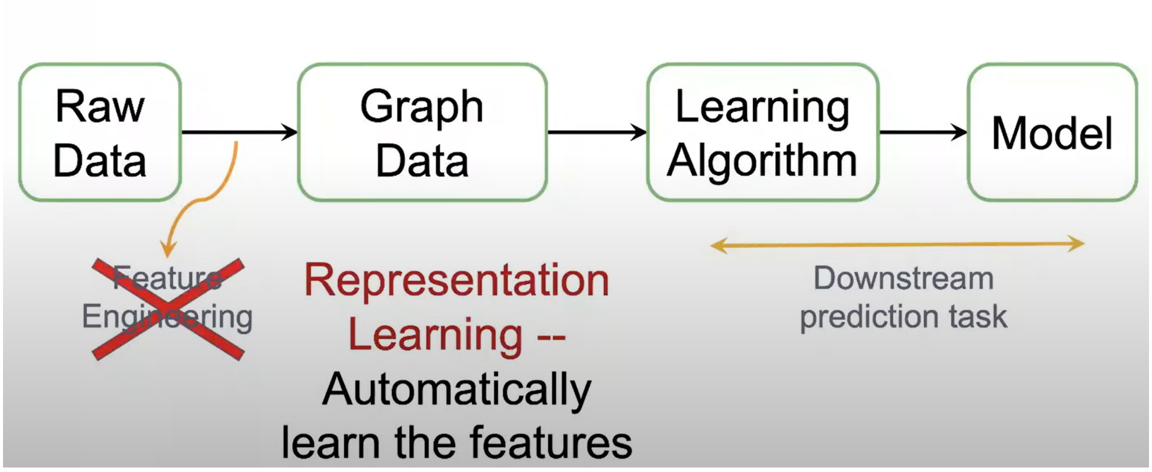
Lecture 1

Motivation for Graph ML

- feature engineering 없이도 모델 구축하기 위해



- 이 수업에서 다룰 것

We are going to cover various topics in Machine Learning and Representation Learning for graph structured data:

- Traditional methods: Graphlets, Graph Kernels
- Methods for node embeddings: DeepWalk, Node2Vec
- Graph Neural Networks: GCN, GraphSAGE, GAT, Theory of GNNs
- Knowledge graphs and reasoning: TransE, BetaE
- Deep generative models for graphs
- Applications to Biomedicine, Science, Industry

