

The background of the slide is a light gray with an abstract geometric pattern. On the left side, there is a dense network of thin black lines connecting various black dots of different sizes, forming a complex web. Scattered across the right side and top are several thin, light gray outlines of triangles of various sizes and orientations. Some of these triangles are isolated, while others are part of small clusters or chains.

Edge Analysis

Antonio Longa^{1,2}

MobS¹ Lab, Fondazione Bruno Kessler, Trento, Italy
SML² Lab, University of Trento, Italy

**Problem
&
Motivations**

01

Link prediction

02

GAE for link prediction

03

TABLE OF CONTENTS

04

Label prediction

05


**Node2Vec for label
prediction**



01 Problem & Motivation

The input graph can have **missing** informations.





01 Problem & Motivation

The input graph can have **missing** informations

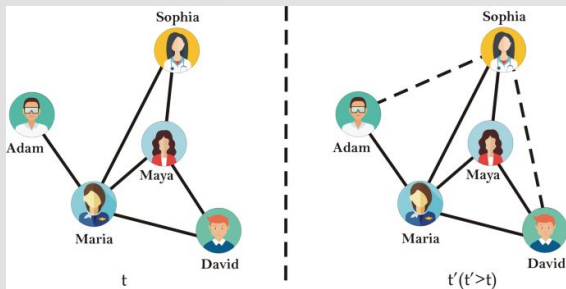
- Edges
- Labels
- Weights
- etc...



01 Problem & Motivation

The input graph can have **missing** informations

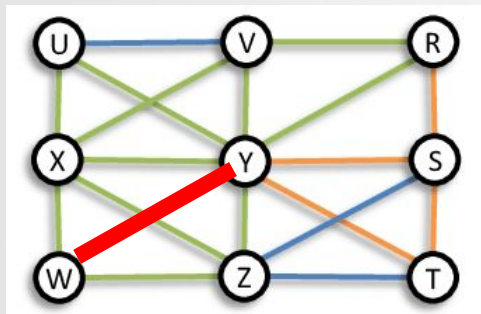
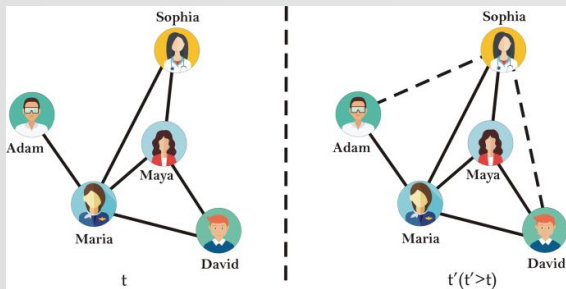
- **Edges**
- Labels
- Weights
- etc...



01 Problem & Motivation

The input graph can have **missing** informations

- Edges
- **Labels**
- Weights
- etc...

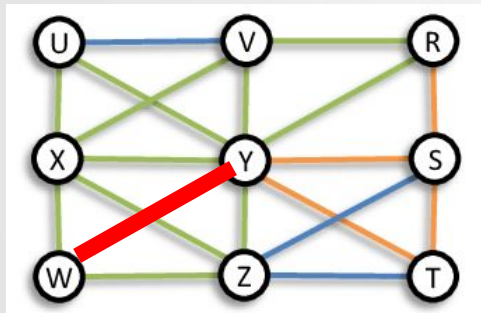
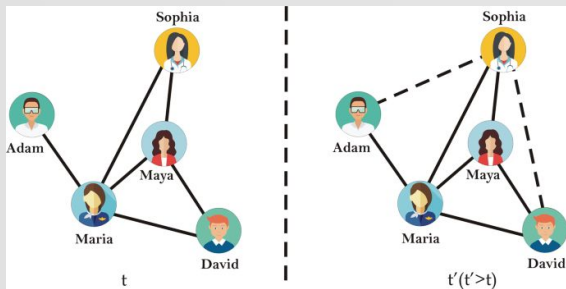


Witch color?

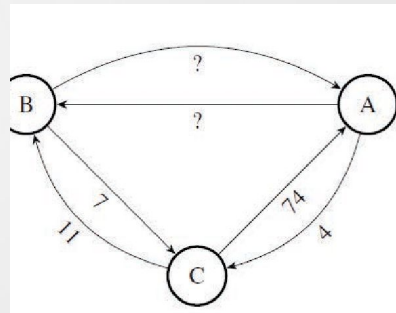
01 Problem & Motivation

The input graph can have **missing** informations

- Edges
- Labels
- **Weights**
- etc...



