



GNNs

Graph Neural Networks

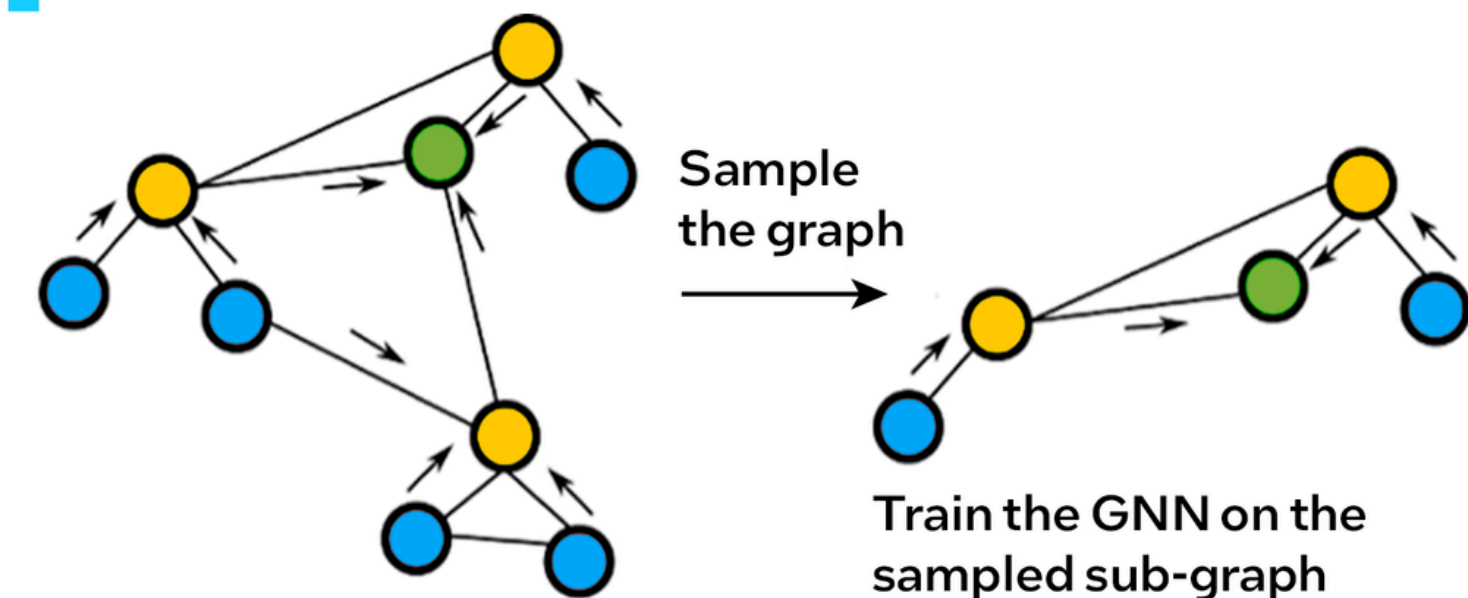
Sumário

- Introdução às Redes Neurais Gráficas (GNNS)
- Arquitetura das GNNs
- Graphic Convolutional Network (GCN)
- Exemplo de GNN



Introdução às Redes Neurais Gráficas (GNNs)

O que são as GNNs?

