Representation Learning for Networks in Biology and Medicine: Part 1

https://arxiv.org/pdf/2104.04883.pdf

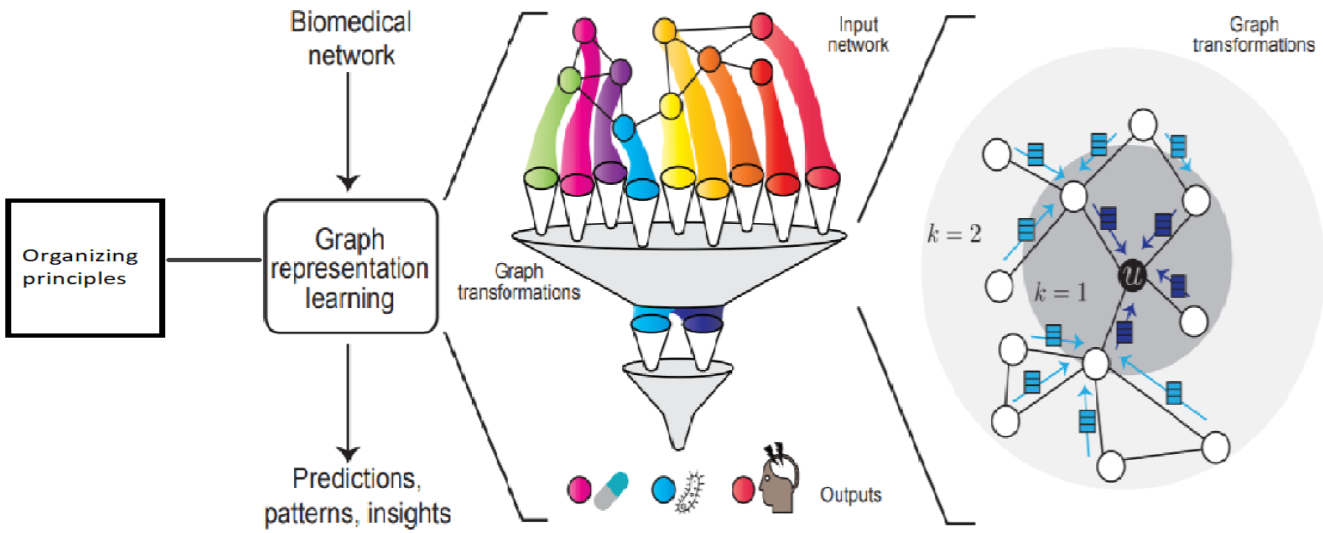
https://github.com/ankith-mohan/GRL_biomedicine_review

19 May 2021

Outline

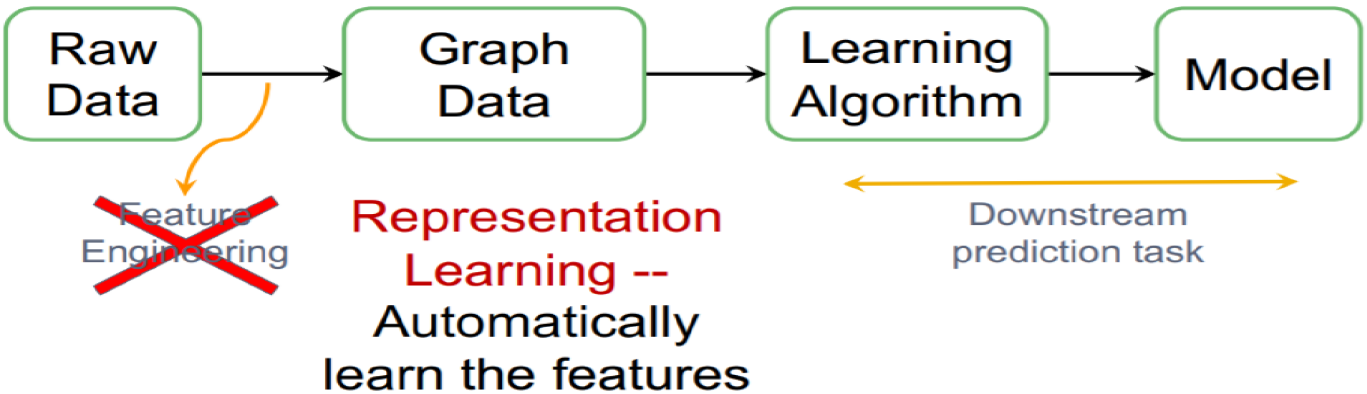
1. Introduction
2. Graph machine learning tasks
3. Graph machine learning paradigms
4. Diffusion state distance
5. GraphWave