Applications of Graph ML

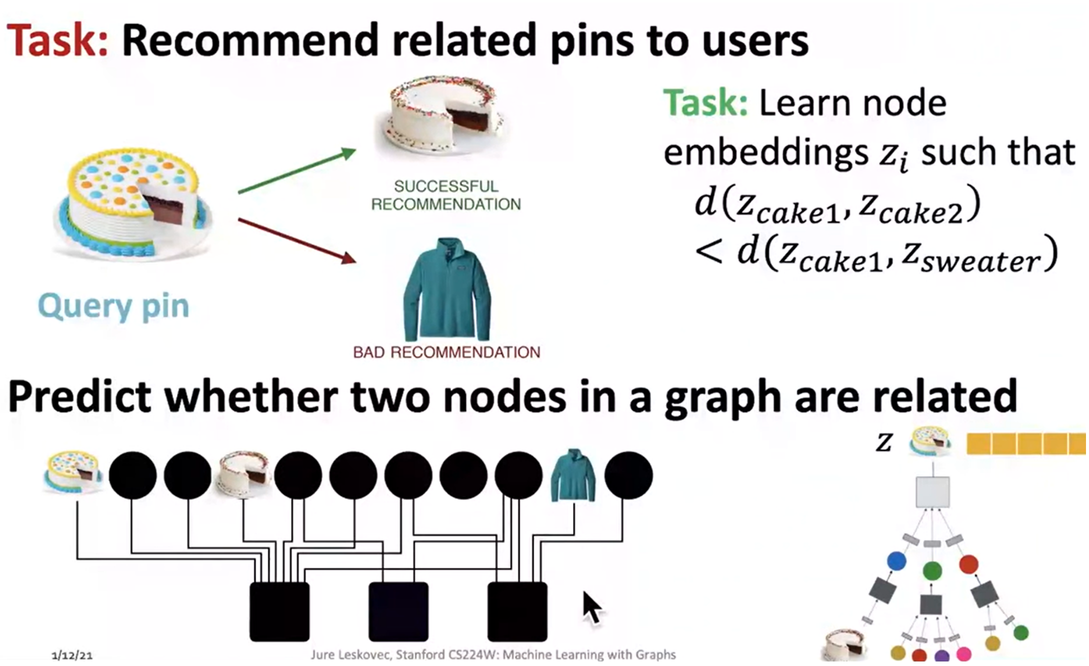
- Classic Graph ML task

- **Node classification**: Predict a property of a node
  - **Example:** Categorize online users / items
- **Link prediction**: Predict whether there are missing links between two nodes
  - **Example:** Knowledge graph completion
- **Graph classification**: Categorize different graphs
  - **Example:** Molecule property prediction
- **Clustering**: Detect if nodes form a community
  - **Example:** Social circle detection
- **Other tasks**:
  - **Graph generation**: Drug discovery
  - **Graph evolution**: Physical simulation

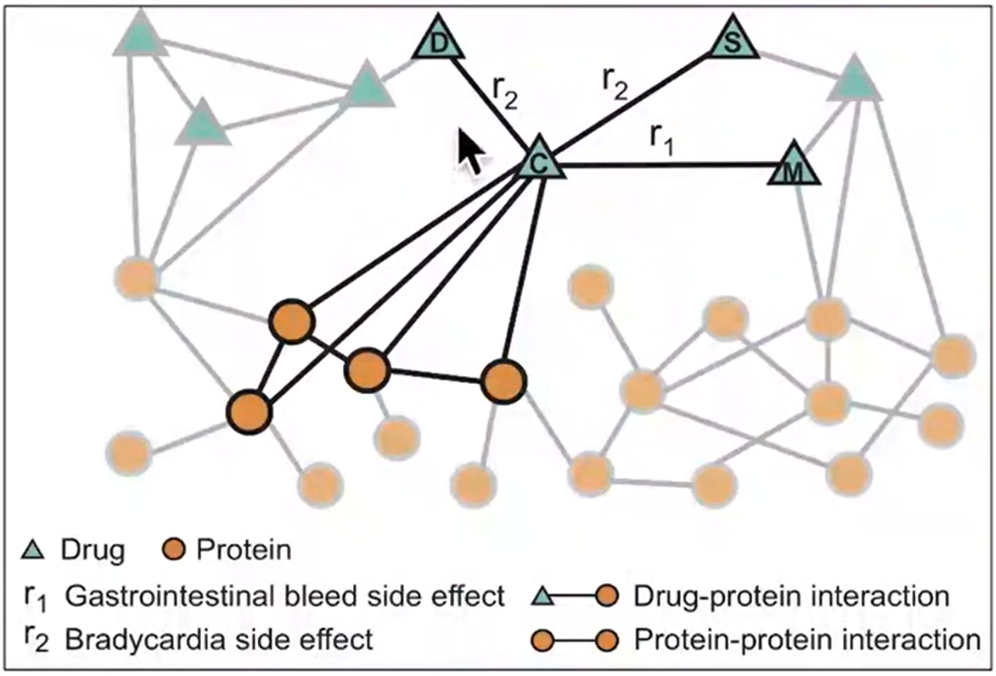
- Example of Node Level

- Protein folding → alphafold로 해결
    - node : protein sequence에서의 amino acid
    - edge : residue들 간 proximity

- Example of Edge Level
  - Recommender systems
    - node : users + items
    - edge : user-item interaction
  - PinSage : Graph-based Recommender



  - Drug side effect
    - 합병증 환자들은 여러 약물들을 한번에 복용하는데 이때 같이 복용할 때의 부작용 예측
    - node는 drug와 protein / edge는 interaction