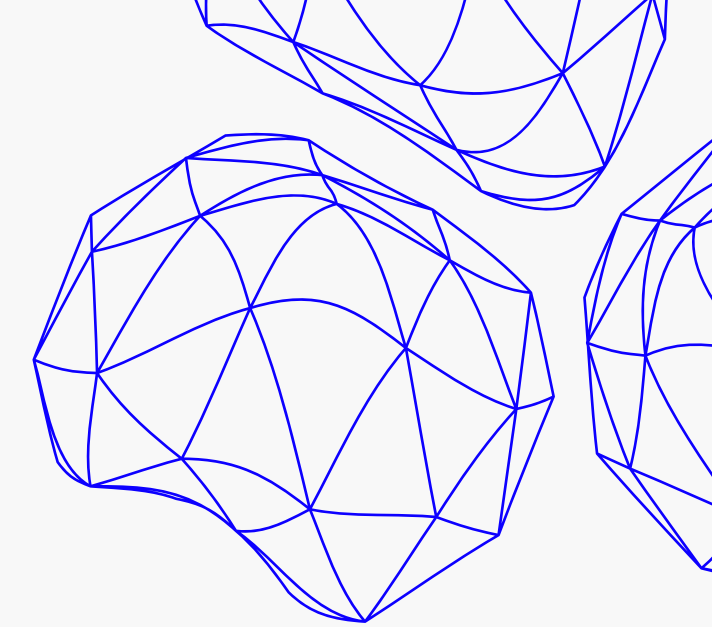


- São uma classe de modelos de aprendizado profundo projetados para operar em dados estruturados em grafos.
- As GNNs são uma extensão das redes neurais convolucionais tradicionais

Diferença entre dados estruturados e dados em grafos

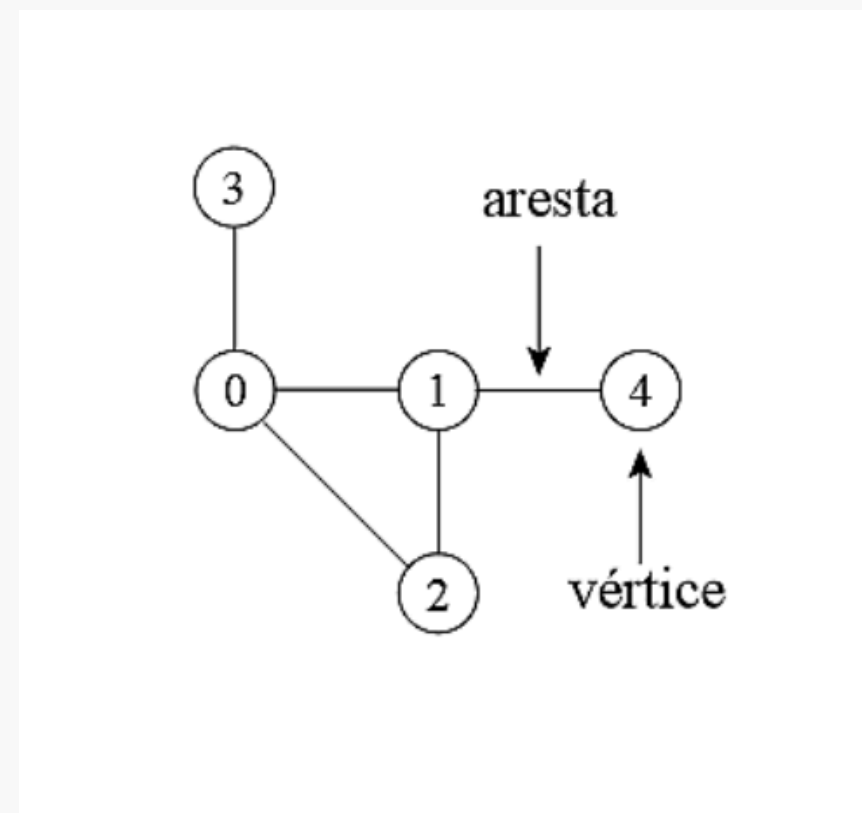
	Dados Estruturados	Dados em Grafos
Estrutura	Regular (linhas/columnas)	Irregular (nós/arestas)
Relações entre dados	Implícitas ou ausentes	Explícitas e complexas
Exemplo	Tabela, imagem, sequência	Rede social, molécula, mapa
Modelos típicos	MLP, CNN, RNN	GNN

Noções básicas de Grafos



Conceitos Básicos:: Nós (vértices), arestas, vizinhos, graus.