Witch color?





02 Link prediction

Given an input graph G , and two nodes u and v , predict if there is an edge between u and v .

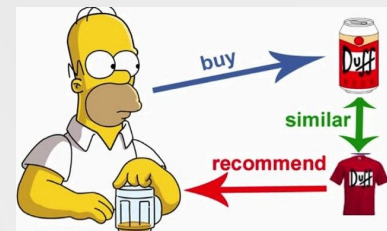


02 Link prediction

Given an input graph G , and two nodes u and v , predict if there is an edge between u and v .

WHY IT IS IMPORTANT?

- Recommendation systems

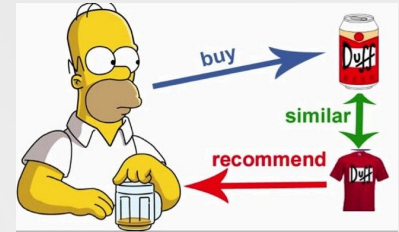


02 Link prediction

Given an input graph G , and two nodes u and v , predict if there is an edge between u and v .

WHY IT IS IMPORTANT?

- Recommendation systems
- Privacy control on social networks

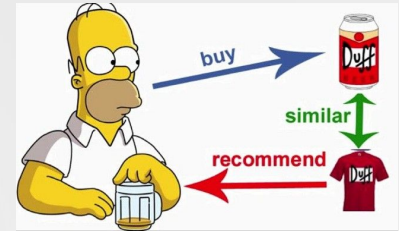


02 Link prediction

Given an input graph G , and two nodes u and v , predict if there is an edge between u and v .

WHY IT IS IMPORTANT?

- Recommendation systems
- Privacy control on social networks
- Influence detection

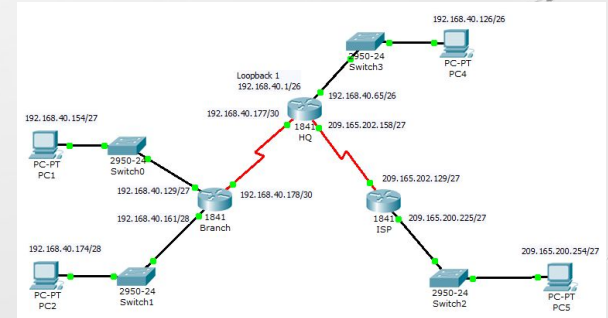
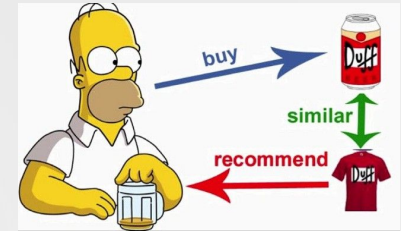


02 Link prediction

Given an input graph G , and two nodes u and v , predict if there is an edge between u and v .

WHY IT IS IMPORTANT?

- Recommendation systems
- Privacy control on social networks
- Influence detection
- Routing networks
- Etc ...

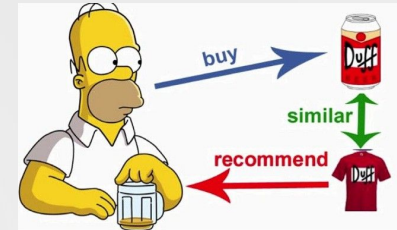


02 Link prediction

Given an input graph G , and two nodes u and v , predict if there is an edge between u and v .

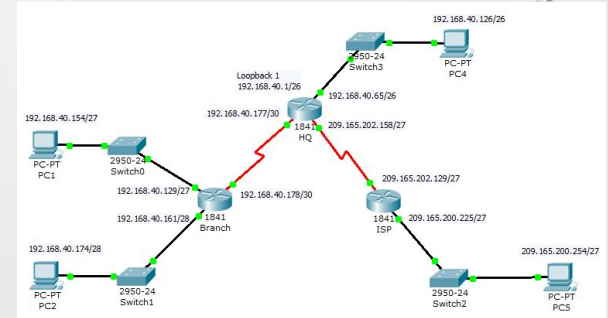
WHY IT IS IMPORTANT?

