in real-time for dynamic campaigns or trends.

Example Use Case

Campaign Optimization for a Fashion Brand:

- Analyze customer reviews and comments on Instagram for new products.
- Identify key influencers discussing the brand and track the reach of their posts.
- Segment users based on shopping behavior (e.g., frequent buyers, occasional shoppers).
- Predict the success of future campaigns by analyzing user engagement on past ads.

Behavior Analytics in Social Media Mining: Example

Scenario: Launching a New Fitness Product

A fitness brand plans to launch a **smart fitness band**. They use behavior analytics on social media platforms like Twitter, Instagram, and Facebook to optimize their campaign and understand user behavior.

Step-by-Step Analysis

1. Data Collection:

- Collect posts, comments, likes, shares, and hashtags related to fitness and wearables using social media mining tools.
- Example data:
    - Tweets: "Looking for a good fitness tracker. Any suggestions? #FitnessGoals"
    - Instagram comments: "Love my current tracker, but I wish it tracked sleep better."
    - Facebook posts: "Top 10 fitness bands of 2025 reviewed."

---

2. Sentiment Analysis:

- Analyze sentiment in the posts to gauge user emotions:
    - **Positive Sentiment**: "Excited about the new smart bands launching this year!"
    - **Negative Sentiment**: "My fitness tracker is inaccurate; I wouldn't recommend it."
- Insights:
    - Users are excited about new features but demand better accuracy.

---

3. Behavioral Segmentation:

- Group users based on their behavior:
    - **Fitness Enthusiasts**: Post frequently about workouts and fitness goals.
    - **Tech-Savvy Users**: Discuss features and specs of wearables.
    - **Occasional Users**: Interact less but engage with posts about fitness trends.
- Insights:
    - Focus marketing efforts on **Fitness Enthusiasts** and **Tech-Savvy Users** for higher engagement.

---

4. Network Analysis:

- Create a graph of interactions (likes, shares, comments) to identify influencers:
    - **Node**: A social media user.
    - **Edge**: Interaction between users (e.g., comments or retweets).
- Example:
    - Influencer "FitLifeGuru" has 500K followers and high engagement (10K likes/post).
    - Target them for product promotion.
- Insights:
    - Collaborate with key influencers to amplify reach.

---

5. Trend Analysis:

- Analyze hashtags like #FitnessBand, #SmartWearables, and #HealthyLiving to identify trends.
- Example:
    - "Sleep tracking" and "calorie counting" are trending features.
- Insights:
    - Highlight these features in the marketing campaign.

---

6. Engagement Metrics:

- Measure engagement levels on previous fitness product campaigns:
    - Posts with video content have **2x higher engagement** than static images.
    - Posts published at 6 PM get **30% more likes** than other times.
- Insights:
    - Use video ads and schedule posts during peak engagement hours.

Outcome

Using behavior analytics, the brand launches a **targeted campaign**:

1. Collaborates with influencers like "FitLifeGuru."
2. Posts video ads highlighting **sleep tracking** and **calorie counting** features.
3. Targets **Fitness Enthusiasts** and **Tech-Savvy Users** using personalized ads.

Results

1. **Increased Engagement**:
    - Posts with the hashtag #SmartFitnessBand2025 receive 50% more engagement.
2. **Higher Conversion Rates**:
    - Users in the target segment (Fitness Enthusiasts) show a 20% purchase rate.
3. **Brand Awareness**:
    - Social media mentions of the brand increase by 70% during the campaign.

5.2.3 Challenges of Recommendation Systems in Social Media

1. Cold-Start Problem

- New users and content lack historical data, making it difficult to generate relevant recommendations.
- Social media platforms often ask for preferences or use general trending content initially.
- When new users join a social media platform (e.g., Instagram, TikTok), the system has no prior data on their preferences. Similarly, when a new content creator starts posting, their content may not get recommended due to a lack of engagement history.
- **Example:** A new TikTok user may initially see random trending videos because the system does not know their interests yet. Platforms often ask users to follow some accounts or select topics to improve recommendations.

2. Data Sparsity

- Many users interact with only a few posts, limiting available data.
- Sparse user-item interactions make it hard to generate diverse and accurate recommendations
- Some users engage heavily with content (liking, sharing, and commenting), while others rarely interact. If many users provide little engagement data, the recommendation system struggles to generate meaningful suggestions.
- **Example:** On Twitter (X), if a user follows many accounts but rarely likes or retweets, the algorithm has limited signals to recommend relevant tweets.

.

3. Popularity Bias

- Algorithms tend to favor already popular content, reducing visibility for niche or new content.
- This creates a feedback loop where viral content keeps getting recommended, limiting diversity.
- Popular content gets recommended more frequently, making it harder for new or niche creators to gain visibility. This leads to a cycle where trending content keeps trending while diverse content remains unseen.
- **Example:** On YouTube, viral videos with millions of views are often recommended over high-quality but less-known videos, reducing content diversity.

4. Echo Chambers & Filter Bubbles

- Users are exposed only to content that aligns with their existing views, limiting diverse perspectives.
- This can reinforce biases and misinformation in social media platforms.
- Users are shown content that aligns with their past behaviors, reinforcing their existing beliefs and limiting exposure to different viewpoints.
- **Example:** If a Facebook user only interacts with politically conservative posts, the algorithm may stop showing them liberal viewpoints, creating a biased content feed.