Tipos de grafos

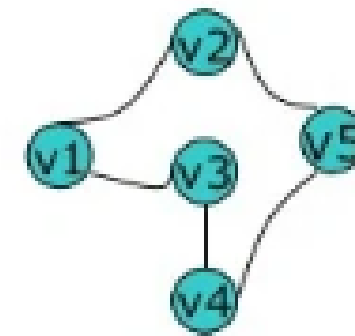


Notação: $G=(V,A)$

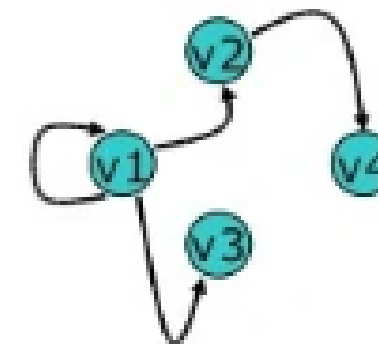
G: Grafo.

V: é o conjunto de vértices (ou nós)

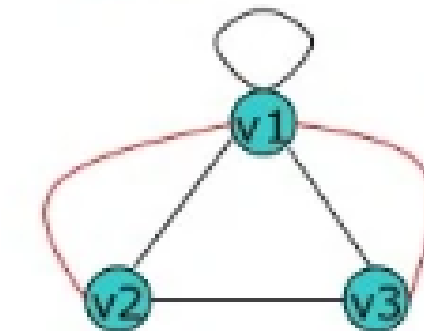
A: é o conjunto de arestas.



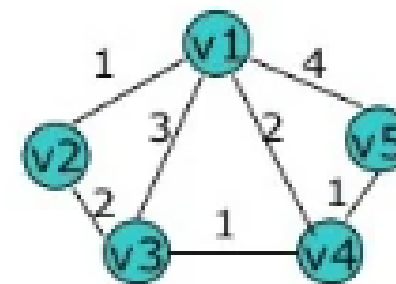
No Dirigido



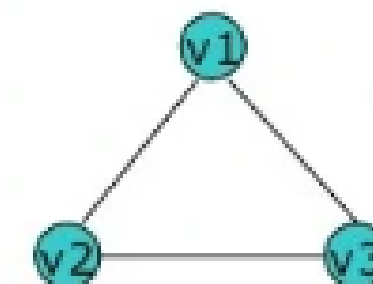
Dirigido



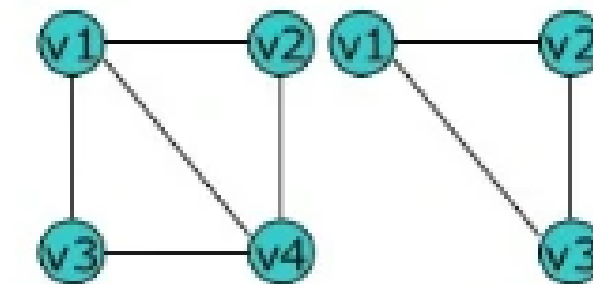
No Simple



Ponderado



Completo



De Similitud

Vantagens das GNNs

Mantém bom desempenho mesmo quando o tamanho do grafo cresce significativamente.