        - missing connection을 찾는 것이 문제

- Example of Subgraph Level
    - Traffic prediction
        - node : road segment
        - edge : connectivity between road segment

- Example of Graph Level
    - antibiotics
        - node : atom
        - edge : chemical bond
    - molecule generation / optimization

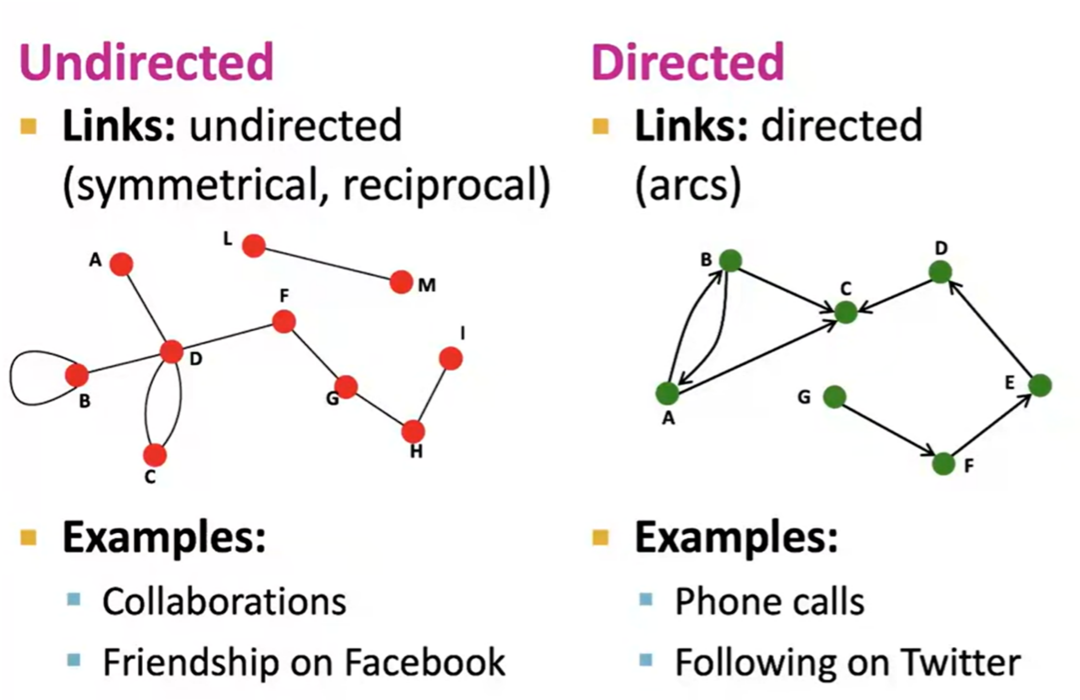
Choice of graph representation

- Component of network
    - objects (N) : nodes, vertices

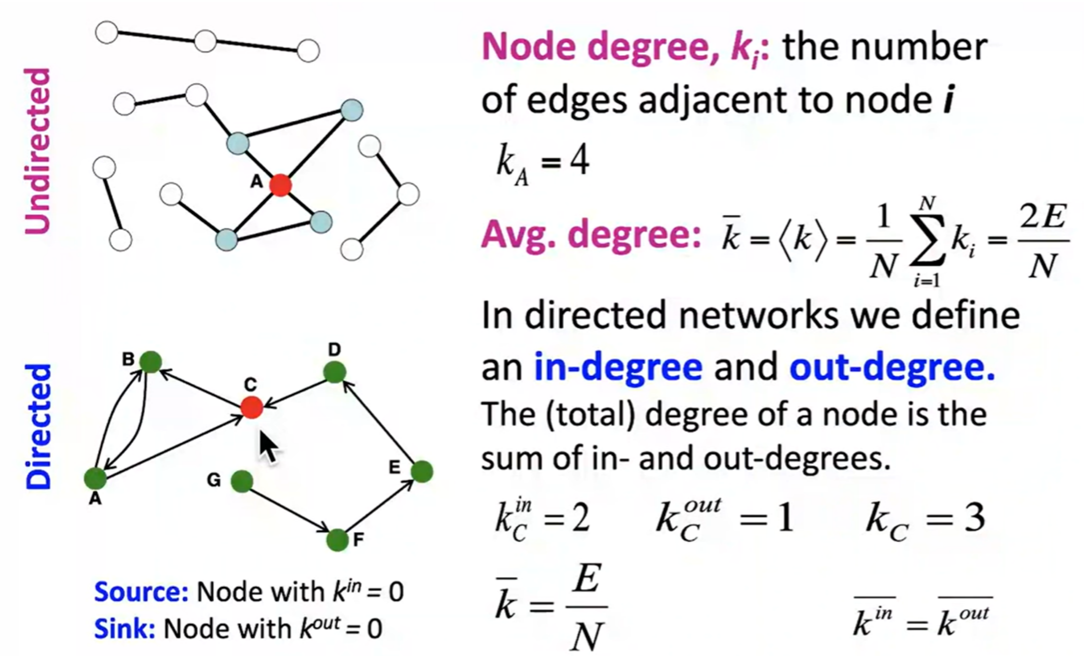
- interaction (E) : links, edges
- system (G(N,E)) : network, graph