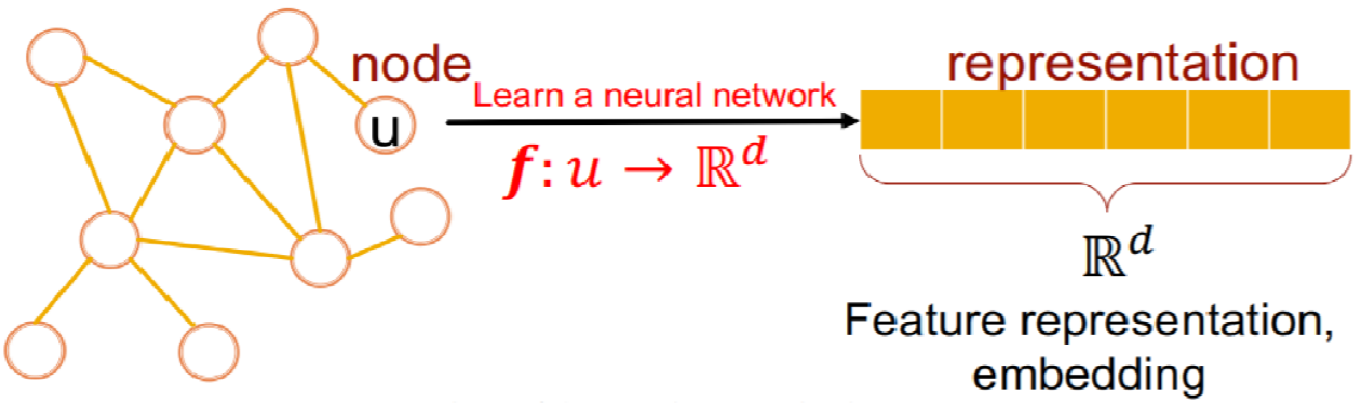
Hypothesis



- Network biology and medicine have identified a series of governing principles that govern biomedical networks.
- These principles can explain successes and limitations of graph representation learning (GRL).
- GRL can realize these organizing principles.

Graph representation learning



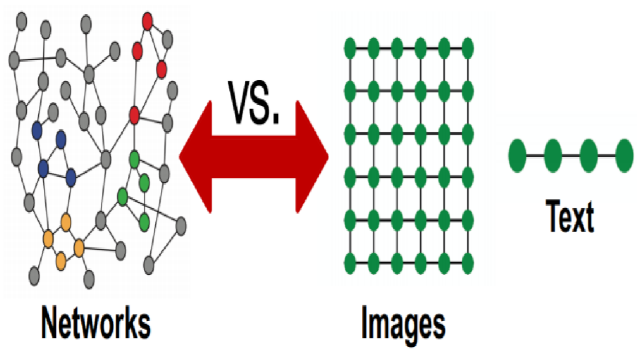


What's the big deal about graph representation learning?

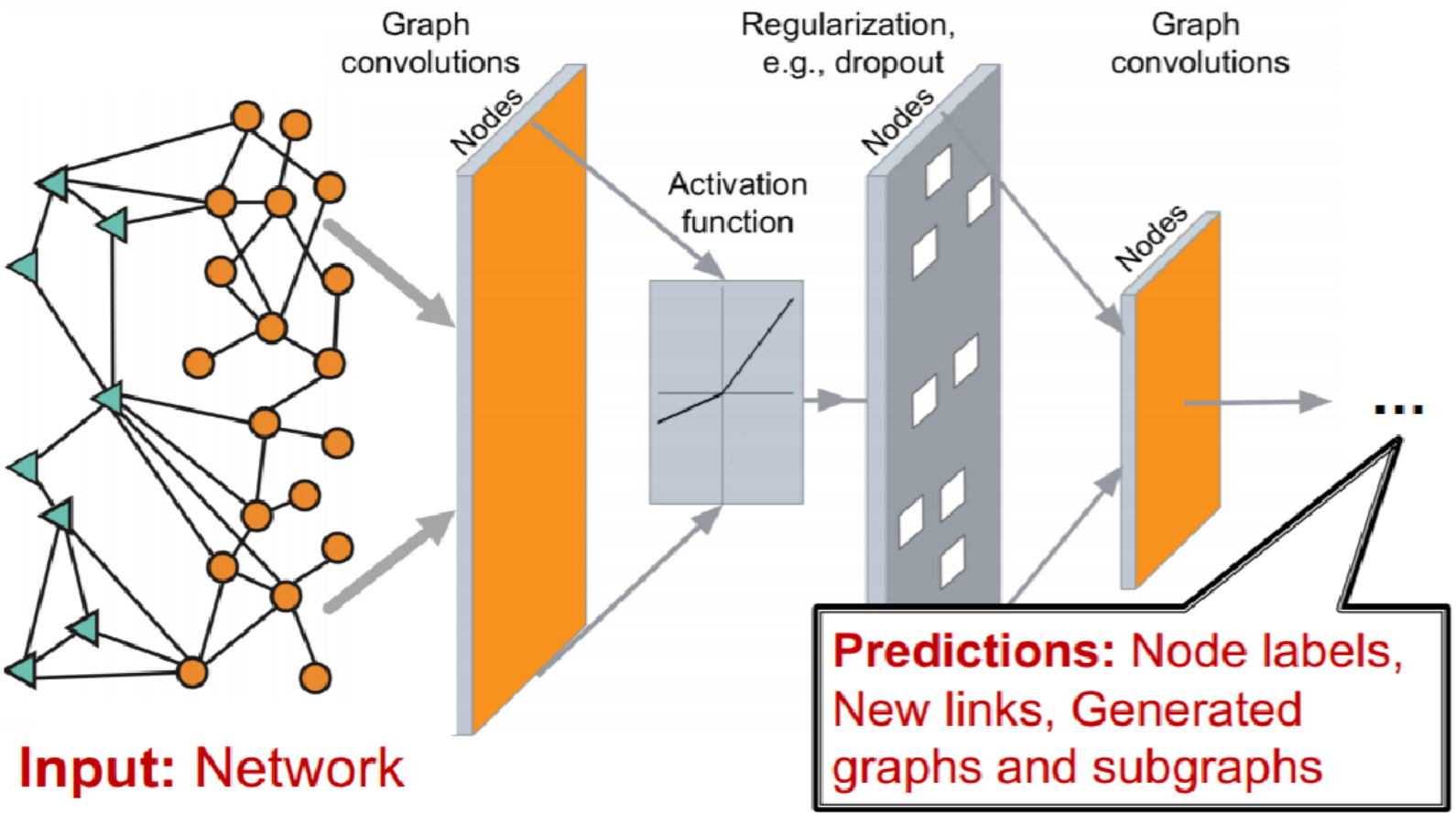
Modern ML toolbox is designed for grids (images) or sequences (speech/language).
Networks are complex:

