



北京邮电大学  
Beijing University of Posts and Telecommunications

# 第一章

## 图机器学习概论

石川 教授

数据科学与服务中心 计算机学院





- 课程介绍
- 讲课内容
  - 1.1 现实生活中的图
  - 1.2 图的应用
  - 1.3 定义引入
  - 1.4 任务分类
  - 1.5 发展历程
  - 课程安排和要求



# 1.1 现实生活中的图

- 图是描述和分析具有关系/交互的实体的通用语言。
- 在现实生活中，图是无处不在的。作为强大的工具，它们在各个领域中用于表示复杂的关系。
  - 在社会科学中，图则用来表示个体之间的关系，例如友谊、同学关系或师生关系，节点代表人，边表示社交关系。
  - 推荐系统中的图同样重要，节点可以表示用户或商品，而边则表示购买行为，从而揭示用户的消费模式。
  - 在化学和材料科学中，图用于描绘分子的结构关系，其中原子被视为节点，化学键作为边，形成化合物的图表示。

# 1.1 现实生活中的图



## ● 多种类型的数据和图(1)



Image credit: [Medium](#)

社交网络

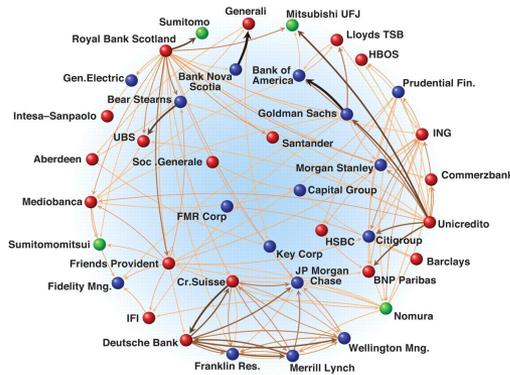


Image credit: [Science](#)

经济网络

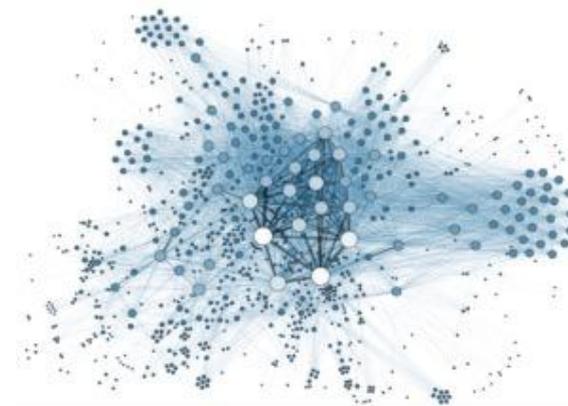
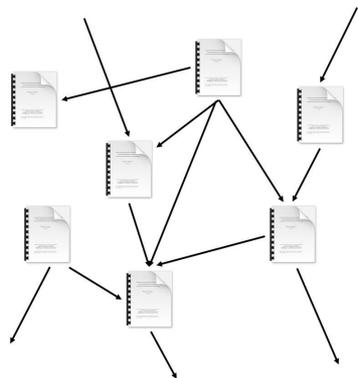


Image credit: [Lumen Learning](#)

通信网络



引文网络



Image credit: [Missoula Current News](#)

互联网

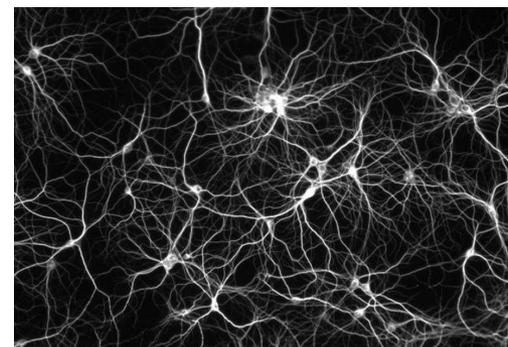


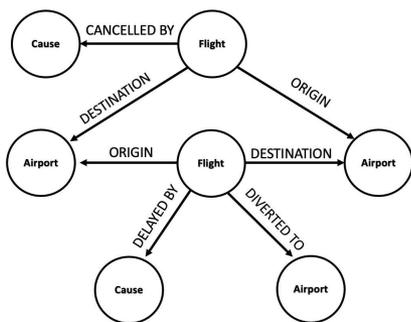
Image credit: [The Conversation](#)

神经元网络

# 1.1 现实生活中的图



## ● 多种类型的数据和图(2)

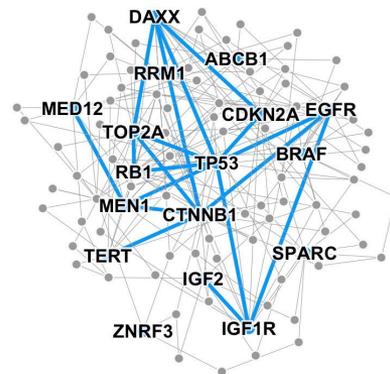


事件图



Image credit: [SalientNetworks](#)

计算机网络



疾病通路

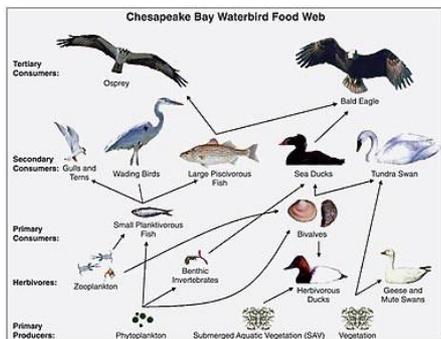


Image credit: [Wikipedia](#)

食物网

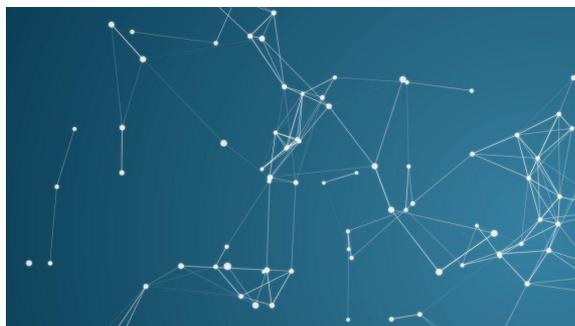


Image credit: [Pinterest](#)

粒子网络



Image credit: [visitlondon.com](#)

地铁网络

# 1.2 图的应用



- 1.2 图的应用
  - 1.2.1 推荐系统
    - 背景及挑战
    - 图神经网络的优势
    - 社交推荐中的图神经网络
  - 1.2.2 天气预测
    - 背景
    - 特殊性
    - 天气预测中的图神经网络
  - 1.2.3 总结

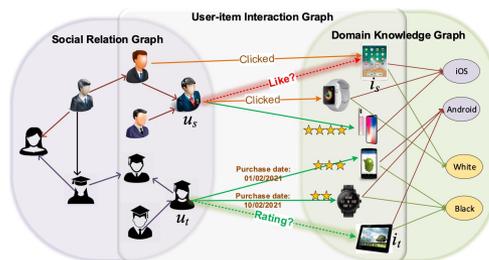
# 1.2 图的应用概述



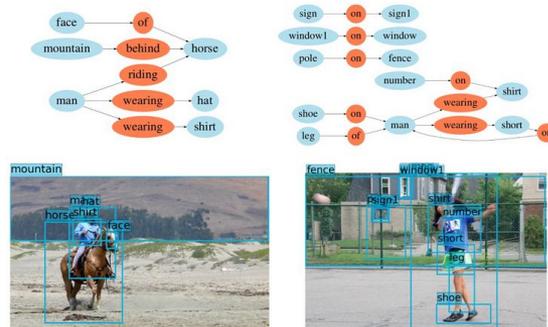
## ● 图的应用场景:



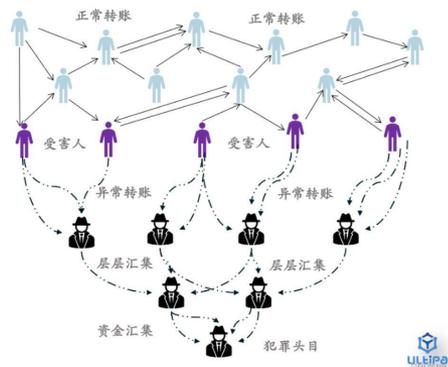
交通



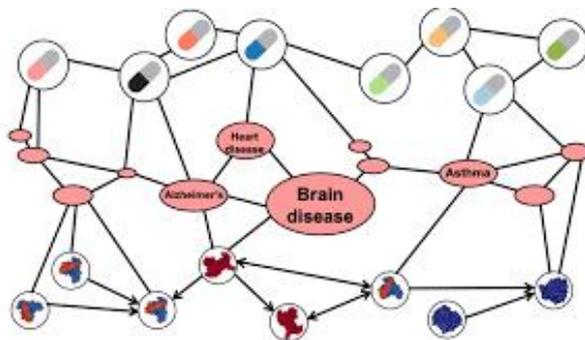
推荐



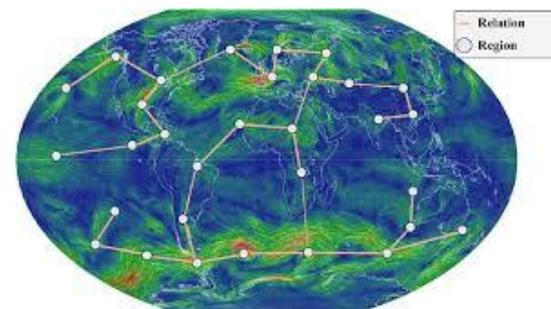
计算机视觉



金融风控



生物医药



天气预测

# 1.2.1 推荐系统

- 推荐系统已成为当今互联网应用的关键组成部分，广泛应用于电子商务、社交网络、内容平台等领域。传统的推荐系统通常依赖于协同过滤和基于内容的推荐方法。
- 然而，随着用户数量和物品种类的不断增加，这些方法面临的挑战也愈加明显：它们无法充分捕捉用户与物品之间复杂的关系，尤其是在高维稀疏数据的情况下，推荐效果显著下降。

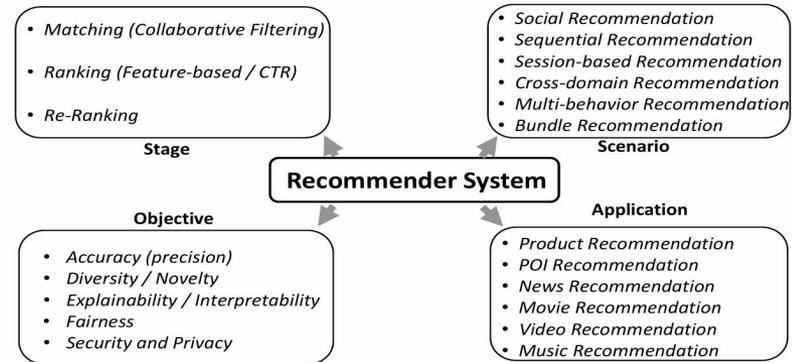
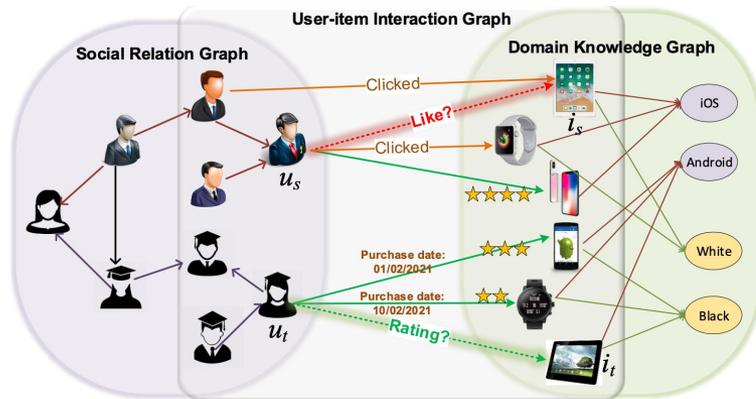
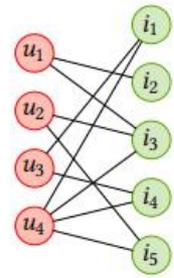


Fig. 1. An illustration of typical recommender systems (stages, scenarios, objectives, and applications)

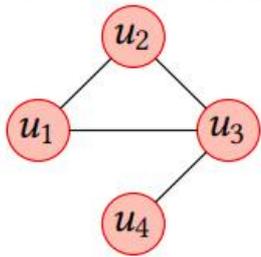
# 1.2.1 推荐系统

- 在推荐系统中，用户和物品通常可以自然地表示为一个图结构，其中节点代表用户或物品，边表示用户与物品之间的交互行为（如点击、购买、评分等）。此外，用户之间的社交关系也可以表示为图中的边。

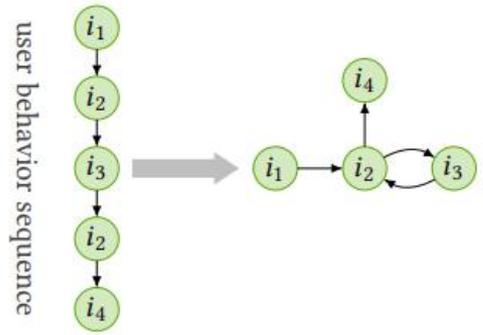
	$i_1$	$i_2$	$i_3$	$i_4$	$i_5$
$u_1$	0	1	1	0	0
$u_2$	0	0	1	0	1
$u_3$	1	0	0	1	0
$u_4$	1	0	1	1	1



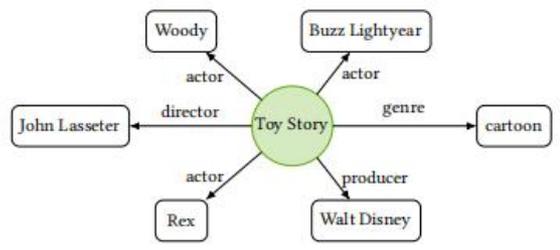
(a) User-item bipartite graph.



(c) Social relationship between users.



(b) Sequence graph.



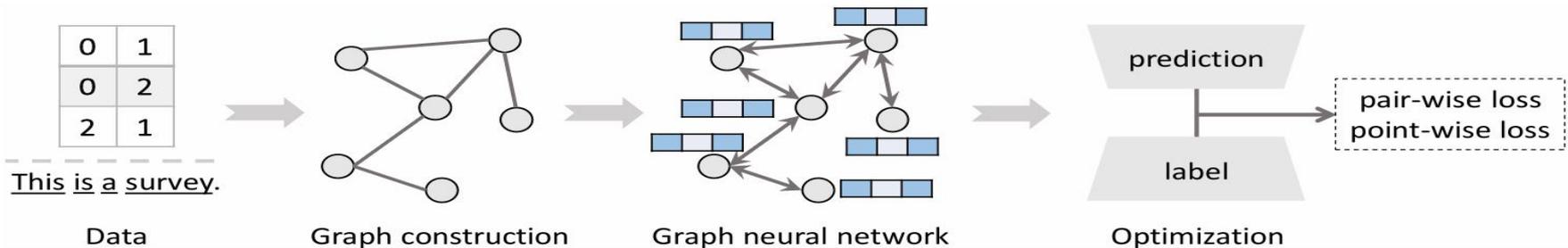
(d) Knowledge graph

Fig. 1. Representative graph structures in recommender systems.



