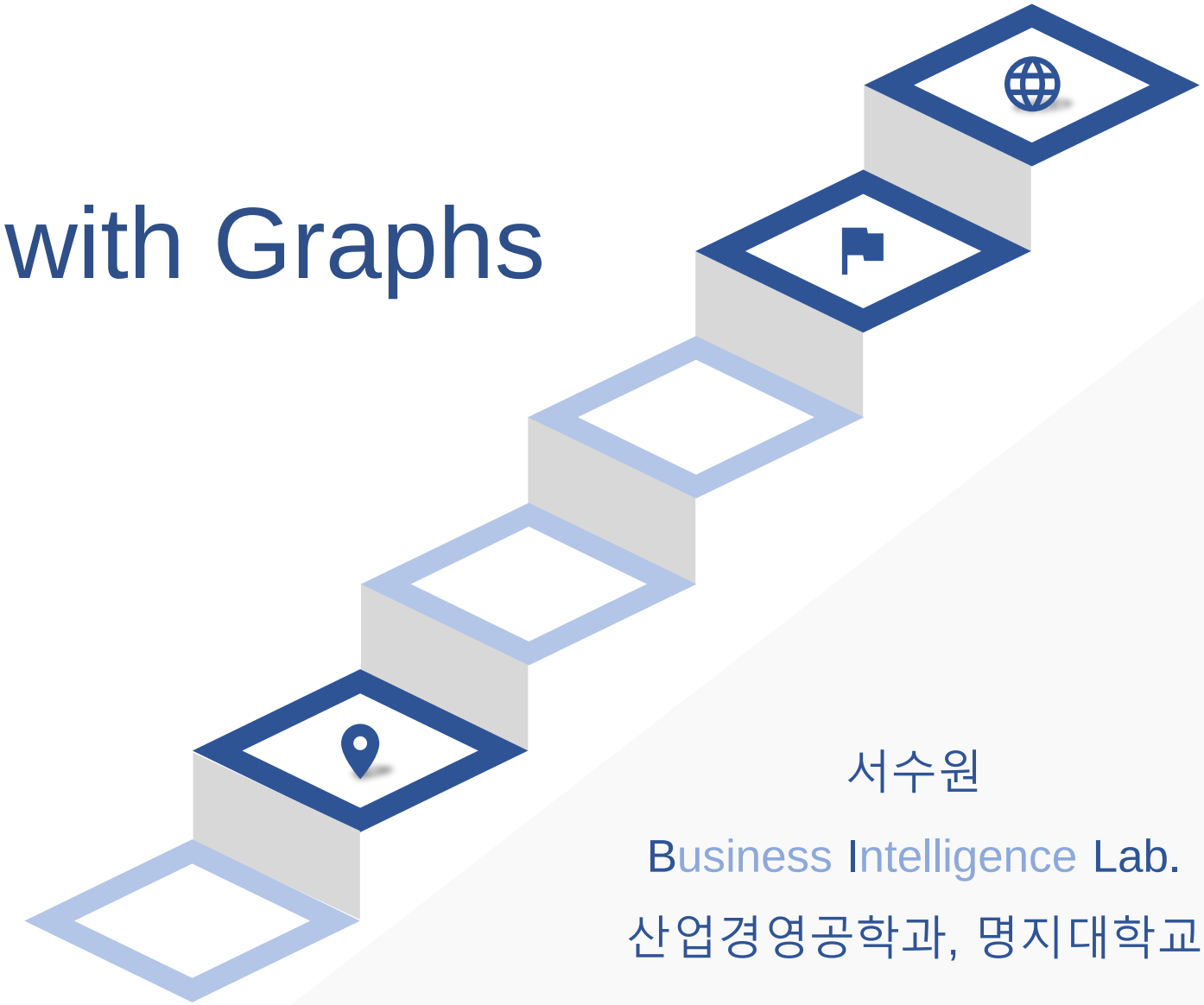
20230629

Machine Learning with Graphs



서수원

Business Intelligence Lab.

산업경영공학과, 명지대학교

01

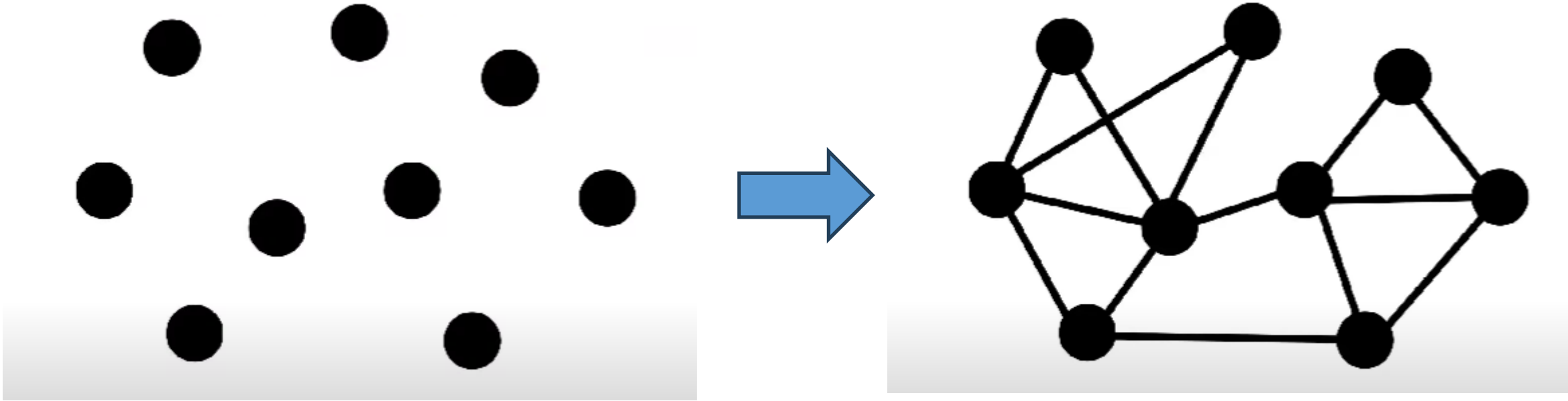
Why Graphs

사전 정보

- Networks vs Graphs
  - Network는 실제 시스템을 의미한다.
    - ✓ Web, Social network
    - ✓ 표현 : Network, node, link
  - Graph는 network의 수학적 표현이다.
    - ✓ Web graph, Social graph
    - ✓ 표현 : Graph, vertex, edge
  - 사실상 Graph = Network, node = vertex, edge = link를 동의어로 봐도 무관하다.

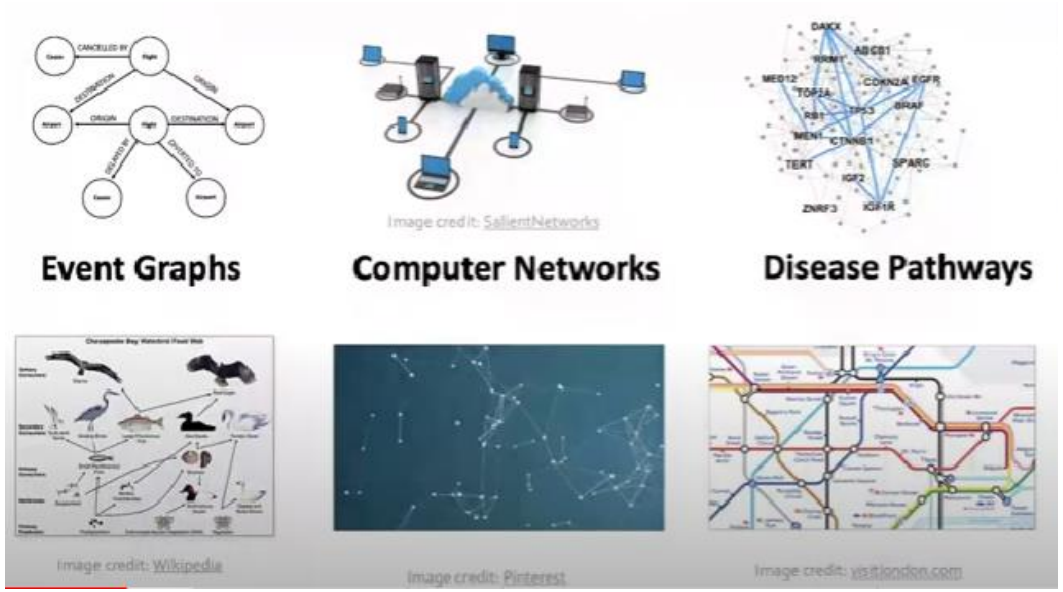
Why Graphs?