5. Privacy Concerns

- Personalized recommendations require user data, raising concerns about data security and consent.
- Stricter data regulations (e.g., GDPR, CCPA) require transparent handling of user information.
- Social media platforms collect vast amounts of user data to improve recommendations, raising concerns about data security and consent. Regulations like GDPR and CCPA require platforms to handle personal data responsibly.
- **Example:** Facebook faced criticism for tracking users' online behavior even outside the app to refine ad and content recommendations, raising concerns about privacy violations.

6. Fake Engagement & Manipulation

- Bots and fake accounts can artificially boost content visibility, distorting recommendation accuracy.
- Influencers and advertisers may game the system to push specific content.
- Some users, bots, or influencers exploit algorithms by artificially increasing engagement (e.g., buying likes, fake followers) to boost content visibility.
- **Example:** Instagram influencers sometimes buy fake followers and engagement to make their posts appear popular, leading the algorithm to recommend their content over genuine, high-quality posts.

7. Context Awareness

- Social media content consumption varies by time, location, and mood, making static recommendations less effective.
- Algorithms struggle to incorporate real-time context effectively.

- User interests change over time, and the system must adapt dynamically. Current algorithms struggle to consider real-time factors such as mood, location, and temporary interests.
- **Example:** A LinkedIn user searching for "data science jobs" may continue receiving job recommendations even after securing a job, making the recommendations irrelevant.

8. Balancing Real-Time & Personalized Recommendations

- Users expect both trending (real-time) and personalized content, requiring a balance between freshness and relevance.
- Too much personalization can lead to outdated recommendations.
- Users expect both trending and personalized content, requiring the system to balance global trends with individual interests.
- **Example:** Twitter (X) shows both “For You” (personalized) and “Trending” (real-time) sections, but sometimes users complain that trending topics are not relevant to them.

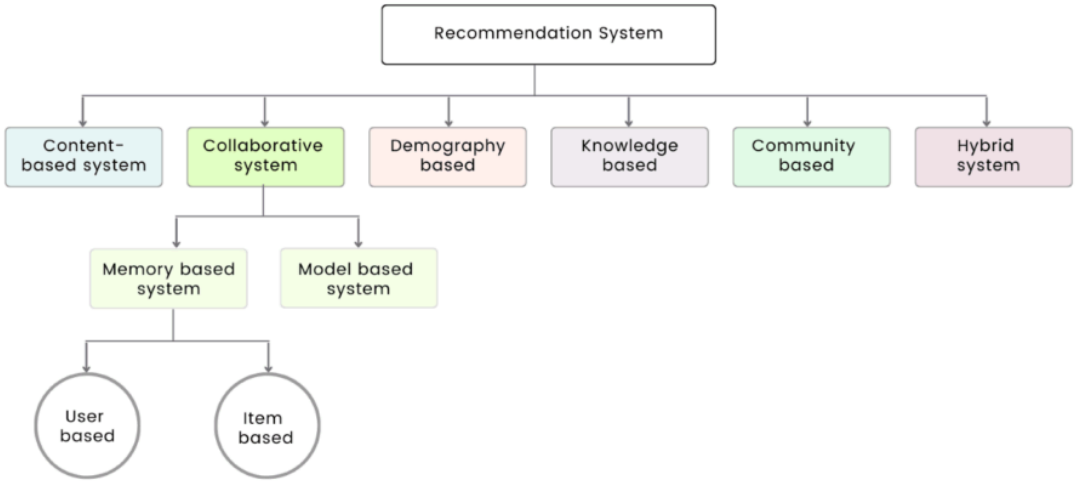
9. Content Moderation & Misinformation

- Recommender systems sometimes amplify harmful, misleading, or inappropriate content.
- Platforms must ensure ethical and responsible content recommendation.
- Algorithms sometimes promote misleading, harmful, or fake news because engagement-driven systems prioritize viral content.
- **Example:** During the COVID-19 pandemic, social media platforms struggled to prevent the spread of misinformation about vaccines despite efforts to fact-check posts.

10. Scalability & Computational Challenges

- Social media generates massive amounts of data, requiring efficient algorithms to process recommendations in real time.
- Ensuring speed and accuracy at scale is a major technical challenge.
- Social media generates massive data streams in real-time, requiring scalable algorithms that process recommendations efficiently.
- **Example:** TikTok’s recommendation engine must analyze millions of user interactions per second to update personalized feeds instantly, requiring high computational power

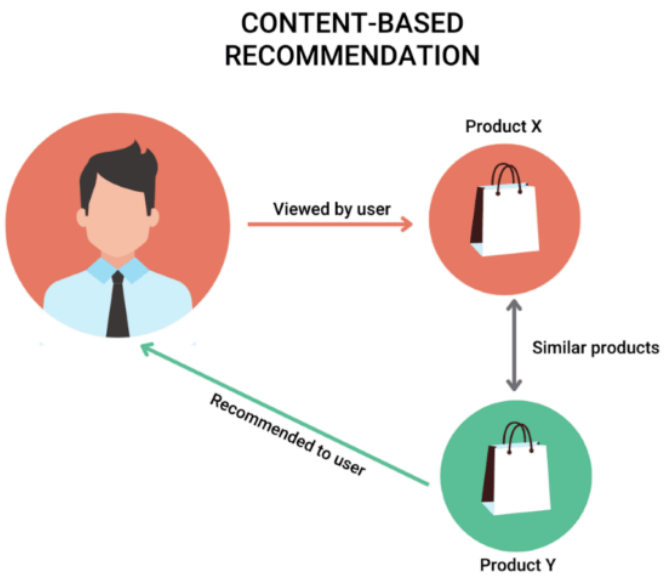
5.2.4 Classical Recommendation Algorithms



Recommendation algorithms help suggest items (e.g., movies, books, products) to users based on their interests and behavior. Classical recommendation algorithms are mainly categorized into **Content-Based Filtering** and **Collaborative Filtering** (User-Based, Item-Based, and Model-Based).