# 1.2.1 推荐系统

- 图神经网络是处理图结构数据的强大工具，也广泛应用于推荐系统。
- 图神经网络通过聚合节点的邻居信息，逐步更新节点的表示，从而捕捉图结构中的复杂关系。与传统的推荐算法相比，具有以下几个优势：
  - 高阶关系建模：GNNs能够通过多层传播捕捉用户与物品之间的高阶关系，如朋友的朋友喜欢的物品。
  - 稀疏数据处理：通过邻居信息的传播，GNNs能够在稀疏数据的情况下生成有效的节点表示。
  - 可扩展性：GNNs可以与其他推荐模型结合，进一步提高推荐效果。

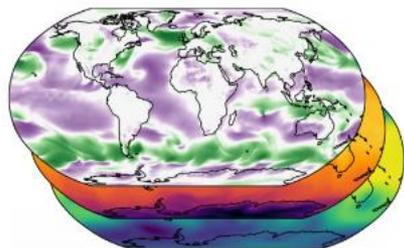


# 1.2.2 天气预测

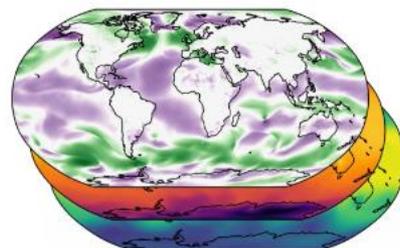


- 气象预测直接关乎人类社会的生产生活，一直以来是人类重点研究的科学领域之一。
- 我们可以将气象站建模成节点，它们之间的位置关系建立连边，将气象预测抽象成一个时空预测问题，尝试通过多维时间序列、图神经网络等方法来解决。

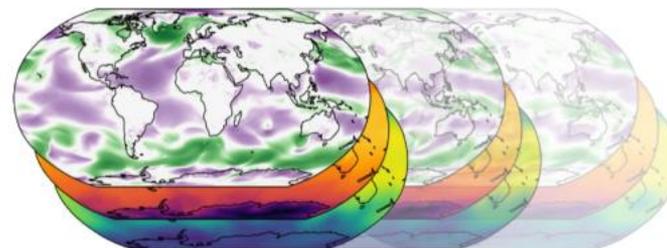
a) Input weather state



b) Predict the next state

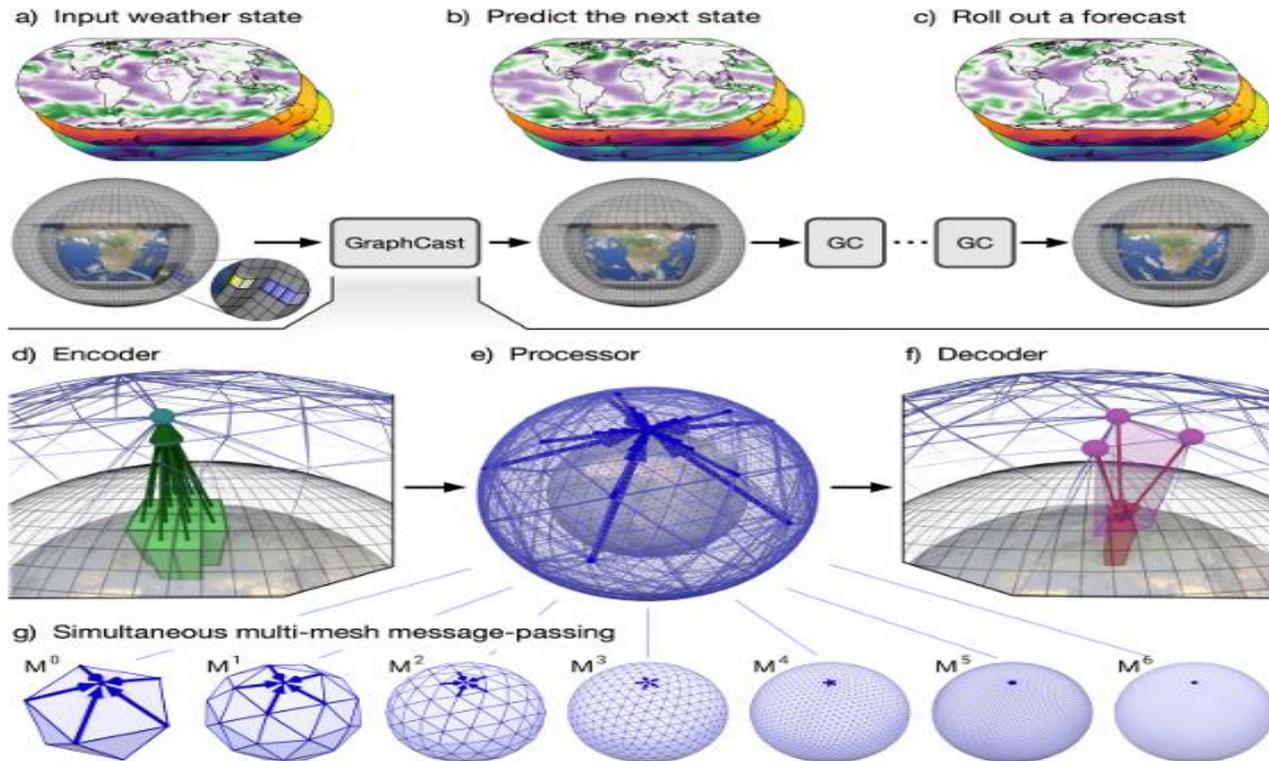


c) Roll out a forecast



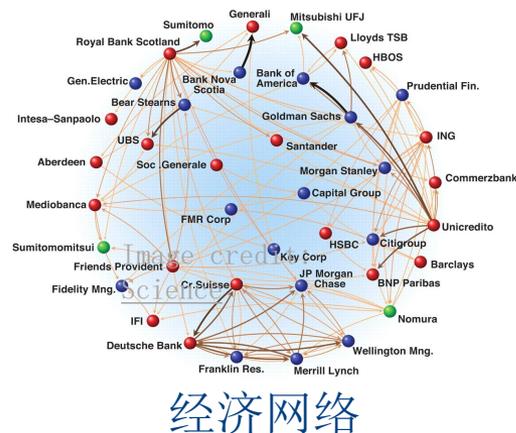
# 1.2.2 天气预测

- Deepmind的GraphCast架构。GraphCast使用图神经网络模型，将原始经纬度网格的输入数据映射到多网格上的学习特征中，该模型60s内可预测10d内的天气状况。



# 1.2.3 总结

- 图是强大的建模工具。图神经网络（GNN）将深度学习的威力拓展到了图结构数据，为理解和挖掘复杂网络中的模式提供了新的视角。
- 除了在推荐系统和天气预测中取得了显著的成就，图神经网络还被广泛应用于社交网络分析、金融风险评估等领域。
  - 在社交网络分析中，GNN可以有效地捕获用户之间的复杂互动关系，揭示社区结构和用户行为模式。
  - 在金融领域，GNN被用于识别欺诈行为和风险评估，通过寻找交易网络中的异常模式，提高了金融系统的安全性。



# 1.3 定义引入

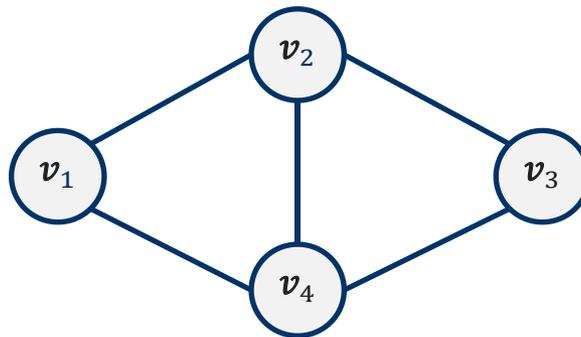


- 1.3 定义引入
  - 1.3.1 图的定义及表示
    - 图的定义
    - 邻接矩阵
    - 邻接表
    - 关联矩阵
    - 拉普拉斯矩阵和标准化拉普拉斯矩阵
  - 1.3.2 图的类型
    - 无向图v. s. 有向图
    - 无权图v. s. 带权图
    - 同质图v. s. 异质图
    - 无属性图v. s. 属性图
    - 静态图v. s. 动态图



# 1.3.1 图的定义及表示—图的定义

- 图可以表示为  $G = \{\mathcal{V}, \mathcal{E}\}$ ，其中  $\mathcal{V} = \{v_1, \dots, v_{|\mathcal{V}|}\}$  表示节点集合， $\mathcal{E} = \{e_1, \dots, e_{|\mathcal{E}|}\}$  表示边集合。其中： $|\mathcal{V}|$  是图中节点的数量， $|\mathcal{E}|$  是图中边的数量。
- 边也可以用其两端的节点进行表示，例如连接节点  $v_1$  和  $v_2$  的边也可以表示为  $(v_1, v_2)$ 。在许多情况下，我们只关心简单无向图，即每对节点之间最多只有一条边，没有节点与自身相连的边，并且这些边都是无向的，即  $(u, v) \in \mathcal{E} \Leftrightarrow (v, u) \in \mathcal{E}$ 。

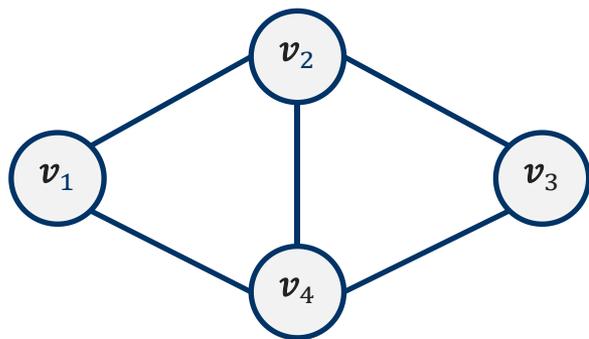


简单无向图



# 1.3.1 图的定义及表示—邻接矩阵

- 给定图  $G$ ，可以使用邻接矩阵  $A \in \{0,1\}^{|\mathcal{V}| \times |\mathcal{V}|}$  表示边的分布。邻接矩阵的第  $(i,j)$  项表示为  $A_{i,j}$ ，其表示节点  $v_i$  与  $v_j$  之间的连接性。如果  $v_i$  与  $v_j$  之间存在一条边，则  $A_{i,j} = 1$ ，否则  $A_{i,j} = 0$ 。
- 邻接矩阵是一种直观、简单且易于理解的图表示方法。其可以方便地检查任意一对节点之间是否存在边，便于找出任一节点的所有邻接点，以及便于计算任一节点的度。
- 邻接矩阵也存在一些缺点。首先，增加或删除节点时需要调整矩阵的行列，操作不便；其次，对于稀疏图，邻接矩阵会浪费大量空间，会有许多无效元素。



(1) 图

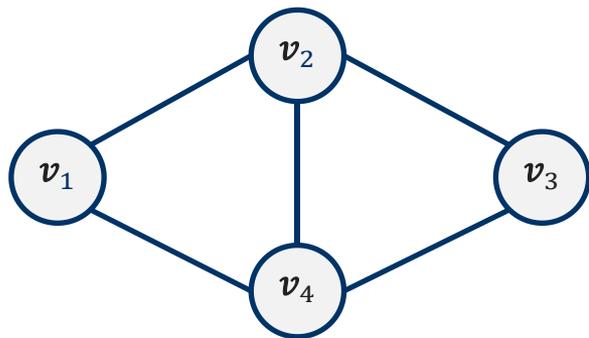
节点	$v_1$	$v_2$	$v_3$	$v_4$
$v_1$	0	1	0	1
$v_2$	1	0	1	1
$v_3$	0	1	0	1
$v_4$	1	1	1	0

(2) 邻接矩阵



# 1.3.1 图的定义及表示—邻接表

- 给定图  $G = (\mathcal{V}, \mathcal{E})$ ，其邻接表是一个包含  $|\mathcal{V}|$  个列表的数组，每个列表对应于图  $G$  的一个节点。对于每个节点  $v \in \mathcal{V}$ ，图  $G$  的邻接表中  $v$  的列表包含所有与  $v$  相邻的节点。
- 邻接表特别适合用于存储稀疏图，这种图中的边相较于节点组合数量要少得多。相比于邻接矩阵，邻接表在存储空间上更为高效，并且能够更快速地找到与某个节点相连的所有邻接节点，邻接表在遍历邻接节点时效率较高。
- 邻接表的缺点是实现相对复杂，且在需要频繁查询两个节点之间是否存在边时效率较低。



(1) 图

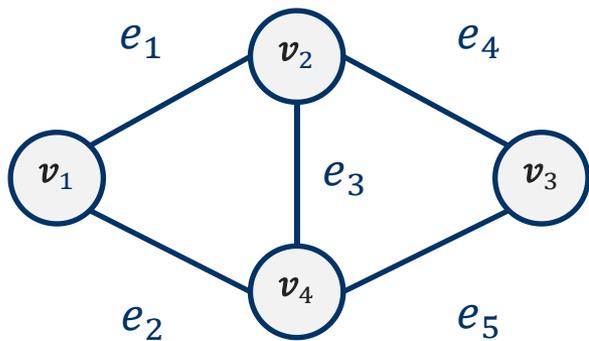
节点	邻接点
$v_1$	$v_2, v_4$
$v_2$	$v_1, v_3, v_4$
$v_3$	$v_2, v_4$
$v_4$	$v_1, v_2, v_3$

(2) 邻接表



# 1.3.1 图的定义及表示—关联矩阵

- 关联矩阵是一个维度为  $|V| \times |E|$  的矩阵。矩阵中的每个元素  $a_{ij}$  表示第  $i$  个节点和第  $j$  条边之间的连接关系：如果第  $i$  个节点与第  $j$  条边关联（即该边连接到该节点），则  $a_{ij} = 1$ 。如果第  $i$  个节点与第  $j$  条边不关联，则  $a_{ij} = 0$ 。
- 关联矩阵适用于稀疏图，可以清晰地表示每个节点与边的关系，适合于处理边的属性和权重，还可以方便地表示节点和边之间的关系，适用于求解边的度数等问题。但对于边数较多的图，关联矩阵的列数会显著增加，导致内存消耗大。



