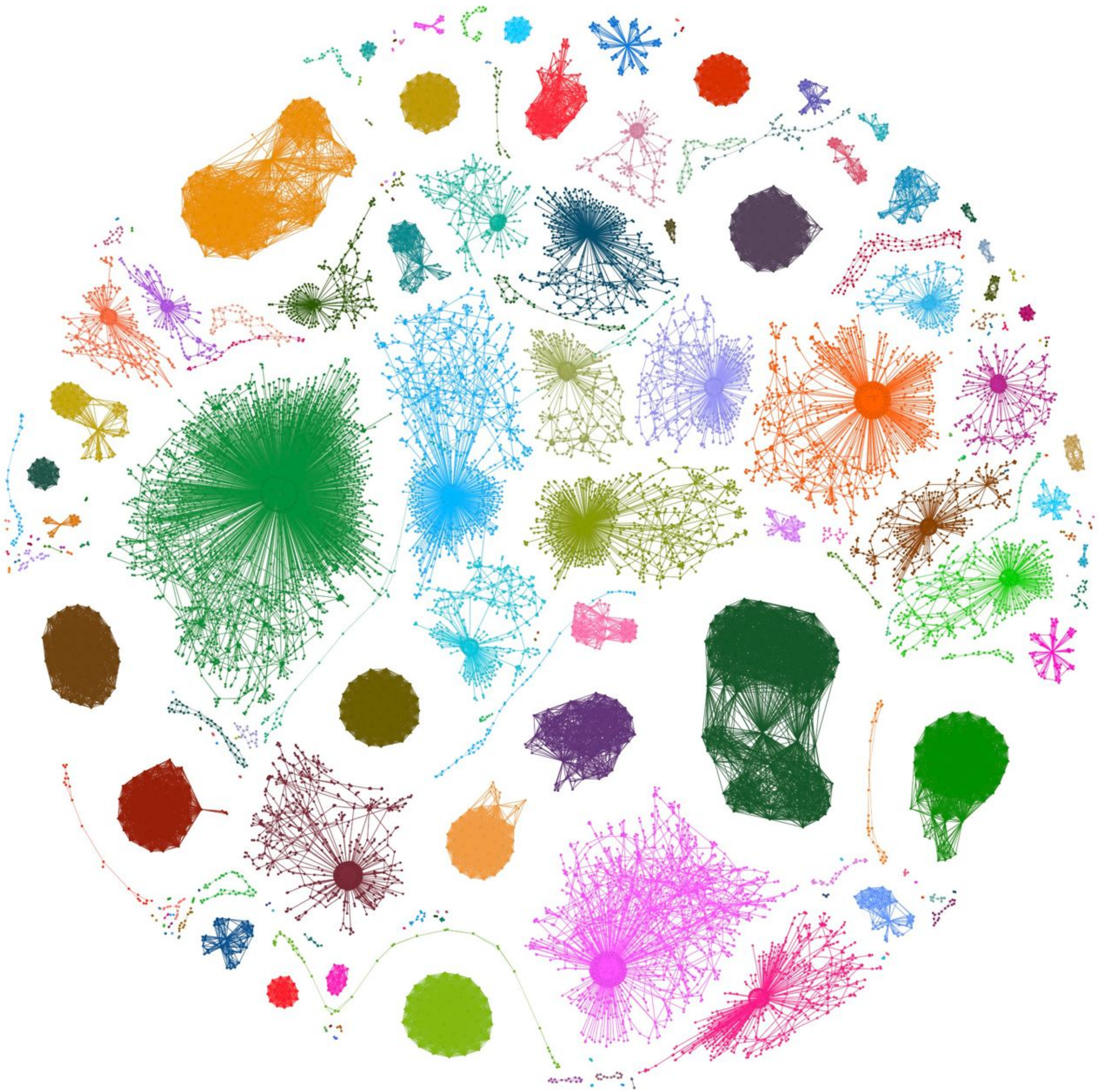
Yandex Research

HSE University Faculty of Computer science

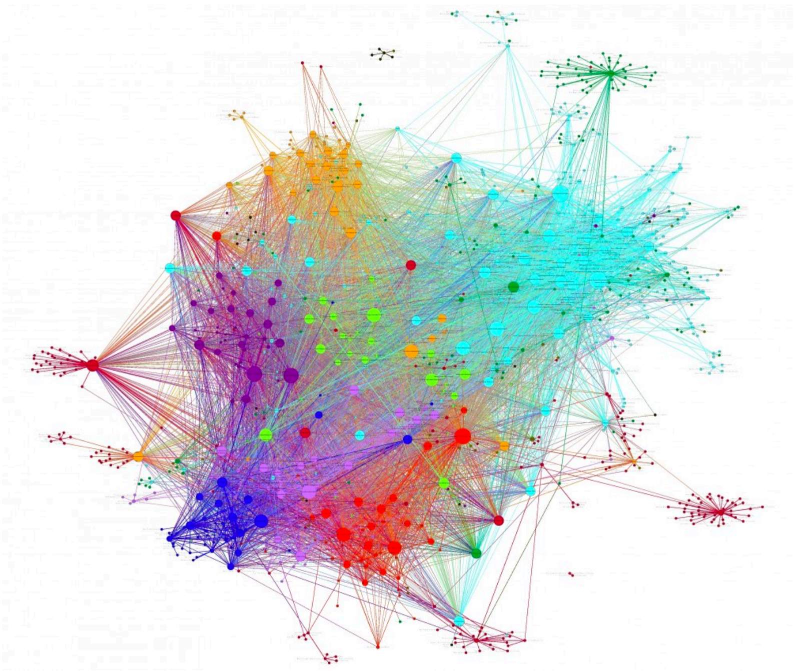
Intro to graph machine learning

Graphs, you know...

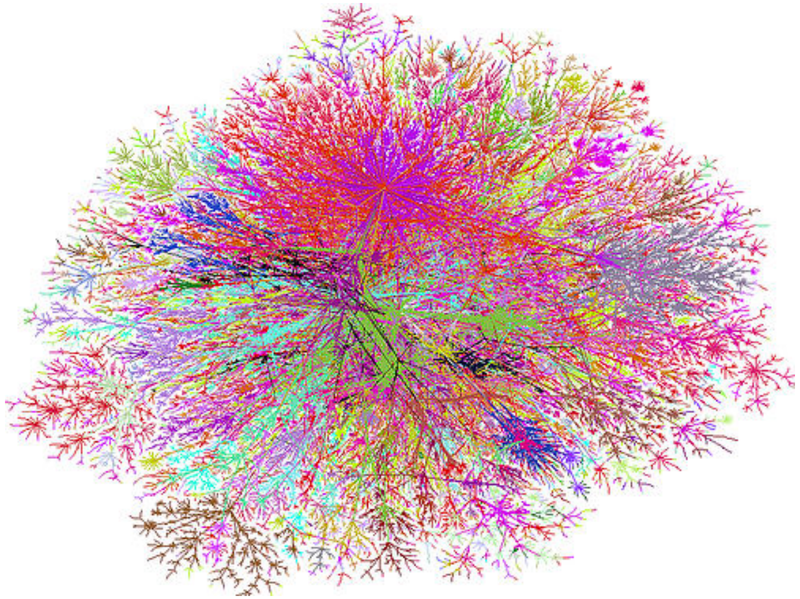
- $G = (V, E)$ — graph
- V — set of objects
- E — set of pairs of related objects
- $(u, v) \in E$ — (maybe directed) edge