---

1. Content-Based Recommendation



Concept:

- This method recommends items that are similar to the ones a user has already liked.
- It analyzes item features and user preferences.

How It Works:

1. Each item is described using features (e.g., genre, director, actors for movies).
2. A user profile is created based on their past interactions.
3. The system calculates similarity between the user's preferred items and new items.
4. Items with the highest similarity score are recommended.

Example:

Let’s say a user has watched and liked the following movies:

- **Movie 1:** "Inception" (Sci-Fi, Action)
- **Movie 2:** "Interstellar" (Sci-Fi, Drama)

Now, we have a new movie:

- **Movie 3:** "The Matrix" (Sci-Fi, Action)

Since "The Matrix" has similar genres to "Inception" and "Interstellar," it will be recommended to the user.

Mathematical Representation:

- Each movie is represented as a vector of its features.

$$\text{Similarity}(A, B) = \frac{A \cdot B}{||A|| \times ||B||}$$

If **Movie 1 (A)** = (1,1,0) (Sci-Fi, Action, No Drama)

and **Movie 3 (B)** = (1,1,0) (Sci-Fi, Action, No Drama),

then **Cosine Similarity** = 1 (perfect match), so Movie 3 is recommended.

Advantages:

✅ Works well for users with some history.
✅ Personalized recommendations.

Disadvantages:

❌ Struggles with **cold start problem** (new users).
❌ Limited diversity (recommends similar items).

---

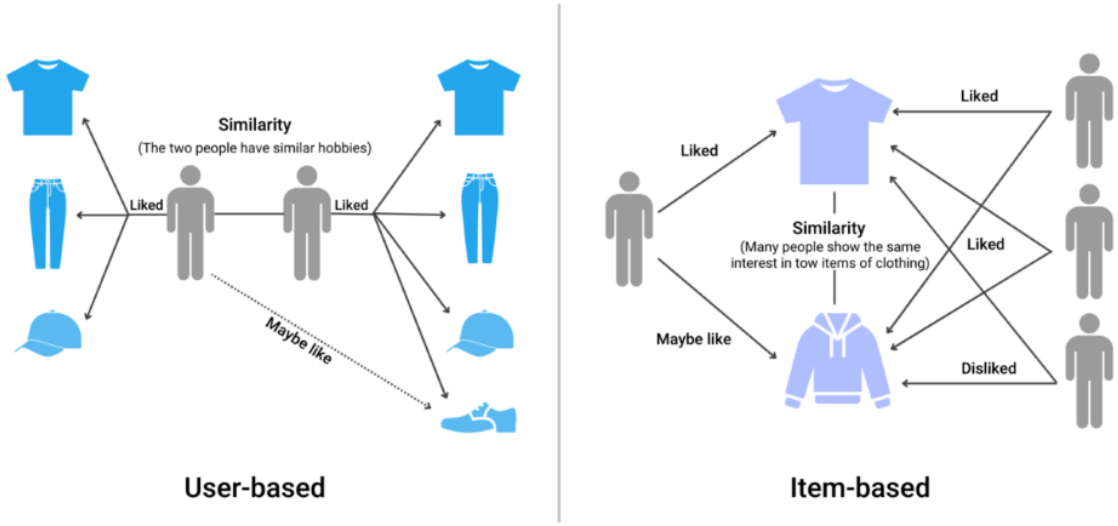
2. Collaborative Filtering (CF)

Instead of relying on item features, CF suggests items based on the preferences of other users. It assumes that **users with similar tastes in the past will have similar tastes in the future**.

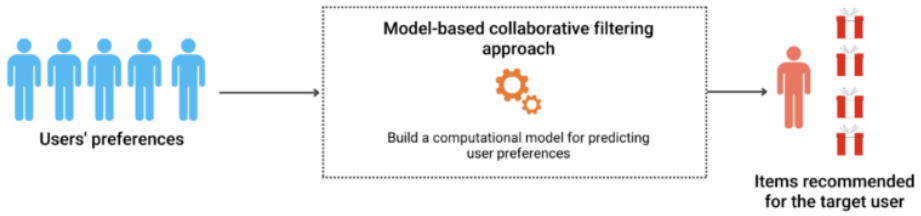
Types of Collaborative Filtering:

A) Memory Based-

1) **User-Based CF**
2) **Item-Based CF**



B) Model-Based CF (e.g., Matrix Factorization - SVD)



2.1 User-Based Collaborative Filtering

Concept:

- Finds users who have rated items similarly and recommends items they liked.
- Uses similarity between users to predict preferences.

Example:

User-Movie Rating Matrix

User	Movie A	Movie B	Movie C	Movie D
User 1	5	4	3	?
User 2	5	5	2	4
User 3	2	2	5	3

- User 1 has not rated **Movie D** yet.
- User 2 is similar to User 1 (their ratings for A, B, and C are close).
- Since **User 2 rated Movie D as 4**, we predict **User 1 will also like Movie D**, so we recommend it.