- Arbitrary size and complex topological structure
- No spatial locality
- No fixed node ordering or reference point
- Often dynamic and have multi-modal features

Refer: `https://bit.ly/3hBbHwD`



Graph machine learning



Outline

1 Introduction

2 Graph machine learning tasks

3 Graph machine learning paradigms

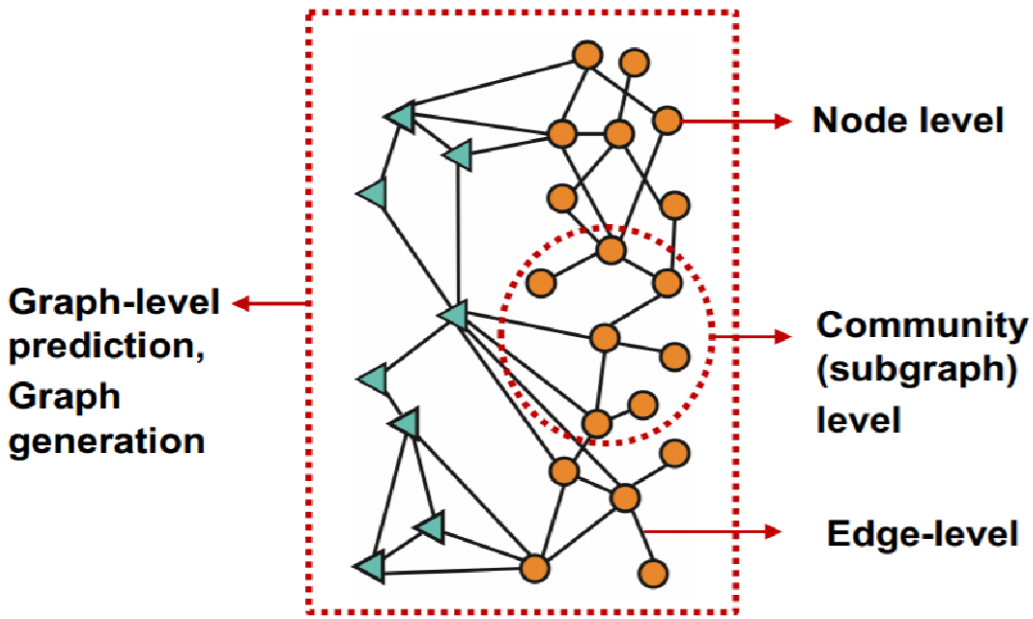
4 Diffusion state distance

5 GraphWave

Canonical graph prediction

Goal: Predict a label in the graph

- *Node classification*: Learn $f : \mathcal{V} \to Y_{\mathcal{V}}$
- *Edge classification*: Learn $f : (u, v) \to Y_{(u,v)}$
- *Subgraph classification*: Learn $f : \mathcal{S} \to Y_{\mathcal{S}}$
- *Graph classification*: Learn $f : \mathcal{G} \to Y_{\mathcal{G}}$



Other graph prediction tasks

- *Link prediction*: Learn $f : (\mathcal{E} \cup \mathcal{E}^c \rightarrow \{0, 1\})$
- *Module detection*: Detect a subgraph S that contributes to a variable
- *Clustering*: Partition the graph into subgraphs such that the intraconnectivity in each partition is greater than the interconnectivity between partitions.
- *Dynamic graph prediction*: Equivalent of the graph classification for dynamic graphs.

Latent graph learning

- **Goal**: Learn $f : \mathcal{V}, X \rightarrow \mathcal{E}$
- Concise latent low-dimensional representation of the graph

Graph generation

Obj: Learn to generate never-before-seen graph with desired properties.
Formally,

- *Training*: Given a set of training graphs $\mathcal{G}$ with certain shared characteristics, learn a function $f : \mathcal{G} \rightarrow \mathcal{D}_{\mathcal{G}}$, where $\mathcal{D}_{\mathcal{G}}$ is the underlying distribution of the training set.
- *Inference*: Use the learned distribution to generate a new graph G', which has the same characteristics as or optimized properties compared to $\mathcal{G}$.

Outline

1 Introduction

2 Graph machine learning tasks

3 Graph machine learning paradigms

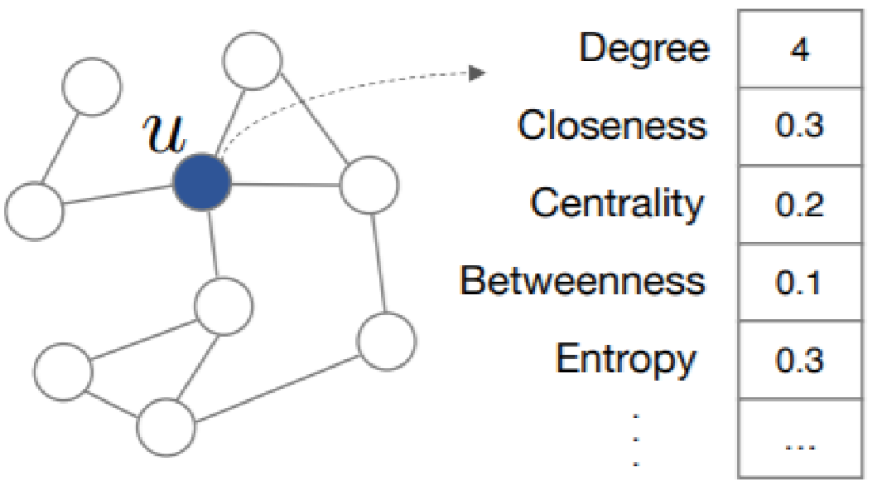
4 Diffusion state distance

5 GraphWave

Notations and definitions

Cheat sheet:
https://github.com/ankith-mohan/GRL_biomedicine_review/blob/main/notations_definitions.pdf