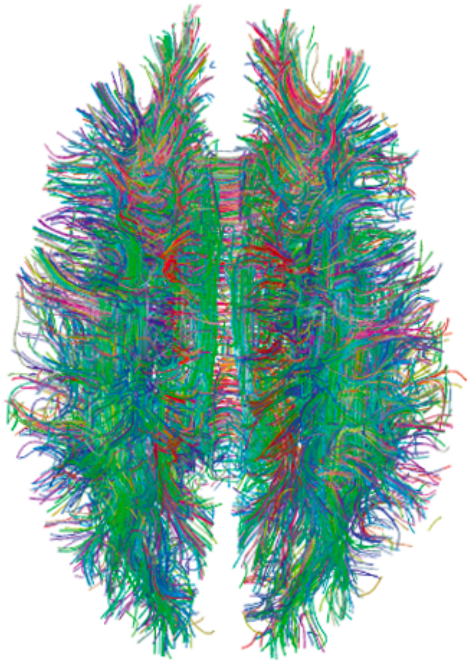
Examples of real-world graphs



social networks



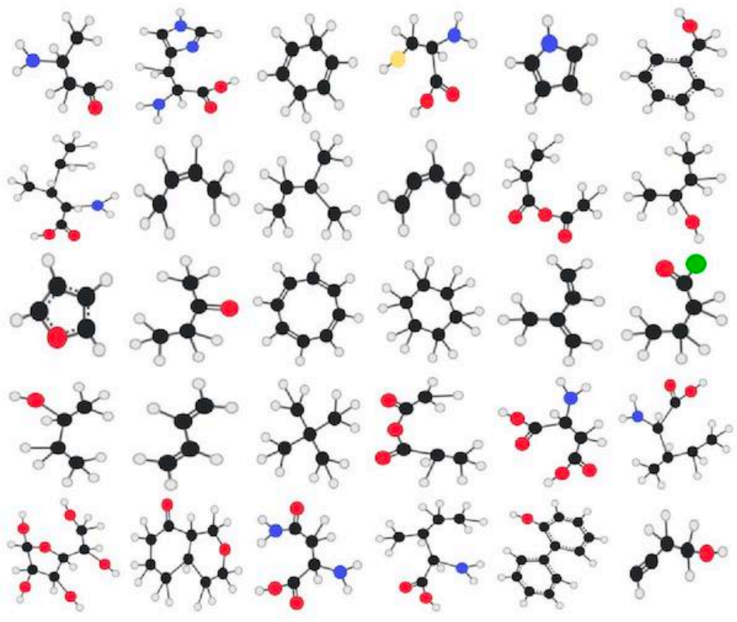
the internet



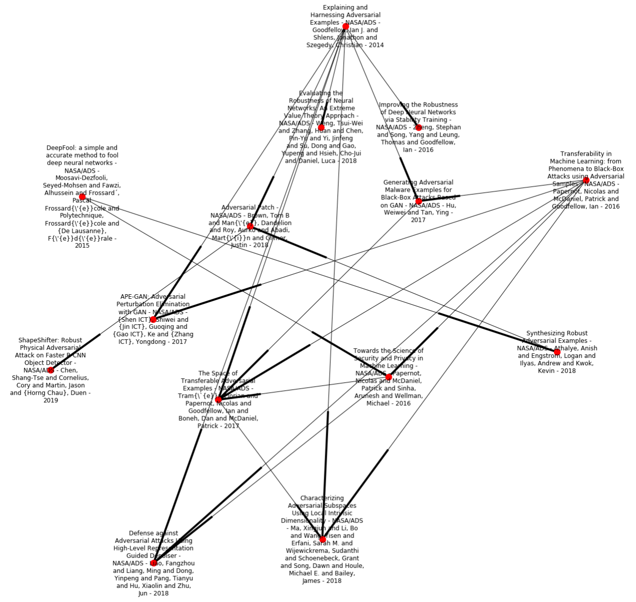
natural neural networks

Examples of real-world graphs

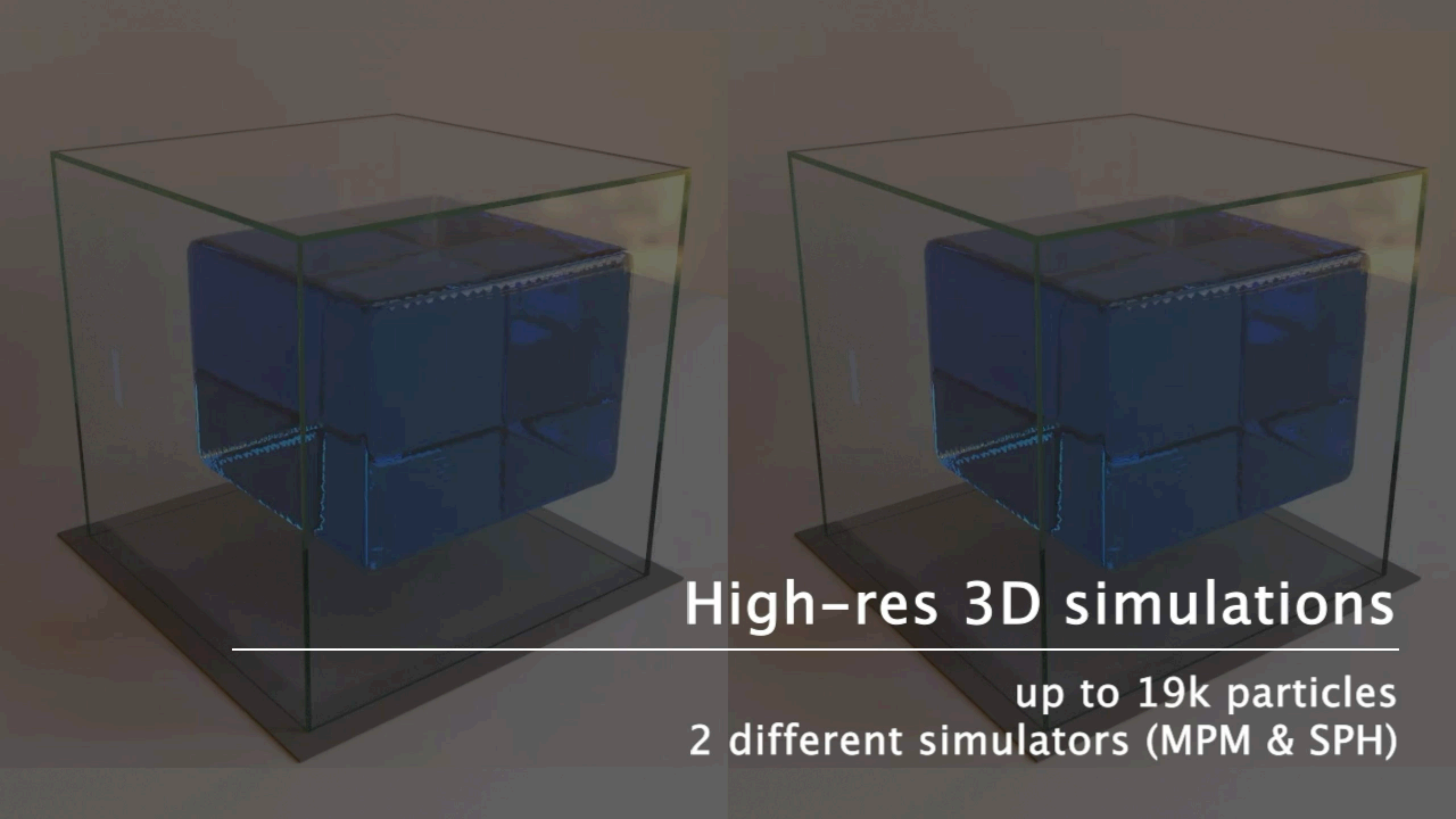
metro networks



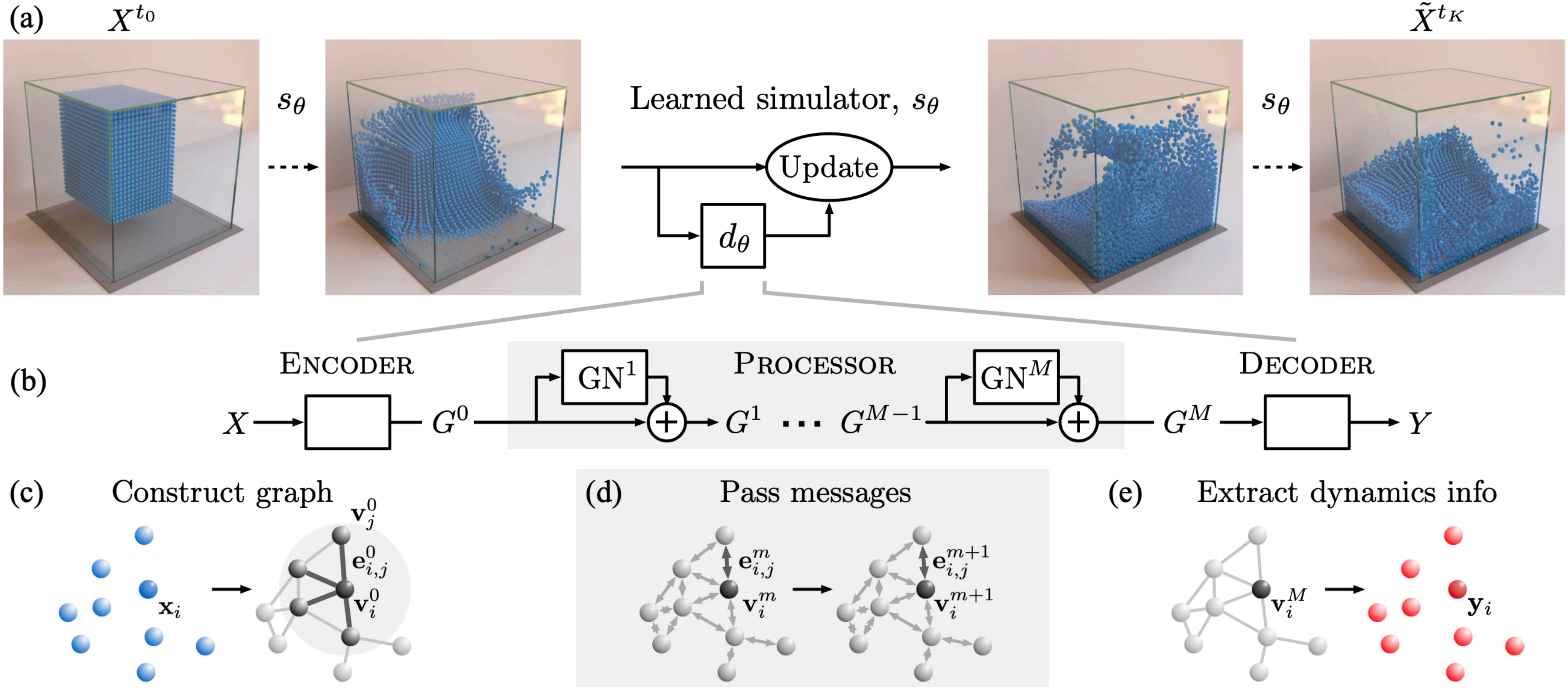
molecular graphs



citation networks


High-res 3D simulations
up to 19k particles
2 different simulators (MPM & SPH)