Mathematical Representation:

- **Pearson Correlation Coefficient** measures similarity between users:

$$\text{Similarity}(U, V) = \frac{\sum (r_{U,i} - \bar{r}_U)(r_{V,i} - \bar{r}_V)}{\sqrt{\sum (r_{U,i} - \bar{r}_U)^2} \sqrt{\sum (r_{V,i} - \bar{r}_V)^2}}$$

where $r_{U,i}$ is User U's rating for item i.

Advantages:

✅ Good for personalized recommendations.
✅ Works well when users have rated multiple items.

Disadvantages:

❌ Struggles with new users (**cold start problem**).
❌ Does not work well with sparse data (when users have rated very few items).

2.2 Item-Based Collaborative Filtering

Concept:

- Instead of finding similar users, this method finds similar items.
- If two items are rated similarly by many users, they are considered similar.

Example:

If many users who watched **"The Dark Knight"** also watched **"Batman Begins,"** then **"Batman Begins"** will be recommended to users who liked "The Dark Knight."

Mathematical Representation:

- **Cosine Similarity** is used to measure item similarity:

$$\text{Similarity}(I, J) = \frac{I \cdot J}{||I|| \times ||J||}$$

- **Prediction formula** (weighted sum of ratings):

$$\hat{r}_{u,j} = \frac{\sum_{i \in N} \text{Similarity}(j,i) \times r_{u,i}}{\sum_{i \in N} |\text{Similarity}(j,i)|}$$

Advantages:

✅ More stable than user-based CF.
✅ Works well even when users rate few items.

Disadvantages:

❌ Cold start problem for new items.

2.3 Model-Based Collaborative Filtering (SVD - Singular Value Decomposition)

Concept:

- Uses **Matrix Factorization** to find hidden patterns in user-item interactions.
- Decomposes the user-item matrix into latent features representing user preferences and item characteristics.

Example:

Netflix applies **SVD** to decompose a **User-Movie rating matrix** into latent factors (e.g., action preference, drama preference) and predicts missing ratings.

Mathematical Representation:

Matrix factorization decomposes matrix **R** (user-item ratings):

$$R = U \times \Sigma \times V^T$$

where:

- **U** = User preferences
- **Σ** = Importance of each latent factor
- **V** = Item properties

The missing ratings are predicted using these latent factors.

Advantages:

✅ Works well with sparse data.
✅ Provides better recommendations than basic CF.

Disadvantages:

❌ Computationally expensive.

Comparison of Methods

Algorithm	Strengths	Weaknesses
Content-Based Filtering	Personalized, works well for known users	Cold start problem, lacks diversity

User-Based CF	Finds similar users, good for personalization	Cold start problem, struggles with sparse data
Item-Based CF	Stable, works well for sparse data	Cold start problem for new items
Model-Based (SVD)	Efficient with large data, finds hidden patterns	Computationally expensive

---

Conclusion

- **Content-Based Filtering** works well when item metadata is available.
- **User-Based CF** is useful when users have many shared preferences.
- **Item-Based CF** is better for stable recommendations.
- **Model-Based CF (SVD)** is powerful for large-scale recommendations but requires computational power.

3) Demographic-Based Recommendation System

- This system suggests things based on a user's characteristics like age, gender, and location.
- **Example**: If a 25-year-old living in Mumbai opens an event app, it may suggest concerts, tech meetups, or weekend getaways popular among people of the same age group in Mumbai.

4) Knowledge-Based Recommendation System

- This system works based on what a user asks or needs rather than their past choices.
- **Example**: When you go to a real estate website and enter details like budget, location, and type of house you want, the system suggests properties that match your requirements.

5) Community-Based Recommendation System

- This system recommends things based on the preferences of a group of people with shared interests.
- **Example**: In a gaming forum, if most members like a new game, the system might recommend that game to new users who join the community.

6) Hybrid Recommendation System

- It combines multiple recommendation methods to improve suggestions.
- **Example**: Netflix uses both content-based (shows similar to what you watched) and collaborative-based (what other users with similar tastes watched) to recommend movies and series.

5.2.5 Recommendation Using Social Context

In social media and recommendation systems, traditional recommendation models rely on user-item interactions, such as ratings or purchase history. However, incorporating **social context**—such as friendships, social influence, or shared preferences—can improve recommendations. Social context can be used in three ways:

1. **Using social context alone** (friendship-based recommendations)
2. **Extending classical recommendation models with social context**
3. **Constraining recommendations using social context**

These methods recognize that users with social connections (e.g., friends) are likely to have similar preferences due to factors like **homophily, influence, or confounding** (discussed in Chapter 8).

---

1) Using Social Context Alone

When we have only the social network (friendship relations) but no explicit user-item rating matrix, we can still generate **friend recommendations**.

How does friend recommendation work?

Many social networking platforms (e.g., Facebook, LinkedIn) suggest potential friends based on existing connections. Some common techniques include:

1. Link Prediction

- Predict whether a **new connection (link)** will form between two individuals.
- Methods include **Common Neighbors, Jaccard Coefficient, Adamic-Adar Index, and Katz Index** (discussed in Chapter 10).