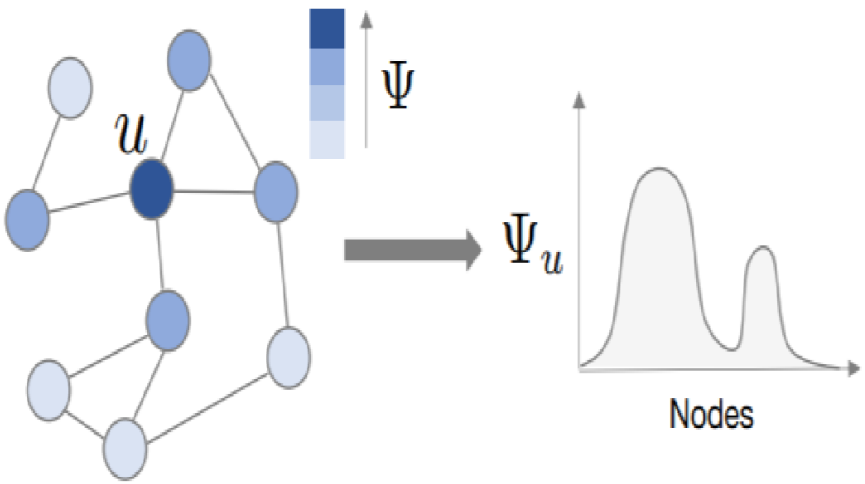
Graph theoretic techniques



Refer lectures:

1. Node-level tasks: `https://bit.ly/3bv5SNt`
2. Edge-level tasks: `https://bit.ly/3hJ7VS3`
3. Graph-level tasks: `https://bit.ly/3hvCBWt`

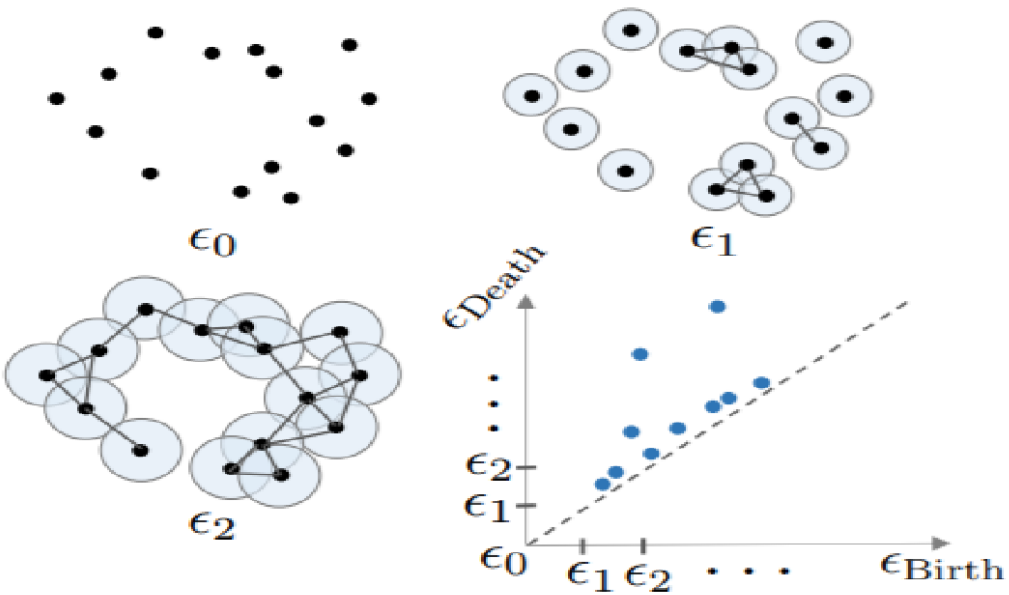
Random walks and diffusion



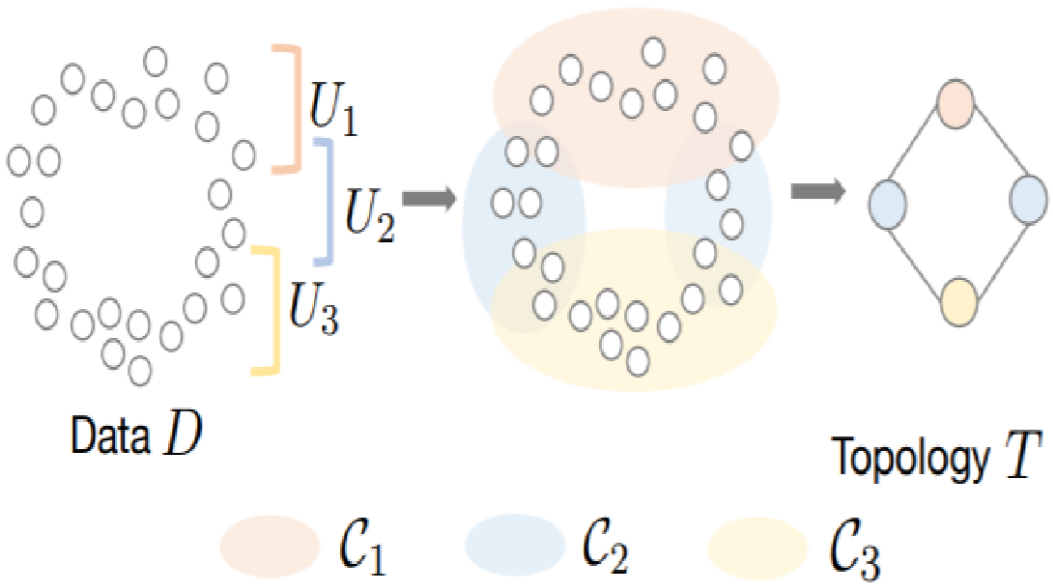
Idea: Nodes influence one another through paths. Diffusion ψ measures this influence