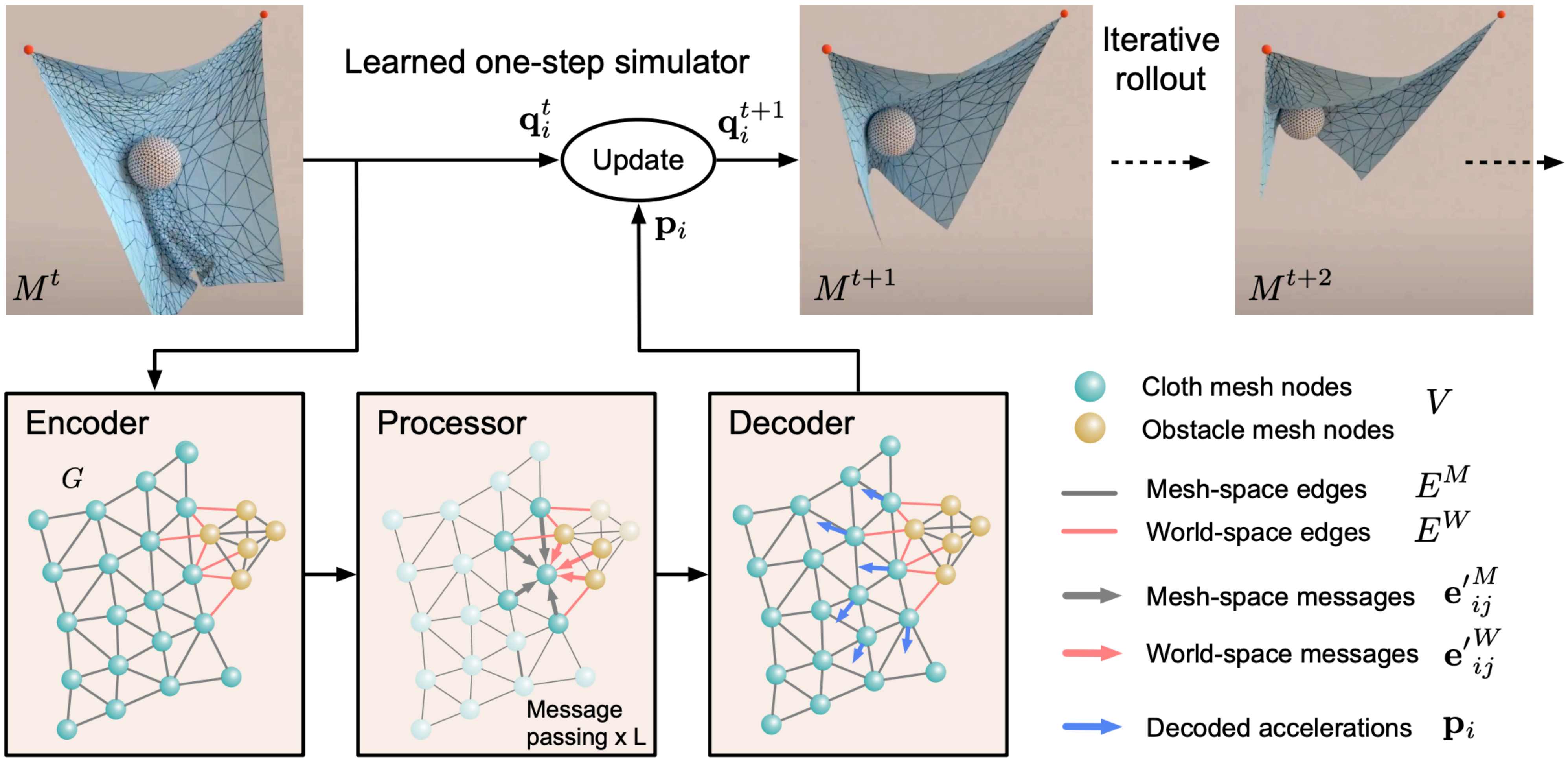
Application: simulating complex physics



A. Sanchez-Gonzalez et al. “Learning to simulate complex physics with graph networks”, ICML 2020

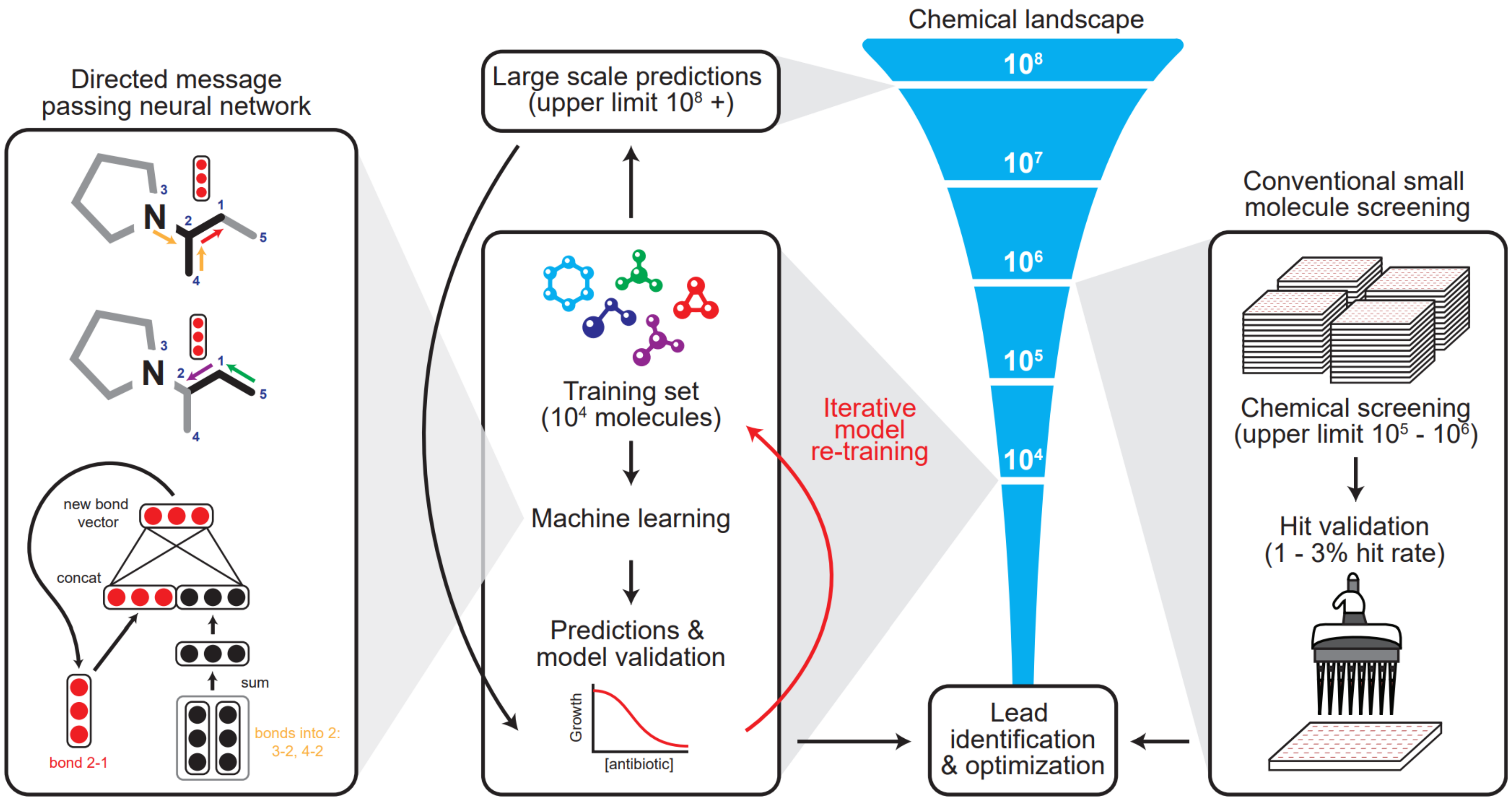
We represent physical systems as meshes

Application: simulating complex physics



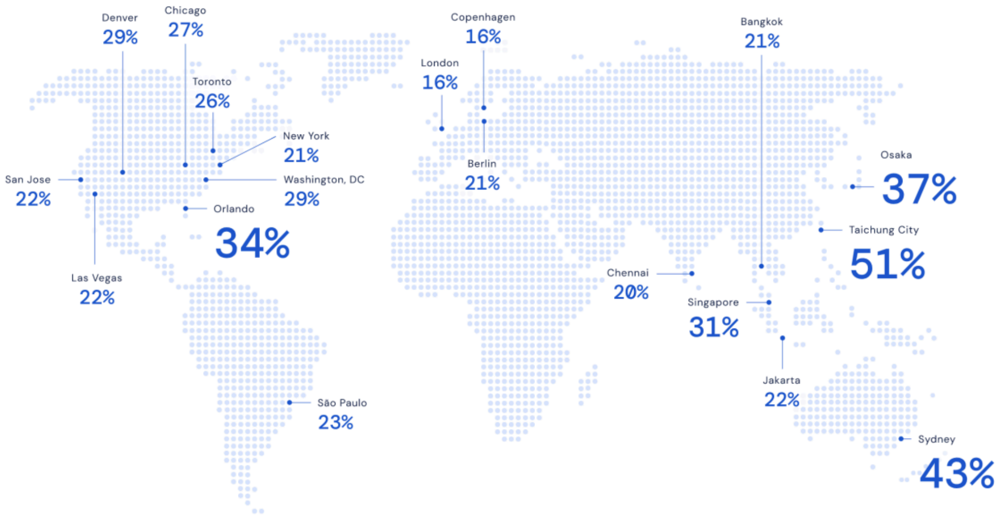
T. Pfaff et al. "Learning Mesh-Based Simulation with Graph Networks", ICLR 2021