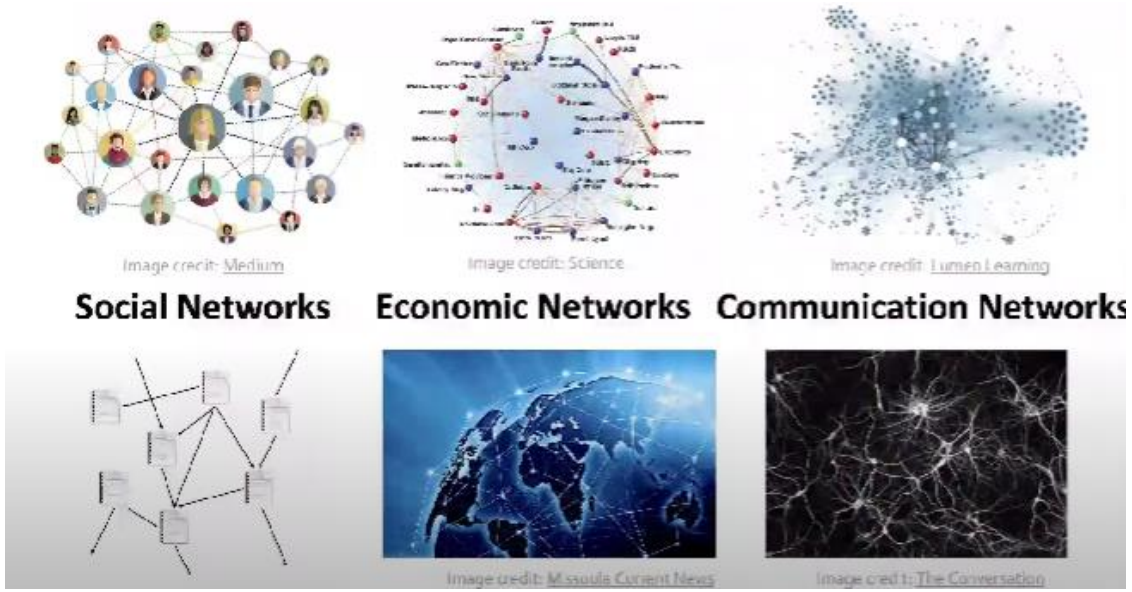
- Why Graphs?
  - 그래프는 관계나 상호작용이 있는 엔티티를 설명하고 분석하기 위한 일반적인 언어이다.
    - ✓ 즉 주어진 엔티티를 고립된 데이터로 생각하기 보다는, 서로 관계가 있는(네트워크가 있는) 측면에서 생각한다는 의미이다.



Why Graphs?

- Many Types of Data are Graphs
  - 따라서 관계가 있는 데이터들은 그래프로 표현이 가능하다.
    - ✓ 많은 시스템 뒤엔 구성 요소간의 상호 작용을 적용하는 다이어그램, 네트워크가 있다.
    - ✓ 예를 들면, 지하철노선도, 질병경로, 컴퓨터네트워크, Social Networks, 인터넷, 신경망 등이 있다.





Why Graphs?

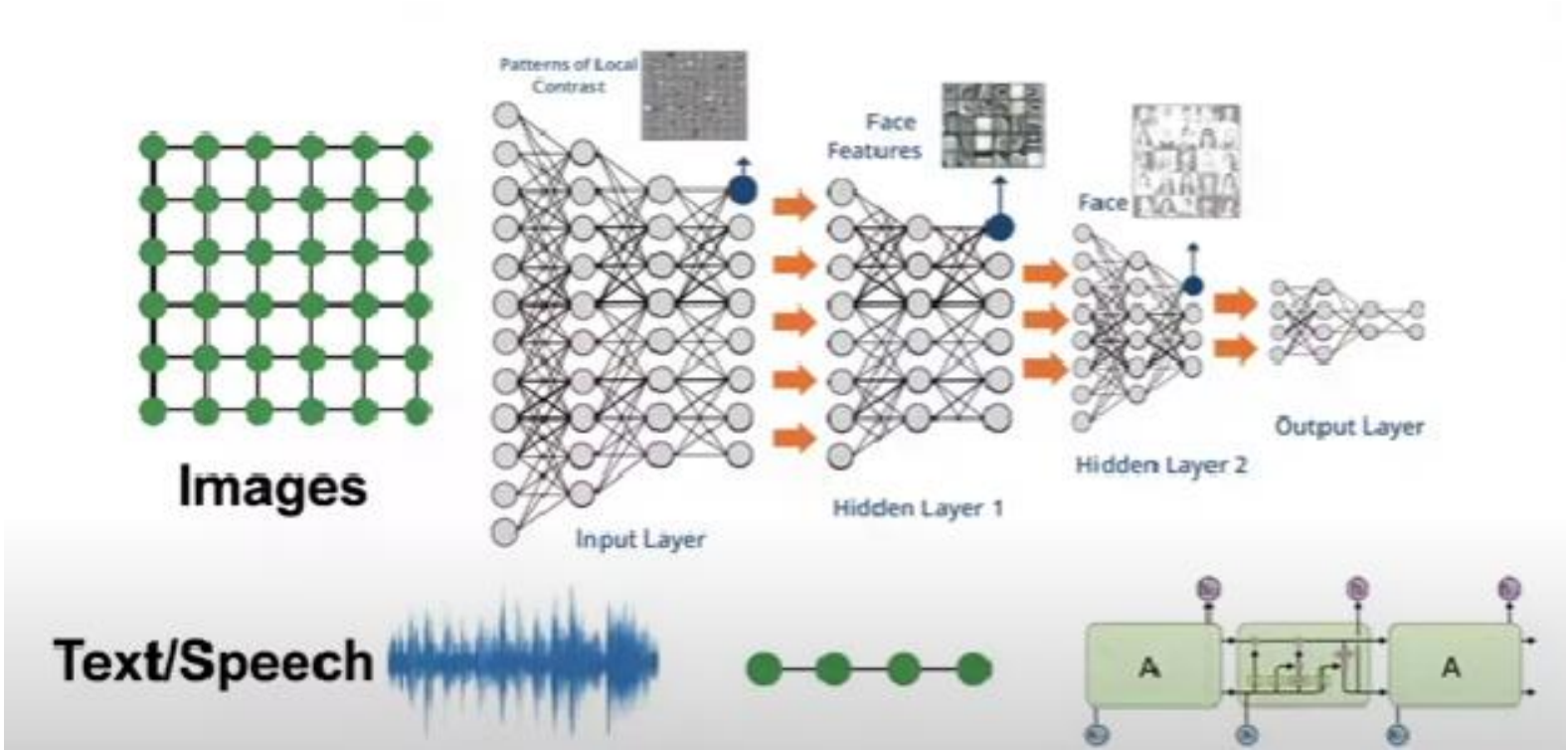
- Networks(also known as Natural Graphs)
  - 관계가 있는 모든 영역에서 근본적인 현상을 잘 설명 할 수 있게 해주는 중요한 부분이다.
  - 네트워크로 표현 될 수 있는 데이터는 두개의 종류가 있다.
    - ✓ Network(Natural Graphs)
      - ❖ 자연스럽게 도메인이 그래프로 표현 될 수 있는 것을 의미한다.
      - ❖ 예를 들면 Social Network, 70억명 이상의 개인이 모인 사회, 전자 장치를 연결하는 통신 시스템, 유전자 단백질의 상호작용, 뇌와 뉴런의 연결이 있다.
    - ✓ Graphs(as a representation)
      - ❖ 정보와 지식은 조직되고 연결된다.
      - ❖ 소프트웨어도 그래프로 표현이 가능하다.
      - ❖ 비슷한 데이터를 연결하여 유사성 네트워크를 만들 수 있다.
        - » 예를 들면 치와와, 리트리버 등 유사한 동물을 연결하여 하나의 네트워크 형성이 가능하다.
    - ✓ 종종 네트워크와 그래프의 구분은 어렵다.
    - ✓ 많은 도메인에 개념을 적용 할 수 있다.

Graphs: Machine Learning

- Main Question
  - 어떻게 이 구조를 활용하여 더 좋은 예측을 할 수 있을까?
- Graphs : Machine Learning
  - 복잡한 도메인은 많은 관계를 가지고 있는 구조를 가지고 있다.
    - ✓ 복잡한 도메인은 관계형 그래프로 표현이 가능하다.
    - ✓ 관계를 명시적으로 모델링 함으로써 우리는 더 정확한 예측을 할 수 있다.