- **PageRank** (`https://bit.ly/3ePMQmL`)
- **Diffusion state distance** (Section 4)
- **GraphWave** (Section 5)

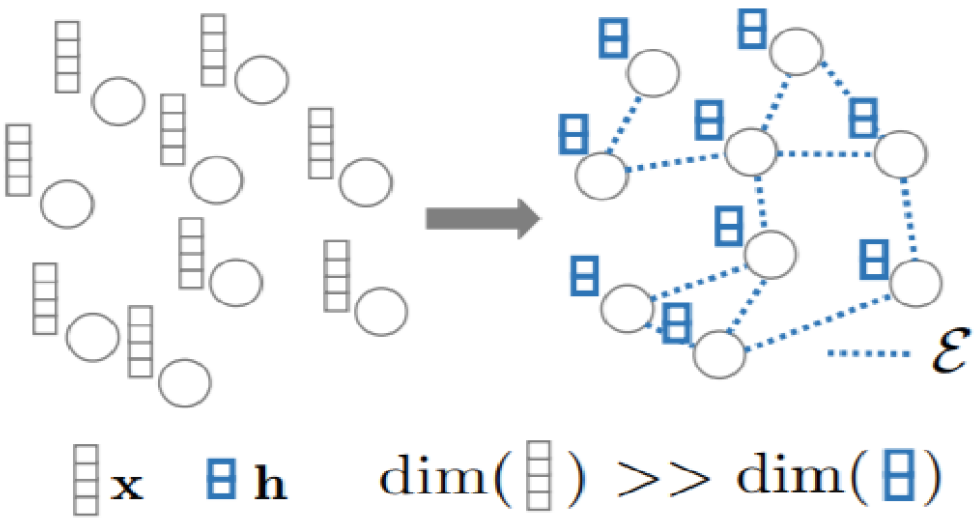
Persistent homology



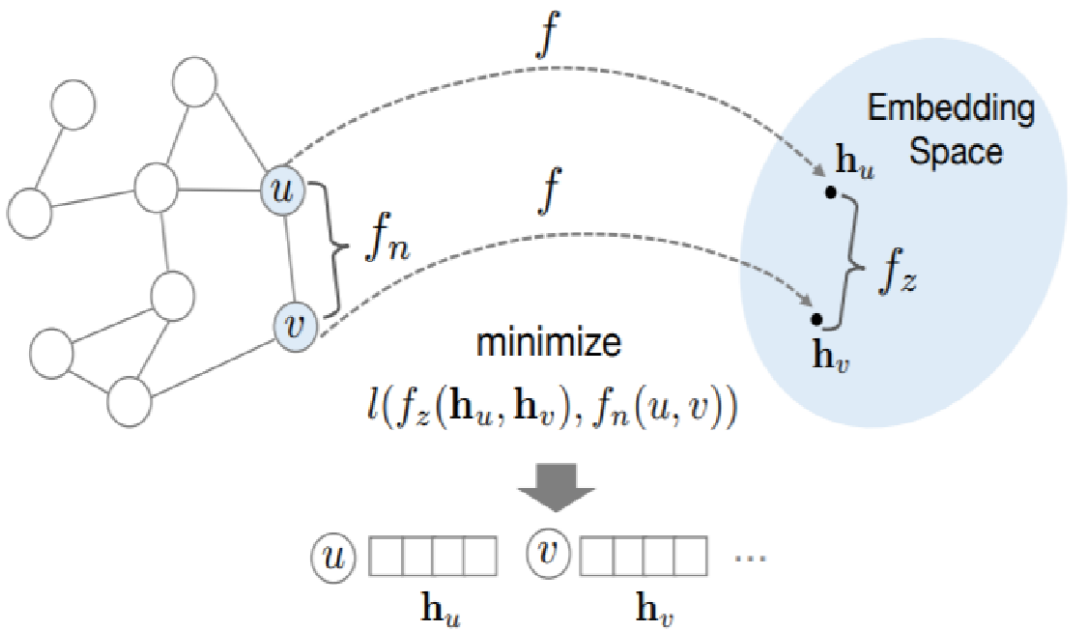
Geometrical and topological representations