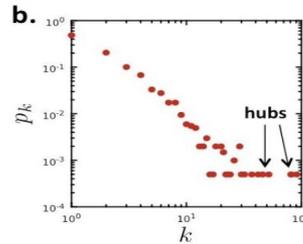
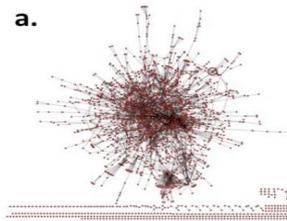
(1) 图

边 \ 节点	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$
$v_1$	1	1	0	0	0
$v_2$	1	0	1	1	0
$v_3$	0	0	0	1	1
$v_4$	0	1	1	0	1

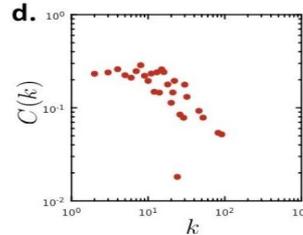
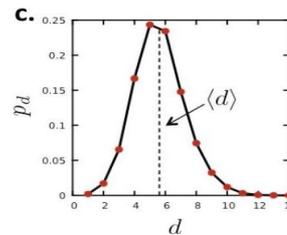
(2) 关联矩阵

# 1.3.1 图的定义及表示

- 图具有一些重要的属性。例如：
  - 邻居:图  $G$  中节点  $v_i$  的邻居集合表示为  $N(v_i)$ ，它包含所有与  $v_i$  相邻的节点。
  - 度:节点  $v_i$  的度数可以表示为  $d_i = \sum_{j=1}^N A_{i,j}$ 。节点  $v_i$  的度数等于  $N(v_i)$  中节点的数量，即  $d_i = |N(v_i)|$ 。对角度数矩阵可以表示为  $D = \text{diag}(d_1, d_2, \dots, d_n)$ 。



**a. Undirected network**  
 N=2,018 proteins as nodes  
 E=2,930 binding interactions as links.

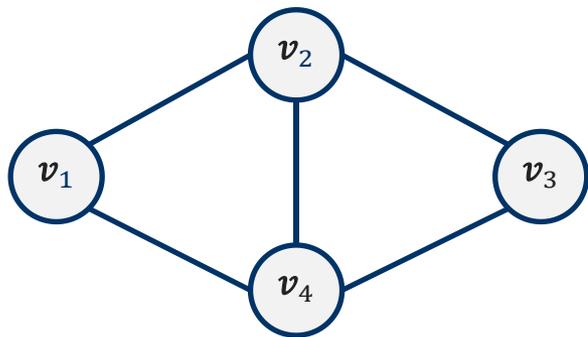


**b. Degree distribution:**  
 Skewed. Average degree  $\langle k \rangle = 2.90$   
**c. Diameter:**  
 Avg. path length = 5.8  
**d. Clustering:**  
 Avg. clustering = 0.12  
**Connectivity:** 185 components  
 the largest component has  
 1,647 nodes (81% of nodes)



# 1.3.1 图的定义及表示—拉普拉斯矩阵

- 对于一个以  $\mathbf{A}$  为邻接矩阵的图  $G$ ，其拉普拉斯矩阵定义为  $\mathbf{L} = \mathbf{D} - \mathbf{A}$ ，其中  $\mathbf{D} = \text{diag}(d(v_1), \dots, d(v_N))$  是对角度矩阵。
- 对于一个给定的以  $\mathbf{A}$  为邻接矩阵的图  $G$ ，其标准化拉普拉斯矩阵记作  $\tilde{\mathbf{L}}$ ，定义为  $\tilde{\mathbf{L}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-\frac{1}{2}} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$ 。



(1) 图

$$\mathbf{L} = \begin{pmatrix} 2 & -1 & -1 & 0 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ 0 & -1 & -1 & 2 \end{pmatrix}$$

(2)拉普拉斯矩阵

$$\tilde{\mathbf{L}} = \begin{pmatrix} 1 & -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} & 0 \\ -\frac{1}{\sqrt{6}} & 1 & -\frac{1}{3} & -\frac{1}{2\sqrt{3}} \\ -\frac{1}{\sqrt{6}} & -\frac{1}{3} & 1 & -\frac{1}{2\sqrt{3}} \\ 0 & -\frac{1}{2\sqrt{3}} & -\frac{1}{2\sqrt{3}} & 1 \end{pmatrix}$$

(2)标准化拉普拉斯矩阵



# 1.3.1 图的定义及表示

- **谱图理论:**

- 结合了图论和线性代数，它通过分析图的矩阵特征值和特征向量来研究图的性质。

- **拉普拉斯矩阵:**

- 是谱图理论中的核心概念，它可以通过考虑节点的度和邻接性来捕获图的几何和拓扑特性。

- **标准化的拉普拉斯矩阵:**

- 消除不同节点度的影响，保持了拉普拉斯矩阵的对称性和半正定性，有助于图信号处理和机器学习工作更好地利用图结构信息。

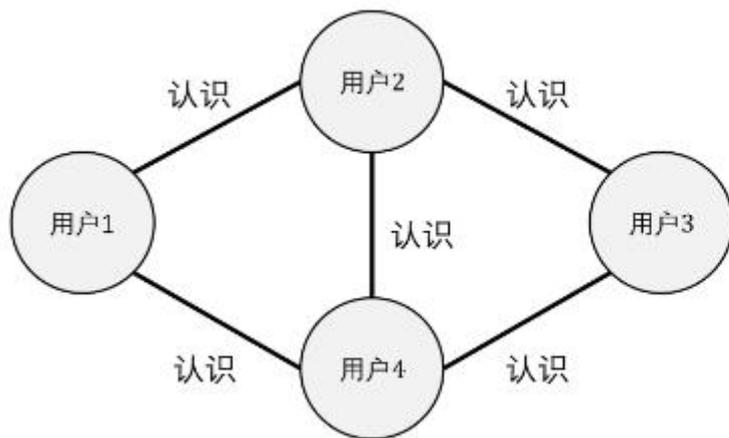
- **应用:**

- 拉普拉斯映射、局部保持投影、谱聚类算法、图神经网络等

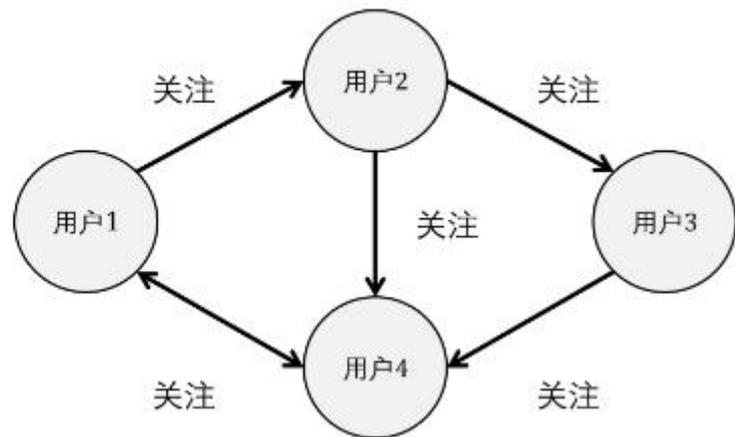
# 1.3.2 图的类型

- 无向图 v. s. 有向图

- 有向图  $G = (\mathcal{V}, \mathcal{E})$  由一个非空的节点集  $\mathcal{V}$  和一组有向边集  $\mathcal{E}$  组成。 $\mathcal{E}$  中的每条边  $e$  是由一个有序顶点对  $u, v \in \mathcal{V}$  指定的。无向图被视为有向图的一种特殊情况，其中如果两个节点之间连接，则存在一对方向相反的边。当且仅当邻接矩阵是对称的时，图才是无向的。



(1) 无向图

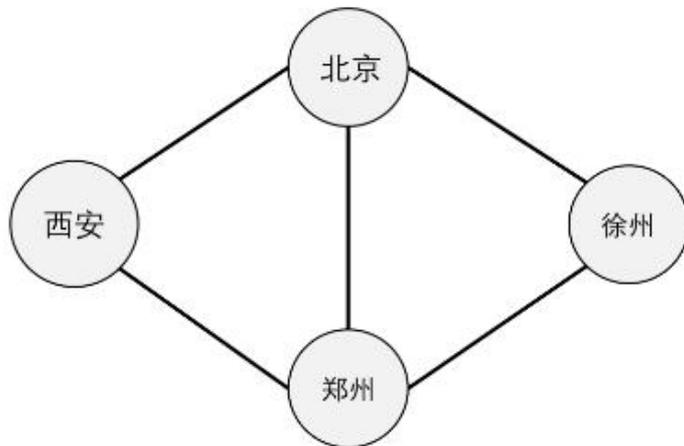


(2) 有向图

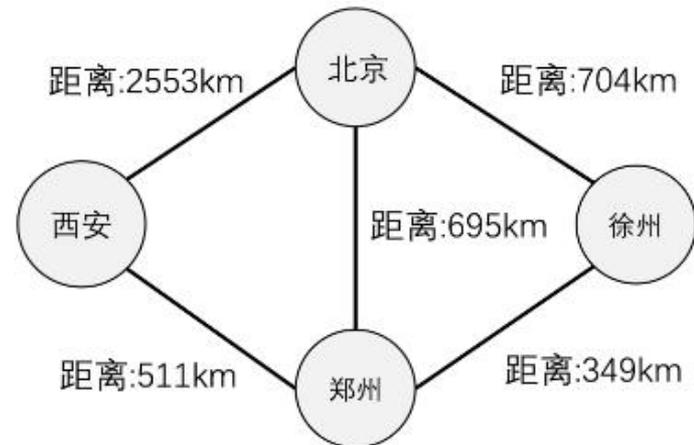
# 1.3.2 图的类型

- 无权图 v. s. 带权图

- 一个带权图是包含一个图  $G = (\mathcal{V}, \mathcal{E})$ ，以及一个权重函数  $w: \mathcal{E} \rightarrow \mathbb{Z}$ 。即：为每条边  $e = (u, v) \in \mathcal{E}$  分配一个整数权重，我们用  $w(e) = w(u, v)$  来表示。



(1) 无权图

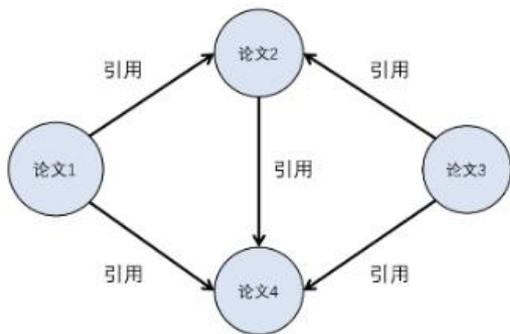


(2) 带权图

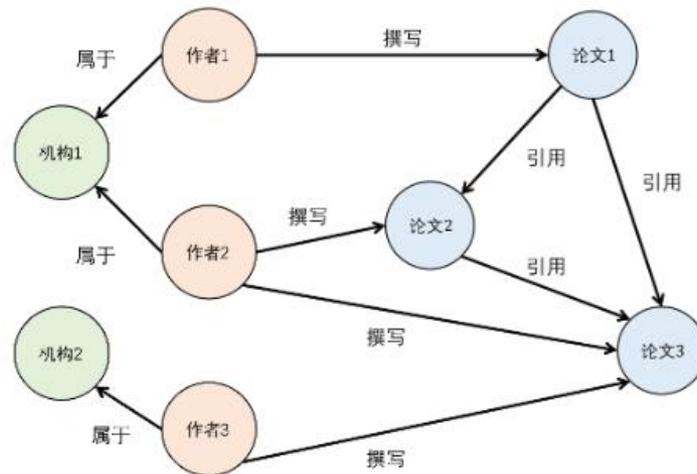
# 1.3.2 图的类型

- 同质图 v. s. 异质图

- 一个异质图  $G$  由一组节点  $\mathcal{V} = \{v_1, \dots, v_N\}$  和一组边  $\mathcal{E} = \{e_1, \dots, e_M\}$  组成。每个节点  $v$  和边  $e$  都与其类型映射函数  $\phi_v: \mathcal{V} \rightarrow \mathcal{T}_v$  和  $\phi_e: \mathcal{E} \rightarrow \mathcal{T}_e$  相关联，其中  $|\mathcal{T}_v| + |\mathcal{T}_e| > 2$ 。其中  $|\mathcal{T}_v|$  表示图中节点的种类数， $|\mathcal{T}_e|$  表示图中边的种类数。



(1) 同质图

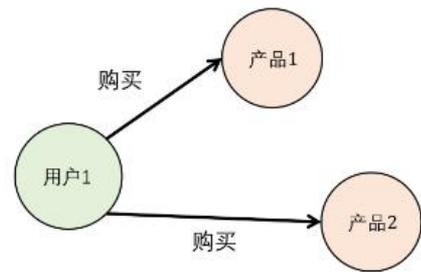


(2) 异质图

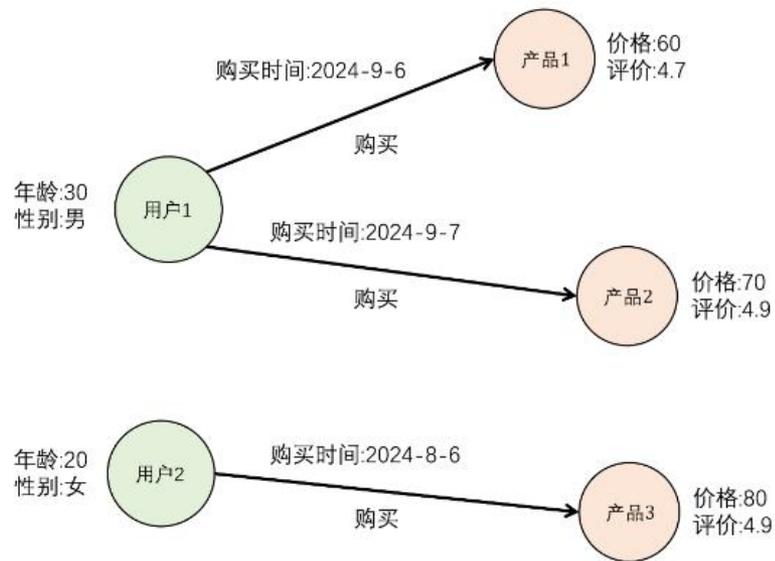
# 1.3.2 图的类型

- 无属性图 v. s. 属性图

- 一个属性图  $G = \{\mathcal{V}, \mathcal{E}, \mathcal{F}\}$  由节点集  $\mathcal{V}$ 、边集  $\mathcal{E}$  以及一组映射  $\mathcal{F} = \{f_1, \dots, f_A\}$  组成, 使得对于  $i \in [1, \dots, A]$ ,  $f_i: \mathcal{V} \rightarrow \text{dom}(a_i)$  将属性  $a_i$  的值  $f_i(v)$  分配给节点  $v$ , 其中  $\text{dom}(a_i)$  是属性  $a_i$  的定义域。