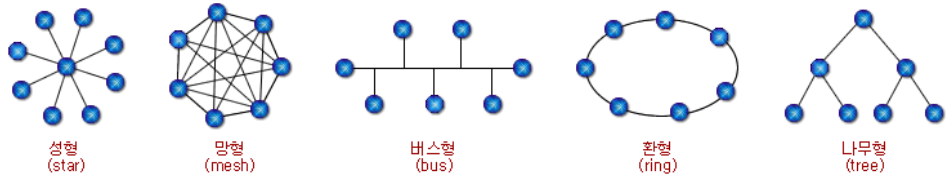
Compare Network and Deep Learning

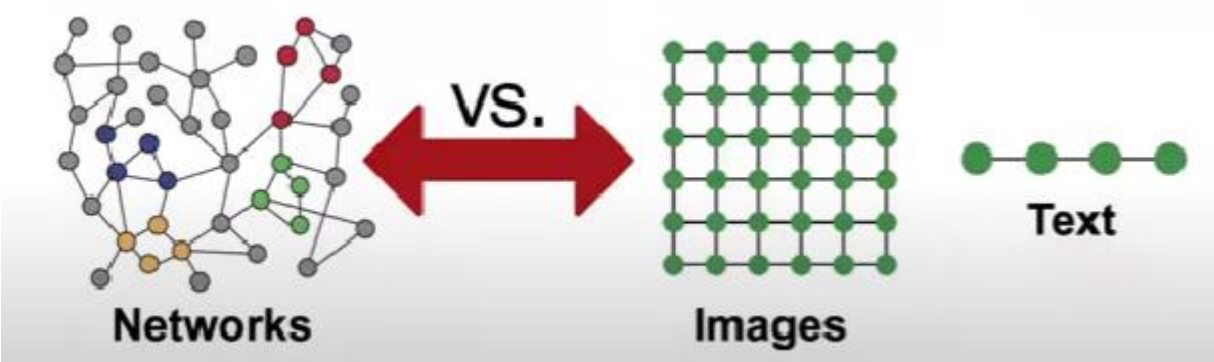
- Deep Learning
  - 단순한 순서가 있는 데이터나, 그리드로 표현될 수 있는 것에 특화 되어 있다.
    - ✓ 문자나 음성은 선형 순서가 있고, 이미지는 크기를 변형하여 고정 그리드(크기)로 나타낼 수 있기 때문에 딥러닝의 이용이 용이하다.



Compare Network and Deep Learning

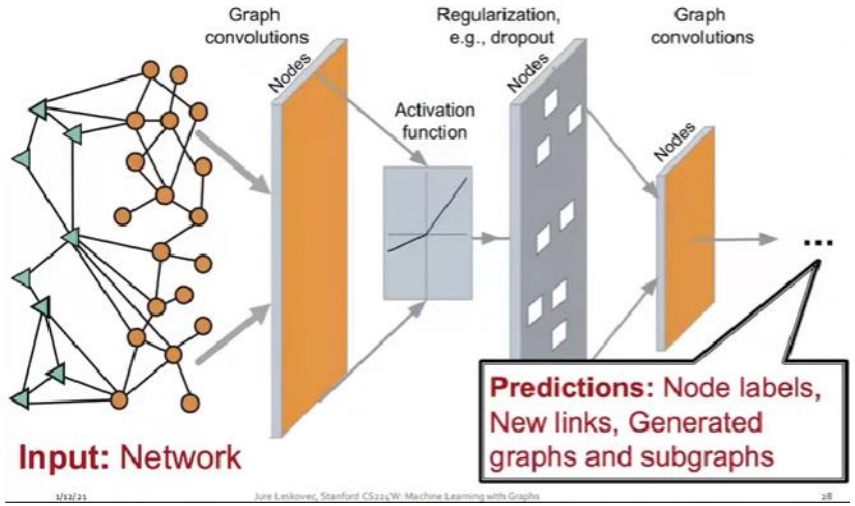
- Networks
  - 임의의 크기와 복잡한 토폴로지를 가지고 있다.
    - ✓ 토폴로지란 노드와 선의 연결을 의미한다.
  - 이미지나 텍스트 데이터와 같은 공간지역성도 없다.
    - ✓ 텍스트는 순서를 알고 이미지는 서로 다른 픽셀의 위치를 알 수 있다.
  - 네트워크에는 기준점이 없다.
    - ✓ 딥러닝을 하기 위한 고정적인 순서가 없다는 것을 의미한다.
  - 종종 네트워크는 동적이며 다중모델 기능을 가지고 있다.

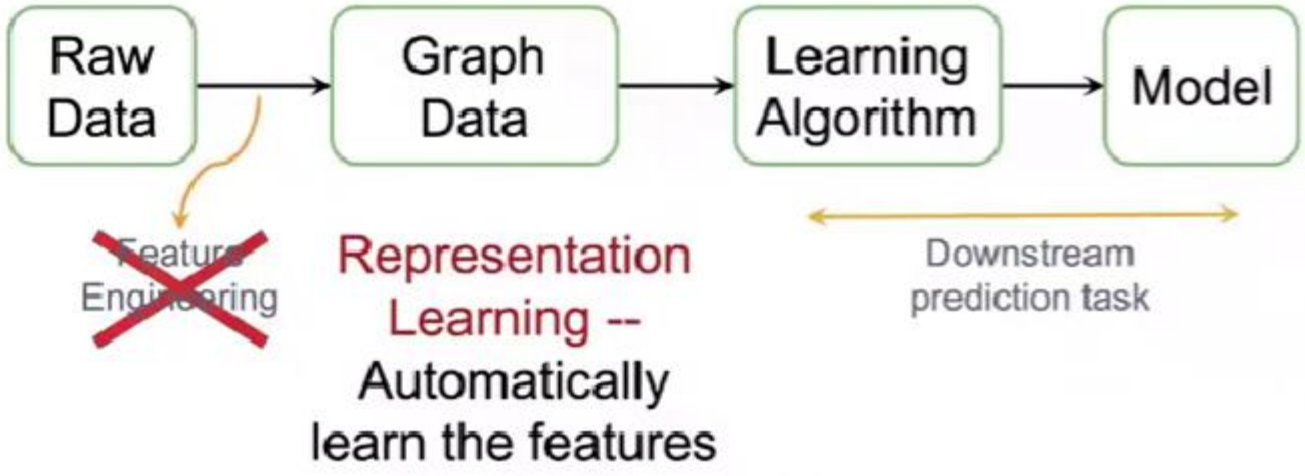




We will learn

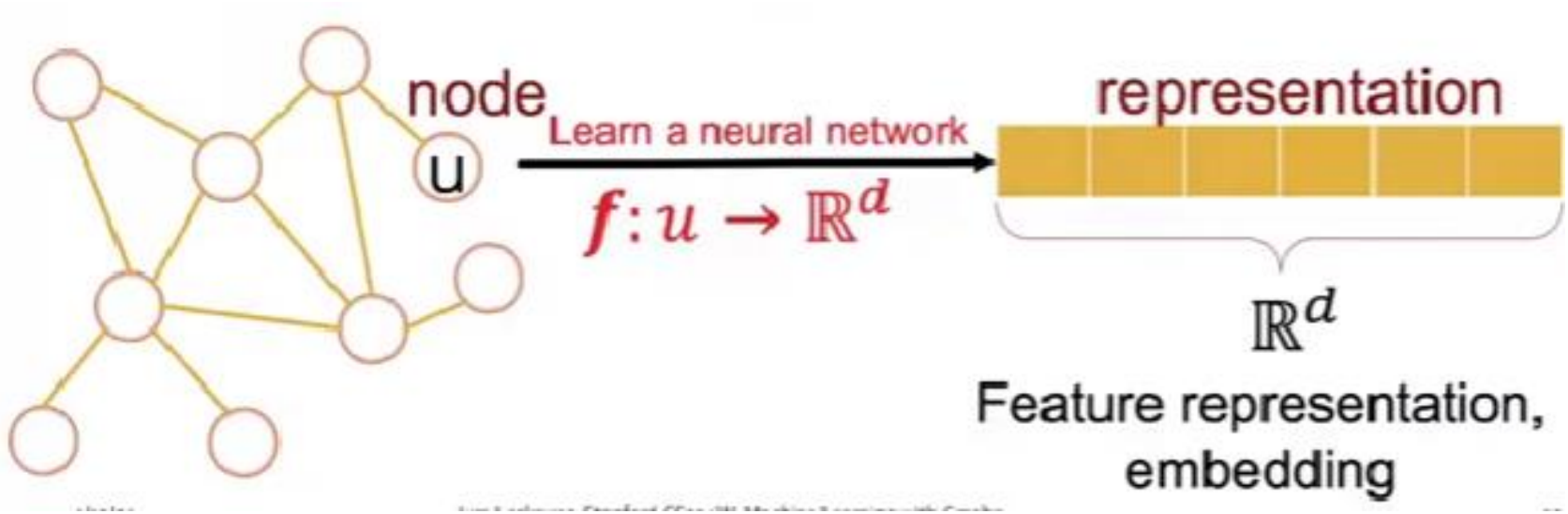
- Graphs
  - 그래프와 같이 복잡한 데이터 유형에 적용 할 수 있는 신경망 방법을 배울 것 이다.
  - 관계형 그래프는 딥러닝의 연구에 새로운 지평을 열 것이다.
  - 그래프를 Input으로 넣고, 그래프에 관한 여러 예측이 가능하다.
    - ✓ 그래프를 활용하면 신경망을 설계 할 때 인간의 개입(Feature Engineering)가 필요 없게 된다.
    - ✓ 그래프의 특징을 자동으로 추출하여 머신러닝에 활용이 가능하다.





We will learn

- Graphs
  - 그래프를 d차원의 벡터로 표현한다.
    - ✓ 비슷한 노드는 비슷한 공간에 임베딩 되게 표현을 할 수 있다.
    - ✓ 이때 사용되는 F를 배울 것이다.
  - 그래프를 분석할 때 적용 가능한 방법론들을 다룰 예정이다.