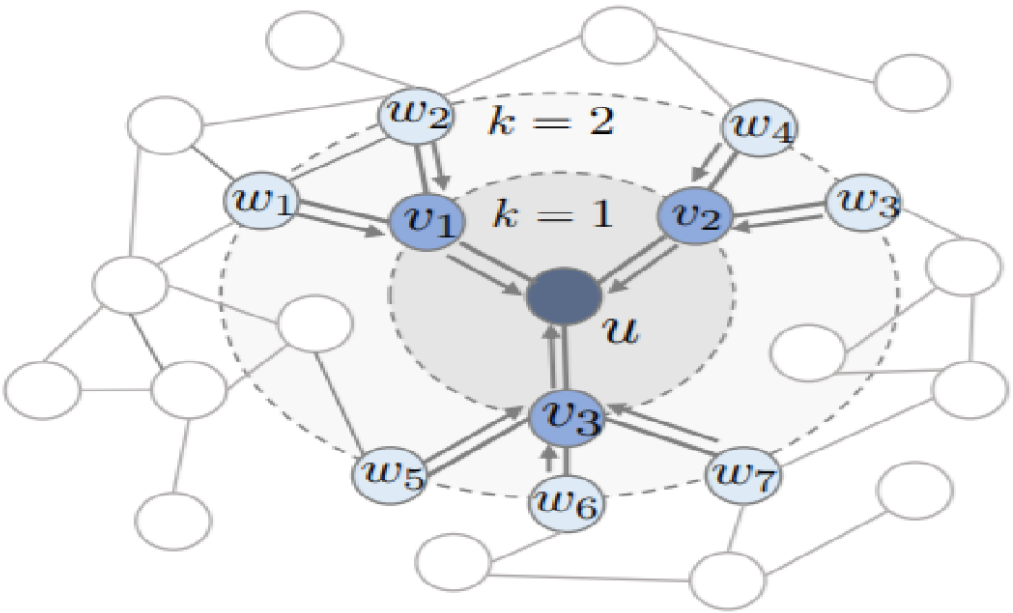
Manifold learning



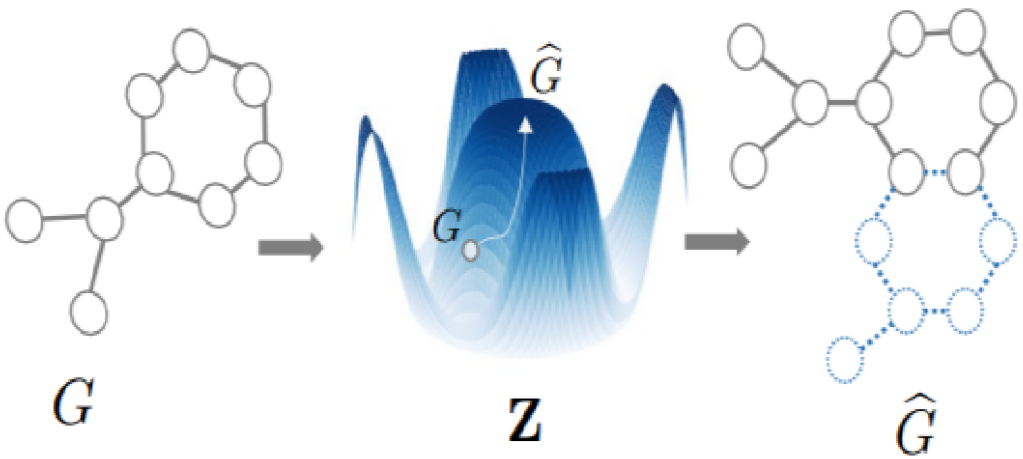
Shallow network embeddings



Graph neural networks



Graph generative models



Outline

1 Introduction

2 Graph machine learning tasks

3 Graph machine learning paradigms

4 Diffusion state distance

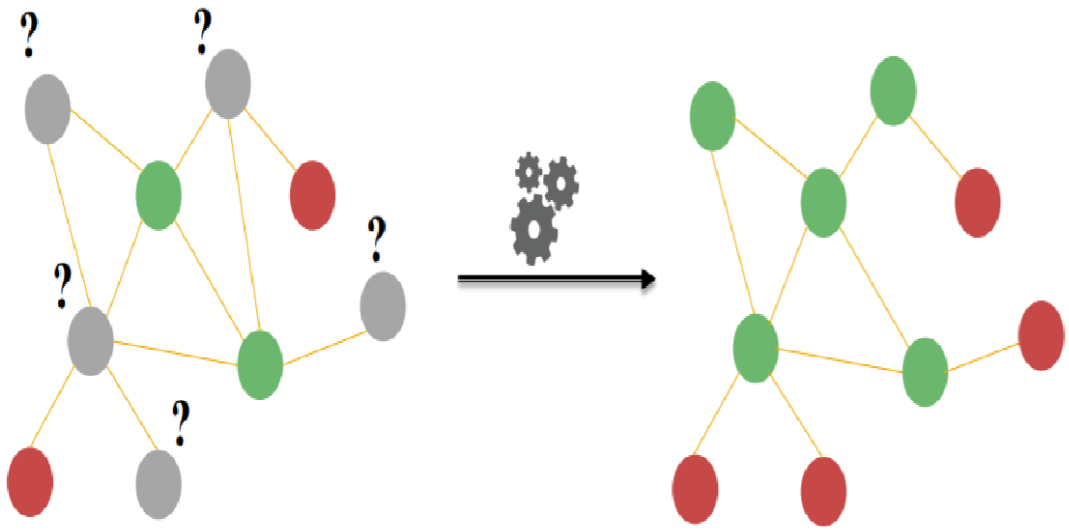
5 GraphWave

Problem statement

Given:

- Network of proteins
- Functions of some of the proteins
- Protein-protein interactions