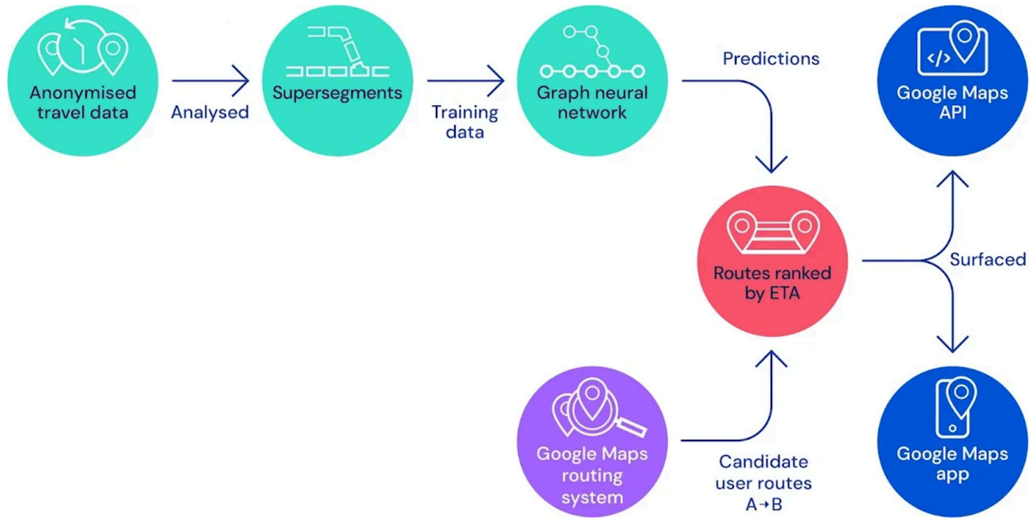
Application: drug discovery

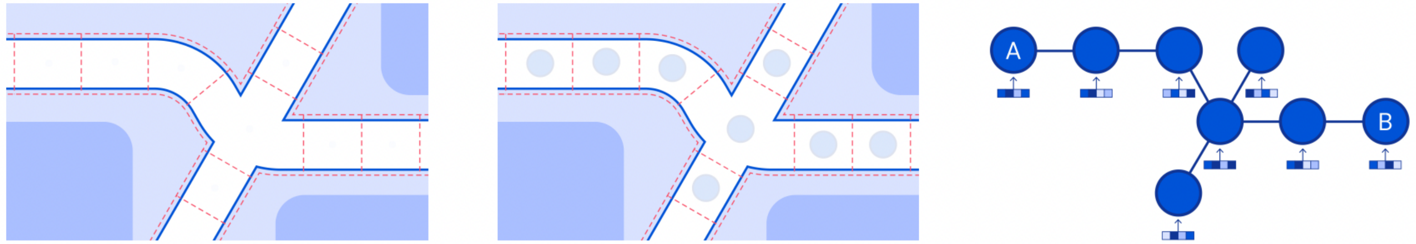


J.M. Stokes et al. "A deep learning approach to antibiotic discovery", Cell 2020

Application: ETA prediction



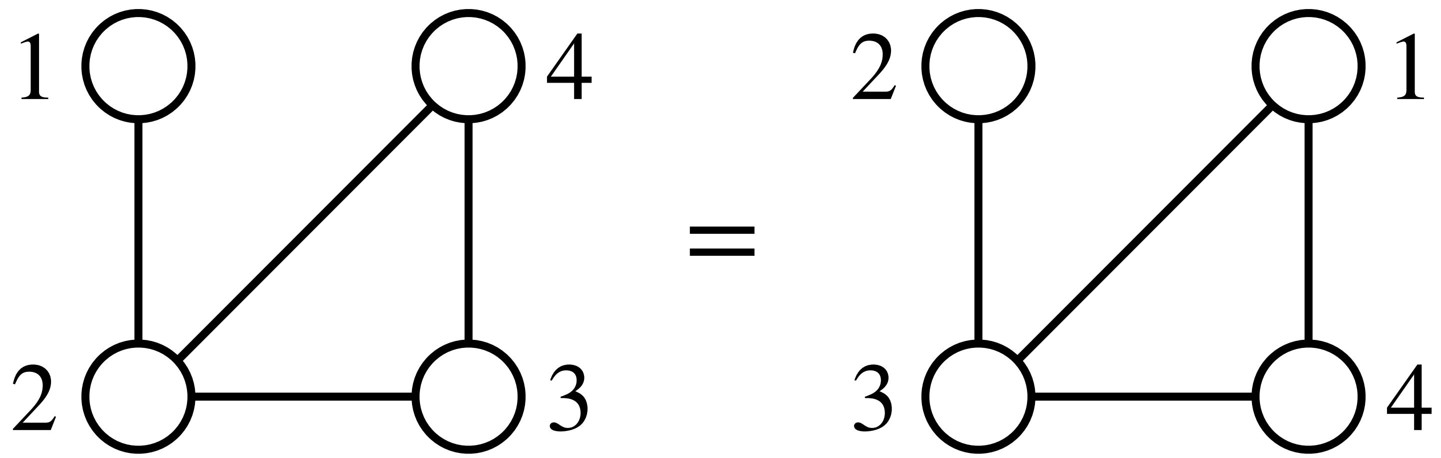




A. Derrow-Pinion et al. “ETA Prediction with Graph Neural Networks in Google Maps”, CIKM 2021

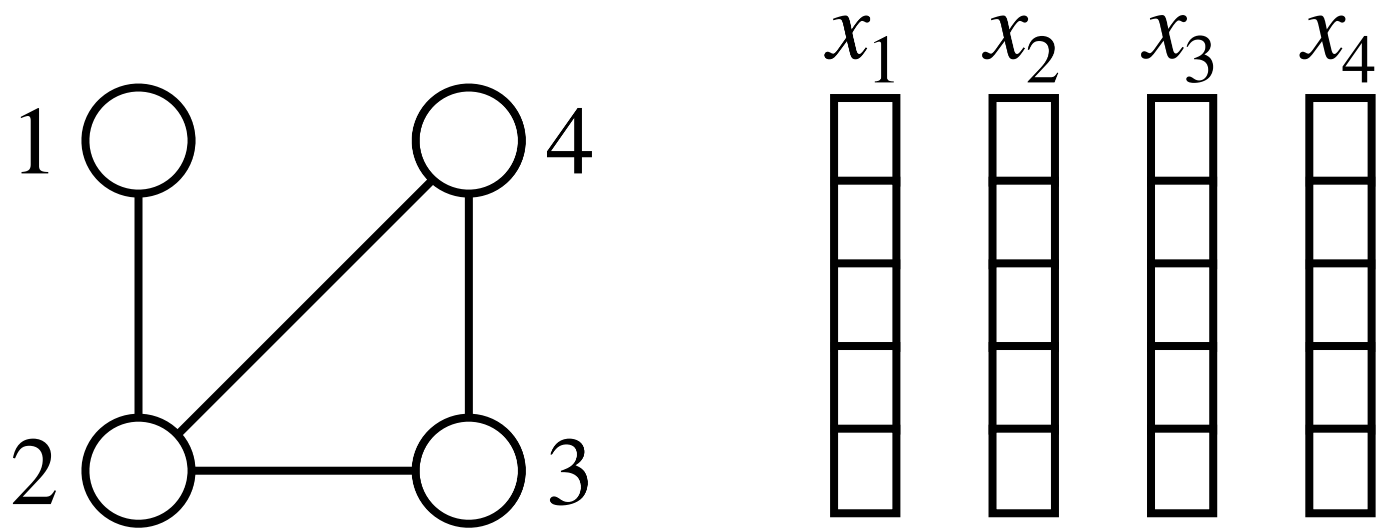
Challenges in learning on graphs

- Nodes in graphs have **no specific order**
- There can be **various features** on nodes, edges and graphs
- Graphs can have **different scale** and **structure**
- And they can even **change over time**



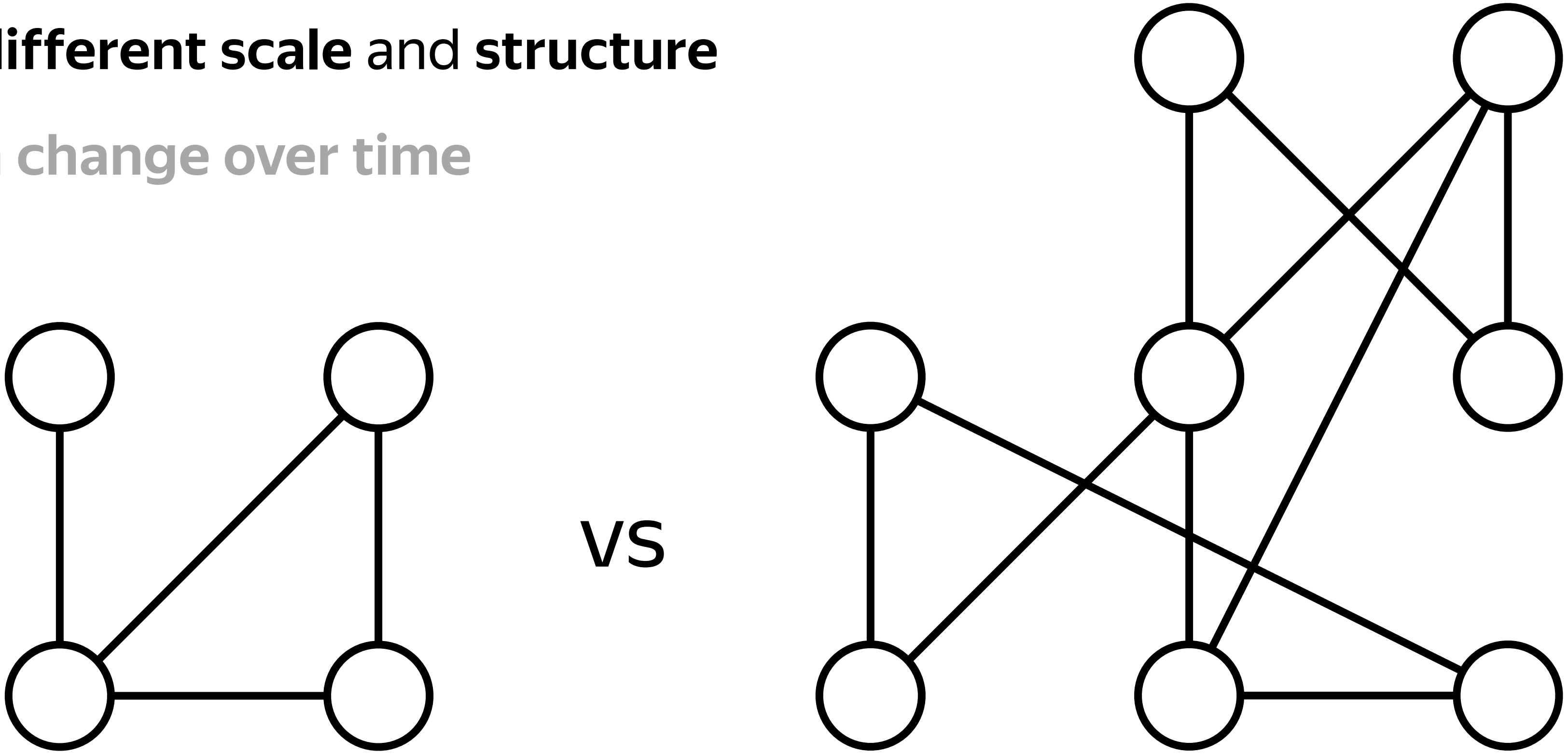
Challenges in learning on graphs

- Nodes in graphs have **no specific order**
- There can be **various features** on nodes, edges and graphs
- Graphs can have **different scale** and **structure**
- And they can even **change over time**



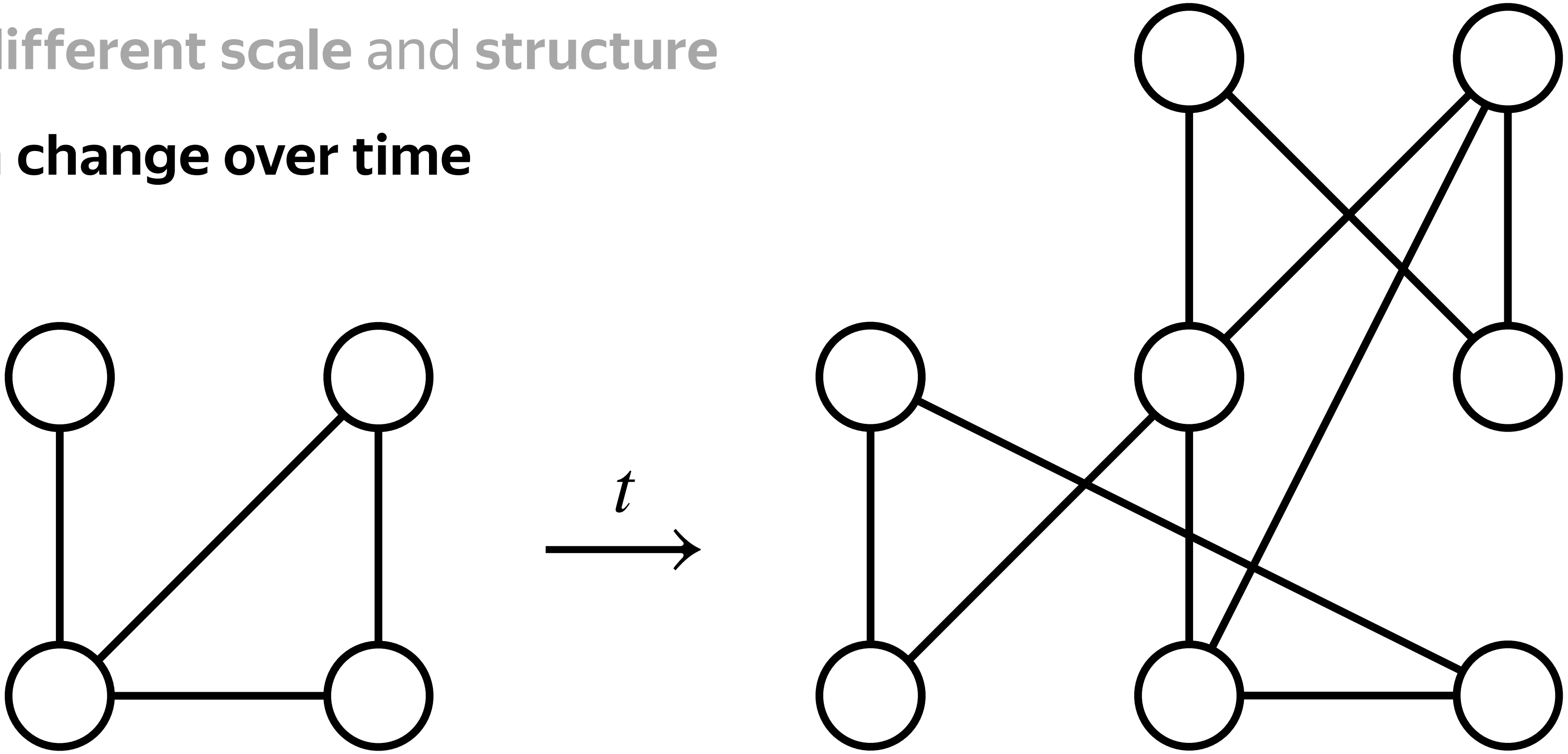
Challenges in learning on graphs

- Nodes in graphs have **no specific order**
- There can be **various features** on nodes, edges and graphs
- Graphs can have **different scale** and **structure**
- And they can even **change over time**