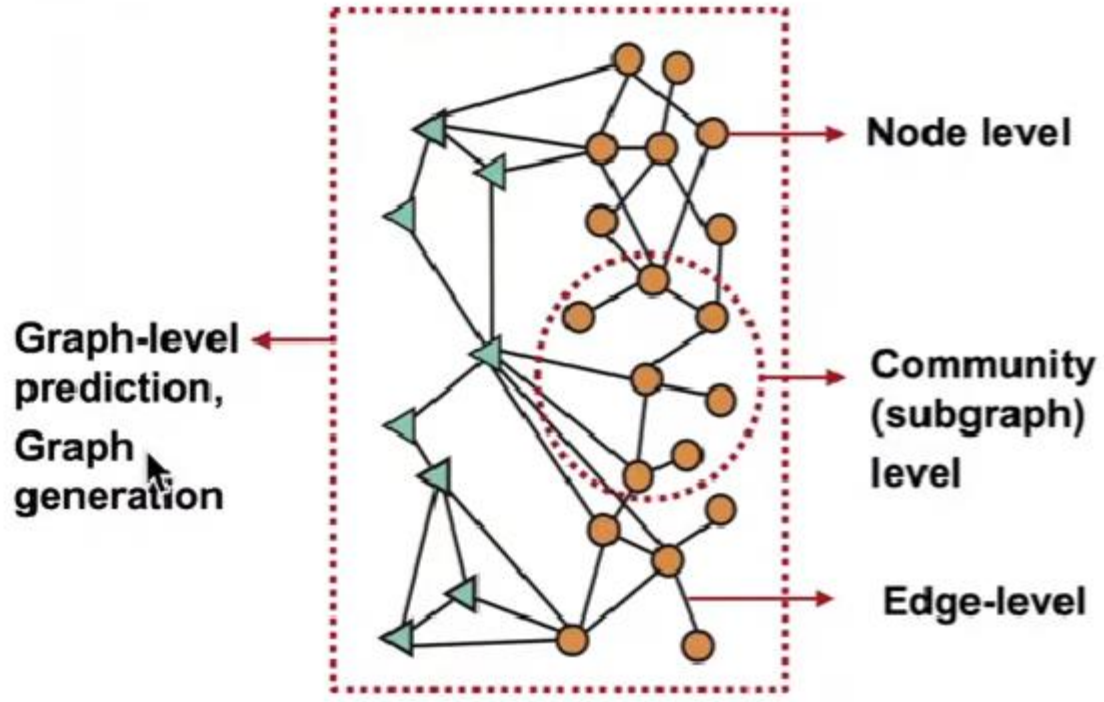


02

Applications of Graph ML

Different Types of Tasks

- Different Types of Tasks
  - 하나의 그래프를 가지고 다양한 유형의 작업을 할 수 있다.
    - ✓ 노드 단위,연결 단위,그룹 단위, 그래프 전체의 단위로 분석을 진행 할 수 있다.



Different Types of Tasks

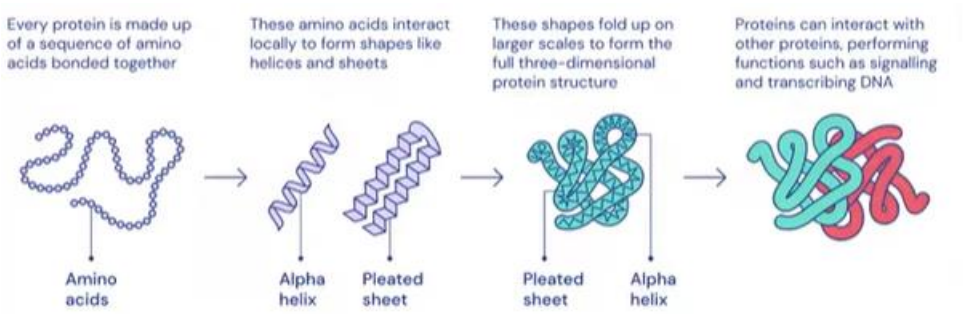
- Different Types of Tasks
  - 노드 분류 : 노드의 특징을 예측한다.
    - ✓ 예 : 온라인 유저나 아이템을 분류 하는 것을 의미한다.
  - 연결 예측 : 두 노드 사이에 빠진 연결이 있는지 확인한다.
    - ✓ 예 : 지식 그래프의 완성을 예로 볼 수 있다.
  - 그래프 분류 : 같은 수준의 다양한 그래프를 분류한다.
    - ✓ 예 : 분자를 그래프로 표현한 다음, 분자의 특징을 예측 할 수 있다.
      이는 분자들을 조합하여 만든 약물의 특징을 예측 하려 하는 설계에 적용될 수 있는 작업이다.
  - 군집화 : 노드가 군집의 형태를 띄는지를 확인한다.
    - ✓ 예: Social circle detection
  - 이런 그래프 분석은 응용의 가능성이 높다.

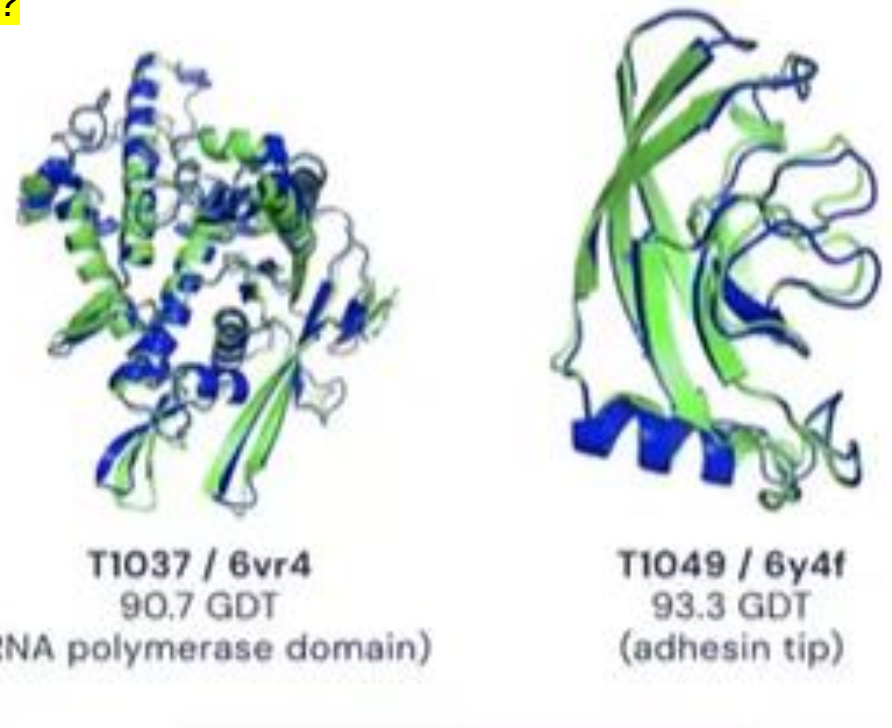
03

Example of Node-level ML Tasks

Example of Node-level ML Tasks

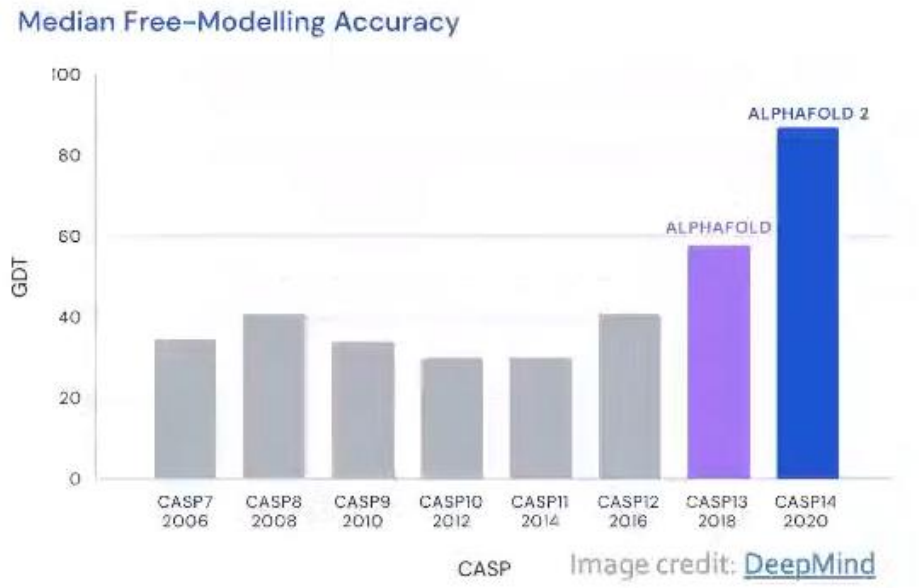
- Protein Folding
  - 단백질은 아미노산으로 구성되어 있다.
    - ✓ 자기력과 같은 힘에 의해, 단백질은 실제로는 복잡한 형태로 접혀 있다.
    - ✓ 이 복잡한 형태의 단백질을 3D로 계산하여 구조를 예측하는 것이 생물학계의 큰 일 이였다.
  - Question
    - ✓ 아미노산 서열이 주어졌을 때 3D로 예측이 가능 한가?