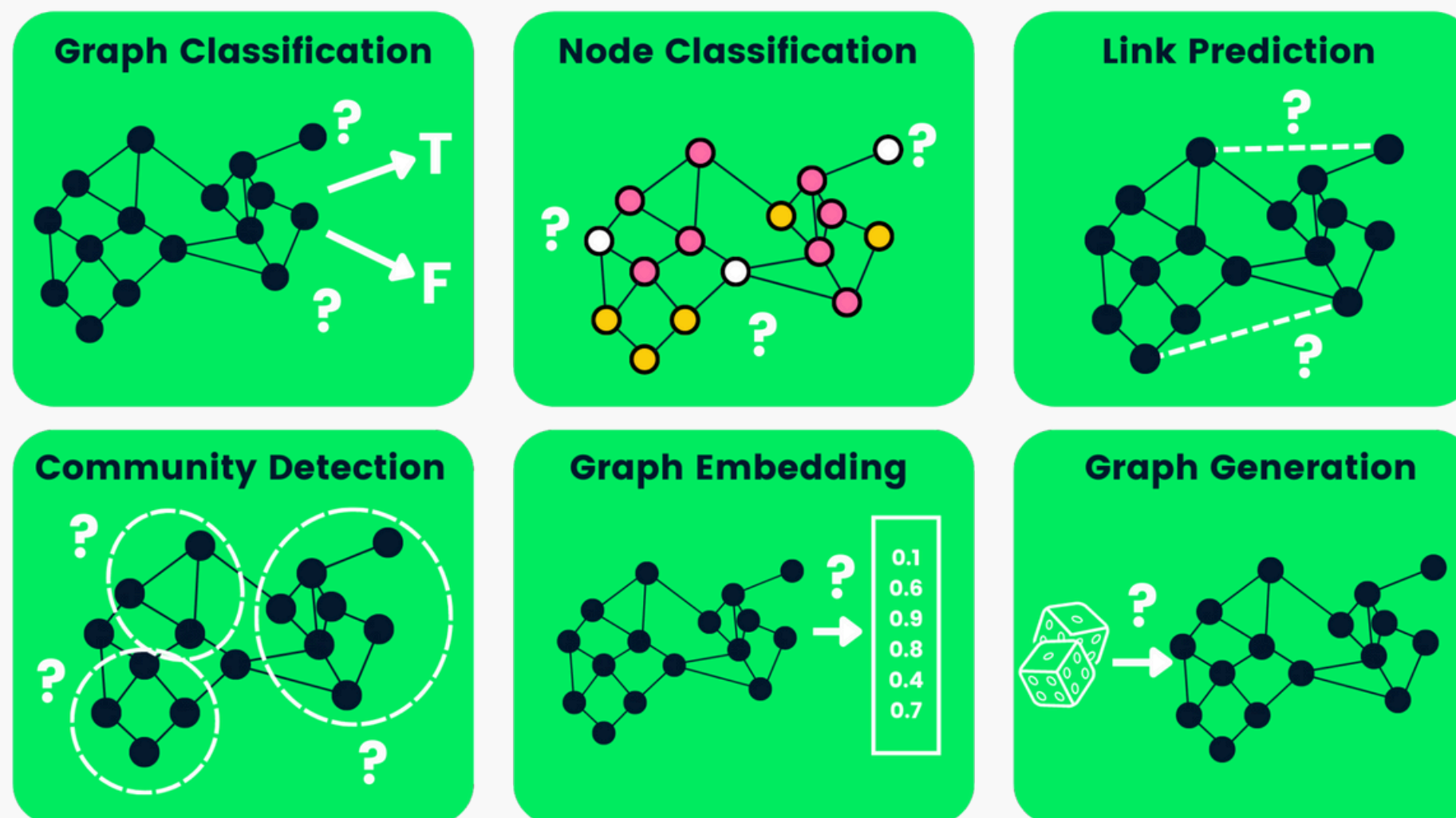
- O número de parâmetros aprendíveis não depende do número de nós do grafo.
- A arquitetura da GNN compartilha pesos entre os nós, permitindo generalização eficiente mesmo em grafos grandes.

Aplicações das GNNs

Aplicações Típicas de GNNs em Grafos

Algumas aplicações:

- **Rede social:** usuários conectados por amizades.
- **Moléculas:** átomos conectados por ligações químicas.
- **Mapas:** locais conectados por estradas.