- Choosing a proper representation 또한 매우 중요함
  - node와 edge를 어떻게 설정할 것인가

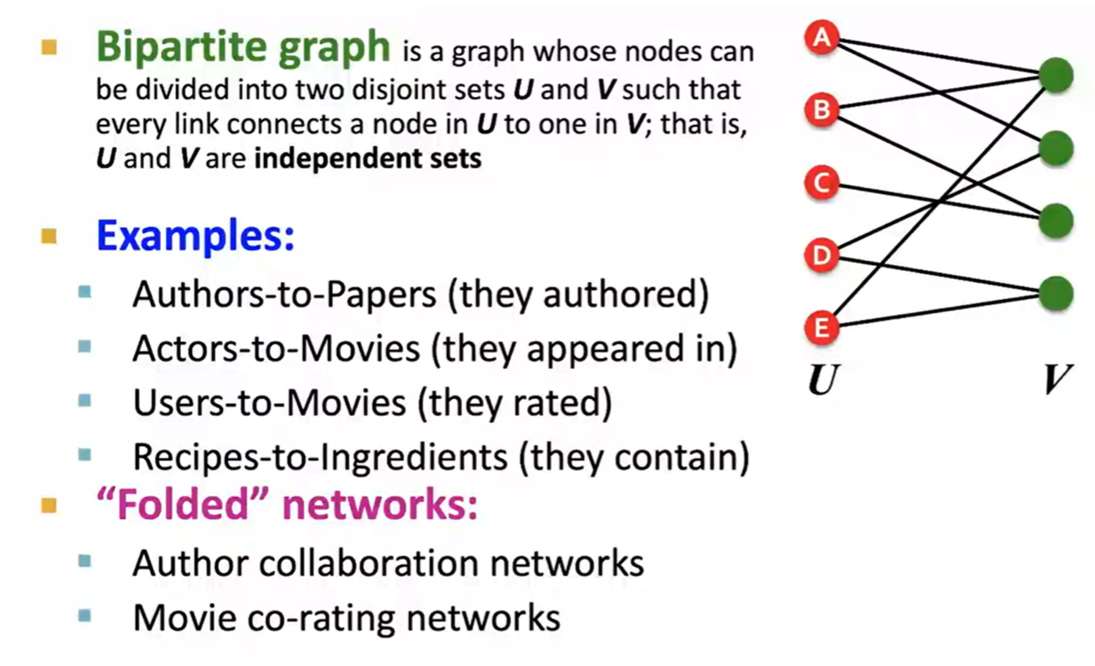
- Directed vs Undirected Graphs