2. Triadic Closure (Open Triads)

- Social networks exhibit a property called **triadic closure**: If **A** is friends with **B**, and **B** is friends with **C**, then **A** is likely to become friends with **C**.
- A **triad** consists of three users with three edges forming a closed loop.
- An **open triad** is missing one of these edges. Friend recommendation suggests closing the loop.

Example

If Alice is friends with Bob and Bob is friends with Charlie, but Alice and Charlie are not yet connected, the system may recommend that Alice and Charlie become friends.

This method is widely used in **friend suggestion systems** in social networks.

When we only have social network data (friendship connections) but no user-item ratings, we can **recommend new friends** to users.

Example 1: Friend Recommendation using Triadic Closure

Consider a social network where users are connected as friends:

Current Network:

- **Alice** is friends with **Bob**.
- **Bob** is friends with **Charlie**.
- **Alice** and **Charlie** are not yet friends.

Triadic Closure Principle:

Since Alice and Charlie have a common friend (Bob), the system **recommends Charlie as a friend to Alice**.

Graph

```
Alice --- Bob --- Charlie
```

- **Alice and Charlie should become friends** since they both know Bob.

Facebook's Friend Suggestion

Facebook uses this approach to suggest **"People You May Know"** based on mutual friends.

2) Extending Classical Methods with Social Context

In traditional recommendation systems, we use a **user-item rating matrix** to predict unknown ratings. Here, social context is added to improve predictions.

A)Social Regularization in Matrix Factorization

What is Social Regularization?

Matrix Factorization (MF) is a technique used in recommendation systems to predict missing ratings. However, it only considers **user-item interactions** and ignores **social influence**.

To improve predictions, **Social Regularization** is added. This ensures that **friends have similar preferences** by minimizing the difference between their learned user preference vectors.

Matrix Factorization for Recommendations

We represent:

- **User preferences** as a **k-dimensional vector** Ui
- **Items** as a **k-dimensional vector** Vj
- The rating of user i for item j is computed as:

$$R_{ij} = U_i^T V_j$$

Matrix Representation

- R is an $n \times m$ **matrix** (users × items)
- U is an $k \times n$ **matrix** (user features)
- V is an $k \times m$ **matrix** (item features)

$$R = U^T V$$

This means that the rating matrix R can be **approximated** using two low-dimensional matrices U and V.

Optimization for Matrix Factorization

To find U and V, we solve the following optimization problem:

$$\min_{U,V} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} I_{ij}(R_{ij} - U_i^T V_j)^2$$

Where:

- I_{ij} is **1 if user i rated item j, otherwise 0**
- This ensures that only observed ratings contribute to the computation.

Overfitting Problem and Regularization

Matrix factorization models can **overfit** because the rating matrix R is usually **sparse** (many missing values). To prevent overfitting, we **regularize** by penalizing large values in U and V:

$$\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} I_{ij}(R_{ij} - U_i^T V_j)^2 + \frac{\lambda_1}{2} ||U||_F^2 + \frac{\lambda_2}{2} ||V||_F^2$$

Where:

- $||U||_F^2$ and $||V||_F^2$ are **Frobenius norms** (sum of squared values).
- λ_1 and λ_2 control the impact of regularization.

Example Scenario

We have **three users (Alice, Bob, and Charlie)** and their movie ratings.
We also have **friendship relationships** that influence their ratings.

Step 1: User-Item Matrix (Ratings Data)

User	Movie A	Movie B	Movie C
Alice	5	?	4
Bob	4	3	?

Charlie	?	2	5

Step 2: Friendship Matrix (Social Graph)

User	Alice	Bob	Charlie
Alice	0	1	1
Bob	1	0	0
Charlie	1	0	0

- **Alice & Bob are friends (1)**
- **Alice & Charlie are friends (1)**
- **Bob & Charlie are NOT friends (0)**

$$R_{Alice,B} = \frac{(sim(Alice, Bob) \times R_{Bob,B}) + (sim(Alice, Charlie) \times R_{Charlie,B})}{sim(Alice, Bob) + sim(Alice, Charlie)}$$

Given:

- $sim(Alice, Bob) = 1.0$
- $sim(Alice, Charlie) = 0.8$
- $R_{Bob,B} = 3$
- $R_{Charlie,B} = 2$

$$R_{Alice,B} = \frac{(1.0 \times 3) + (0.8 \times 2)}{1.0 + 0.8}$$

$$= \frac{3 + 1.6}{1.8} = \frac{4.6}{1.8} = 2.56$$

So, **Alice's predicted rating for Movie B = 2.56.**

Why Use Social Regularization?

✅ Improves recommendations by incorporating **friendship influence**.
✅ More realistic—people **trust** recommendations from friends.
✅ Reduces **overfitting** by smoothing user preferences.

B)Trust-Based Recommendation Model

$$R_{u,i} = \frac{\sum_{v \in F(u)} T(u,v) R_{v,i}}{\sum_{v \in F(u)} T(u,v)}$$

where:

- $R_{u,i}$ = Predicted rating for user u on item i.
- $F(u)$ = Friends of user u.
- $T(u,v)$ = Trust score between user u and friend v.

Collaborative Filtering (CF) recommends items (movies, products, songs, etc.) based on what similar users like. However, it **doesn't consider social relationships**—for example, if your **friend likes something, you might like it too**.

By **adding social context** (friendship, trust, influence), we improve recommendations.

---

1. Classical Collaborative Filtering (CF) – Without Social Context

Example:

Imagine an **online movie platform** where users rate movies from **1 to 5 stars**.