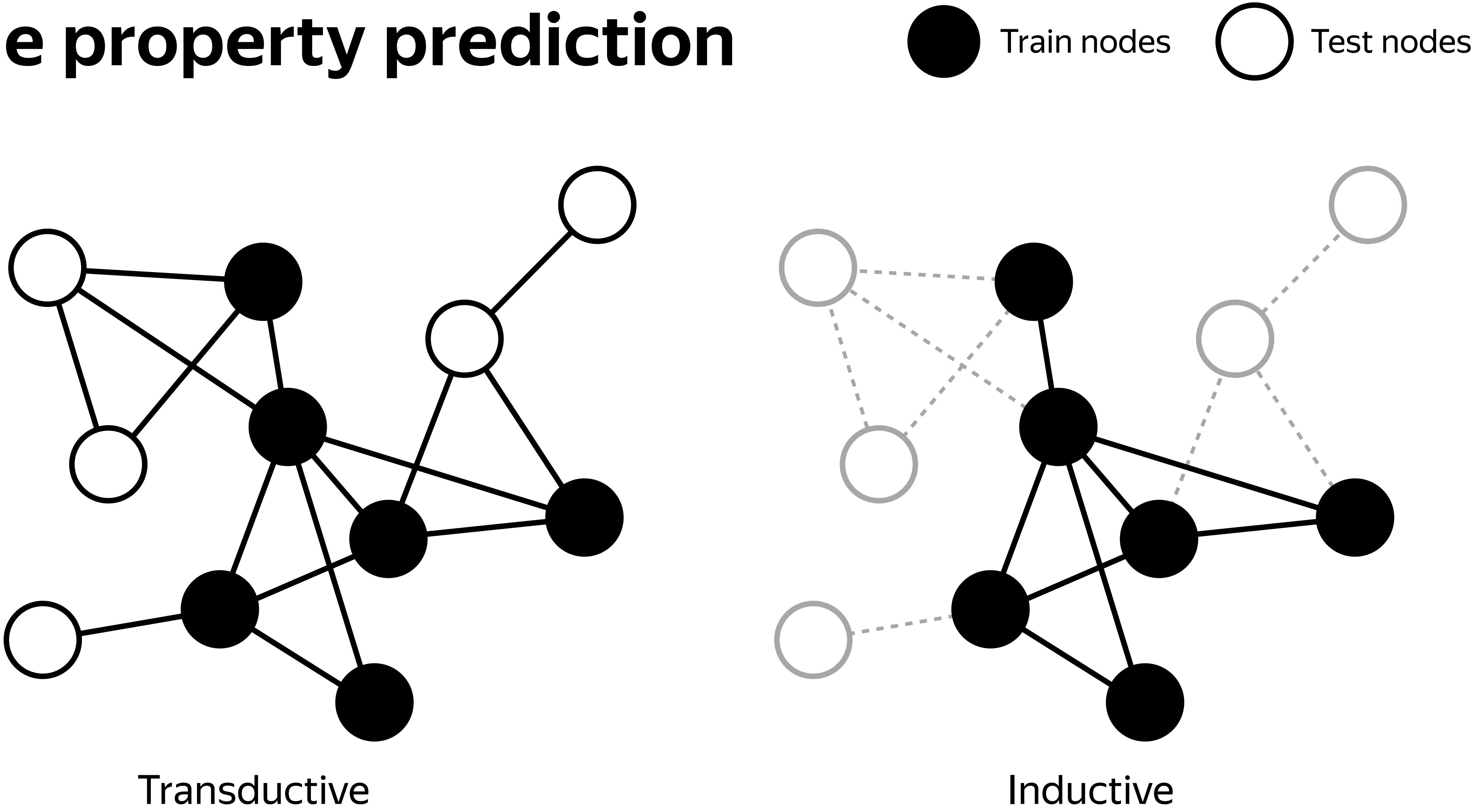


Challenges in learning on graphs

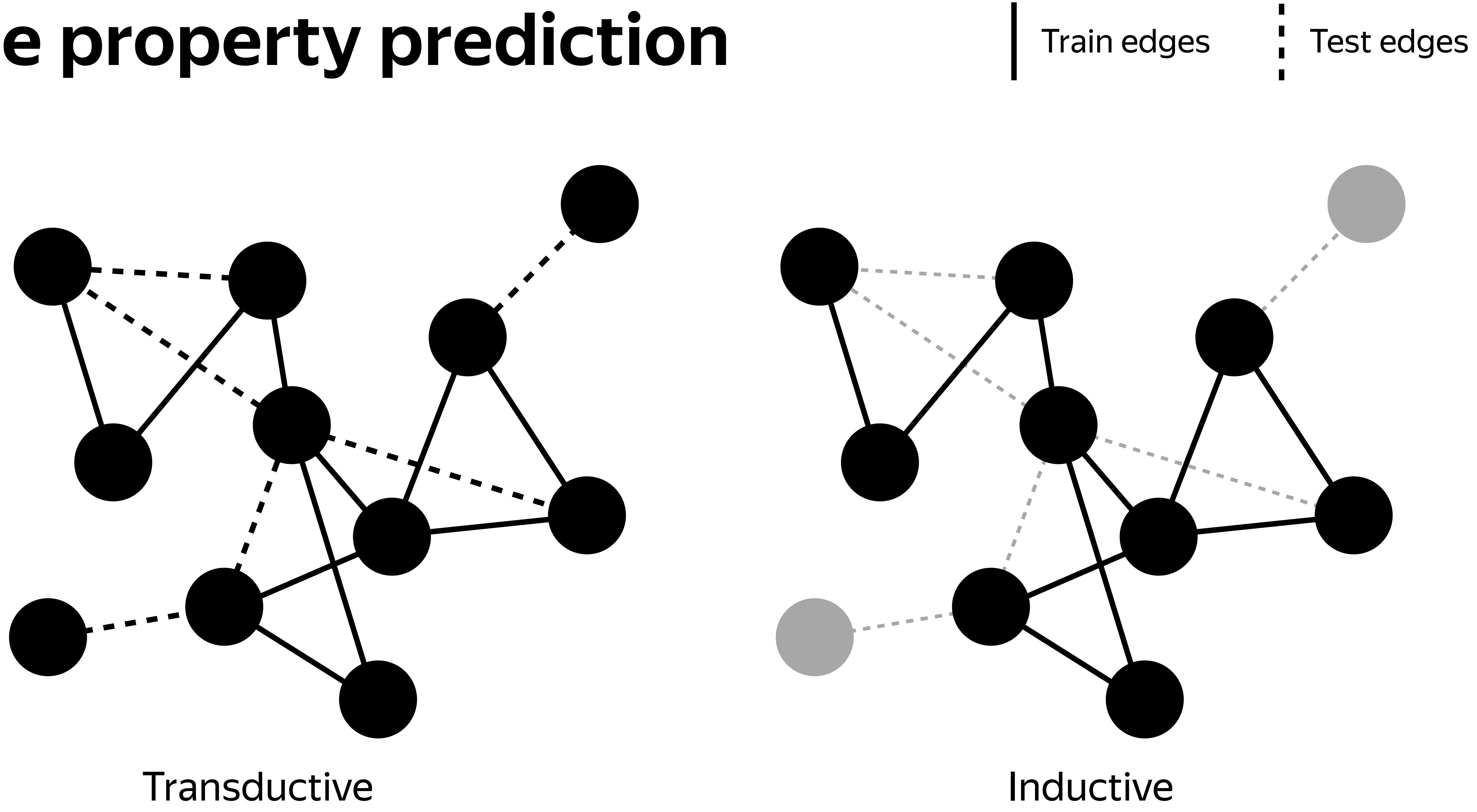
- Nodes in graphs have **no specific order**
- There can be **various features** on nodes, edges and graphs
- Graphs can have **different scale** and **structure**
- And they can even **change over time**