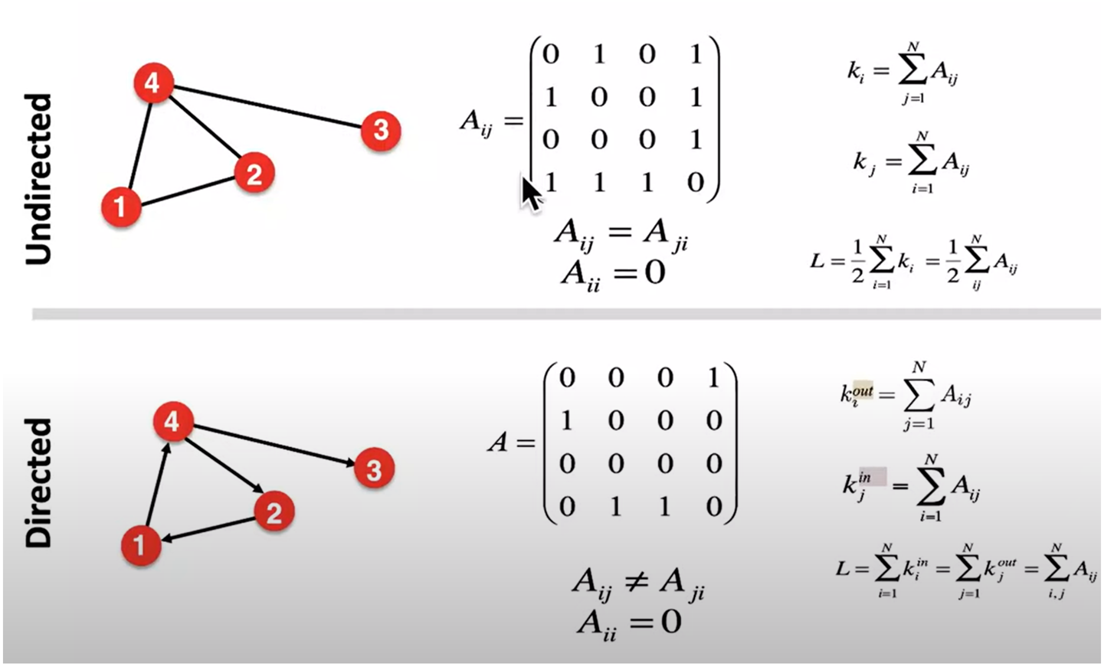
  - Node degree



- Bipartite Graph

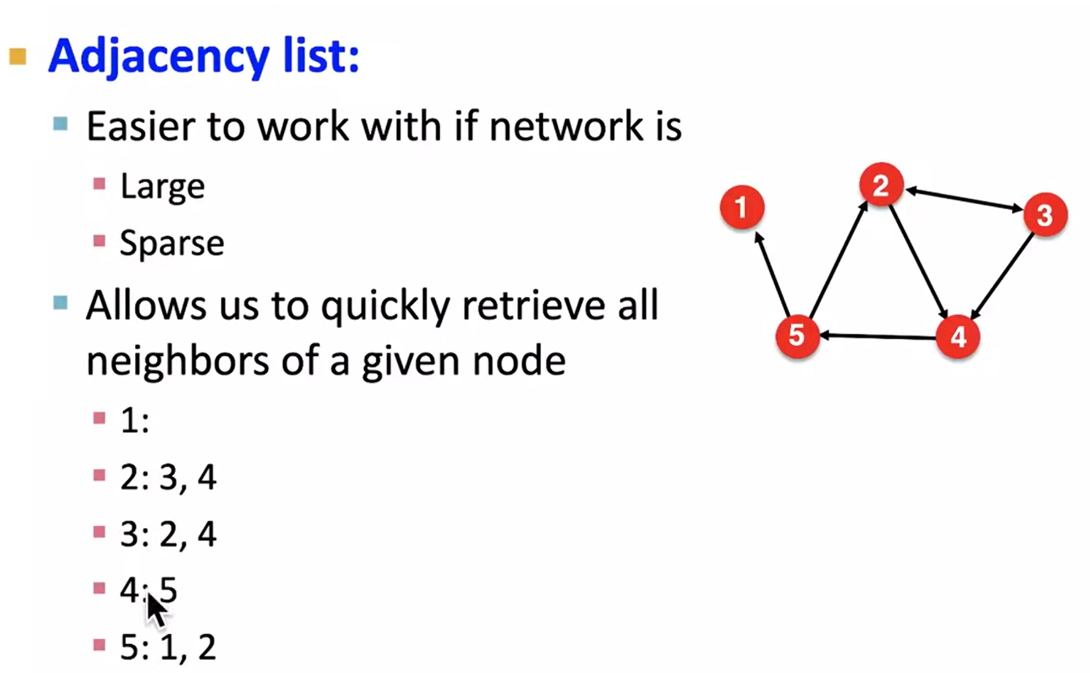


- Representing Graph : Adjacency Matrix