- Recommendation systems
- Privacy control on social networks
- Influence detection
- Routing networks
- Etc ...



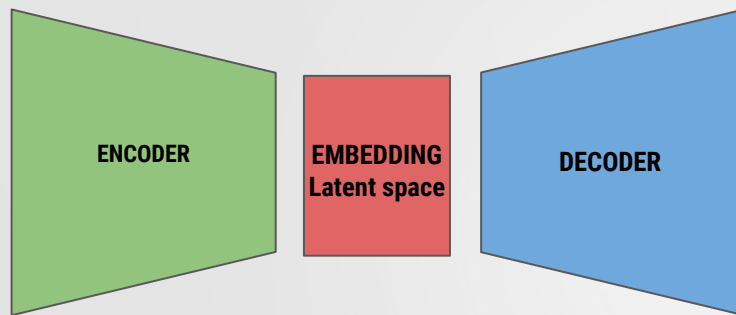
HOW TO SOLVE IT

- Topology based methods
- Node attribute based methods
- Based on embeddings
- Probabilistics relationship models
- Etc ...



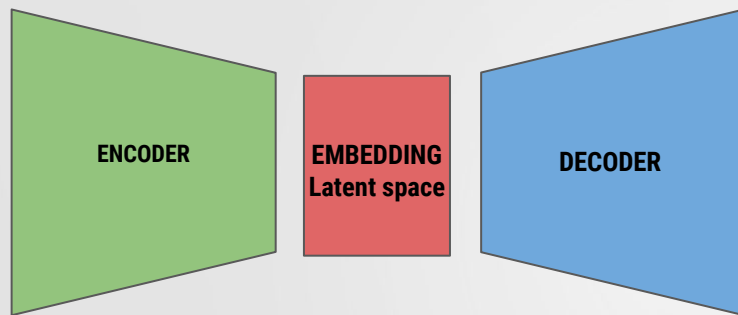
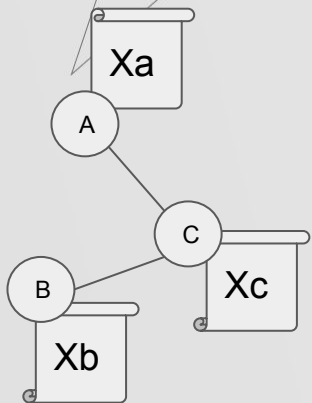
03 GAE for link prediction