Arquitetura das GNNs

Arquitetura das GNNs

1

Representação do Grafo como Entrada

Entrada de uma GNN é composta por:

- Matriz de adjacência (estrutura do grafo)
- Matriz de atributos dos nós
- Exemplo: com atributos em cada nó

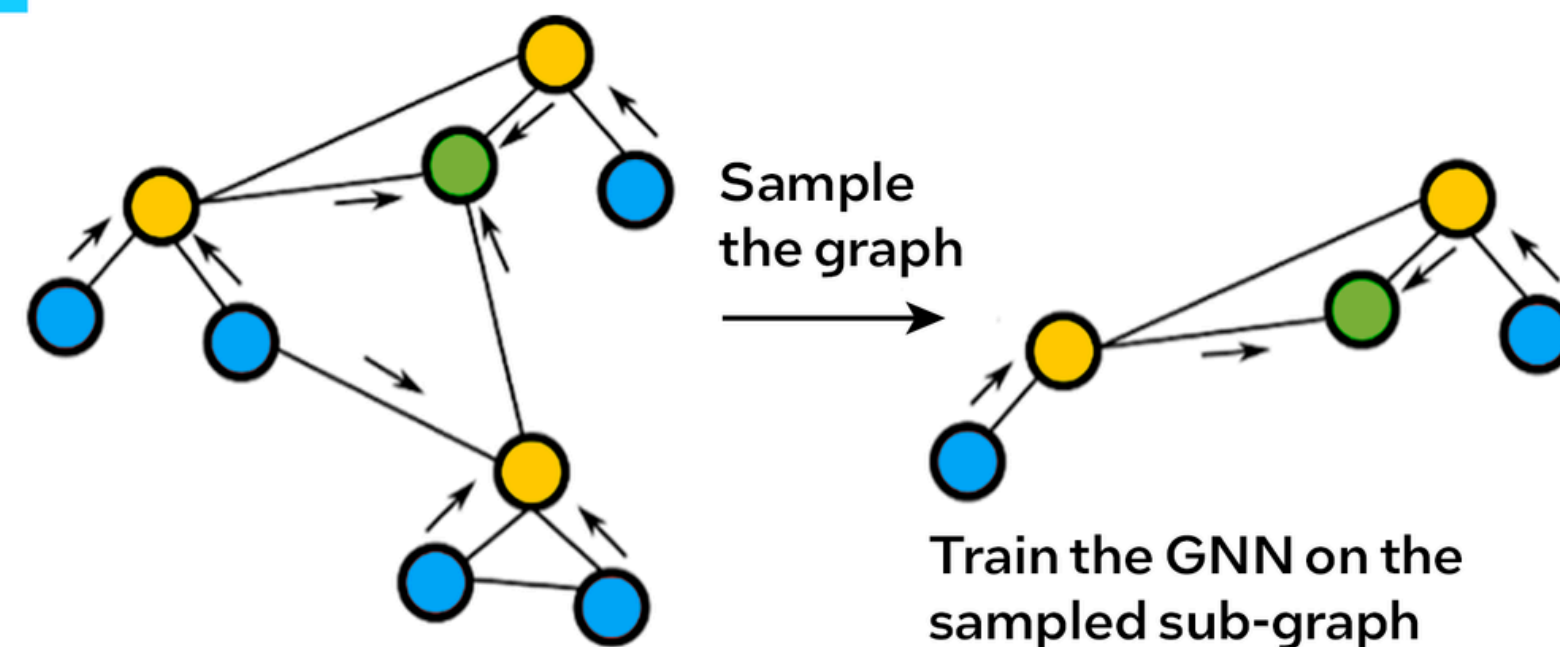
Estratégias de Amostragem

- GraphSAGE, neighbor sampling, cluster sampling, e

2

Propagação de Mensagens (Message Passing)

- Cada vetor do nó é atualizado com base em seus próprios atributos e nas mensagens agregadas dos vizinhos:
- Duas etapas principais:
 - Agregação: soma/média/máximo das features dos vizinhos
 - Combinação: atualiza o vetor do nó alvo