(1) 无属性图

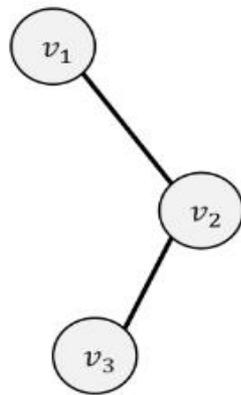


(2) 属性图

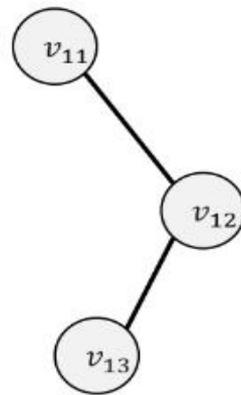
# 1.3.2 图的类型

- 静态图 v. s. 动态图

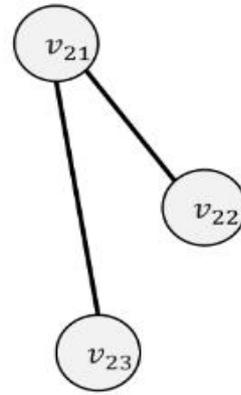
- 动态图  $G = \{\mathcal{V}, \mathcal{E}, \mathcal{T}\}$  由节点集  $\mathcal{V}$ 、边集  $\mathcal{E}$  和时间映射  $\mathcal{T}$  组成。具体来说，每个节点或每条边都与时间戳信息相关联，这些时间戳指示它们出现的时间。时间映射  $\mathcal{T}: (\mathcal{V}, \mathcal{E}) \rightarrow (\mathcal{V}', \mathcal{E}')$ ，表示节点集  $\mathcal{V}$  和边集  $\mathcal{E}$  及其属性随时间的变化关系，其中  $\mathcal{V}'$  是映射后的节点集， $\mathcal{E}'$  是映射后的边集。



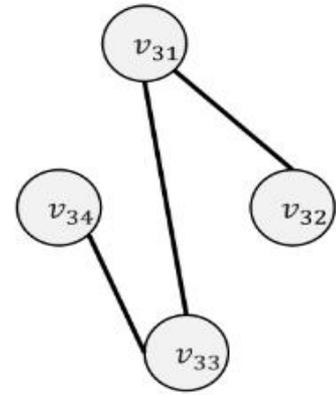
(1) 静态图



时间 $T_1$



时间 $T_2$



时间 $T_3$

(2) 动态图

# 1.4 图机器学习

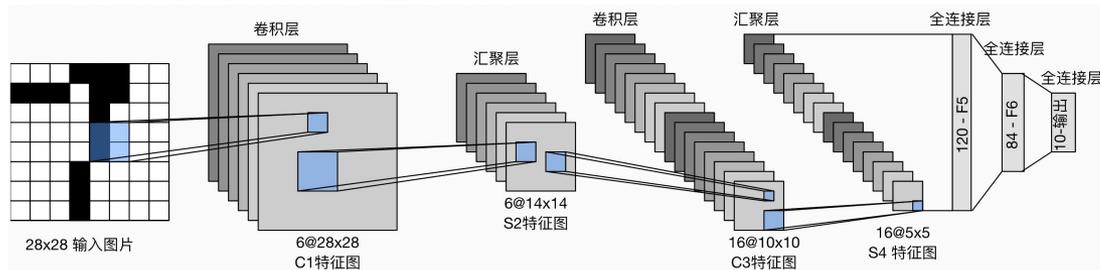


- 1.4 图机器学习
  - 1.4.1 基本概念
  - 1.4.2 节点级任务
    - 节点分类
    - 社区发现
    - 异常检测
  - 1.4.3 边级任务
    - 链接预测
    - 边分类
  - 1.4.4 图级任务
    - 图分类
    - 图生成

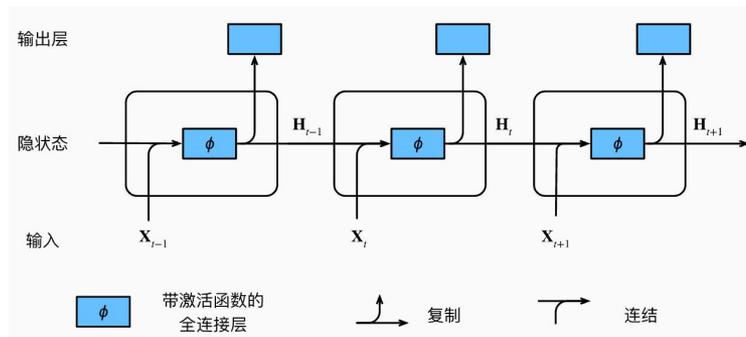
# 1.4.1 基本概念

- 机器学习：是一种**数据驱动**的计算方法，它通过让计算机从大量数据中自动学习模式和规律来解决实际问题。
- 在处理**图像数据**和**文本数据**相关任务上取得了巨大成功：

- 针对图像数据的卷积神经网络CNN

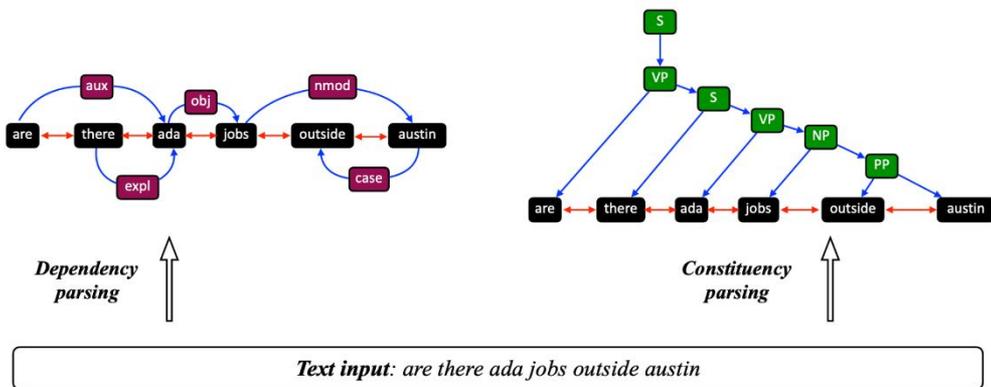


- 针对文本数据的循环神经网络RNN

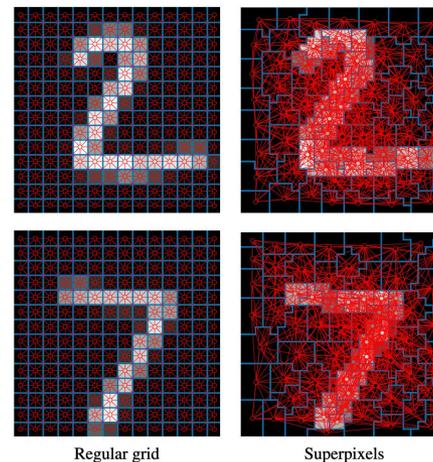


# 1.4.1 基本概念

- 机器学习有效性在很大程度上依赖于针对数据特性所设计的归纳偏置 (Inductive Bias)。
- 然而传统机器学习模型往往缺乏针对图结构数据的特殊设计，因此，图机器学习的概念应运而生。
- 图数据结构可以被视为更通用的数据组织结构。



文本数据—>语法关系图



图像数据—>网格图

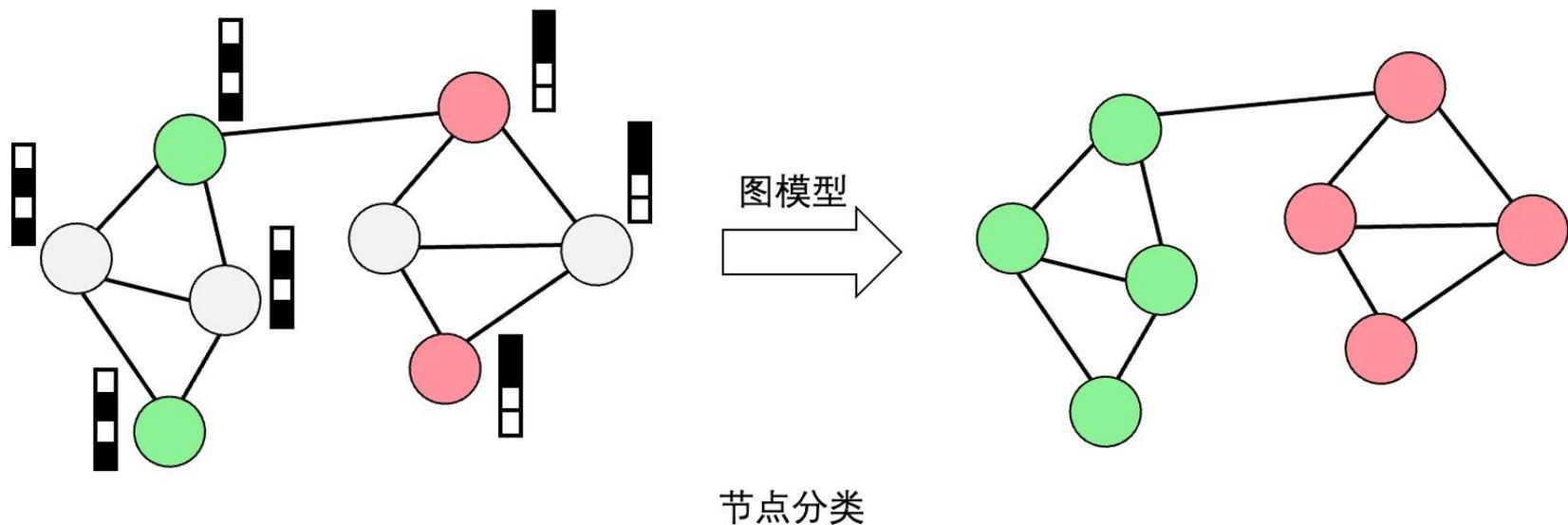


## 1.4.2 节点级任务

- 定义：现实场景中，常常需要对网络中的节点进行预测，这样的任务被称为**节点级任务**。例如：
  - 在社交网络中预测用户的兴趣类别；
  - 在推荐系统中预测用户对商品的喜好程度；
  - 在金融网络中的检查出出现异常交易的账户。
- 特点：
  - **数据集通常为一整张图** $G = (\mathcal{V}, \mathcal{E})$ ，如社交网络，金融网络，电商网络等。
  - 图中的节点  $v \in \mathcal{V}$  被定义为**样本点**。
  - 模型进行推断时，会将图数据作为**整体进行输入**，来预测目标节点的标签信息。

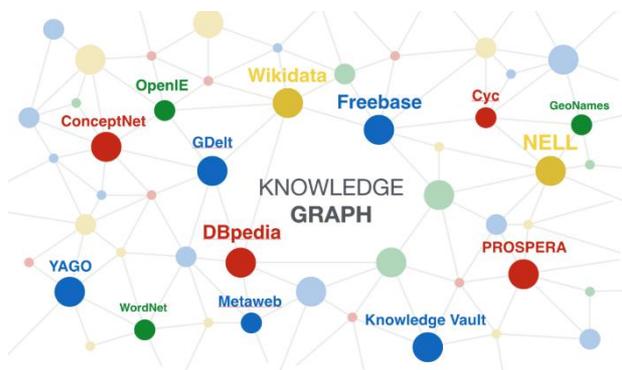
## 1.4.2 节点级任务—节点分类

- **节点分类**任务旨在利用图中少量节点的标签信息，推断剩余未标注节点的标签。
- 样本（节点）往往是**相互依赖**的，这打破了监督分类任务中对样本数据**独立同分布**（Independent and Identically Distributed, i.i.d）的假设。

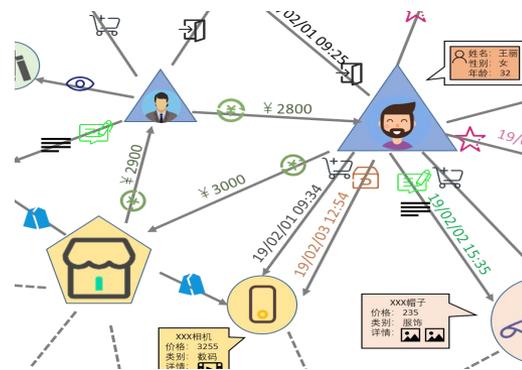


# 1.4.2 节点级任务—节点分类

- 节点分类任务重要假设：
  - 同配性假设 (Homophily Hypothesis)
  - 结构等价性 (Structural Equivalence)
  - 异配性假设 (Heterophily Hypothesis)
- 应用



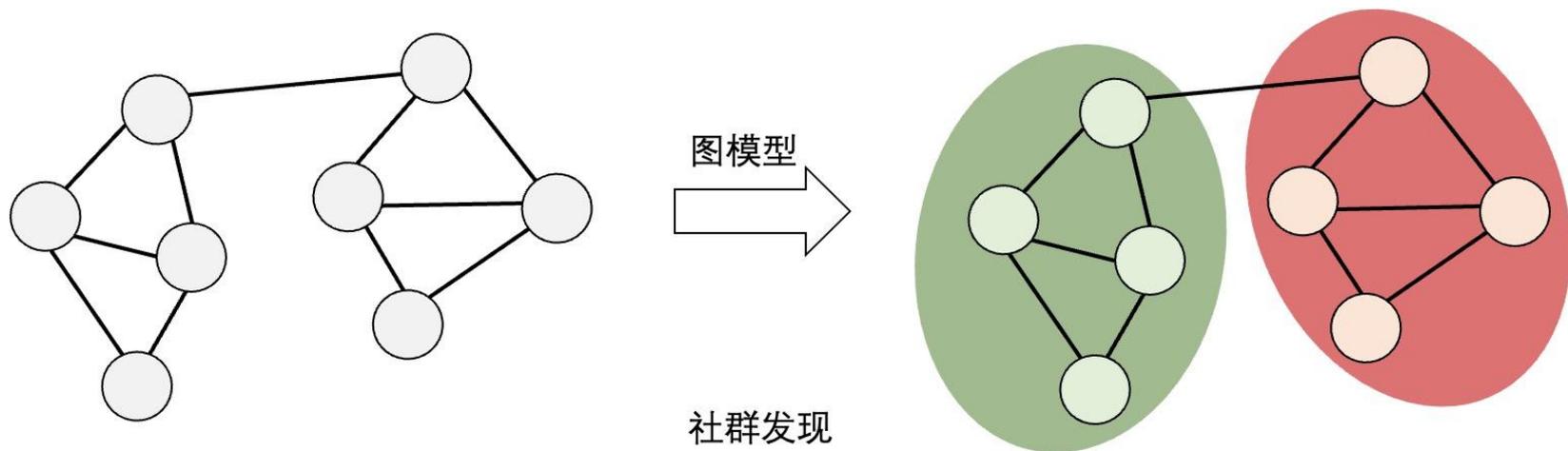
知识图谱



电商网络

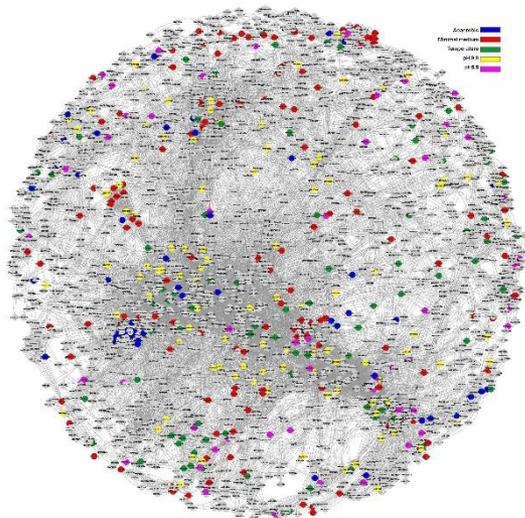
## 1.4.2 节点级任务—社区发现

- 现实世界的图中经常会出现这样的连接模式，一些节点彼此之间连接较为紧密，而与外部其他节点之间的连接则较为稀疏，这些彼此紧密连接的节点集合被定义为社区。
- 社区发现任务旨在无标注情况下，自主发掘识别图中的社区结构。