Recap GAE



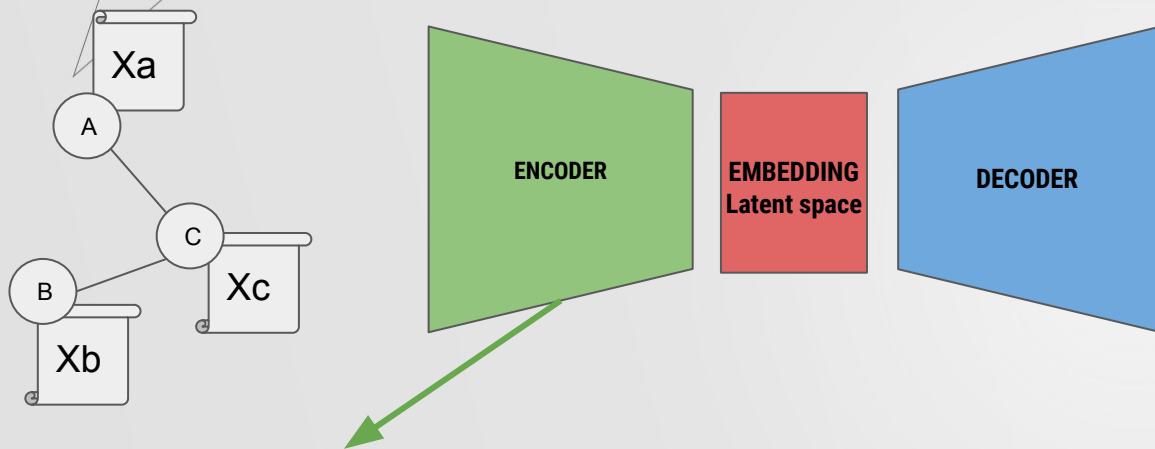
03 GAE for link prediction

Recap GAE

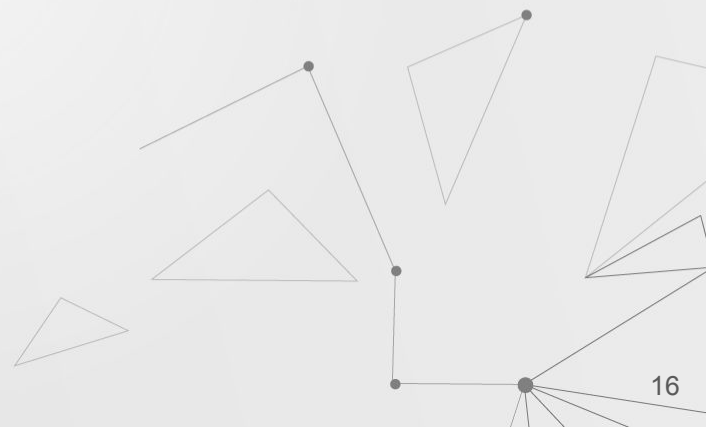


03 GAE for link prediction

Recap GAE

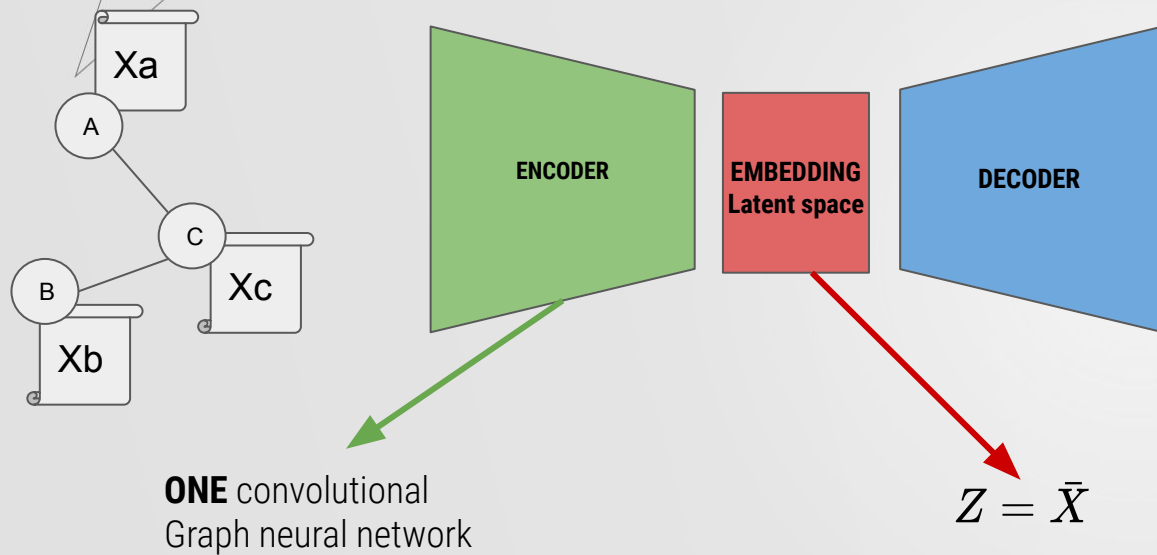


ONE convolutional
Graph neural network



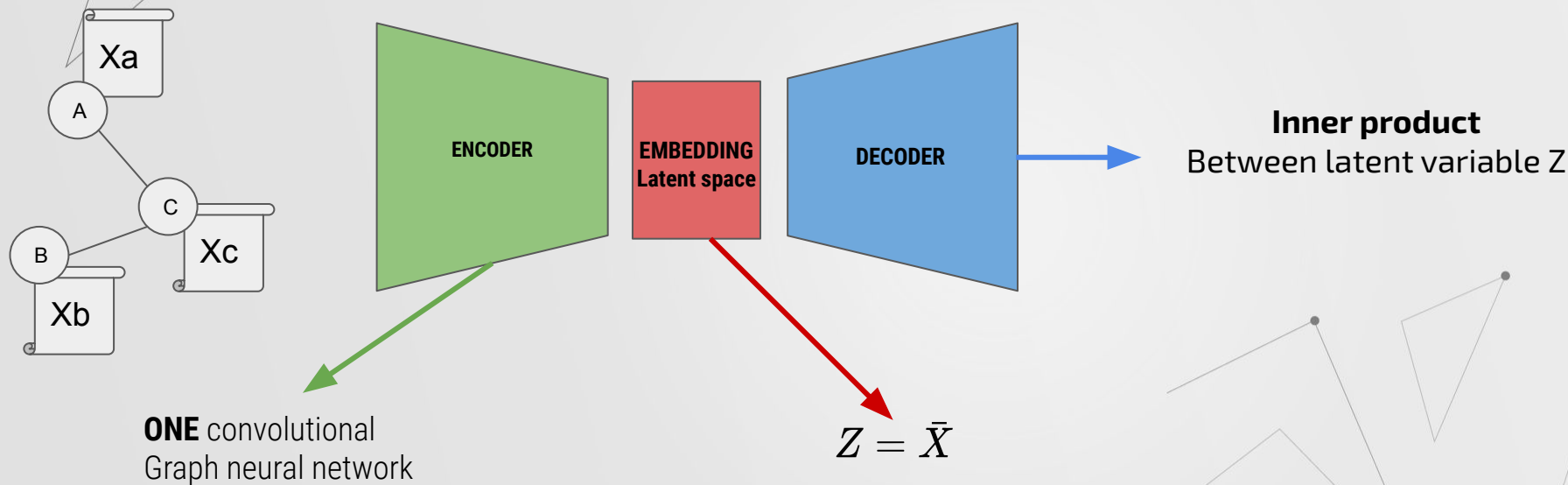
03 GAE for link prediction

Recap GAE



03 GAE for link prediction

Recap GAE

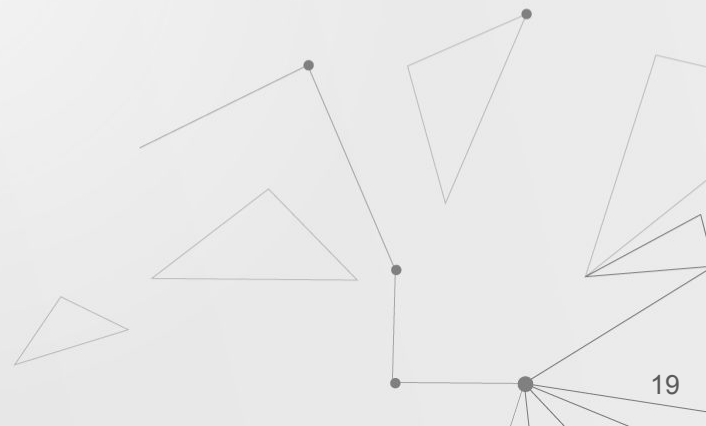




03 GAE for link prediction

But..