User	Movie A	Movie B	Movie C
Alice	5	?	4
Bob	4	3	?

Charlie	?	2	5

👉 Alice hasn't rated **Movie B**. CF predicts Alice's rating by looking at similar users (Bob & Charlie). If Bob and Charlie **liked Movie B**, Alice is likely to like it too.

Limitation: It **doesn't** consider that Alice and Bob might be **close friends**, meaning Bob's rating should influence Alice's rating more than Charlie's.

2. Extending CF with Social Context

We improve CF by **including social connections**:

- If **Alice trusts Bob**, his ratings should influence her recommendations more.
- If **Bob and Charlie are strangers**, Charlie's opinion should matter less.

Example:

Now, let's add **friendship strength**:

User	Movie A	Movie B	Movie C	Trust Score (with Alice)
Alice	5	?	4	-
Bob	4	3	?	0.9 (close friend)
Charlie	?	2	5	0.3 (distant friend)

Now, Alice's predicted rating for **Movie B** depends more on Bob's opinion because he is a **closer friend**.

Formula (Trust-Based Prediction):

$$R_{Alice,B} = \frac{(0.9 \times 3) + (0.3 \times 2)}{0.9 + 0.3}$$

$$= \frac{2.7 + 0.6}{1.2} = 2.75$$

Prediction: Alice's rating for **Movie B is 2.75**, influenced **more by Bob's opinion.**

3. Why is Social Context Important?

- Friends have **similar tastes** (if your best friend loves a movie, you might too).
- Users **trust** recommendations from friends more than strangers.
- Improves accuracy by reducing dependence on **only past ratings**.

✅ **Real-world Example:**

- **Netflix & Prime Video:** "Because your friend liked this show..."
- **Amazon & Flipkart:** "Recommended based on what your friends bought."
- **Spotify & Apple Music:** "Playlists from your friends."

3. Recommendation Constrained by Social Context

Instead of **modifying the recommendation model**, we can **restrict recommendations to a user's social network**.

How It Works?

1. **Find Similar Users** → Identify users with similar preferences.
2. **Apply Social Constraint** → Only use **friends** from the similar users' group.
3. **Predict Ratings** using information from these **socially constrained neighbors**.

Example Scenario

We have **five users (John, Joe, Jill, Jane, Jorge)** and their ratings for **four movies**.

Step 1: User-Item Ratings Matrix

User	Lion King (LK)	Aladdin (A)	Mulan (M)	Anastasia (AN)
John	4	3	2	2
Joe	5	2	1	5
Jill	2	5	?	0
Jane	1	3	4	3
Jorge	3	1	1	2

Jill's rating for **Mulan (M) is missing (?).**
We will predict it using **socially constrained collaborative filtering**.

Step 2: Friendship Matrix (Social Graph)

User	John	Joe	Jill	Jane	Jorge
John	0	1	0	0	1
Joe	1	0	1	0	0
Jill	0	1	0	1	1
Jane	0	0	1	0	0
Jorge	1	0	1	0	0

- **Jill's friends**: **Joe, Jane, Jorge**.
- We will only use **Jill's friends** when predicting her missing rating for Mulan.

Step 3: Compute Similarities

We calculate the **cosine similarity** between Jill and her friends based on their ratings.

$$\text{sim}(Jill, Joe) = \frac{(2 \times 5) + (5 \times 2) + (0 \times 5)}{\sqrt{(2^2 + 5^2 + 0^2)} \times \sqrt{(5^2 + 2^2 + 1^2 + 5^2)}}$$

$$= \frac{(10 + 10 + 0)}{\sqrt{29} \times \sqrt{54}} = 0.50$$

$$\text{sim}(Jill, Jane) = \frac{(2 \times 1) + (5 \times 3) + (0 \times 3)}{\sqrt{(2^2 + 5^2 + 0^2)} \times \sqrt{(1^2 + 3^2 + 4^2 + 3^2)}}$$

$$= \frac{(2 + 15 + 0)}{\sqrt{29} \times \sqrt{19}} = 0.72$$

$$\text{sim}(Jill, Jorge) = \frac{(2 \times 3) + (5 \times 1) + (0 \times 2)}{\sqrt{(2^2 + 5^2 + 0^2)} \times \sqrt{(3^2 + 1^2 + 1^2 + 2^2)}}$$

$$= \frac{(6 + 5 + 0)}{\sqrt{29} \times \sqrt{14}} = 0.54$$

Step 4: Predict Jill's Rating for Mulan Using Social Constraint

$$r_{Jill,Mulan} = \bar{r}_{Jill} + \frac{\sum_{v \in S(Jill)} \text{sim}(Jill, v)(r_{v,Mulan} - \bar{r}_v)}{\sum_{v \in S(Jill)} \text{sim}(Jill, v)}$$

Step 4.1: Compute Average Ratings

$$\bar{r}_{Jill} = \frac{(2+5+0)}{3} = 2.33$$

$$\bar{r}_{Joe} = \frac{(5+2+1+5)}{4} = 3.25$$

$$\bar{r}_{Jane} = \frac{(1+3+4+3)}{4} = 2.75$$

$$\bar{r}_{Jorge} = \frac{(3+1+1+2)}{4} = 1.75$$

Step 4.2: Compute Weighted Sum for Prediction

$$r_{Jill,Mulan} = 2.33 + \frac{(0.72 \times (4 - 2.75)) + (0.54 \times (1 - 1.75))}{0.72 + 0.54}$$

$$= 2.33 + \frac{(0.72 \times 1.25) + (0.54 \times -0.75)}{1.26}$$

$$= 2.33 + \frac{0.90 - 0.405}{1.26}$$

$$= 2.33 + \frac{0.495}{1.26} = 2.33 + 0.393 = 2.72$$

Final Prediction

- **Jill's predicted rating for Mulan = 2.72 (rounded).**

Why Use Socially Constrained Recommendations?