Example of Node-level ML Tasks

- Protein Folding
  - 단백질은 아미노산으로 구성되어 있다.
    - ✓ 자기력과 같은 힘에 의해, 단백질은 실제로는 복잡한 형태로 접혀 있다.
    - ✓ 이 복잡한 형태의 단백질을 3D로 계산하여 구조를 예측하는 것이 생물학계의 큰 일 이였다.
  - Question
    - ✓ 아미노산 서열이 주어졌을 때 3D로 예측이 가능 한가?
      - ❖ 단백질을 그래프로 나타내는 방법을 통해 정확도를 90%까지 올렸다.

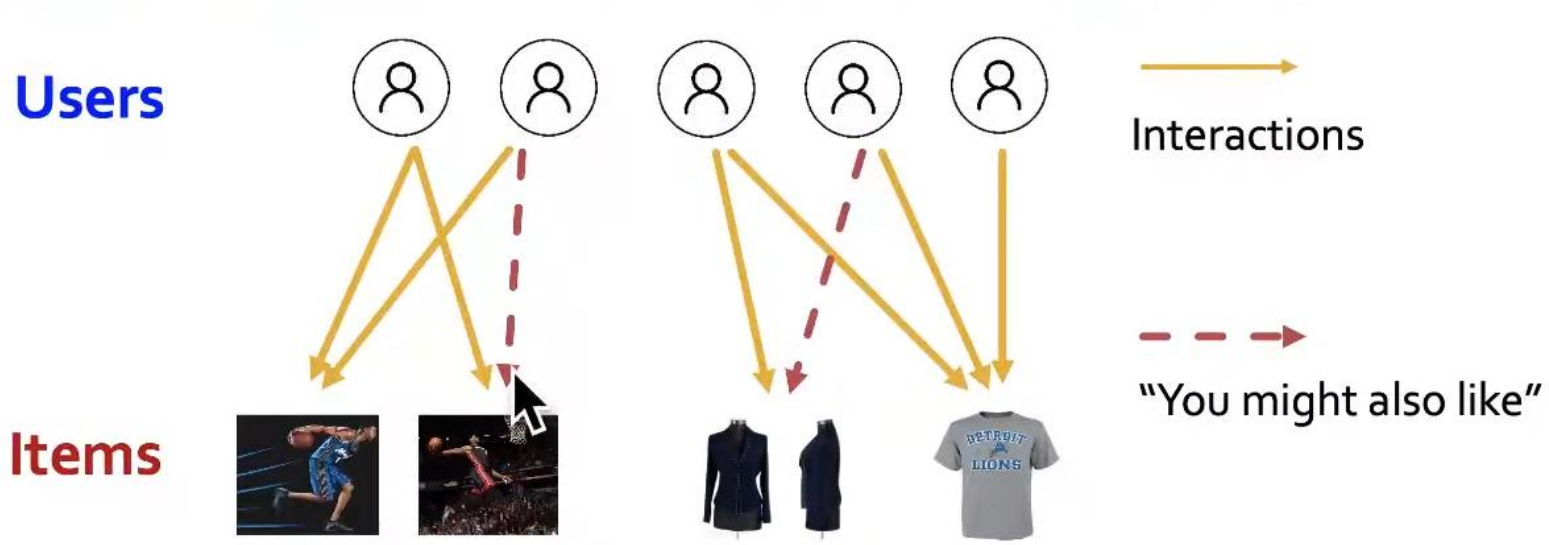


04

Examples of Edge-level ML tasks

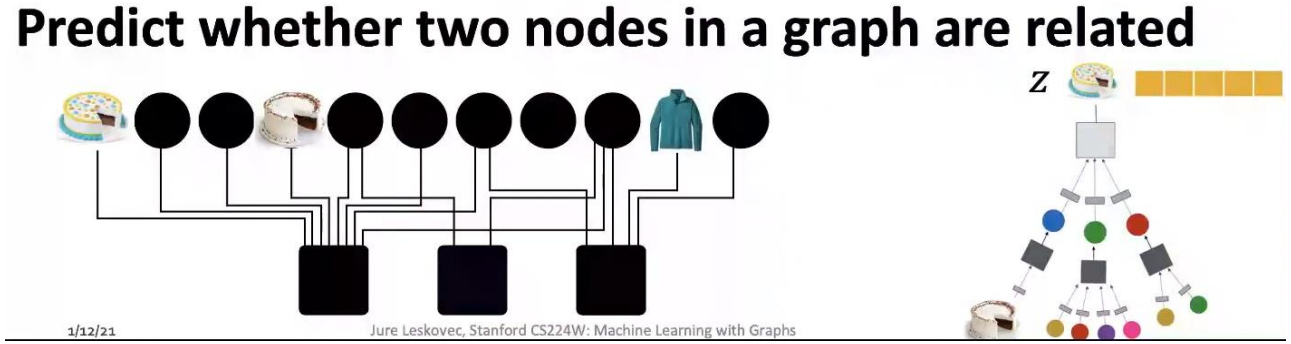
Example of edge-level ML Tasks

- Recommender Systems
  - 사용자가 아이템과 상호작용 하는 것을 예로 들 수 있다.
    - ✓ 아이템은 영화를 보거나, 상품을 사거나 하는 등의 행위를 의미한다.
    - ✓ 노드는 사용자와 아이템, 선은 유저와 아이템의 상호작용을 나타낸다.
  - 목표는 사용자가 좋아할만한 아이템을 찾아 주는 것 이다.
    - ✓ 페이스북, 알리바바 등 여러 기업에서 사용하는 방법이다.



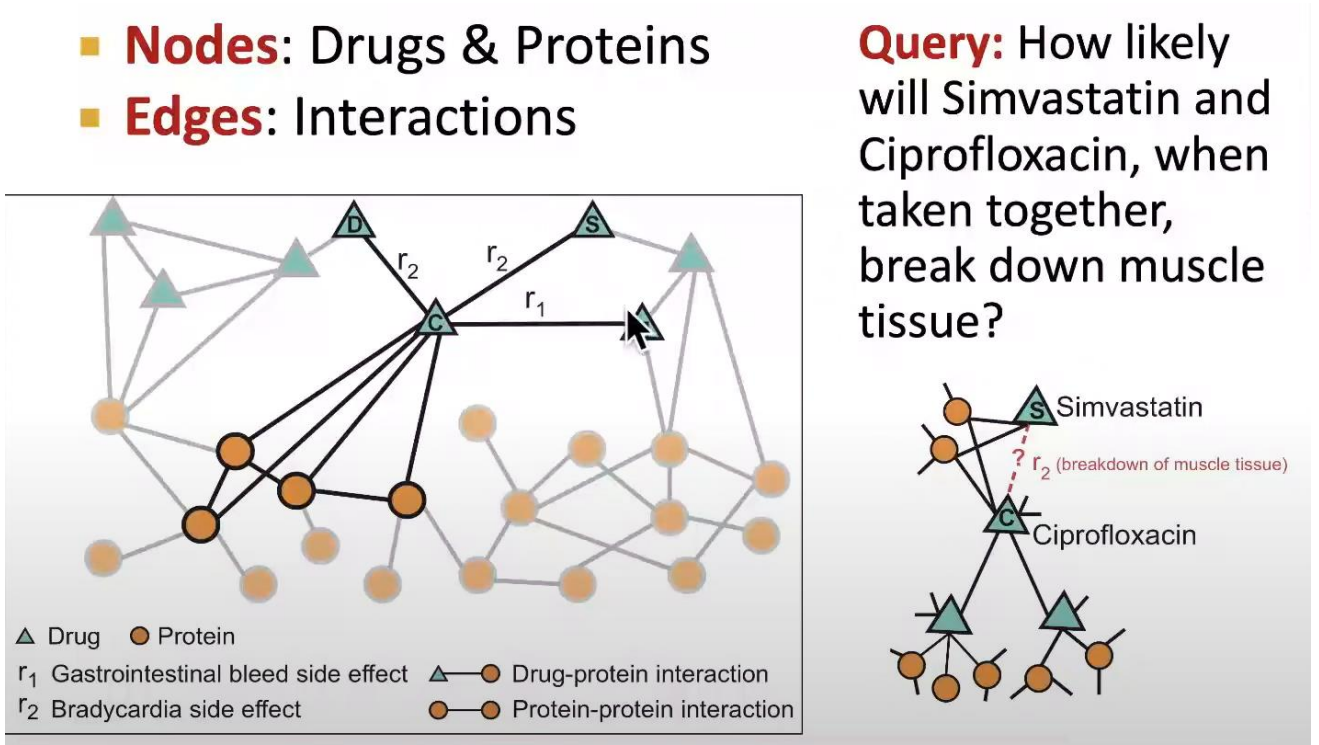
Example of edge-level ML Tasks

- Recommender Systems
  - Pinterest의 예
    - ✓ 케잌과 옷은 각각의 노드이다.
    - ✓ 케잌들 끼리 더 가까이 임베딩 되는 것이 바람직 하다.
  - 이를 수행하는 데는 이분 네트워크를 만드는 것이 중요하다.
    - ✓ 이미지는 상단, 사용자 또는 Pinterest보드는 하단에 있다.
    - ✓ 이미지를 그래프로 하단에 가져와서 임베딩을 시킨다.
  - 이는 단순히 이미지를 고려 하는 것 보다 훨씬 정확하다.