# 1.4.2 节点级任务—社区发现

- 社区发现任务重要指标：
  - 模块度 (Modularity)
  - 标准化互信息 (Normalized Mutual Information)
- 应用



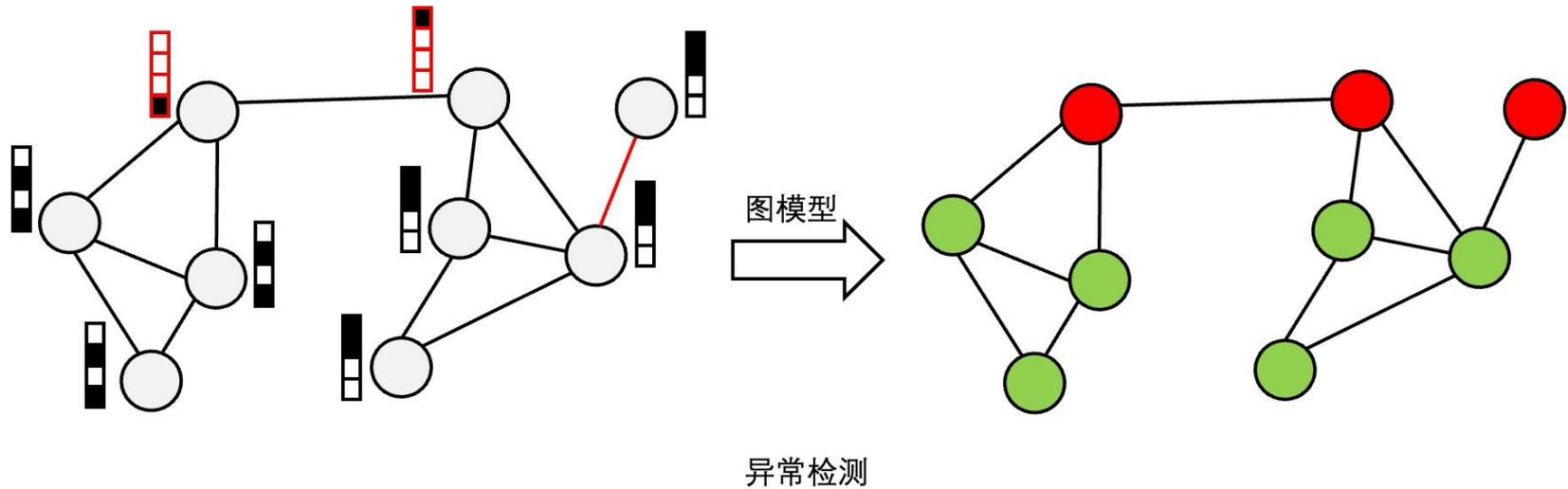
基因交互网络



社交网络

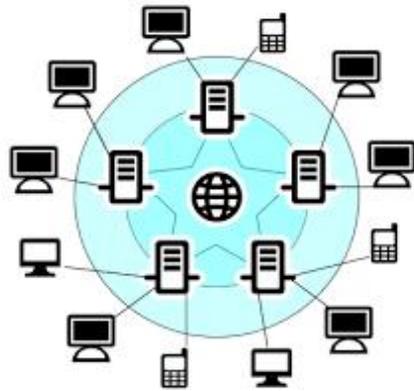
## 1.4.2 节点级任务—异常检测

- 异常检测是十分常见的机器学习任务，旨在从给定样本集合中找出与其他样本在某些方面显著不同的样本。
- 图上异常节点检测则需要额外考虑节点间的异常连接模式，因此会基于特征和结构进行综合推断。

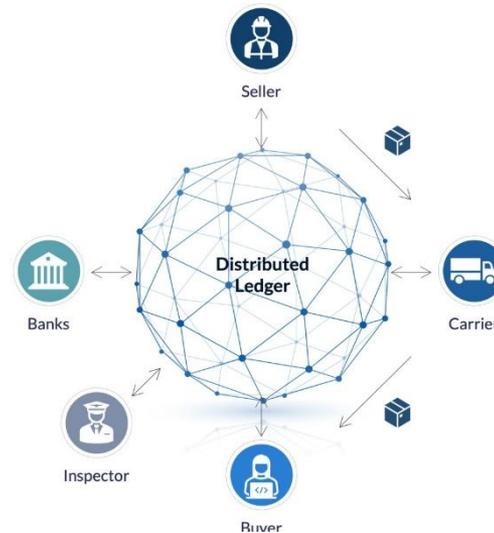


# 1.4.2 节点级任务—异常检测

- 异常检测任务常用假设：
  - 异常样本在特征空间中往往较为孤立
  - 模型难以捕捉异常样本分布
- 应用



互联网



金融网络

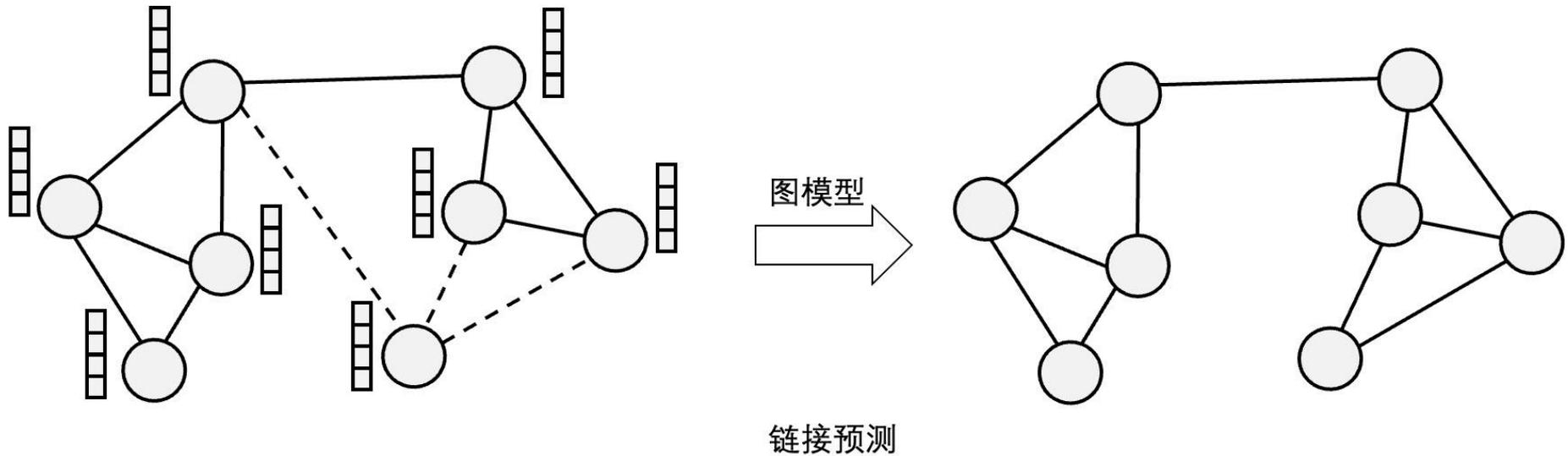


## 1.4.3 边级任务

- 定义：现实场景中，常常需要判断某两个节点之间是否存在连边，或者预测图中边的类型，这样的任务被称为**边级任务**。例如：
  - 推荐系统中预测用户和商品之间是否会产生交互（点击，购买等）
  - 社交网络中预测两个交互过的用户之间的关系类别（熟人，朋友，恋人等）。
- 特点：
  - **数据集通常为一整张图** $G = (\mathcal{V}, \mathcal{E})$ ，如社交网络，金融网络，电商网络等。
  - 图中的节点对  $(v_i, v_j)_{v_i, v_j \in \mathcal{V}}$  或者**边**被定义为样本点。
  - 模型进行推断时，会将图数据作为**整体进行输入**，来预测目标节点对或边的标签信息。

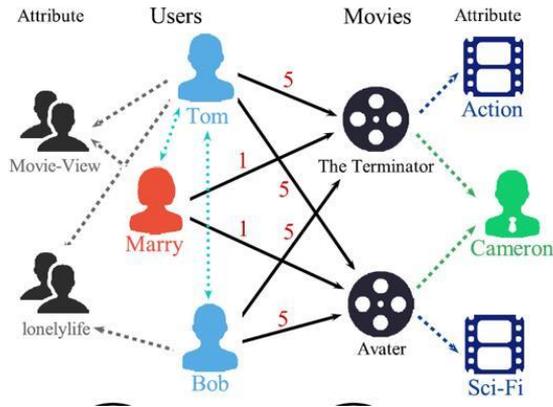
# 1.4.3 边级任务—链接预测

- **链接预测**是最常见的边级任务，其目标是预测图中节点对之间是否会形成边，因此可以被看做一个二分类任务。
- 训练时，**正样本**通常采用图中已经存在连边的节点对，**负样本**则通过在未形成连边的节点对中进行采样。