✅ **More reliable recommendations** → We trust friends' opinions.
✅ **Avoids random similar users** → Focuses only on **socially relevant** recommendations.
✅ **Improves personalization** → Users get recommendations from people they relate to.

5.2.6 Evaluating Recommendations

Evaluating recommendation systems is crucial to measure their effectiveness. The evaluation can be categorized into three main aspects:

1️⃣ **Accuracy of Predictions** – Measures how close the predicted ratings are to the actual ratings using metrics like:

- **Mean Absolute Error (MAE)**: Average absolute difference between predicted and true ratings.
- **Root Mean Squared Error (RMSE)**: Penalizes larger errors more than MAE.
- **Normalized MAE (NMAE)**: Adjusts MAE based on rating range.

2️⃣ **Relevancy of Recommendations** – Assesses if recommended items are useful to users using:

- **Precision**: Proportion of recommended items that are relevant.
- **Recall**: Proportion of relevant items that were recommended.
- **F-measure**: Harmonic mean of precision and recall.

3️⃣ **Ranking of Recommendations** – Evaluates the order of recommendations using:

- **Spearman's Rank Correlation**: Measures correlation between predicted and true rankings.
- **Kendall's Tau**: Assesses ranking consistency by comparing concordant and discordant pairs.

1 Evaluating Accuracy of Predictions

We measure how **close the predicted ratings** are to the **actual user ratings**.

1. Mean Absolute Error (MAE)

It calculates the **average absolute difference** between **predicted ratings** ($\hat{r}_{ij}$) and **true ratings** (r_{ij}).

$$MAE = \frac{\sum_{i,j} |\hat{r}_{ij} - r_{ij}|}{n}$$

Where:

- n = number of ratings
- $\hat{r}_{ij}$ = predicted rating
- r_{ij} = true rating

2. Normalized Mean Absolute Error (NMAE)

This normalizes the MAE by the **range of ratings** ($r_{\max} - r_{\min}$).

$$NMAE = \frac{MAE}{r_{\max} - r_{\min}}$$

3. Root Mean Squared Error (RMSE)

RMSE **penalizes large errors more** than MAE by squaring the differences before averaging.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i,j} (\hat{r}_{ij} - r_{ij})^2}$$

Accuracy Calculation

Item	Predicted Rating	True Rating
1	1	3
2	2	5
3	3	3
4	4	2

5	4	1

Step 1: Compute MAE

$$MAE = \frac{|1-3| + |2-5| + |3-3| + |4-2| + |4-1|}{5}$$
$$= \frac{2+3+0+2+3}{5} = \frac{10}{5} = 2$$

Step 2: Compute NMAE

Given that the rating range is **1 to 5**,

$$NMAE = \frac{2}{5-1} = \frac{2}{4} = 0.5$$

Step 3: Compute RMSE

$$RMSE = \sqrt{\frac{(1-3)^2 + (2-5)^2 + (3-3)^2 + (4-2)^2 + (4-1)^2}{5}}$$
$$= \sqrt{\frac{4+9+0+4+9}{5}}$$
$$= \sqrt{\frac{26}{5}} = \sqrt{5.2} \approx 2.28$$

🔹 **Final Results:**
✅ **MAE** = 2
✅ **NMAE** = 0.5
✅ **RMSE** = 2.28

These evaluation metrics indicate how well the recommendation system's predicted ratings match the true user ratings.

1. **Mean Absolute Error (MAE) = 2**
    - On average, the predicted ratings differ from the actual ratings by **2 points**.
    - A lower MAE means the model makes more accurate predictions.
2. **Normalized Mean Absolute Error (NMAE) = 0.5**
    - The error (MAE) is **50% of the total rating range** (which is from 1 to 5).

    - A higher NMAE means larger relative errors in predictions.
3. **Root Mean Squared Error (RMSE) = 2.28**
    - RMSE penalizes large errors more than MAE.
    - Since RMSE is slightly larger than MAE, it suggests some predictions have **higher deviations** from true ratings.

Overall Interpretation

- The **errors are relatively high**, meaning the recommendation system is **not very accurate** in predicting user ratings.
- The model needs improvement, such as using **better collaborative filtering**, **adding social context**, or **using deep learning models**.

2 Evaluating Relevancy of Recommendations

We evaluate whether **recommended items** are actually **useful** to users.