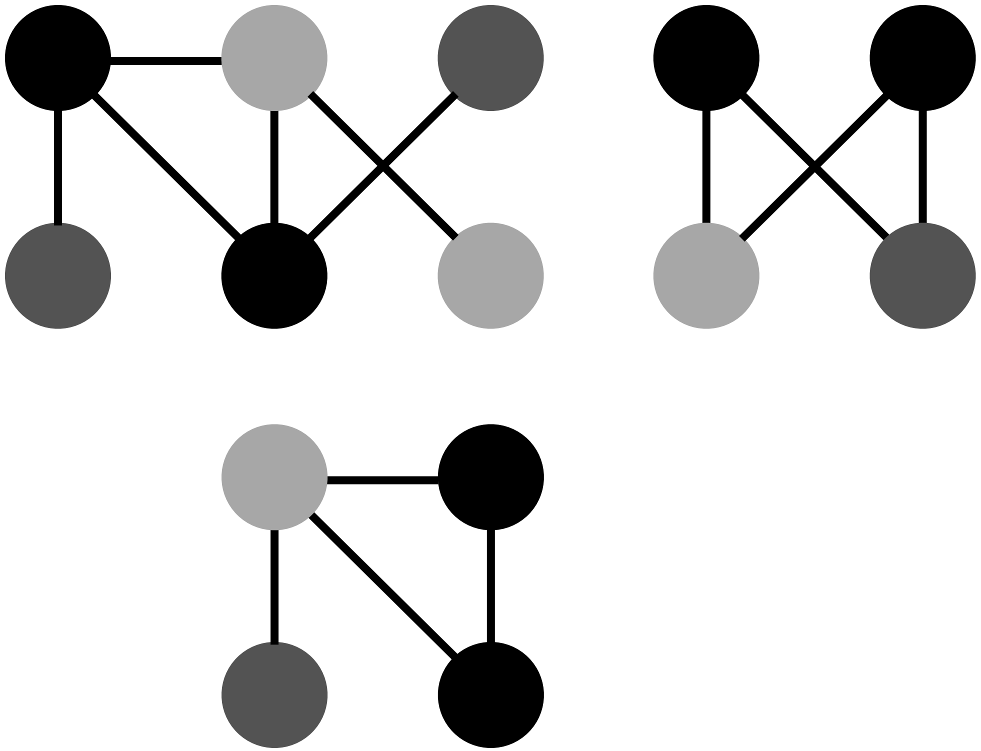
Node property prediction



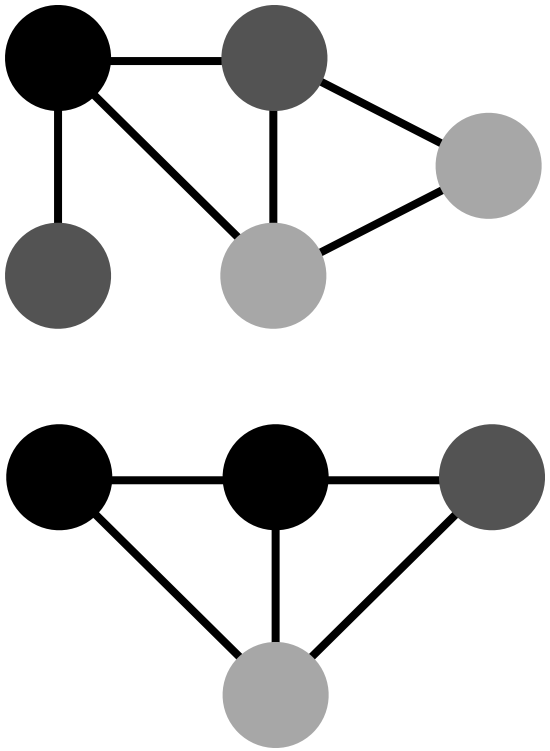
Edge property prediction



Graph property prediction



Train graphs

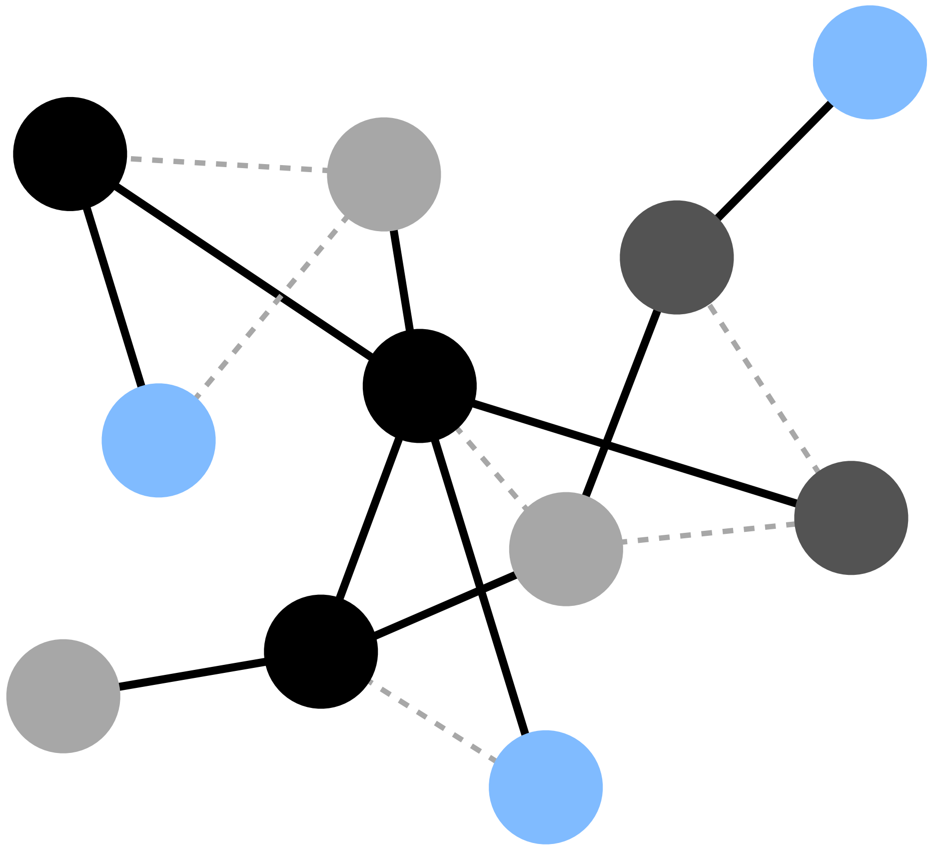


Test graphs

Link prediction

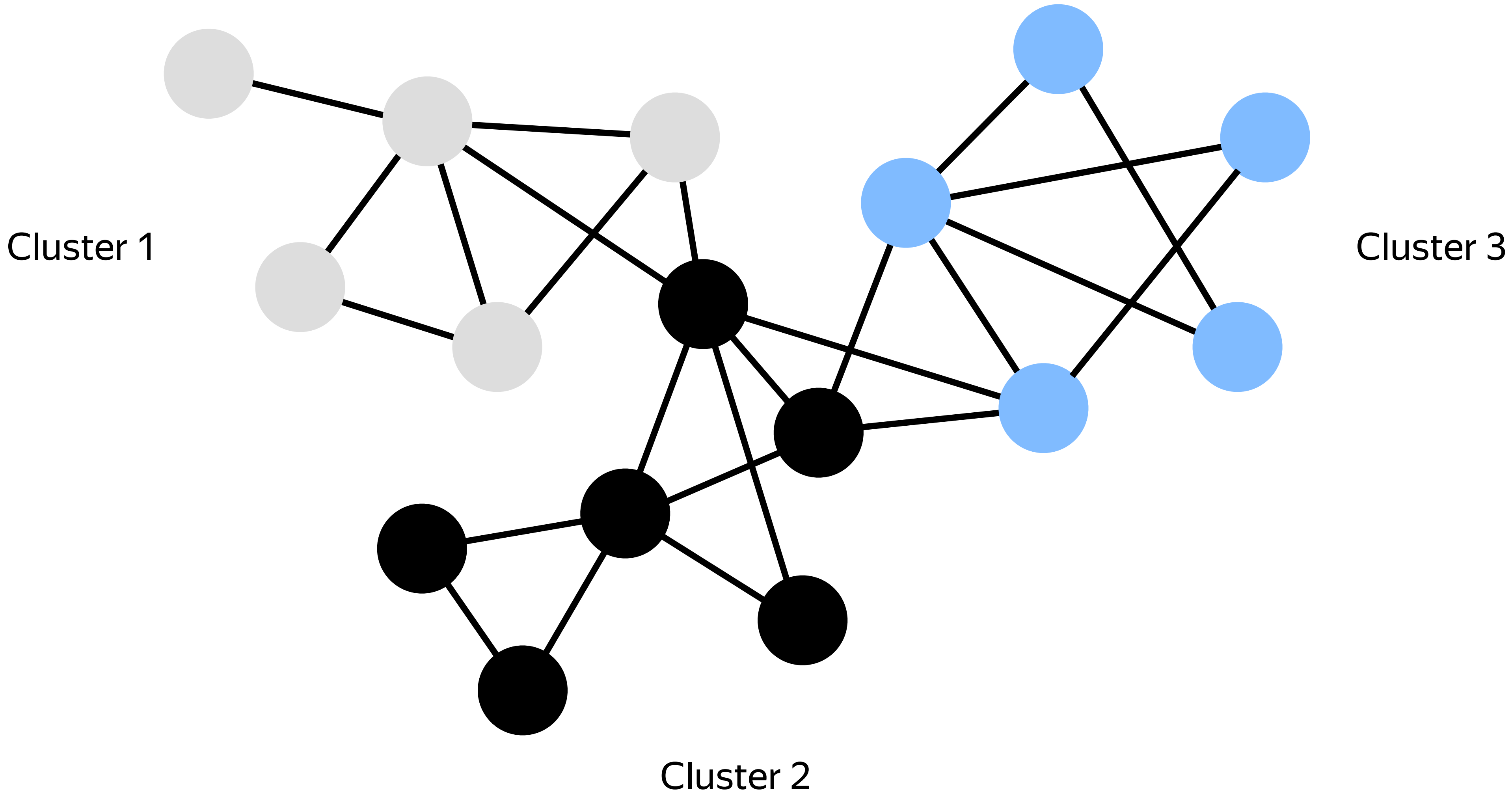
Existing edges

New edges



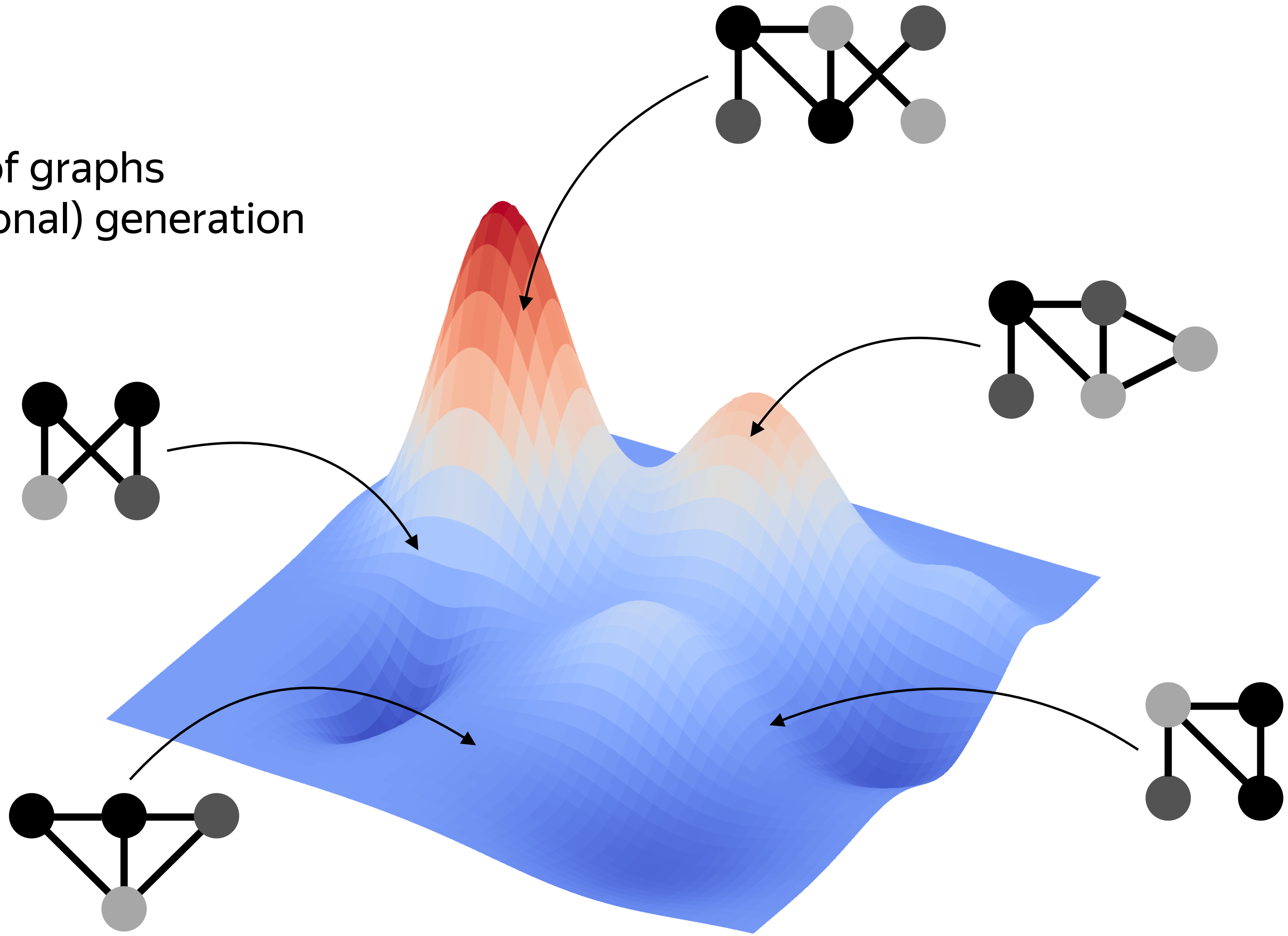
Potentially different types of nodes

Node clustering



Graph generation

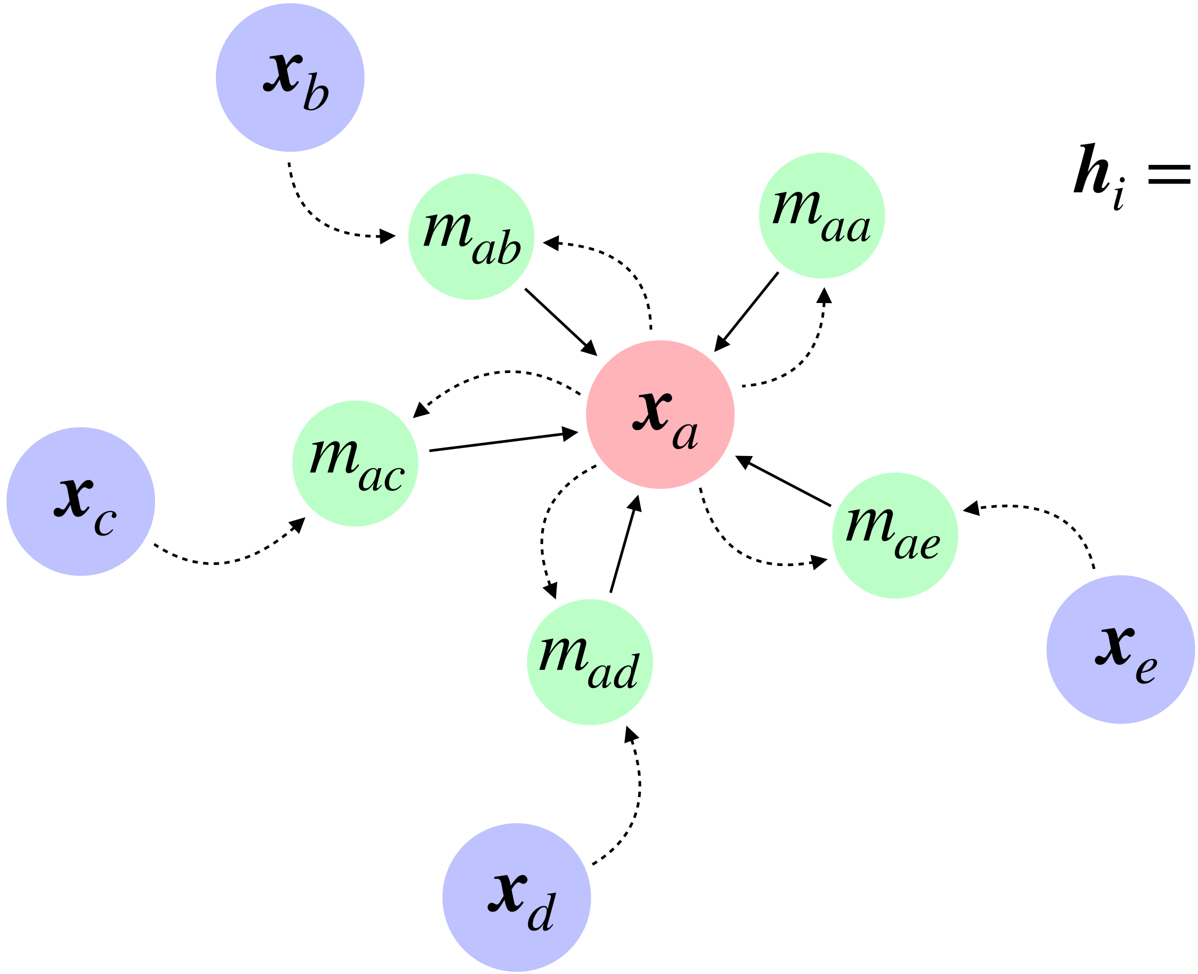
Learn to **model distribution** of graphs
and make (potentially conditional) generation