- 대부분의 adjacency matrix는 sparse matrix임

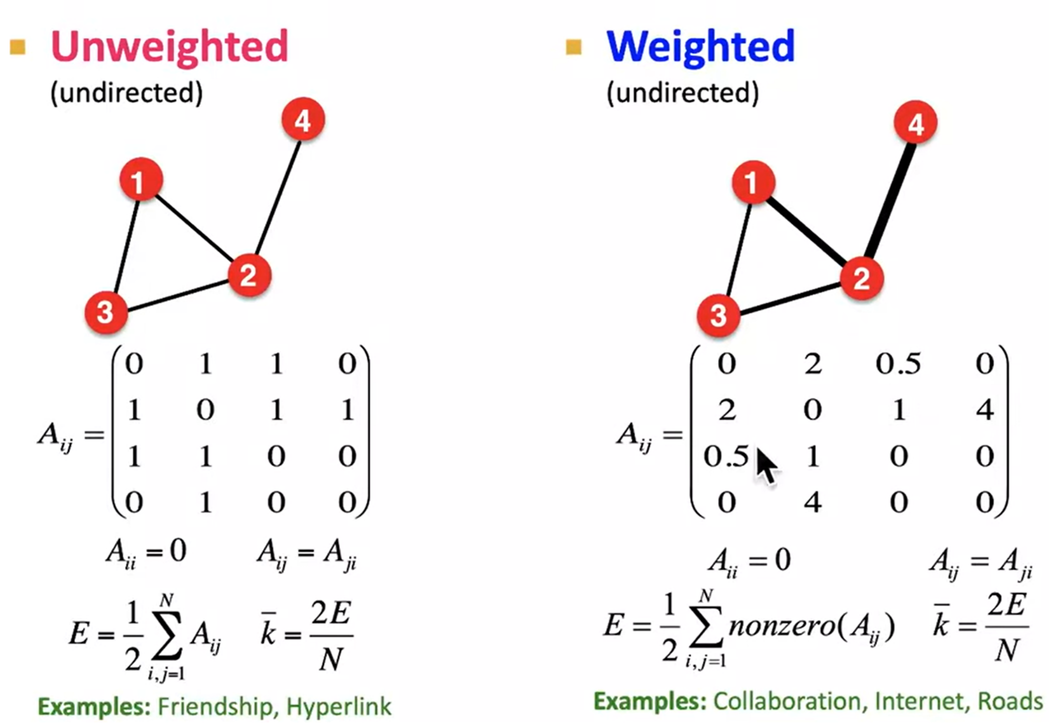
- Representing Graph : Adjacency list

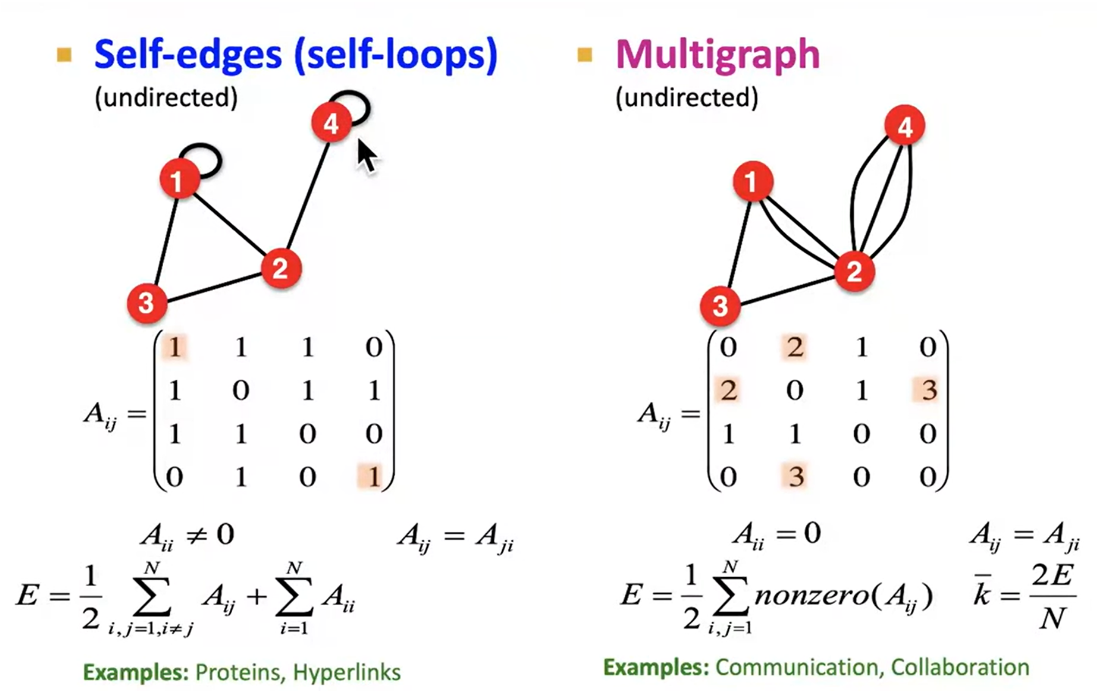