Task: Predict the function of the unlabeled proteins



Approach

Idea

Functional similarity $\propto$ Proximity in network

Q. How to quantify proximity?

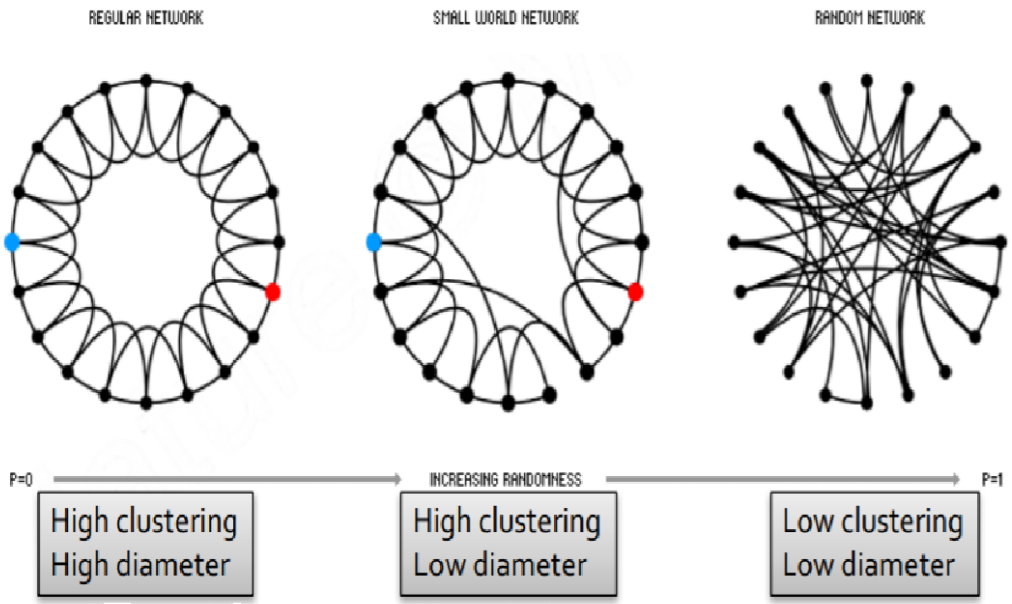
Shortest-path distance

Note: Focus on methods that rely solely on network structure.

Limitations of shortest-path distance in PPI networks

1. Does not rely on the degree of neighborhood overlap.
2. PPI networks are *small world* networks.

Small world networks

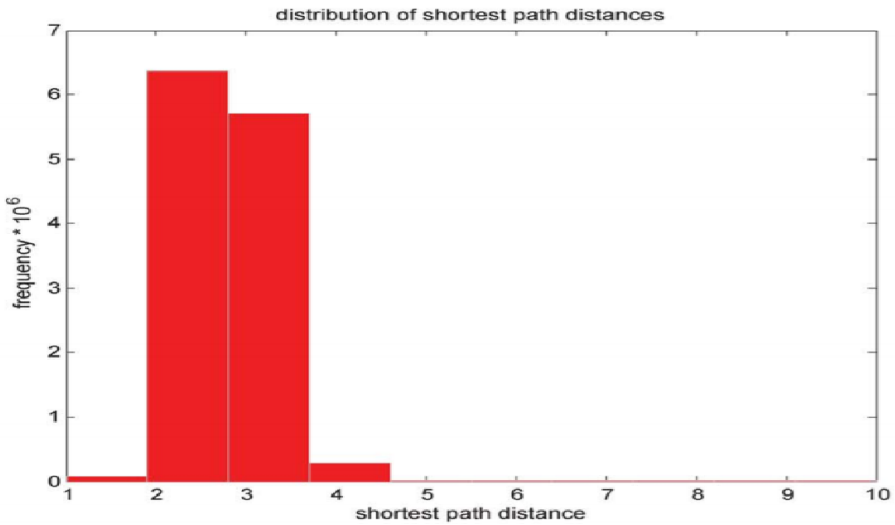


$\Longrightarrow$ Large fraction of nodes is proximate to any typical node

Refer: `https://stanford.io/3osc8dS`

Small world phenomena exhibited by PPI network

Yeast PPI network



Majority of the nodes are 2 or 3 hops away from any node.
$\implies$ $\mathcal{N}^2(v)$ will contain about half the network.

Alternate approach

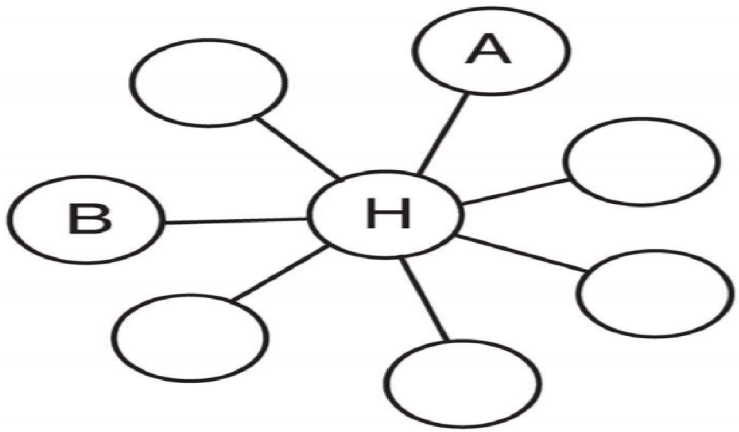
Clearly, shortest-path distance is an unreliable measure of proximity in PPI networks.

Q. Why do PPI networks have so many short paths?

Due to the presence of hubs

Hub

Hubs are high degree nodes.



1. B and H are one hop apart but H is a hub while B is not.
2. A and B are two hops apart but are not necessarily functionally related, since they are connected through H.

Note: Refer section "Motivation for DSD" for biological reason behind the presence of hubs in PPI networks.

Accessibility

A random walk starting from A is not likely to reach B quickly.
If the walk is of length 3, then we will not reach B from A at all.

Idea (revisited)

Functional similarity $\propto$ Accessibility in network

Nodes that co-occur more frequently in random walks are more functionally similar.
Same idea that inspired random walk-based shallow node embedding approaches such as DeepWalk, node2vec, etc.

Pairwise measure of node similarity

$He^{(k)}(A, B)$: Expected number of times that a random walk starting from A and proceeding k steps, will visit B.

Under this metric, A and B are relatively far because the influence of hub H decreases the likelihood of a random walk from A landing at B.
However:

- $He^{(k)}(H, B)$ will still be high.
- $He^{(k)}()$ is not symmetric $\implies$ its not a metric.