Basic terminology & notation

- $G = (V, E)$ — graph
- $N_i = \left\{ j : (i, j) \in E \right\}$ — neighbours of node $i \in V$
- $d_i = |N_i|$ — degree of node $i \in V$
- $\boldsymbol{A} \in \{0, 1\}^{|V|}$ — adjacency matrix with $a_{ij} = \left[(i, j) \in E \right]$
- $\boldsymbol{D} = \mathrm{diag}(d_i)$ — diagonal matrix of node degrees
- $\boldsymbol{x}_i \in \mathbb{R}^f$ — features of node $i \in V$

Message Passing Network



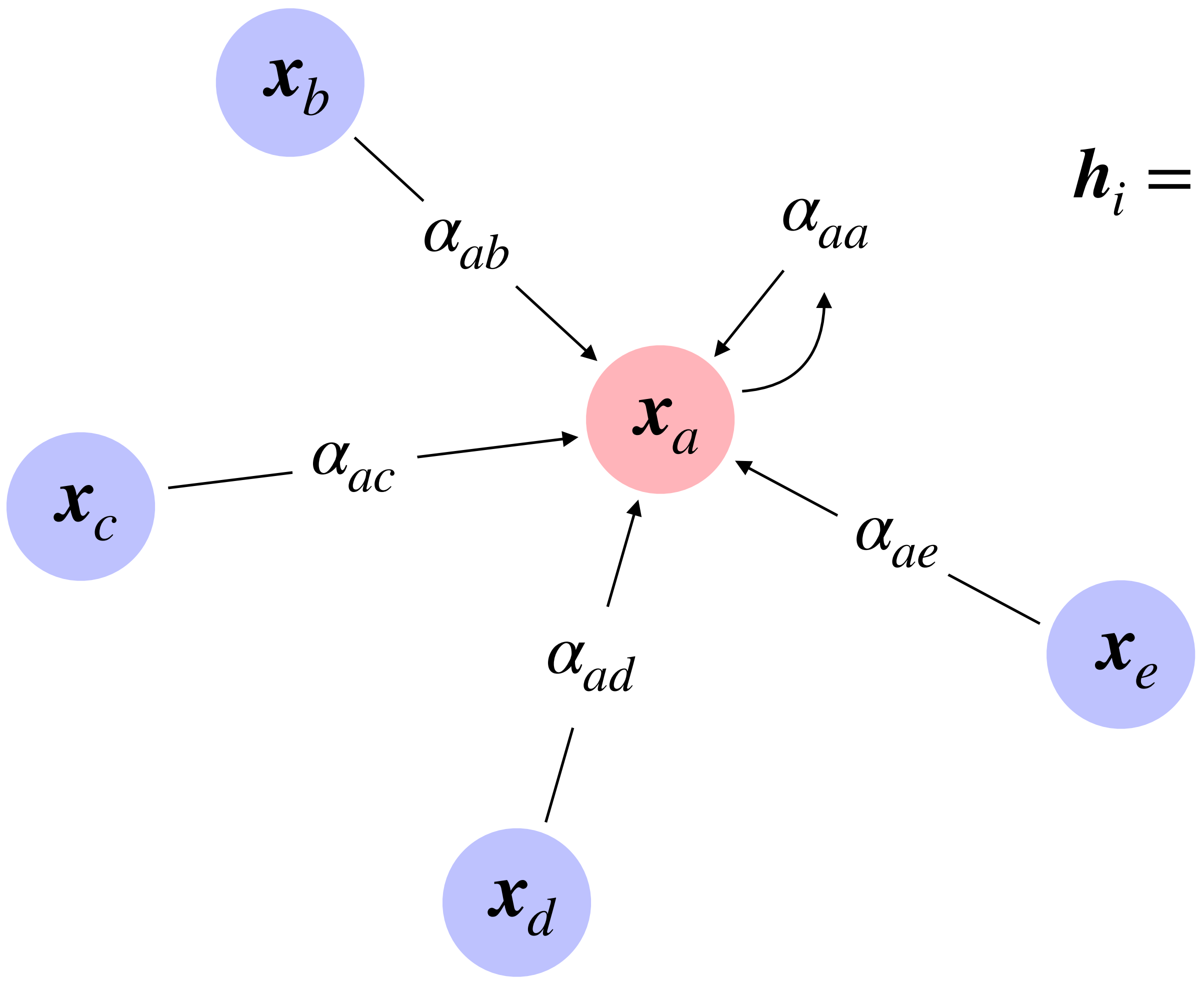
$$\boldsymbol{h}_i = \phi\left(\boldsymbol{x}_i, \bigoplus_{j \in N_i} m_{ij}\right), \text{ where } m_{ij} = \psi(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

$\bigoplus$ — invariant aggregation function

ϕ and ψ — learnable functions

J. Gilmer et al. "Neural Message Passing for Quantum Chemistry", ICML 2017

Graph Convolutional Network



$$\boldsymbol{h}_i = \sigma\left(\sum_{j \in N_i} \alpha_{ij} \boldsymbol{x}_j \boldsymbol{W}\right), \text{ where } \alpha_{ij} = \frac{1}{\sqrt{d_i d_j}}$$

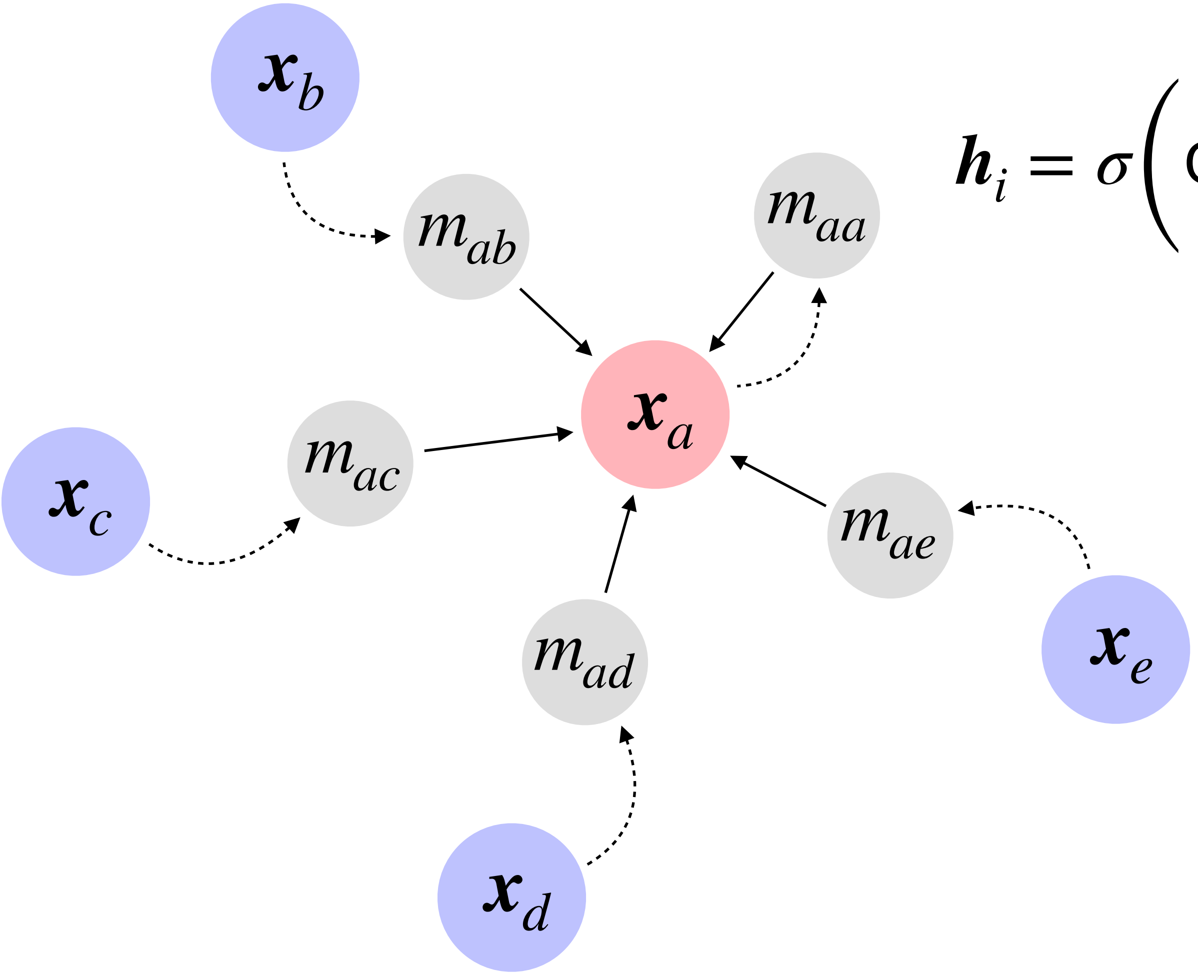
$$\boldsymbol{H} = \sigma\left(\boldsymbol{D}^{-1/2} \boldsymbol{A} \boldsymbol{D}^{-1/2} \boldsymbol{X} \boldsymbol{W}\right)$$

σ — some activation function

$\boldsymbol{W}$ — learnable parameters

Thomas N. Kipf et al. "Semi-Supervised Classification with Graph Convolutional Networks", ICLR 2017

GraphSAGE



$$\boldsymbol{h}_i = \sigma\left(\text{Concat}\left(\boldsymbol{x}_i, \sum_{j \in N_i} m_{ij}\right) \boldsymbol{W} \right), \text{ where } m_{ij} = \frac{1}{d_i}\boldsymbol{x}_j$$