Arquitetura das GNNs

3

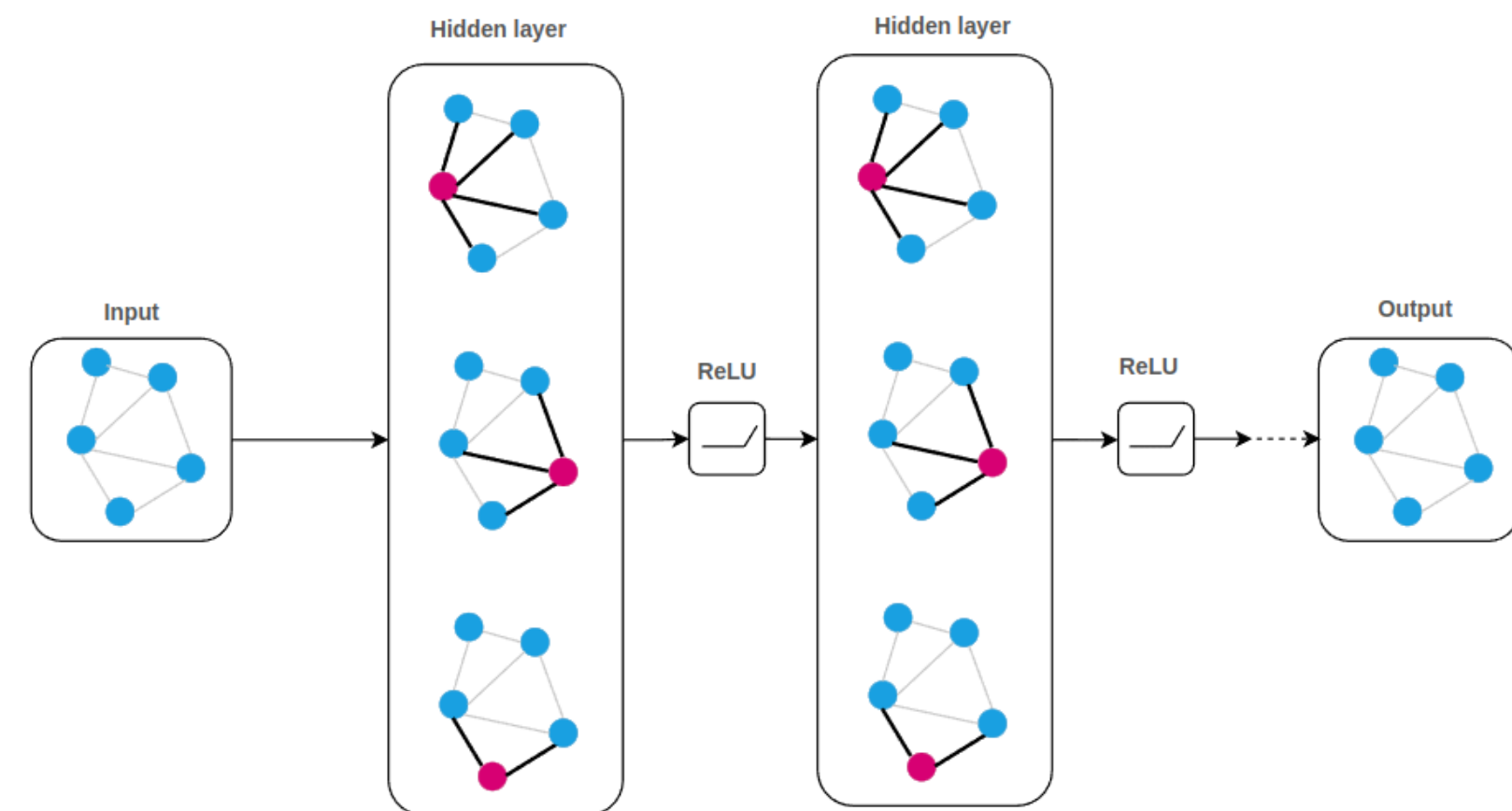
Camadas da GNN

- Cada camada de uma GNN atualiza o embedding dos nós, incorporando informações cada vez mais distantes da estrutura do grafo:
- A primeira camada considera os vizinhos imediatos de cada nó. As camadas seguintes integram informações de vizinhos mais distantes (2ª ordem, 3ª ordem, etc.)