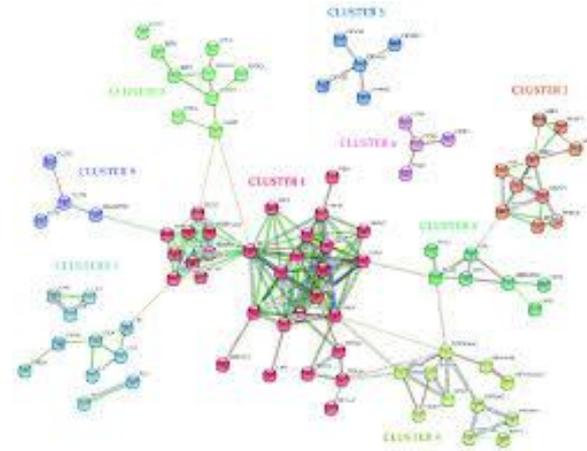


# 1.4.3 边级任务—链接预测

- **链接预测**任务关注的是图中的潜在连接，因此其任务复杂度可能高达  $O(|V|^2)$ ，因此链接预测往往也更加需要关注模型的计算开销。
- 应用



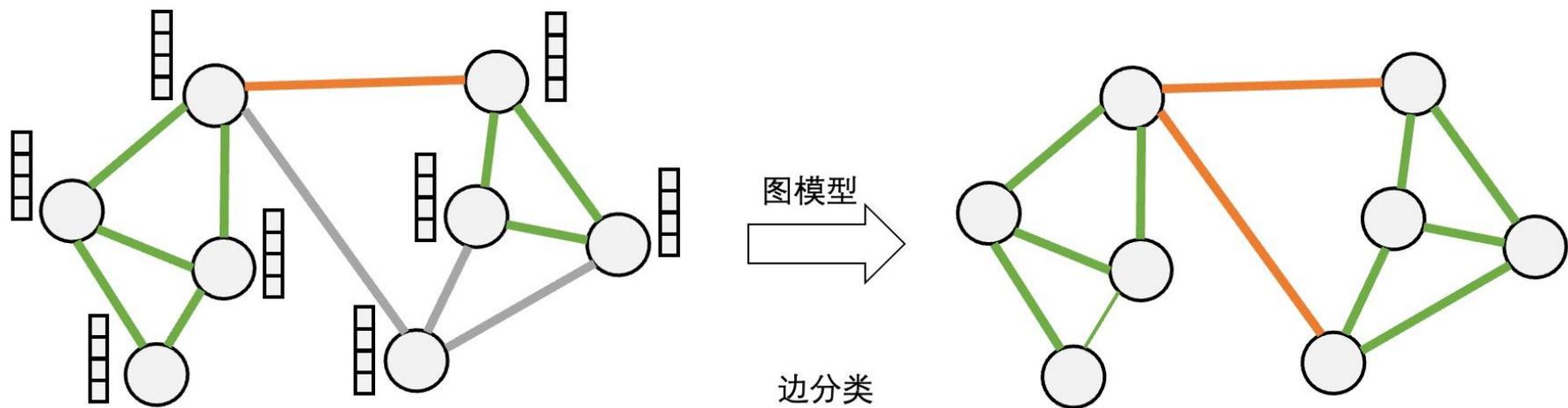
推荐系统



药物交互网络

# 1.4.3 边级任务—边分类

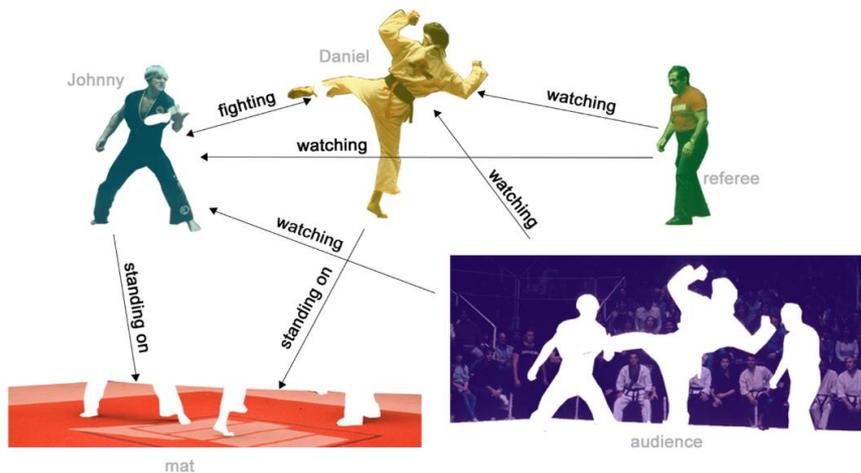
- 边分类旨在利用图中少量边的标签信息，推断剩余未标注边的标签。



# 1.4.3 边级任务一边分类

- 与链接预测主要不同点在于：
  - 采用半监督的学习范式
  - 只考虑图中已有边，任务复杂度较低。
  - 除了边两端节点信息外，还需要利用边信息。

- 应用



实体交互图



交通网络

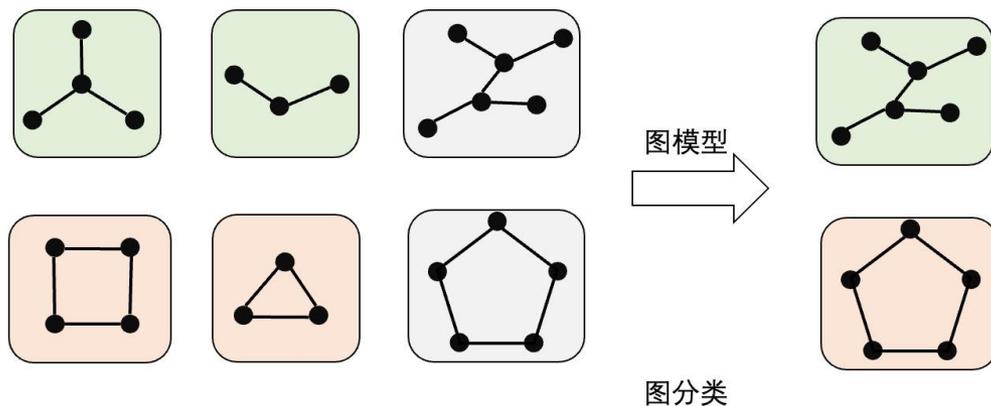


## 1.4.4 图级任务

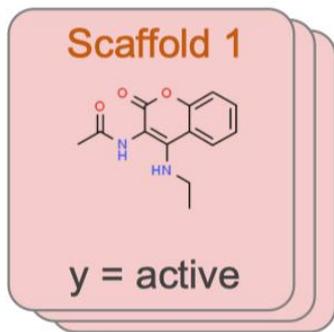
- 定义：现实场景中，许多任务会将图本身作为预测的对象，而非图中的部分节点或者边，这样的任务被称为图级任务。如：
  - 在生物制药领域，常常会将原子表示为节点，原子间的化学键表示为边，使用整张图来建模药物分子，并依此预测分子的理化性质等；
  - 在人体姿态识别任务中，会将人体的关节视为节点，关节间连接表示为边，使用整张图来建模人体骨架，并依此预测姿态类别等。
- 特点：
  - 图级任务的数据集往往不止一张图，我们将其表示为图集合的形式，即  $\mathcal{G} = \{G_1, \dots, G_{|\mathcal{G}|}\}$ 。
  - 图本身被定义为样本点。
  - 模型在进行推断时，通常会将提取出图的整体特征进行预测。

# 1.4.4 图级任务—图分类

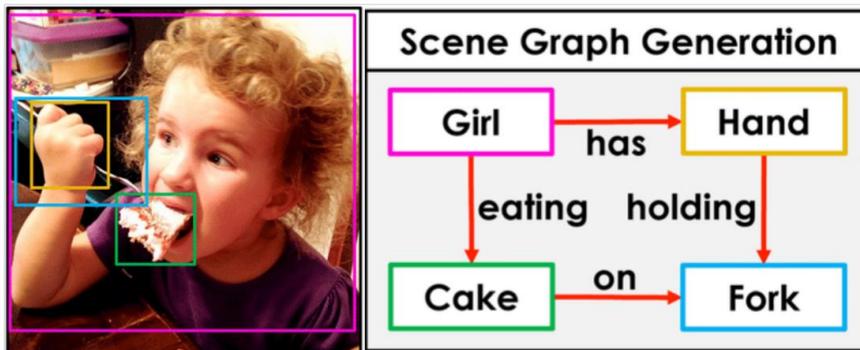
- 图分类是最常见的图级任务，其目标是预测给定图的标签类别，并且往往采用监督学习的训练范式。



- 应用



分子性质预测

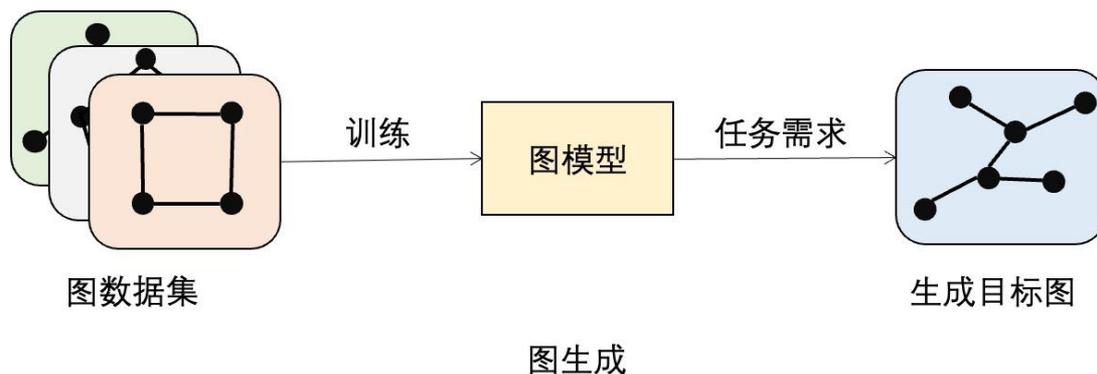


scene graph generation

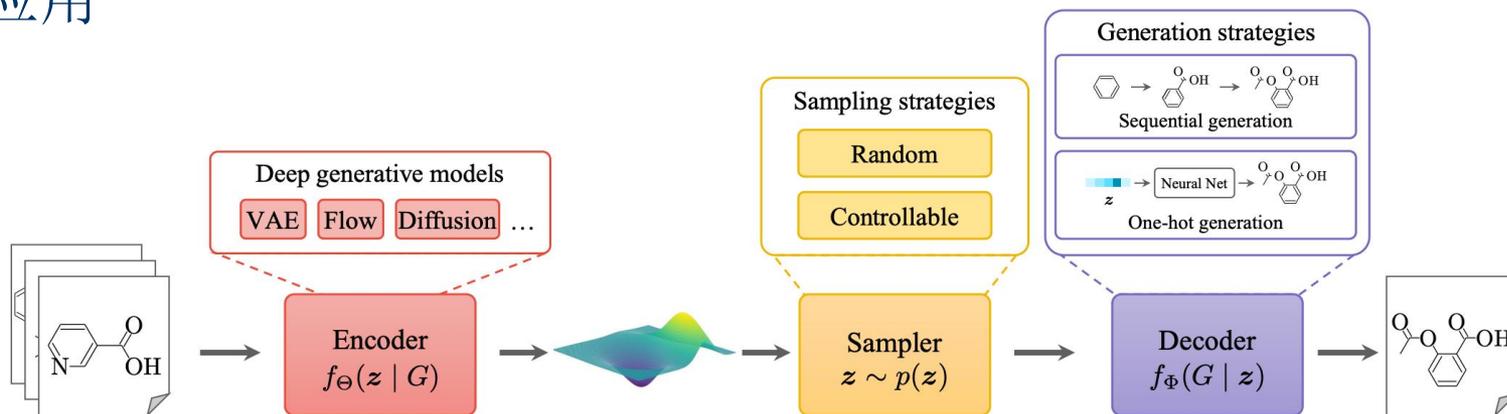
图像理解

# 1.4.4 图级任务—图生成

- **图生成**旨在利用模型捕捉给定图的分布，并通过采样的方式生成具有类似分布的图数据。



- **应用**



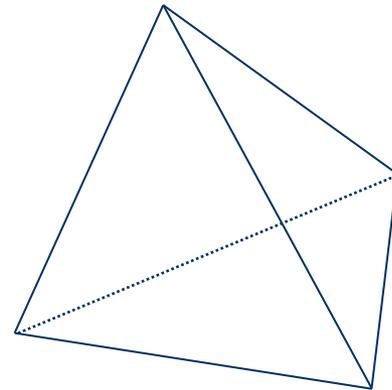
# 1.5 发展历程



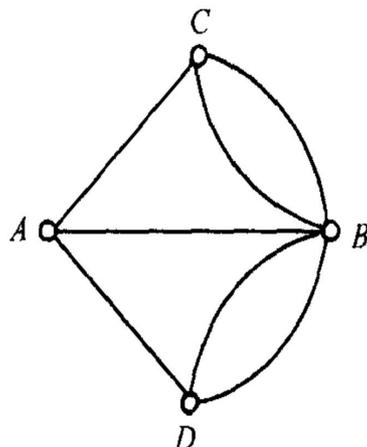
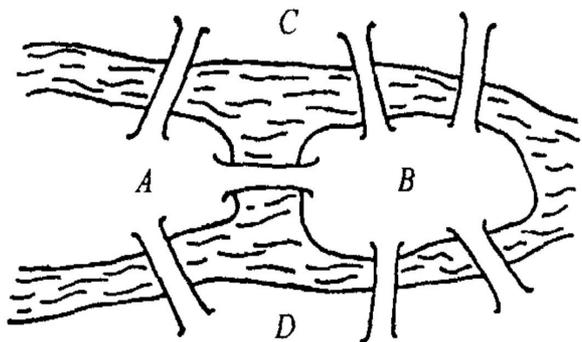
- 1.5.1 图论时期 1736年~至今
- 1.5.2 图算法时期 19世纪中期~至今
- 1.5.3 复杂网络时期 20世纪90年代~至今
- 1.5.4 社交网络分析时期 20世纪90年代~至今
- 1.5.5 图嵌入时期 21世纪00年代~至今
- 1.5.6 图神经网络时期 21世纪00年代~至今
- 1.5.7 未来展望

# 1.5.1 图论时期

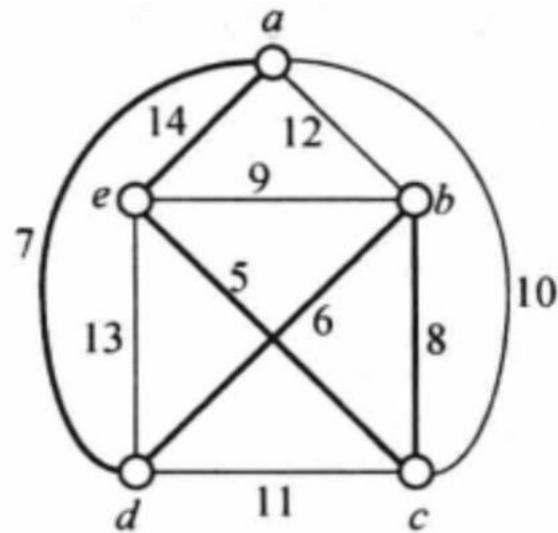
- 时间：1736年~至今。
- 该阶段学者的研究主要在：
  - 对点、线、面进行定义。
  - 对现实问题进行图的建模。
  - 并基于此诞生许多图论基本定理。



欧拉定理： $V+F-E=2$



七桥问题



巡回售货员问题

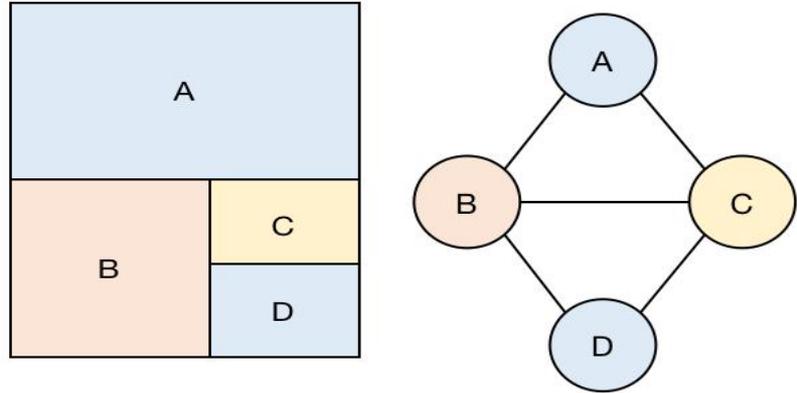
## 1.5.2 图算法时期



- 时间：19世纪中期～至今。
- 该阶段对图的定义愈发丰富与严谨。
- 通过为点、线、图赋予属性，映射现实问题，并产生了多种图算法。
  - 点属性：图着色问题。
  - 边属性：网络流问题：Dijkstra、Bellman-Ford、Floyd-Warshall、Prime、Ford-Fulkerson、最大流最小割定理。
  - 图属性：二分图匹配问题：Hopcroft-Karp、匈牙利算法。

# 1.5.2 图算法时期—图着色问题

- 图的形式化建模
  - 涂色区域—>节点
  - 区域相邻—>连边

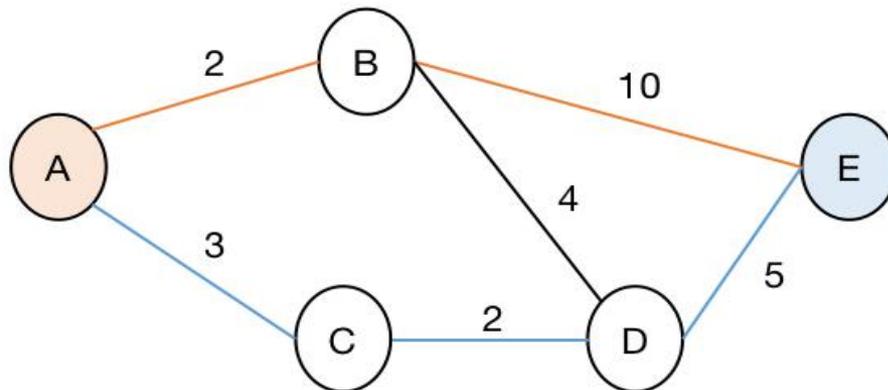


- 判定性问题
  - 给定一定数量的颜色，是否存在相邻区域不同色的涂色方案。
- 优化问题
  - 为使相邻区域不同色，至少需要多少种颜色。



## 1.5.2 图算法时期—网络流问题

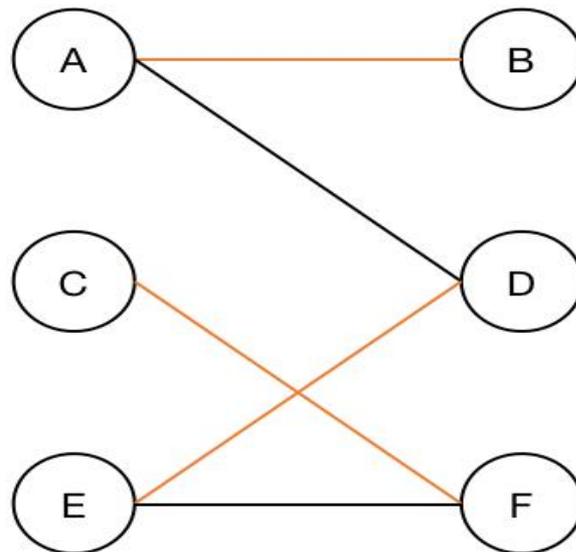
- 为边赋予**权重**，按需求规划起始节点到目标节点的路径。
- 按照需求不同分为：
  - **最大路径**：路径权重和最大（红色）。
  - **最小路径**：路径权重和最小（蓝色）。
  - **最短路径**：权重均为1的最小路径问题。