Example of edge-level ML Tasks

- Drug Side Effects
  - 여러 약물을 한번에 복용 할 때 약물들의 부작용을 전부 알 순 없다.
    - ✓ R은 약물을 동시에 복용 했을 때 생기는 부작용을 의미한다.
    - ✓ 한 사람이 s와 d라는 약물을 같이 복용 했을 때 부작용이 생길지 안 생길지는 모른다.
      - ❖ 이때 edge-level ML을 통해 부작용을 예측 할 수 있다.

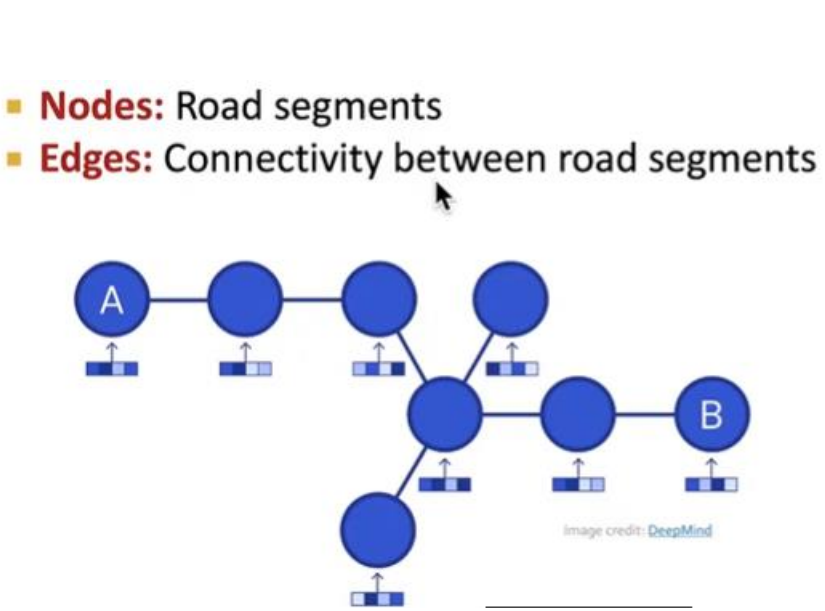


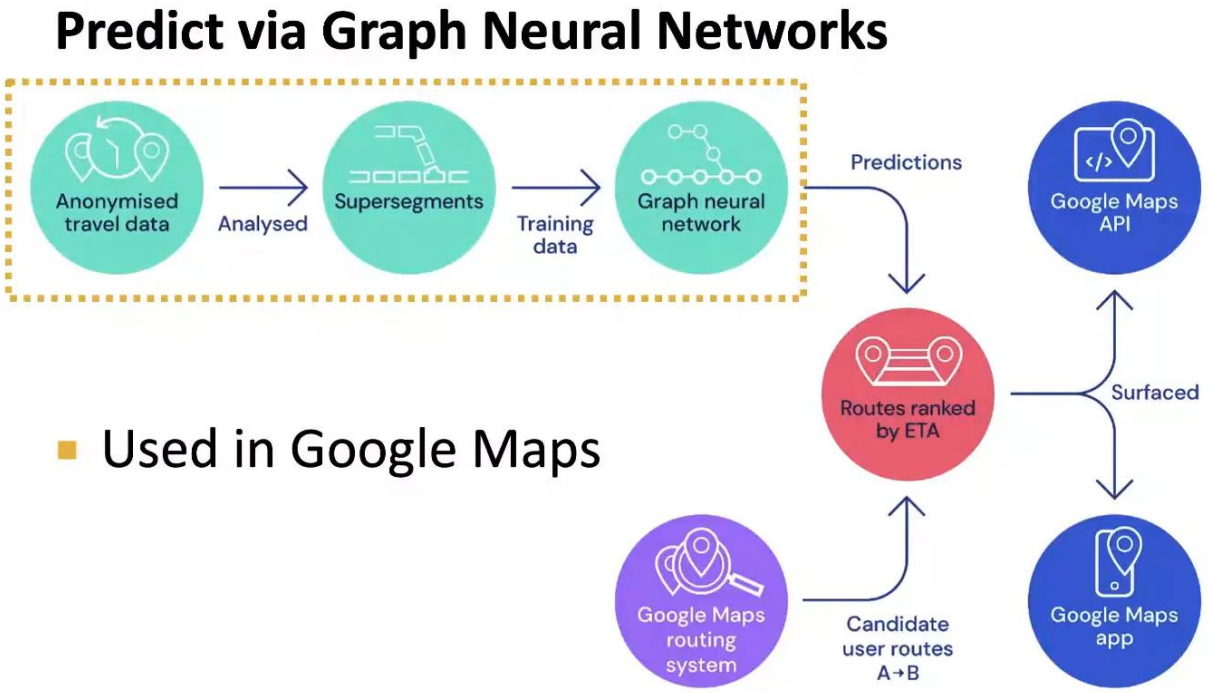
05

Examples of Subgraph-level ML tasks

Example of Subgraph-level ML Tasks

- Traffic Prediction
  - 구글맵의 예
    - ✓ 도착시간을 예측 할 때 사용한다.
    - ✓ 노드는 도로의 구간을 나타내고, 엣지는 도로 사이의 연계를 의미한다.





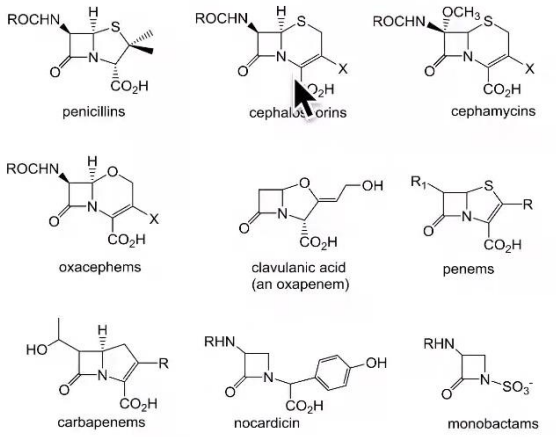
06

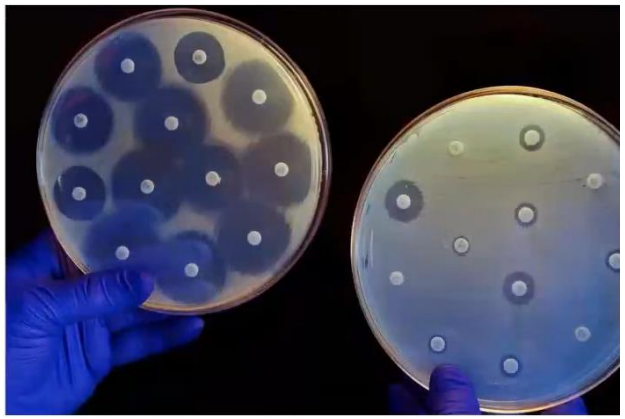
Examples of Graph-level ML tasks

Example of Graph-level ML Tasks

- Drug Discovery
  - 노드가 원자이다. 분자는 그래프이다. 엣지는 화학결합에 해당한다.
    - ✓ 수십억개의 분자중 어떤 것이 치료 효과가 있는지가 궁금하다.
      - ❖ MIT의 한 팀은 그래프 신경망을 사용하여 새로운 항생제를 발견 했다.