- Node and Edge attribute
  - weight

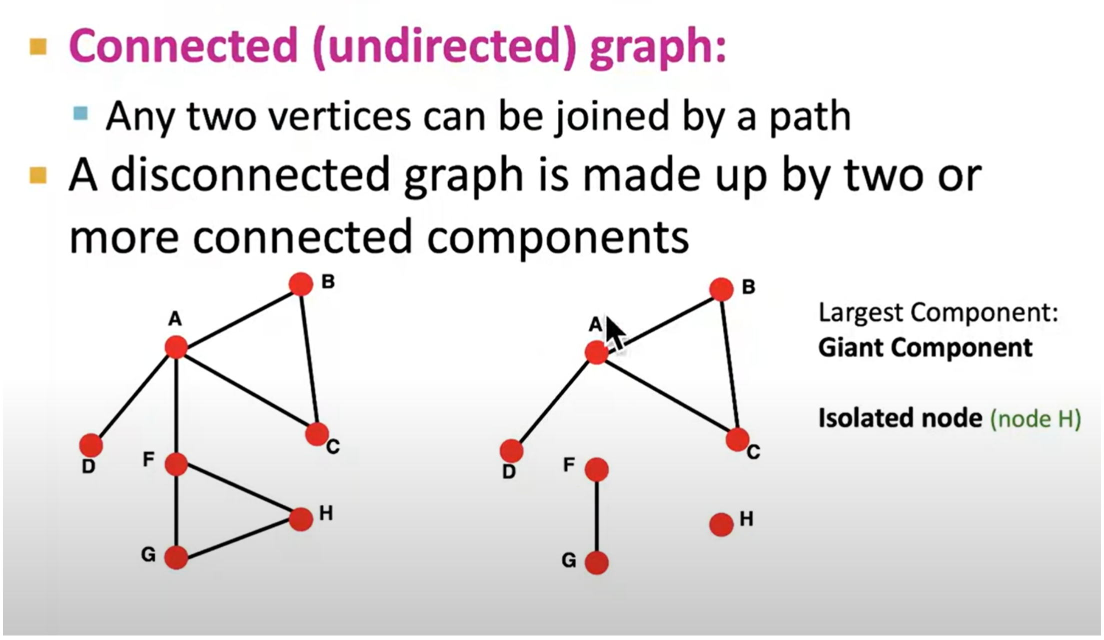
- ranking
- type
- sign
- properties depending on the structure of the rest of the graph