Definitions

	Selected for Recommendation	Not Selected	Total
Relevant	N_{rs} (correctly recommended)	N_{rn} (missed relevant)	N_r
Irrelevant	N_{is} (wrongly recommended)	N_{in} (correctly ignored)	N_i
Total	N_s (total recommended)	N_n (not recommended)	N

1. Precision (P)

Fraction of **relevant** recommendations among all **recommended** items.

$$P = \frac{N_{rs}}{N_s}$$

2. Recall (R)

Fraction of **relevant items** that were successfully recommended.

$$R = \frac{N_{rs}}{N_r}$$

3. F-measure

Harmonic mean of **precision and recall**.

$$F = \frac{2PR}{P + R}$$

Relevancy Calculation

	Selected	Not Selected	Total
Relevant	9	15	24
Irrelevant	3	13	16
Total	12	28	40

Step 1: Compute Precision

$$P = \frac{9}{12} = 0.75$$

Step 2: Compute Recall

$$R = \frac{9}{24} = 0.375$$

Step 3: Compute F-Measure

$$F = \frac{2 \times 0.75 \times 0.375}{0.75 + 0.375}$$

$$= \frac{0.5625}{1.125} = 0.5$$

🔹 **Final Results:**

✅ **Precision** = 0.75

✅ **Recall** = 0.375

✅ **F-measure** = 0.5

These evaluation metrics indicate how well the recommendation system identifies **relevant items** for users.

Interpretation of Metrics

1. **Precision = 0.75** (75%)
    - Out of all the recommended items, **75% were actually relevant** to the user.
    - A high precision means **fewer irrelevant recommendations** were made.
2. **Recall = 0.375** (37.5%)
    - Out of all relevant items, the system was able to **recommend only 37.5%** of them.
    - A low recall suggests that the system **missed many relevant items**.
3. **F-measure = 0.5**
    - This is the **harmonic mean of precision and recall**, balancing both metrics.
    - Since recall is much lower than precision, the F-measure is also **moderate (0.5)**.

Overall Interpretation

- The recommendation system is **good at avoiding irrelevant recommendations** (high precision)

- However, it **fails to recommend many relevant items** (low recall)
- To improve, the system needs **better coverage** of relevant items, such as:
  ✅ Expanding the recommendation pool
  ✅ Using hybrid filtering (Collaborative + Content-based)
  ✅ Incorporating social/contextual data

3 Evaluating Ranking of Recommendations

We evaluate how well **predicted rankings** match the **true rankings**.

1. Spearman's Rank Correlation (ρ)

Measures **rank similarity** between predicted and true rankings.

$$\rho = 1 - \frac{6 \sum_{i=1}^{n} (x_i - y_i)^2}{n^3 - n}$$

Where:

- x_i = predicted rank
- y_i = true rank

Spearman's Rank Correlation measures the relationship between predicted and true rankings by calculating the correlation between them.

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

where:

- d_i = difference between predicted rank and true rank
- n = total number of items

Step 1: Compute Rank Differences (d_i) and Squared Differences (d_i^2)

Item	Predicted Rank (x_i)	True Rank (y_i)	Difference ($d_i = x_i - y_i$)	Squared Difference (d_i^2)
i_1	1	1	$1 - 1 = 0$	$0^2 = 0$
i_2	2	4	$2 - 4 = -2$	$(-2)^2 = 4$
i_3	3	2	$3 - 2 = 1$	$1^2 = 1$
i_4	4	3	$4 - 3 = 1$	$1^2 = 1$

Step 2: Sum of Squared Differences

$$\sum d_i^2 = 0 + 4 + 1 + 1 = 6$$

Step 3: Apply Formula

$$\rho = 1 - \frac{6(6)}{4(4^2 - 1)}$$

$$\rho = 1 - \frac{36}{4(16 - 1)}$$

$$\rho = 1 - \frac{36}{60}$$

$$\rho = 1 - 0.6$$

$$\rho = 0.4$$

Final Result:

✅ **Spearman's Rank Correlation (ϱ) = 0.4**

This indicates a **moderate positive correlation** between the predicted rankings and true rankings. The recommendation system is somewhat aligned with actual user preferences but has room for improvement

2. Kendall's Tau (τ)

Measures **order agreement** between pairs of rankings.

$$\tau = \frac{c - d}{\frac{n(n-1)}{2}}$$

Where:

- c = number of **concordant** pairs
- d = number of **discordant** pairs

Rank Correlation Calculation

Item	Predicted Rank	True Rank
i_1	1	1
i_2	2	4
i_3	3	2
i_4	4	3

Step 1: Identify Concordant/Discordant Pairs

✅ (i_1, i_2) → concordant
✅ (i_1, i_3) → concordant
✅ (i_1, i_4) → concordant
❌ (i_2, i_3) → discordant
❌ (i_2, i_4) → discordant
✅ (i_3, i_4) → concordant

Step 2: Compute Kendall's Tau

$$\tau = \frac{4 - 2}{6} = 0.33$$

🔹 **Final Result: Kendall's Tau** = 0.33

Interpretation of Kendall's Tau = 0.33

🔹 **Kendall's Tau (τ)** measures the **correlation between predicted and true rankings** in a recommendation system.
🔹 The value of **τ ranges from -1 to 1**:

- **+1** → Perfect agreement (all pairs are concordant) ✅
- **0** → No correlation (random ranking) 🤷
- **-1** → Perfect disagreement (all pairs are discordant) ❌

What Does τ = 0.33 Indicate?

✔ **Weak Positive Correlation**:

- The predicted rankings are somewhat aligned with the true rankings.
- There is **some agreement**, but there are also **several incorrect orderings** (discordant pairs).

❗ **Improvement Needed**:

- Since τ = **0.33 is closer to 0 than 1**, the ranking system is **not very strong**.

- There are **more discordant pairs than expected**, meaning some recommended items are ranked incorrectly.