What is the difference between the GAE for node embedding?



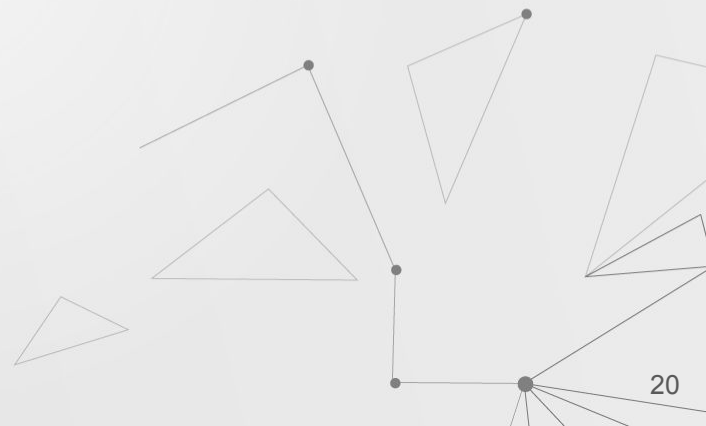


03 GAE for link prediction

But..

What is the difference between the GAE for node embedding?

The LOSS function



03 GAE for link prediction

GAE for node embedding

`recon_loss (z, pos_edge_index, neg_edge_index=None)` [\[source\]](#)

Given latent variables `z`, computes the binary cross entropy loss for positive edges `pos_edge_index` and negative sampled edges.