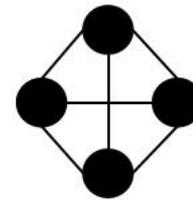
## 1.5.2 图算法时期—二分图匹配问题

- 二分图：将图中节点集分为两个互不相交的子集，子集内节点无边相连（子图性质）。
- 匹配：任意两个选中边无共同顶点。
- 二分图最大匹配：求两个子集间的最大匹配情况（选中边最多）。

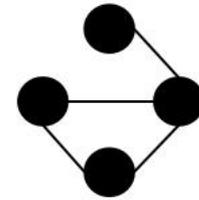


# 1.5.3 复杂网络时期

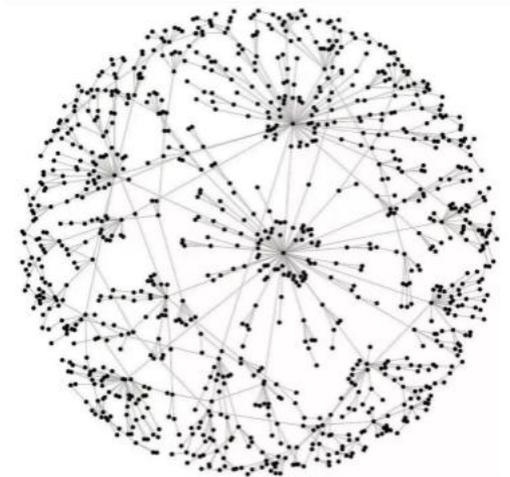
- 时间：20世纪90年代~至今。
- 规则图与随机图的折中。
- 小世界特性：
  - 任意两个节点都有一条很短的路径。
- 无尺度网络：
  - 大部分节点与少部分节点相连，少部分节点与大部分节点相连。
  - 二八定理、长尾效应。
  - 脆弱性→蓄意破坏枢纽节点。
  - 鲁棒性→随机破坏任意节点。



规则图



随机图



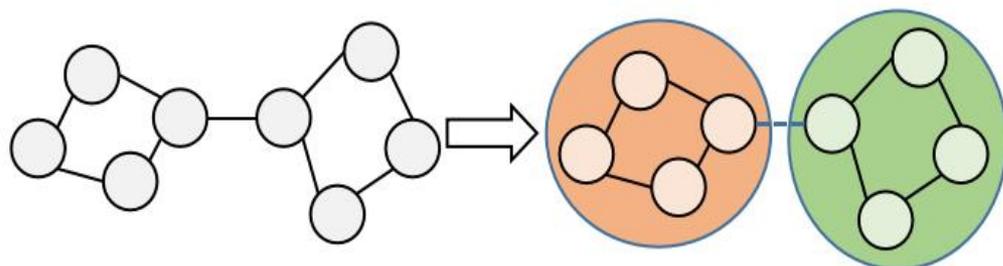
复杂网络

# 1.5.4 社交网络分析时期

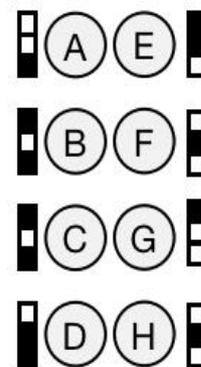
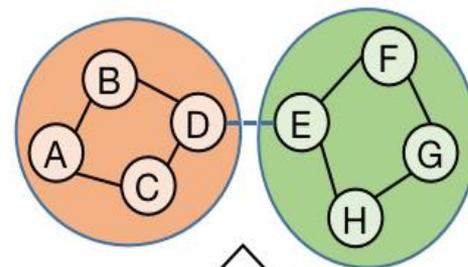


- 时间：21世纪10年代~至今。
- 背景：随着在线社交网络广泛兴起
- 枢纽节点→节点重要性：
  - 使用各种指标进行衡量（度中心性、中介中心性、连接中心性）。
- 小世界→社群发现：
  - 分裂式算法
  - 聚合式算法
  - 共享多标签（节点具有多个社群标签）
  - 随机游走（适应动态图）

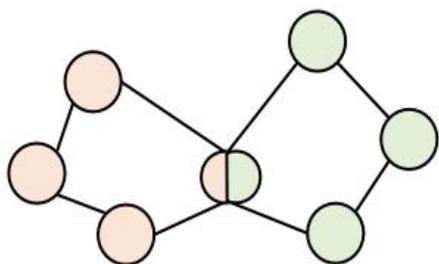
# 1.5.4 社交网络分析时期



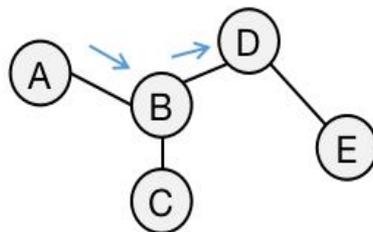
分裂式算法



聚合式算法



共享多标签



随机游走



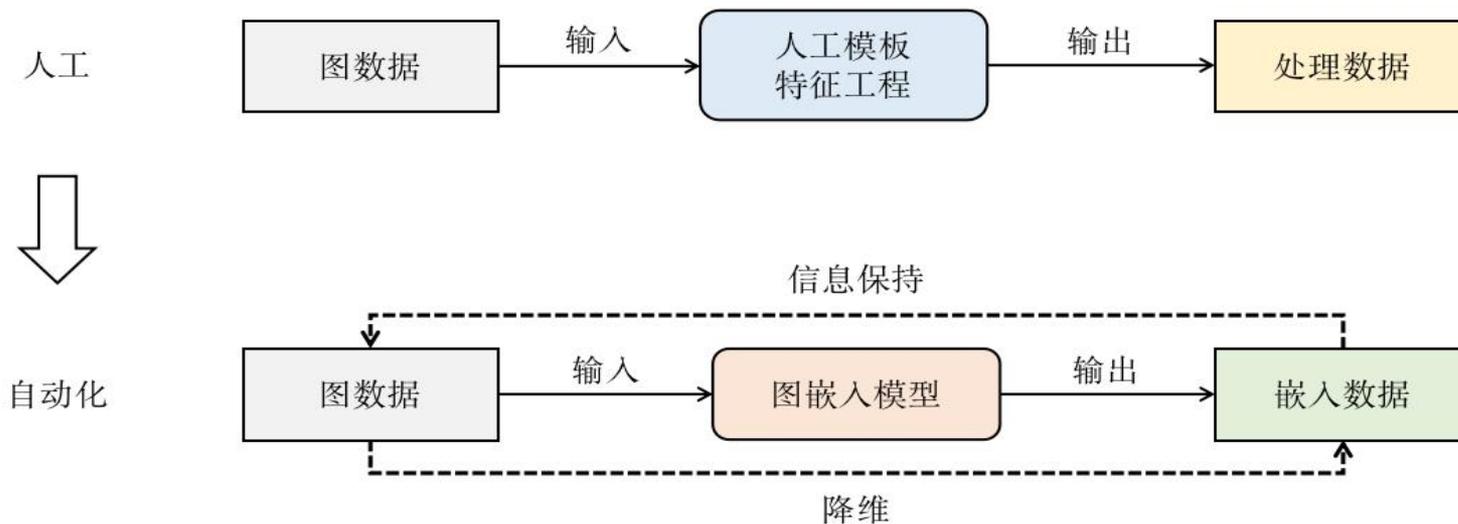
## 1.5.5 图嵌入时期

- 时间：21世纪10年代~至今。
- 背景：图数据广泛存在，如何高效计算与利用。
- 目的：
  - 降维，信息保持（结构信息、侧信息），自动学习特征
- 主流方法：
  - 矩阵分解（拉普拉斯矩阵分解、HOPE、LINE等）
  - 随机游走（DeepWalk、Node2Vec）
  - 侧信息保持（MMDW对标签信息、TADW对属性信息）



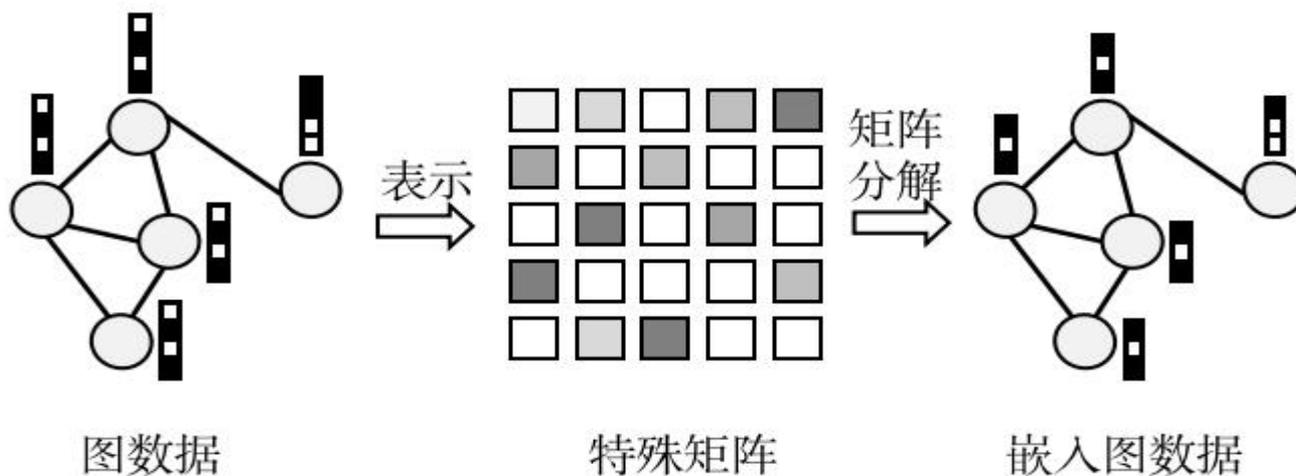
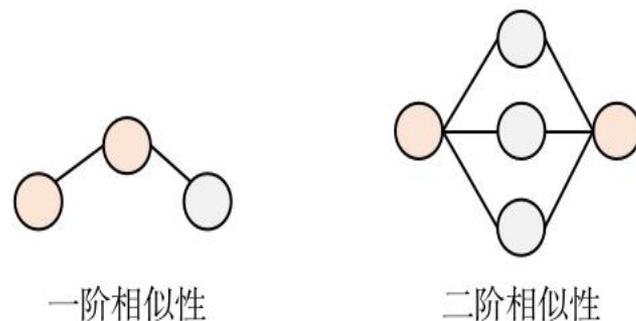
# 1.5.5 图嵌入时期

- 目的：
  - 降维
  - 信息保持（结构信息、侧信息）
  - 自动学习特征



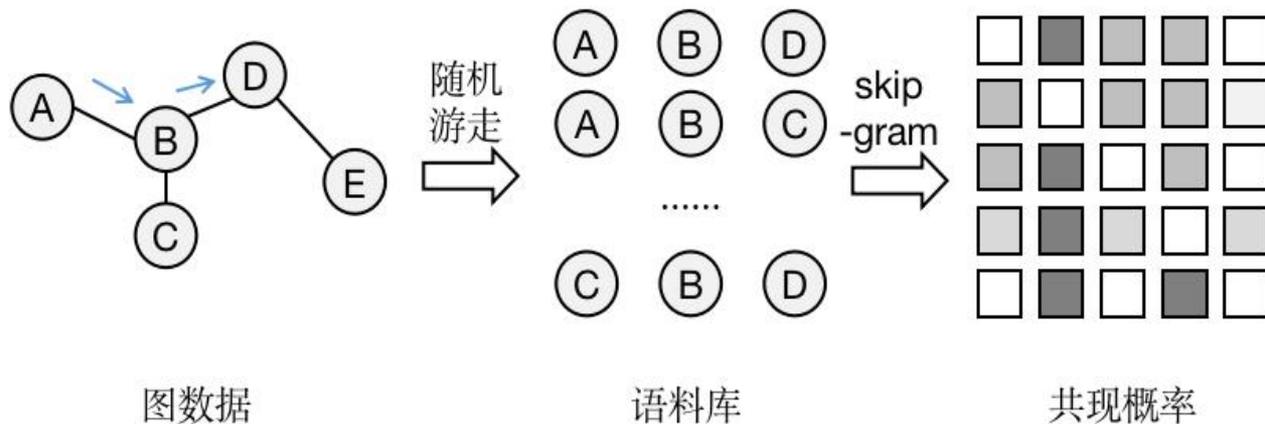
# 1.5.5 图嵌入时期—矩阵分解

- 通过分解矩阵进行图嵌入，根据矩阵不同（拉普拉斯矩阵、度矩阵等），衍生出不同的模型
- 不同模型关注的结构信息也不同：
  - 一阶相似性：邻近节点相似。
  - 二阶相似性：邻域相似的节点相似。



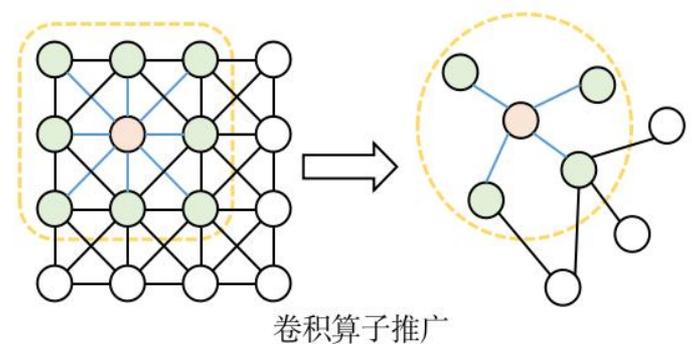
# 1.5.5 图嵌入时期—随机游走

- 用**随机游走**的方式获得**语料库**，通过学习语料库来理解节点的**相似性**并进行嵌入。
  - 采样→语料库                      路径→句子                      节点→单词
- 用**共现概率**去衡量相似性。
- 随机游走本质是一种**矩阵分解**。