- **Antibiotics are small molecular graphs**
  - **Nodes:** Atoms
  - **Edges:** Chemical bonds



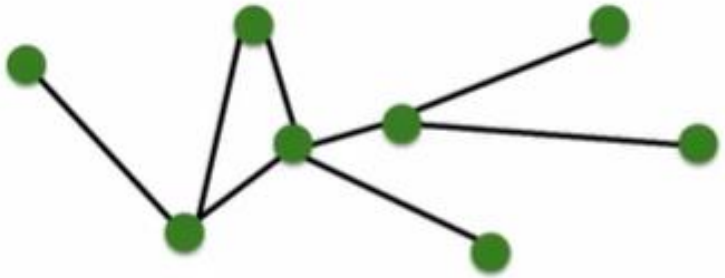


07

Choice of Graph Representation

Choice of Graph Representation

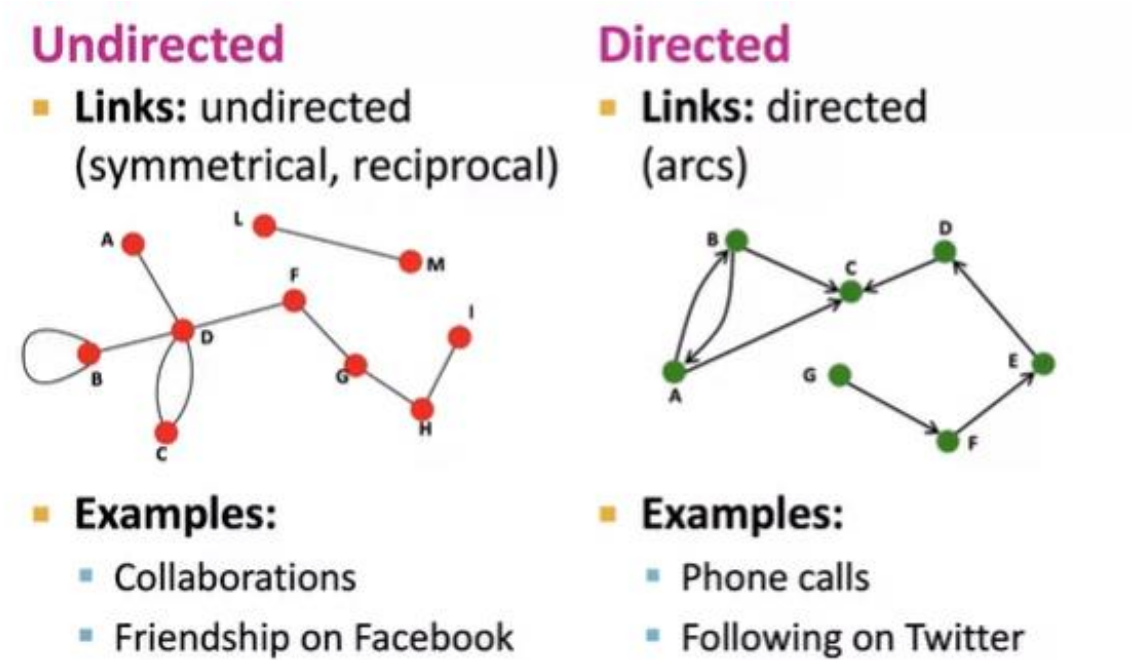
- Components of a Network
  - Objects : nodes vertices
  - Interactions : links, edges
  - System : network, graph



- **Objects:** nodes, vertices — N
- **Interactions:** links, edges — E
- **System:** network, graph — $G(N,E)$

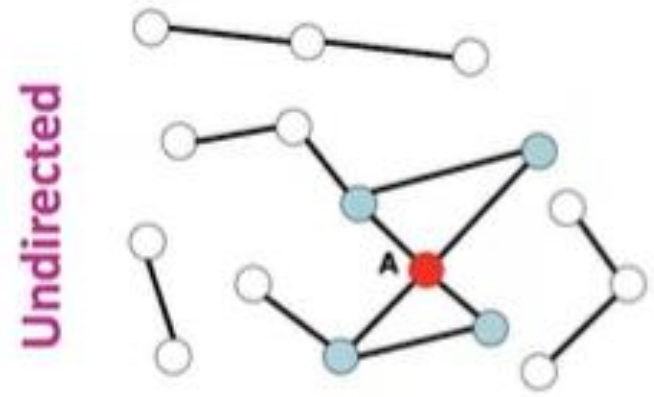
Choice of Graph Representation

- How to build a graph
  - 그래프를 표현하는 데는 여러 방법이 있다.
    - ✓ 단순히 도메인과 같은 이름의 네트워크(논문 네트워크..) 라는 식으로 이름을 붙이고 사용하는 것은 별로다.
  - Directed vs Undirected Graphs



Choice of Graph Representation

- Undirected
  - Node degree
    - ✓ 인접한 노드의 수를 말한다.
      - ❖ 예시에서 A는 degree 가 4 이다.
  - Avg.degree
    - ✓ 단순히 모든 노드의 degree의 평균이다.
    - ✓ 2Edge인 이유는, 양방향이기 때문이다.(방향성이 없다.)



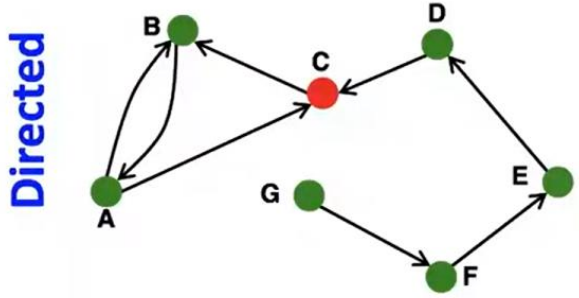
Node degree, k_i: the number of edges adjacent to node ***i***

$k_A = 4$

Avg. degree: $\bar{k} = \langle k \rangle = \frac{1}{N}\sum_{i=1}^{N} k_i = \frac{2E}{N}$

Choice of Graph Representation

- directed
  - Node degree
    - ✓ In-degree와 out-degree로 나뉜다.
    - ✓ 한 노드의 총 degree는 in-degree와 out-degree 의 합이다.



Source: Node with $k^{in} = 0$
Sink: Node with $k^{out} = 0$

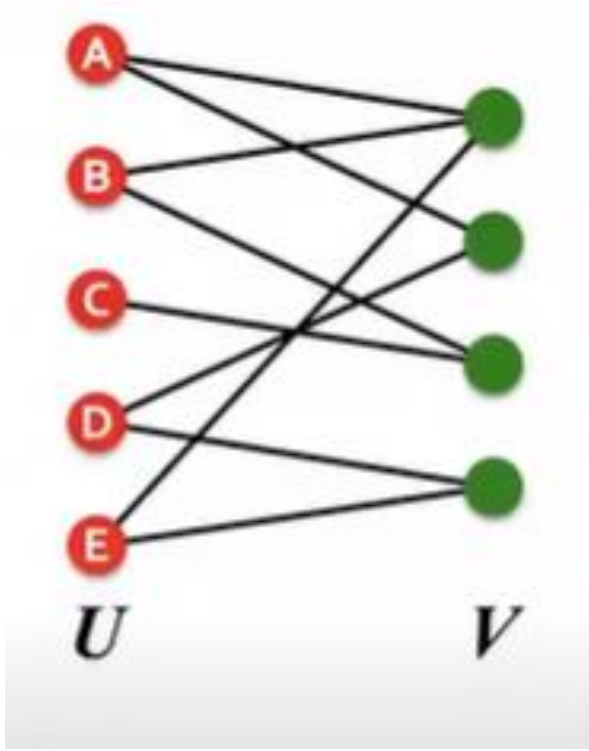
In directed networks we define an **in-degree** and **out-degree.**
The (total) degree of a node is the sum of in- and out-degrees.

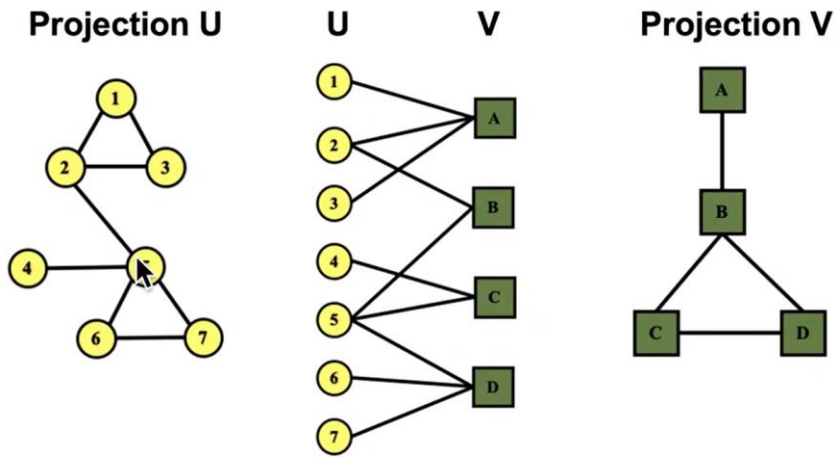
$$k_C^{in} = 2 \qquad k_C^{out} = 1 \qquad k_C = 3$$

$$\bar{k} = \frac{E}{N} \qquad \overline{k^{in}} = \overline{k^{out}}$$

Choice of Graph Representation

- Bipartite graph
  - 이분그래프는 두집단으로 나뉜 노드가 같은 집단에는 링크가 되지 않는 것을 의미한다.
    - ✓ 예를 들면 저자와 책, 배우와 영화 등의 관계가 있다.
- Folded/Projected Bipartite Graphs
  - 이분 그래프의 각각의 성분으로 분해가 가능하다.
    - ✓ 밑 예시를 v는 논문, U는 저자로 보면, v1은 1,2,3이 공동 저자 이기 때문에 서로 링크된 것을 볼 수 있다.(3,4는 공동 저자 한 것이 없어 링크 되어있지 않다.)