PARAMETERS:

- `z` (*Tensor*) – The latent space **Z**.
- `pos_edge_index` (*LongTensor*) – The positive edges to train against.
- `neg_edge_index` (*LongTensor, optional*) – The negative edges to train against. If not given, uses negative sampling to calculate negative edges. (default: `None`)

03 GAE for link prediction

GAE for node embedding

`recon_loss (z, pos_edge_index, neg_edge_index=None)` [\[source\]](#)

Given latent variables `z`, computes the **binary cross entropy loss** for positive edges `pos_edge_index` and negative sampled edges.

PARAMETERS:

- `z` (*Tensor*) – The latent space **Z**.
- `pos_edge_index` (*LongTensor*) – The positive edges to train against.
- `neg_edge_index` (*LongTensor, optional*) – The negative edges to train against. If not given, uses negative sampling to calculate negative edges. (default: `None`)

03 GAE for link prediction

GAE for node embedding

`recon_loss (z, pos_edge_index, neg_edge_index=None)` [\[source\]](#)

Given latent variables `z`, computes the **binary cross entropy loss** for positive edges `pos_edge_index` and negative sampled edges.

PARAMETERS:

- `z` (*Tensor*) – The latent space **Z**.
- `pos_edge_index` (*LongTensor*) – The positive edges to train against.
- `neg_edge_index` (*LongTensor, optional*) – The negative edges to train against. If not given, uses negative sampling to calculate negative edges. (default: `None`)

GAE for link prediction

```
link_labels = get_link_labels(data.train_pos_edge_index, neg_edge_index)
loss = F.binary_cross_entropy_with_logits(link_logits, link_labels)
loss.backward()
optimizer.step()
```

03 GAE for link prediction

GAE for node embedding

`recon_loss (z, pos_edge_index, neg_edge_index=None)` [\[source\]](#)

Given latent variables `z`, computes the **binary cross entropy loss** for positive edges `pos_edge_index` and negative sampled edges.

PARAMETERS:

- `z` (*Tensor*) – The latent space **Z**.
- `pos_edge_index` (*LongTensor*) – The positive edges to train against.
- `neg_edge_index` (*LongTensor, optional*) – The negative edges to train against. If not given, uses negative sampling to calculate negative edges. (default: `None`)

GAE for link prediction

```
link_labels = get_link_labels(data.train_pos_edge_index, neg_edge_index)
loss = F.binary_cross_entropy_with_logits(link_logits, link_labels)
loss.backward()
optimizer.step()
```