# 1.5.6 图神经网络时期

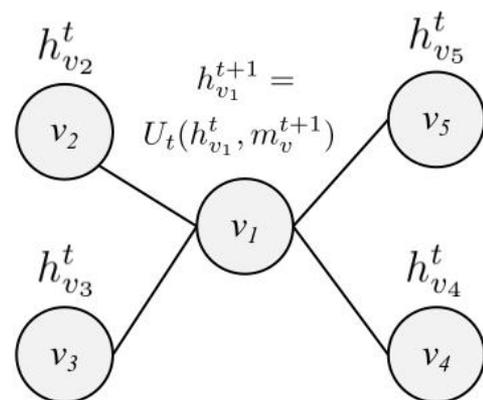
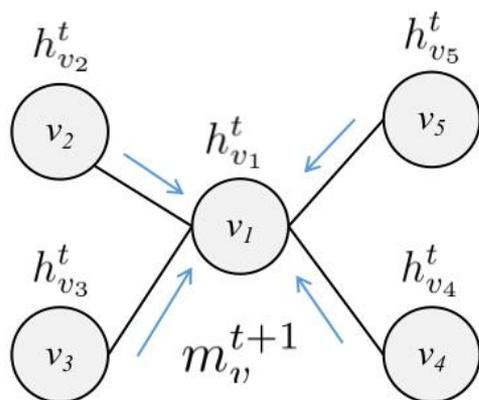
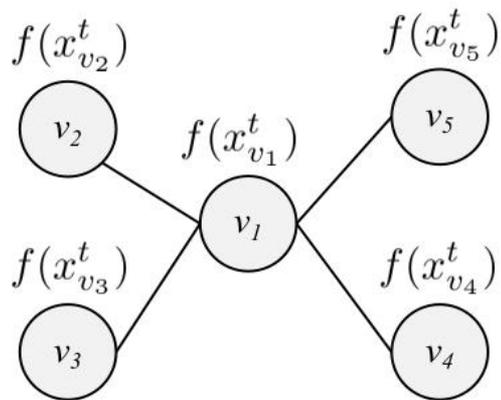
- 时间：21世纪10年代~至今。
- 学者们受RNN、CNN启发，将神经网络迁移到图上：
  - RNN→循环迭代（受图规模限制）
  - CNN→卷积算子通过多层网络、共享参数提取特征
- 比起图嵌入算法，对节点特征有着**更强的表达能力**。
- 给图增加额外属性：
  - 时间——动态图神经网络
  - 时空——时空图神经网络
  - 节点、边的种类——异质图神经网络





# 1.5.6 图神经网络时期—图卷积网络

- 消息传递范式
- 邻居节点特征变换, 聚合邻居节点信息, 更新目标节点特征。



$$h_v^t = f(x_v^t) \quad m_v^{t+1} = \sum_{u \in N(v)} M_t(h_v^t, h_u^t, e_{vu}) \quad h_v^{t+1} = U_t(h_v^t, m_v^{t+1})$$

# 1.5.7 未来展望



- 可信图神经网络：
  - 提高图模型的隐私性、鲁棒性、公平性、可解释性。
- 多模态图模型：
  - 以图的角度处理图数据，或者融合图的多模态数据处理。
- 图基础模型：
  - 通过大量图数据进行预训练，使得模型能快速迁移到下游任务。
- 以数据为中心的图模型：
  - 以模型为中心—模型增大带来性能提升与成本增高，迁移性能差。
  - 以数据为中心—用合适的数据增强模型能力。

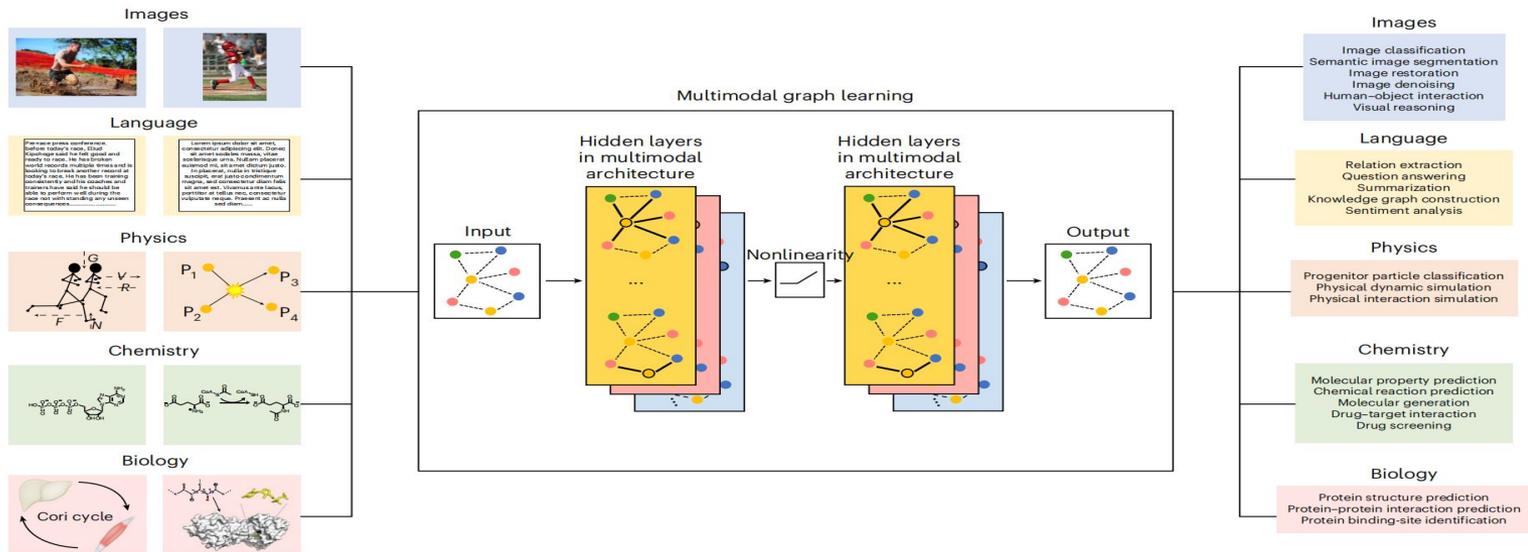
# 1.5.7 未来展望—可信图神经网络



- 定义：
  - 可信图神经网络是指在设计和训练过程中，特别关注**模型的隐私性、鲁棒性、公平性和可解释性**的一类图神经网络。
- 可信度主要有下面四部分：
  - **隐私性**：避免模型在遭受攻击时泄露训练、使用数据的**敏感信息**。（病患信息、金融账户信息等）。
  - **鲁棒性**：减少系统在遭受蓄意攻击而崩溃的情况，确保系统能**稳定运行**。（金融系统、交通系统等）。
  - **公平性**：图数据中难免存在人种、性别、年龄等**偏见**，而图神经网络又可能加深其影响并做出带偏见的决策。
  - **可解释性**：神经网络具有**高度非线性性**，**可解释性差**，在应用于现实时不够放心。（缺少预测合理性、因果逻辑等）。

# 1.5.7 未来展望—多模态模型

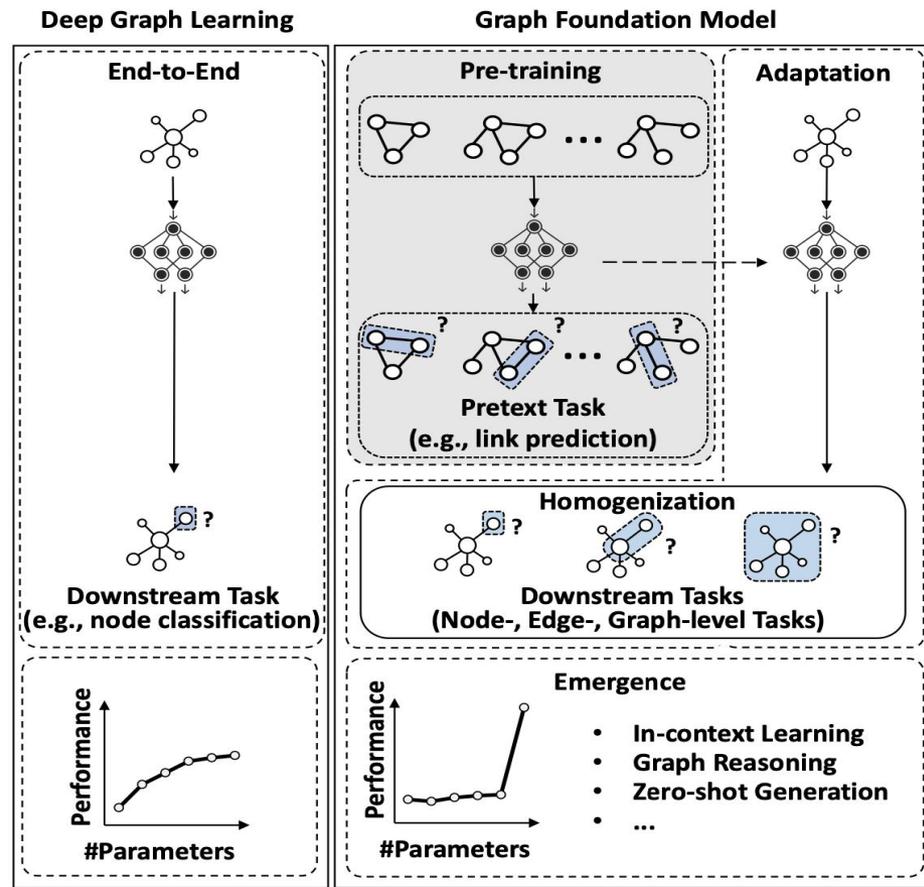
- 定义：
  - 多模态模型是能够处理多种模态（文本、图像、音频等）数据的模型。
- 图的优势：
  - 图能够更有效的帮助模型理解不同模态数据之间关系





# 1.5.7 未来展望—图基础模型

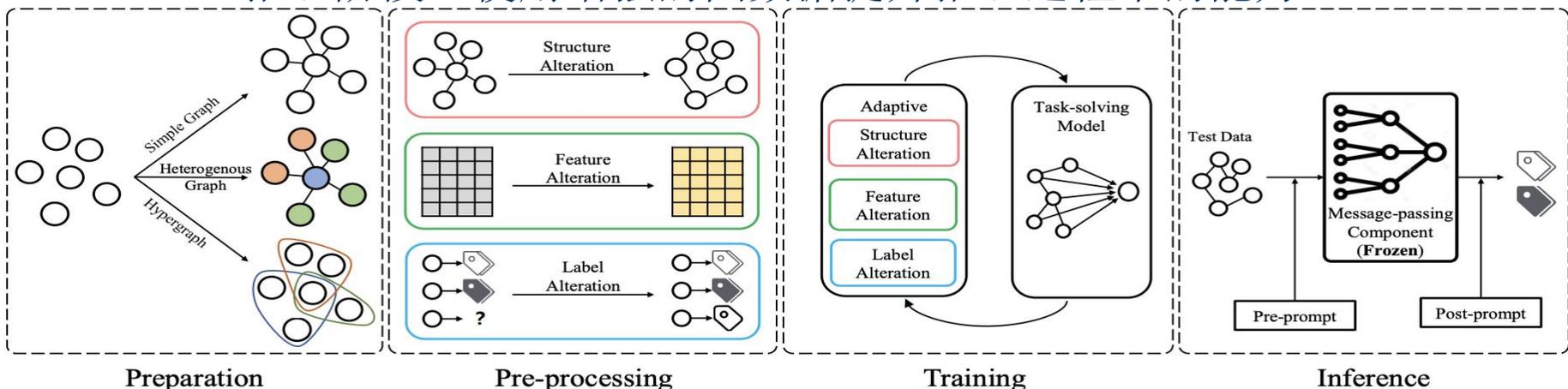
- 定义：
  - 图基础模型是指能在广泛的图数据上进行预训练，并能快速适应各种不同的下游图任务上的图模型。
- 核心：
  - 通过预训练适应下游任务。
- 原理：
  - 大量预训练带来涌现能力。
  - 图任务间可迁移的同质性。





# 1.5.7 未来展望—以数据为中心的图模型

- 定义：
  - 从数据的角度出发，使用合适的数据来增强模型性能的一类模型。
- 核心：
  - 1) 数据准备阶段：按需构建图数据；2) 预处理阶段：增强结构、特征、标签信息；3) 训练阶段：预处理数据与目标任务的协同增强；4) 推理阶段：使用增强的图数据提升推理过程中的能力。





- 课程内容：
  - 图机器学习基础 ；
  - 高级图机器学习；
  - 图机器学习平台与应用 。
- 前沿论文阅读：
  - 阅读相关最新论文；
  - 并进行代表性邀请学者演讲。
- 实践
  - 基于OpenHGNN/GammaGL算法库实现前沿图机器学习模型；
  - 将图机器学习应用到实际任务中。



- 阅读论文：10%
  - 每位同学阅读一篇图机器学习相关的前沿论文
- 课堂作业：30%
  - 2-3次的记录课程评分的作业
- 课程项目：60%
  - 以下两个项目中选择其一进行完成并提交报告：
    - 利用现有图机器学习算法解决实际应用任务，完成训练和评估。
    - 利用图神经网络算法库实现前沿的图机器学习模型的复现。

## 课程老师



石川



杨成



- Graph Neural Networks: Foundations, Frontiers, and Applications
  - <https://graph-neural-networks.github.io/index.html>
- Deep learning on Graphs. Yao Ma and Jiliang Tang.
  - <https://cse.msu.edu/~mayao4/dlg-book>
- Heterogeneous Graph Representation Learning and Applications
- Advances in Graph Neural Networks
  - On my webpage [www.shichuan.org](http://www.shichuan.org)



- 在线课程：
  - C224W|Home <http://web.stanford.edu/class/cs224w/>
- 图机器学习相关论文和代码网址：
  - 图机器学习顶会论文 <https://github.com/naganandy/graph-based-deep-learning-literature>
  - Papers with Code - Papers With Code : Search for graph neural networks
    - [https://paperswithcode.com/search?q\\_meta=&q=graph+neural+networks](https://paperswithcode.com/search?q_meta=&q=graph+neural+networks)
  - [www.shichuan.org](http://www.shichuan.org)

- 1、给出一个图的现实应用的例子，并指出其中点、线、图的含义。
- 2、给出节点级、边级、图级任务例子各一个。
- 3、计算下面图的邻接矩阵、邻接表、关联矩阵、拉普拉斯矩阵以及标准化拉普拉斯矩阵。