$He^{(k)}()$ is a pairwise measure of node similarity.
We require a global measure of similarity.

Diffusion State Distance

Assume k is fixed.
Let $V = [v_1, v_2, \ldots, v_n]$ be the vertex set.
When there is no ambiguity, denote $He^{(k)}(u, v)$ by $He(u, v)$, $\forall u, v \in V$.

$$He(v) = [He(v, v_1), He(v, v_2), \ldots, He(v, v_n)] \in \mathbb{R}^n, \forall v \in V$$

DSD

$$DSD(u, v) = ||He(u) - He(v)||_1$$

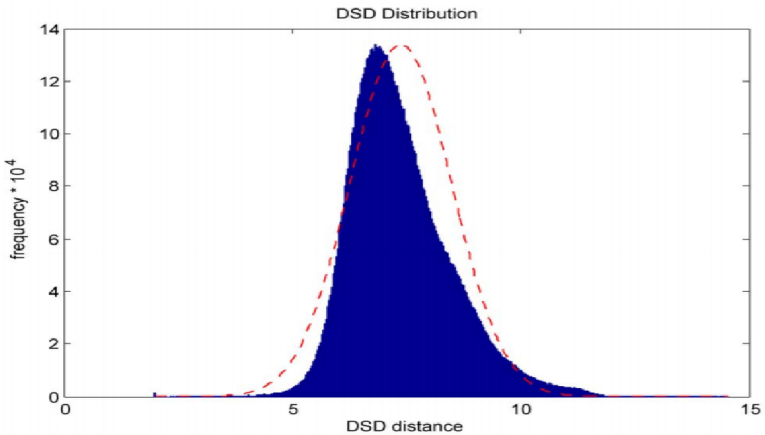
Properties of DSD

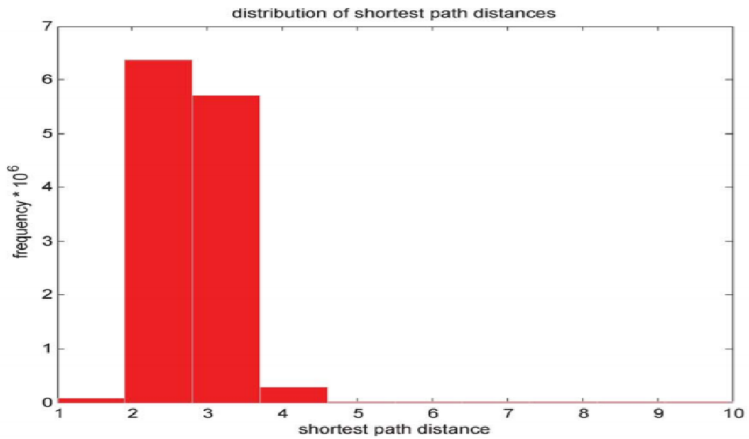
- DSD is a metric for any fixed value of k.
- DSD converges as k goes to infinity.

Note: Refer section "Formal Properties of DSD" for proofs.

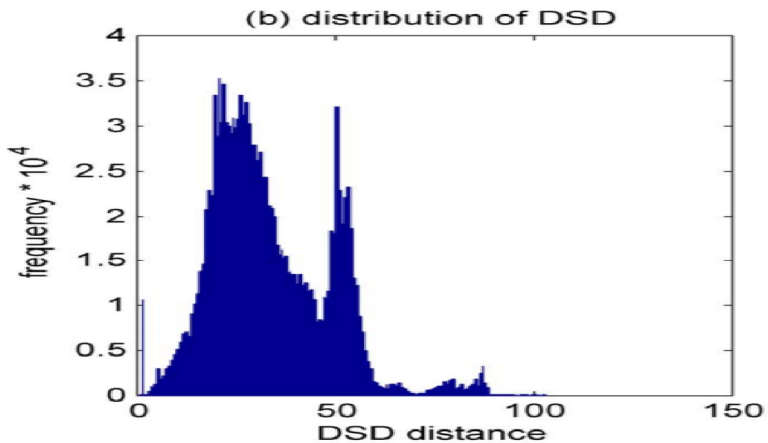
Characteristics of DSD

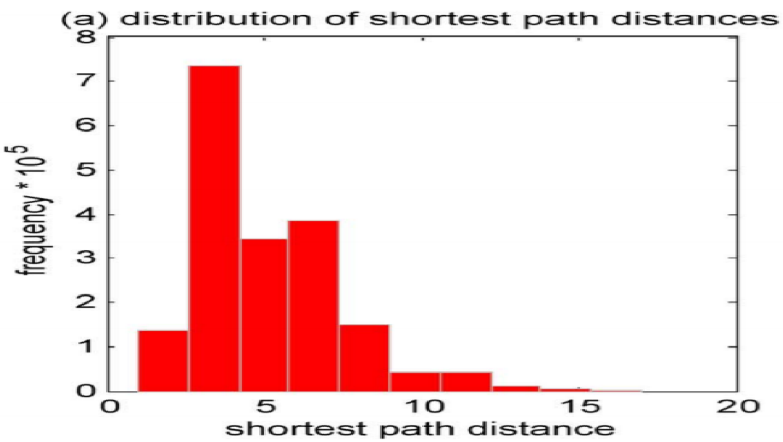
S. cerevisiae





S. pombe





Material

- Paper: `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0076339`
- Tool: `http://dsd.cs.tufts.edu/server/`

Outline

1 Introduction

2 Graph machine learning tasks

3 Graph machine learning paradigms

4 Diffusion state distance

5 GraphWave

GraphWave

Next episode!