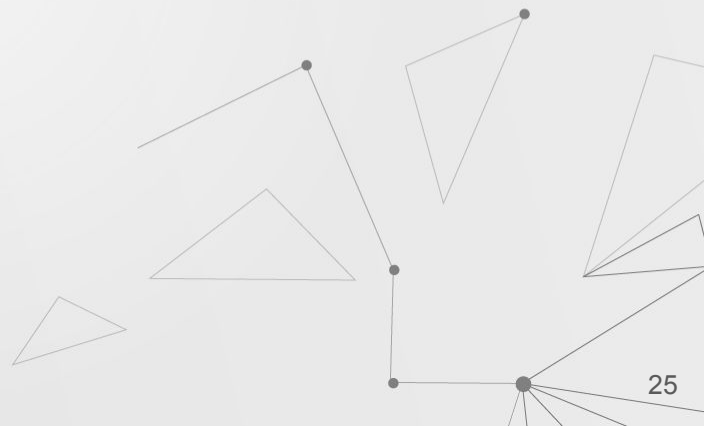

03 GAE for link prediction

GAE for node embedding

Binary cross entropy loss

GAE for link prediction

Binary cross entropy with logits loss





03 GAE for link prediction

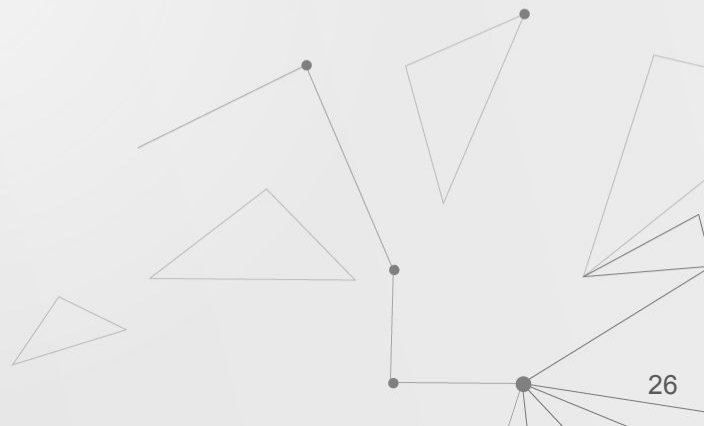
GAE for node embedding

Binary cross entropy loss

$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top$$

GAE for link prediction

Binary cross entropy with logits loss





03 GAE for link prediction

GAE for node embedding

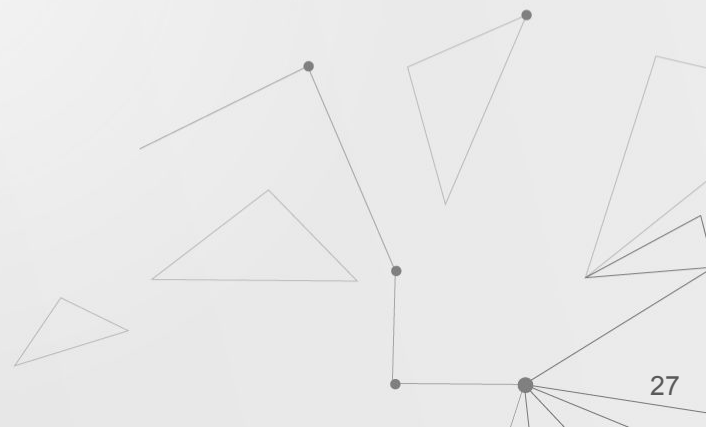
Binary cross entropy loss

$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top$$

$$l_n = -w_n [y_n \cdot \log x_n + (1 - y_n) \cdot \log(1 - x_n)]$$

GAE for link prediction

Binary cross entropy with logits loss





03 GAE for link prediction

GAE for node embedding

Binary cross entropy loss

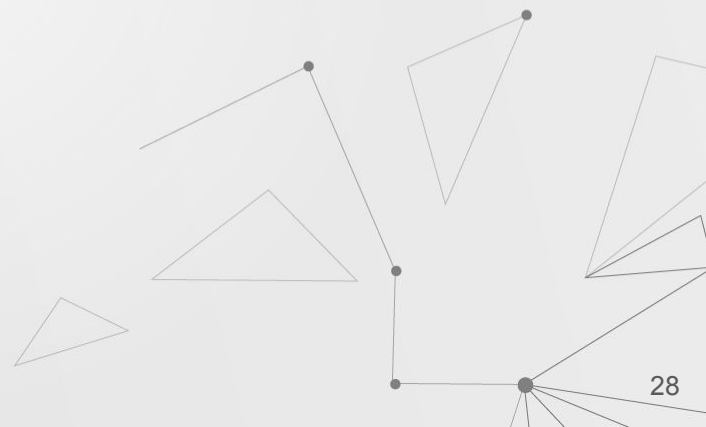
$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top$$

$$l_n = -w_n [y_n \cdot \log x_n + (1 - y_n) \cdot \log(1 - x_n)]$$

GAE for link prediction

Binary cross entropy with logits loss

$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top$$





03 GAE for link prediction

GAE for node embedding

Binary cross entropy loss

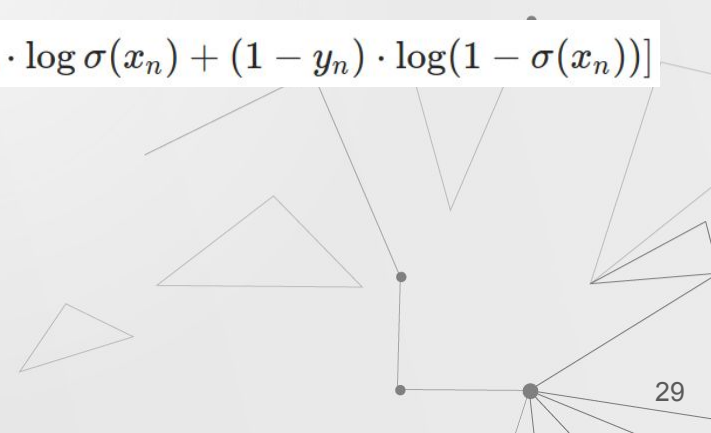
$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top$$

$$l_n = -w_n [y_n \cdot \log x_n + (1 - y_n) \cdot \log(1 - x_n)]$$

GAE for link prediction

Binary cross entropy with logits loss

$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top$$

$$l_n = -w_n [y_n \cdot \log \sigma(x_n) + (1 - y_n) \cdot \log(1 - \sigma(x_n))]$$