Choice of Graph Representation

- Adjacency Matrix
  - 그래프를 인접 행렬로 나타내자는 것이다.
    - ✓ Undirected의 경우 링크 되어 있는 것을 전부 행렬로 표시한다.
    - ✓ Directed의 경우 Out만 표시한다.
      - ❖ 이때 가로는 Out, 세로는 in을 의미한다.
    - ✓ 희소성의 문제가 발생한다.

$A_{ij} = 1$ if there is a link from node i to node j

$A_{ij} = 0$ otherwise

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \qquad A = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

$$A_{ij} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

$$k_i = \sum_{j=1}^{N} A_{ij}$$

$$k_j = \sum_{i=1}^{N} A_{ij}$$

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

$$k_i^{out} = \sum_{j=1}^{N} A_{ij}$$

$$k_j^{in} = \sum_{i=1}^{N} A_{ij}$$

Choice of Graph Representation

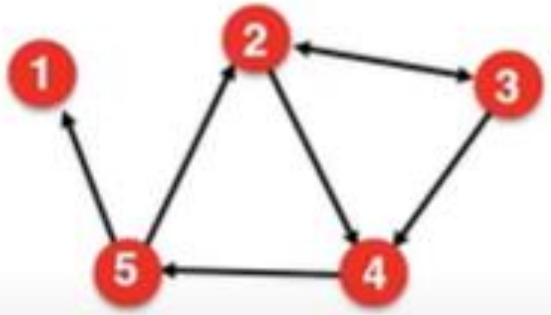
- Networks are Sparse Graphs
  - 희소함을 보여주는 예 이다.
    - ✓ K는 링크되어 있는 비율을 의미한다.

NETWORK	NODES	LINKS	DIRECTED/ UNDIRECTED	N	L	<k>
Internet	Routers	Internet connections	Undirected	192,244	609,066	6.33
WWW	Webpages	Links	Directed	325,729	1,497,134	4.60
Power Grid	Power plants, transformers	Cables	Undirected	4,941	6,594	2.67
Phone Calls	Subscribers	Calls	Directed	36,595	91,826	2.51
Email	Email Addresses	Emails	Directed	57,194	103,731	1.81
Science Collaboration	Scientists	Co-authorship	Undirected	23,133	93,439	8.08
Actor Network	Actors	Co-acting	Undirected	702,388	29,397,908	83.71
Citation Network	Paper	Citations	Directed	449,673	4,689,479	10.43
E. Coli Metabolism	Metabolites	Chemical reactions	Directed	1,039	5,802	5.58
Protein Interactions	Proteins	Binding interactions	Undirected	2,018	2,930	2.90

Choice of Graph Representation

- Edge list
  - 단순히 2차원의 행렬로 만든다.
    - ✓ 이는 분석을 하기 어렵다는 문제가 있다.
      - ❖ 주어진 노드의 차수를 분석 하는 것도 사실 어렵기 때문이다.

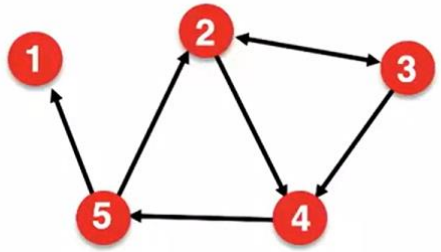
- **Represent graph as a list of edges:**
  - (2, 3)
  - (2, 4)
  - (3, 2)
  - (3, 4)
  - (4, 5)
  - (5, 2)



Choice of Graph Representation

- Adjacency list
  - 인접한 노드들을 표현한다.
    - ✓ 네트워크가 크고 희소할 때 효과적이다.

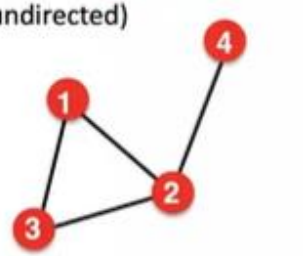
- Easier to work with if network is
  - Large
  - Sparse
- Allows us to quickly retrieve all neighbors of a given node
  - 1:
  - 2: 3, 4
  - 3: 2, 4
  - 4: 5
  - 5: 1, 2



Choice of Graph Representation

- Node and Edge Attributes
  - 가중치를 줄 수 있다.
  - 순위를 둘 수 있다.
  - 타입을 정할 수 있다.
  - 자체 루프를 둘 수 있다.
  - 여러 엣지를 둘 수도 있다.

Unweighted
(undirected)

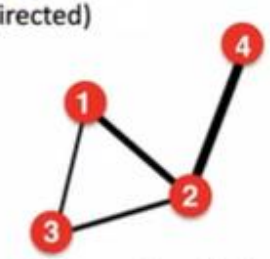


$$A_{ij} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$A_{ii} = 0 \qquad A_{ij} = A_{ji}$

$E = \frac{1}{2}\sum_{i,j=1}^{N} A_{ij} \qquad \bar{k} = \frac{2E}{N}$

Weighted
(undirected)

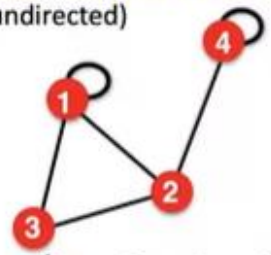


$$A_{ij} = \begin{pmatrix} 0 & 2 & 0.5 & 0 \\ 2 & 0 & 1 & 4 \\ 0.5 & 1 & 0 & 0 \\ 0 & 4 & 0 & 0 \end{pmatrix}$$

$A_{ii} = 0 \qquad A_{ij} = A_{ji}$

$E = \frac{1}{2}\sum_{i,j=1}^{N} nonzero(A_{ij}) \qquad \bar{k} = \frac{2E}{N}$

Self-edges (self-loops)
(undirected)

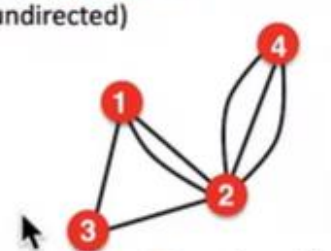


$$A_{ij} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

$A_{ii} \neq 0 \qquad A_{ij} = A_{ji}$

$E = \frac{1}{2}\sum_{i,j=1,i\neq j}^{N} A_{ij} + \sum_{i=1}^{N} A_{ii}$

Multigraph
(undirected)



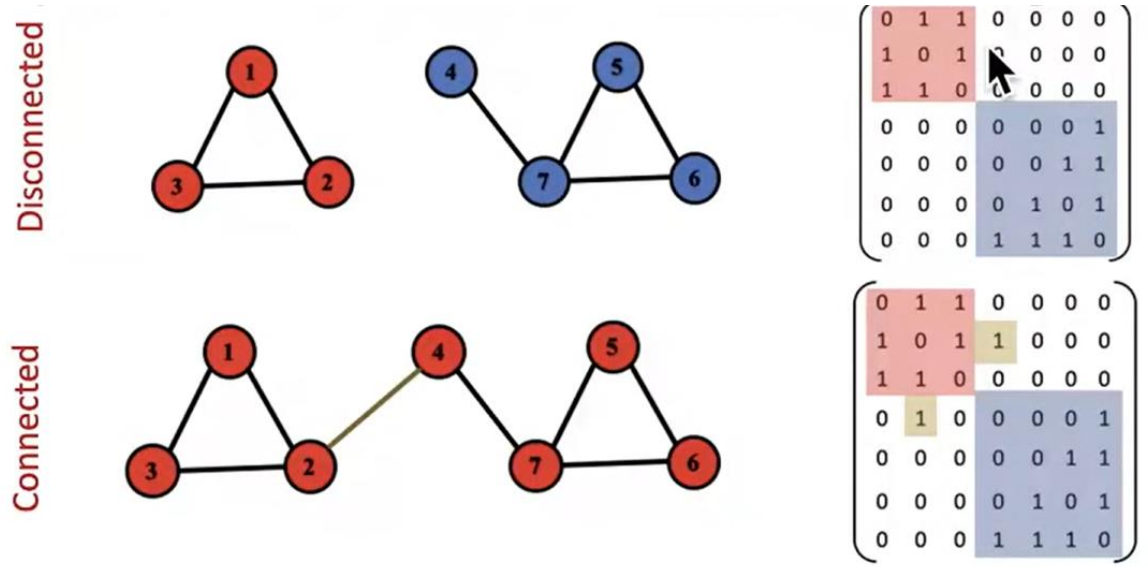
$$A_{ij} = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 3 \\ 1 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 \end{pmatrix}$$

$A_{ii} = 0 \qquad A_{ij} = A_{ji}$

$E = \frac{1}{2}\sum_{i,j=1}^{N} nonzero(A_{ij}) \qquad \bar{k} = \frac{2E}{N}$

Choice of Graph Representation

- Connectivity of Undirected Graphs
  - 연결이 끊어져 있으면 행렬의 사각형이 대각선으로 연결 되어 있다.
  - 연결이 잘 되어 있으면 대각선으로 연결 되어 있는 사각형에, 노란색의 튀는 것이 생긴다.



Choice of Graph Representation

- Connectivity of directed Graphs
  - Strong connected
    - ✓ A to B, B to A, A to B to C to A 도 가능하다.