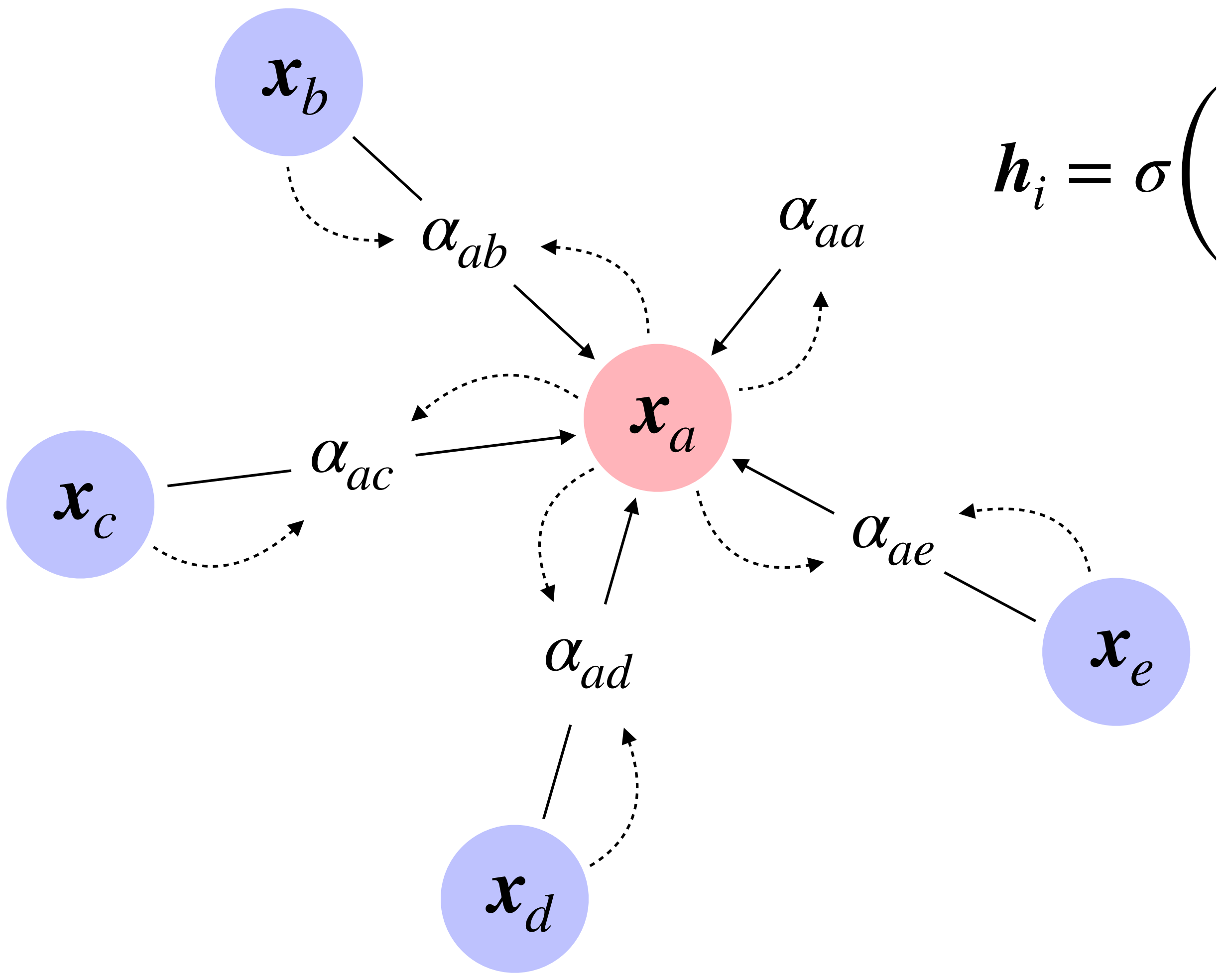
$$\boldsymbol{H} = \sigma\left(\text{Concat}(\boldsymbol{X}, \boldsymbol{D}^{-1}\boldsymbol{A}\boldsymbol{X})\boldsymbol{W}\right)$$

σ — some activation function

$\boldsymbol{W}$ — learnable parameters

William L. Hamilton et al. "Inductive Representation Learning on Large Graphs", NeurIPS 2017

Graph Attention Network v2



$$\boldsymbol{h}_i = \sigma\left(\sum_{j \in N_i} \alpha_{ij} \boldsymbol{x}_j \boldsymbol{W}\right), \text{ where } \alpha_{ij} = \operatorname{softmax}_j(m_{ij})$$

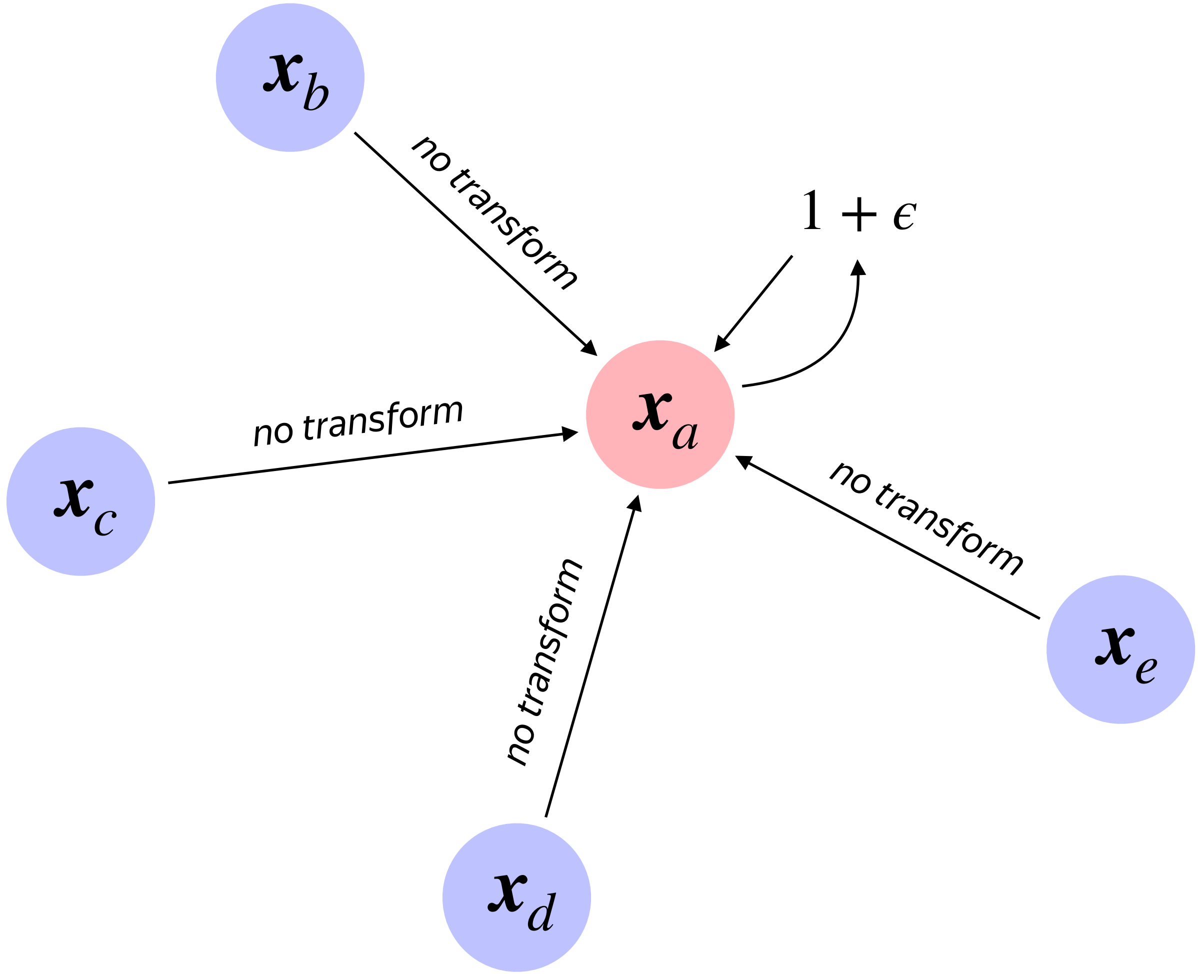
$$\text{and } m_{ij} = \text{LeakyReLU}\big(\boldsymbol{x}_i \boldsymbol{Q} + \boldsymbol{x}_j \boldsymbol{K}\big)\boldsymbol{a}$$

σ — some activation function

$\boldsymbol{W}$, $\boldsymbol{Q}$, $\boldsymbol{K}$ and $\boldsymbol{a}$ — learnable parameters

S. Brody et al. "How Attentive are Graph Attention Networks?", ICLR 2022

Graph Isomorphism Network



$$\boldsymbol{h}_i = \phi\left((1+\epsilon)\,\boldsymbol{x}_i + \sum_{j \in N_i} \boldsymbol{x}_j\right)$$

ϕ — learnable transformation

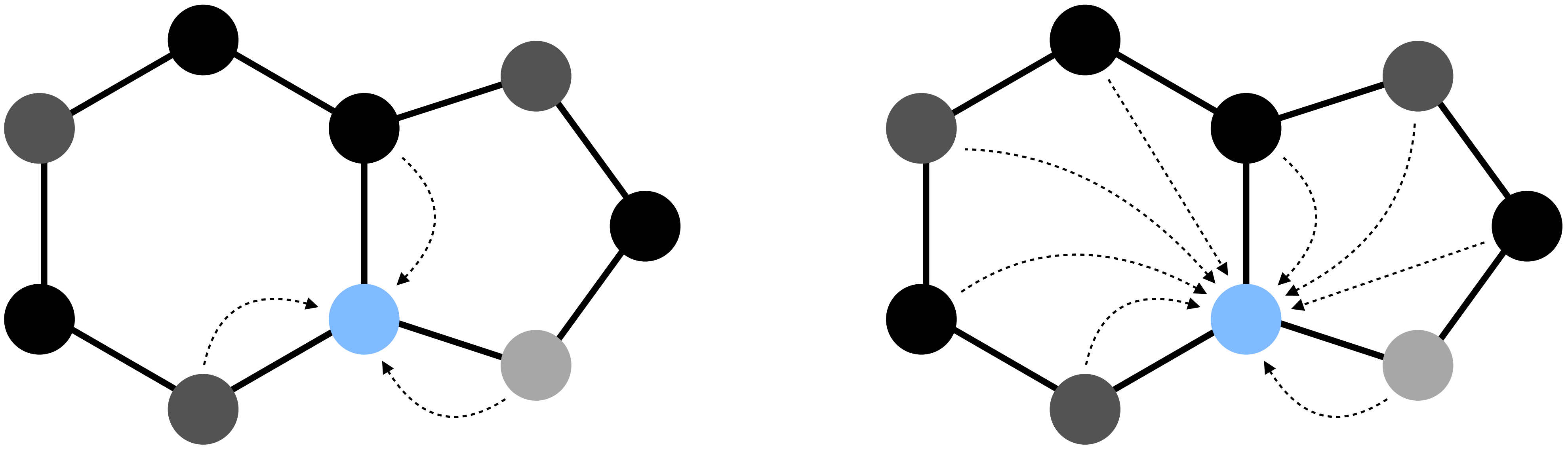
This neural architecture can be as powerful as WL test for graph isomorphism

K. Xu et al. "How Powerful are Graph Neural Networks?", ICLR 2019

There are also Graph Transformers...

Local Message Passing

Global Attention Mechanism



information flow in a single layer of GNN

Trending & challenging setups

- Heterogeneous graphs (e.g., question answering in knowledge graphs)
- Dynamic graphs (e.g., recommendations in temporal graphs)
- Time series forecasting with graphs (e.g., forecasting road traffic, weather, etc.)
- Learning on large-scale graphs (e.g., working on >1B nodes)
- Graph Foundation Models

Bonus: efficiency Challenges

- Graph convolutions are sparse operations
- Modern GPUs are built for dense operations

  => sparse operations are computed MUCH slower
- How can we utilize the power of GPUs to accelerate GNNs?