- More types of graphs

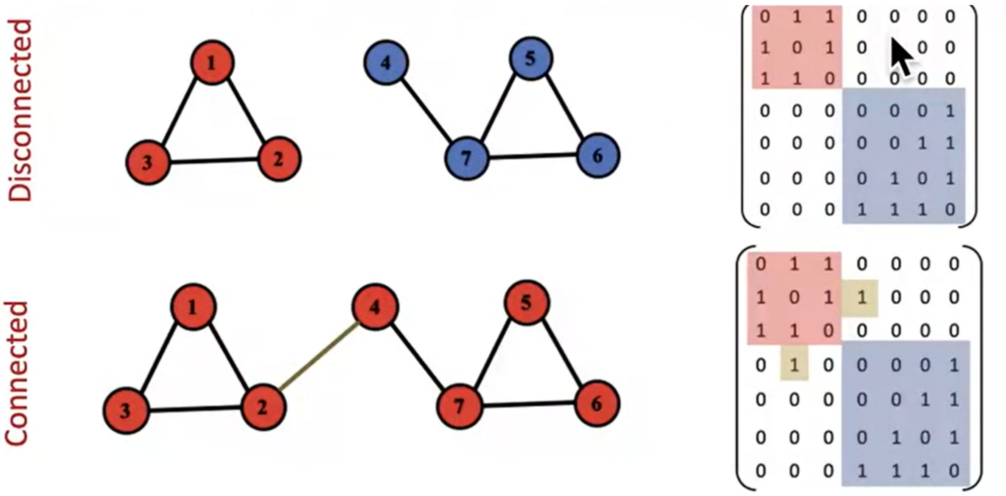




- Connectivity of Undirected graphs



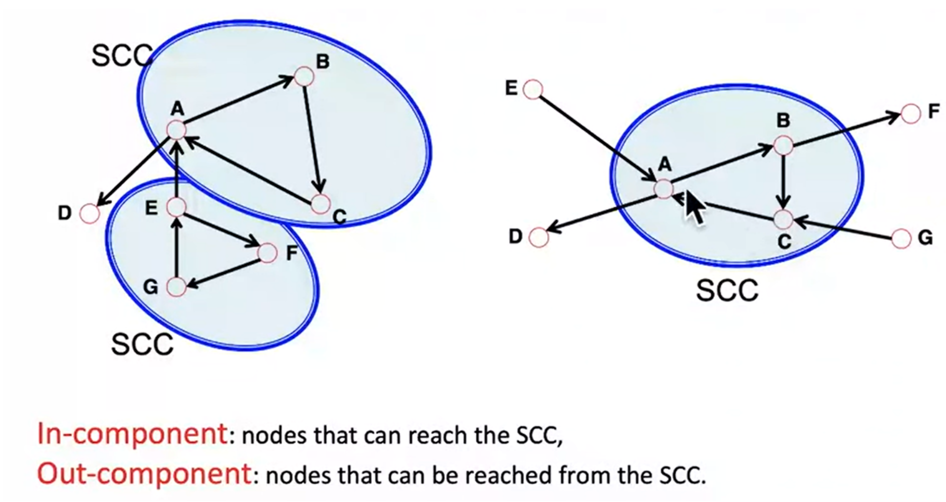
  - adjacency matrix가 block diagonal form으로 만들어질 수 있음



- 종류
  - strongly connected directed graph

    노드들끼리 link 있음 (A-B path와 B-A path 둘 다 존재

    SCC (strongly connected components)



  - Weakly connected directed graph