03 GAE for link prediction

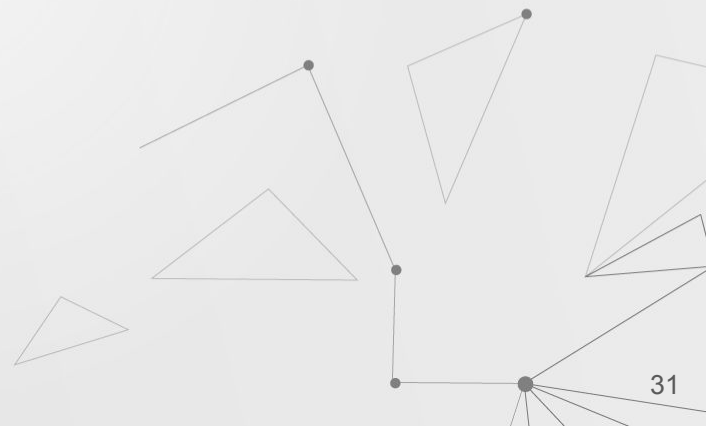
Jupyter - notebook





04 Label prediction

Given an input graph G , and two nodes u and v , predict the label of the edge between u and v .



04 Label prediction

Given an input graph G , and two nodes u and v , predict the label of the edge between u and v .

WHY IT IS IMPORTANT?

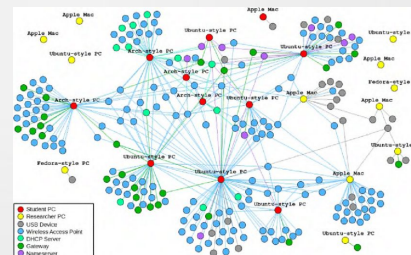
- Human mobility forecast



An abstract geometric pattern composed of numerous thin, dark gray lines that intersect to form a complex web of triangles and polygons. The lines vary in thickness and orientation, creating a sense of depth and movement. Several small, solid dark gray circles are placed at key intersection points, serving as focal points or nodes within the network. The overall composition is non-representational and emphasizes the relationships between the lines and the spaces they define.

WHY IT IS IMPORTANT?

- Human mobility forecast
- Type of relationship in social networks

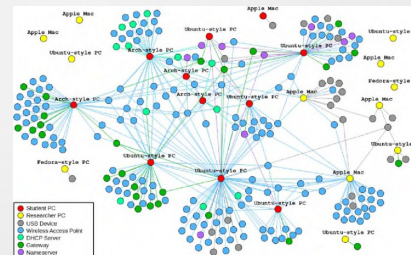


04 Label prediction

Given an input graph G , and two nodes u and v , predict the label of the edge between u and v .

WHY IT IS IMPORTANT?

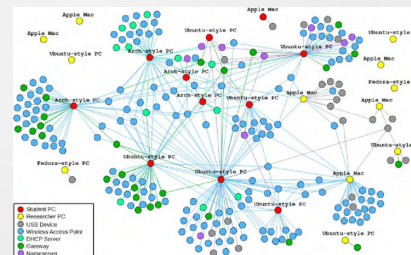
- Human mobility forecast
- Type of relationship in social networks
- Traffic congestion



An abstract geometric pattern composed of numerous thin, dark gray lines that intersect to form a complex web of triangles and polygons. The lines vary in thickness and orientation, creating a sense of depth and movement. Several small, solid dark gray circles are placed at key intersection points, serving as focal points or nodes within the network. The overall composition is non-representational and emphasizes the relationships between the lines and the spaces they define.

WHY IT IS IMPORTANT?

- Human mobility forecast
- Type of relationship in social networks
- Traffic congestion





05 Node2Vec for Label Prediction

Jupyter - notebook