4

Função de Ativação

- Após a combinação, aplica-se uma função não linear (ReLU, tanh etc.)

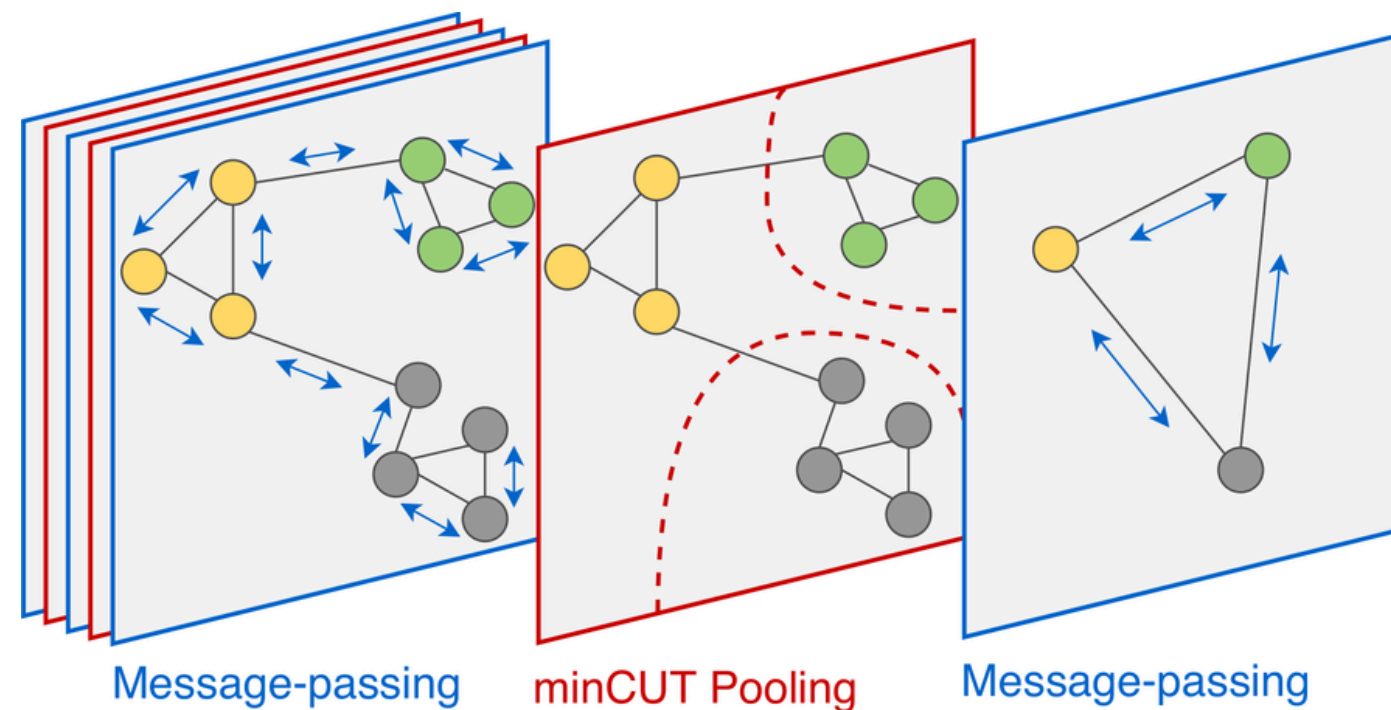


Arquitetura das GNNs

5

Pooling

- Para tarefas globais (ex: classificação de grafos), é preciso gerar um embedding geral:
- Pooling: média, soma ou max global dos embeddings dos nós.



6

Treinamento da GNN

- Perda computada conforme a tarefa (cross-entropy, MSE etc.)
- Otimização por backpropagation
- Algoritmos: SGD, Adam, etc.



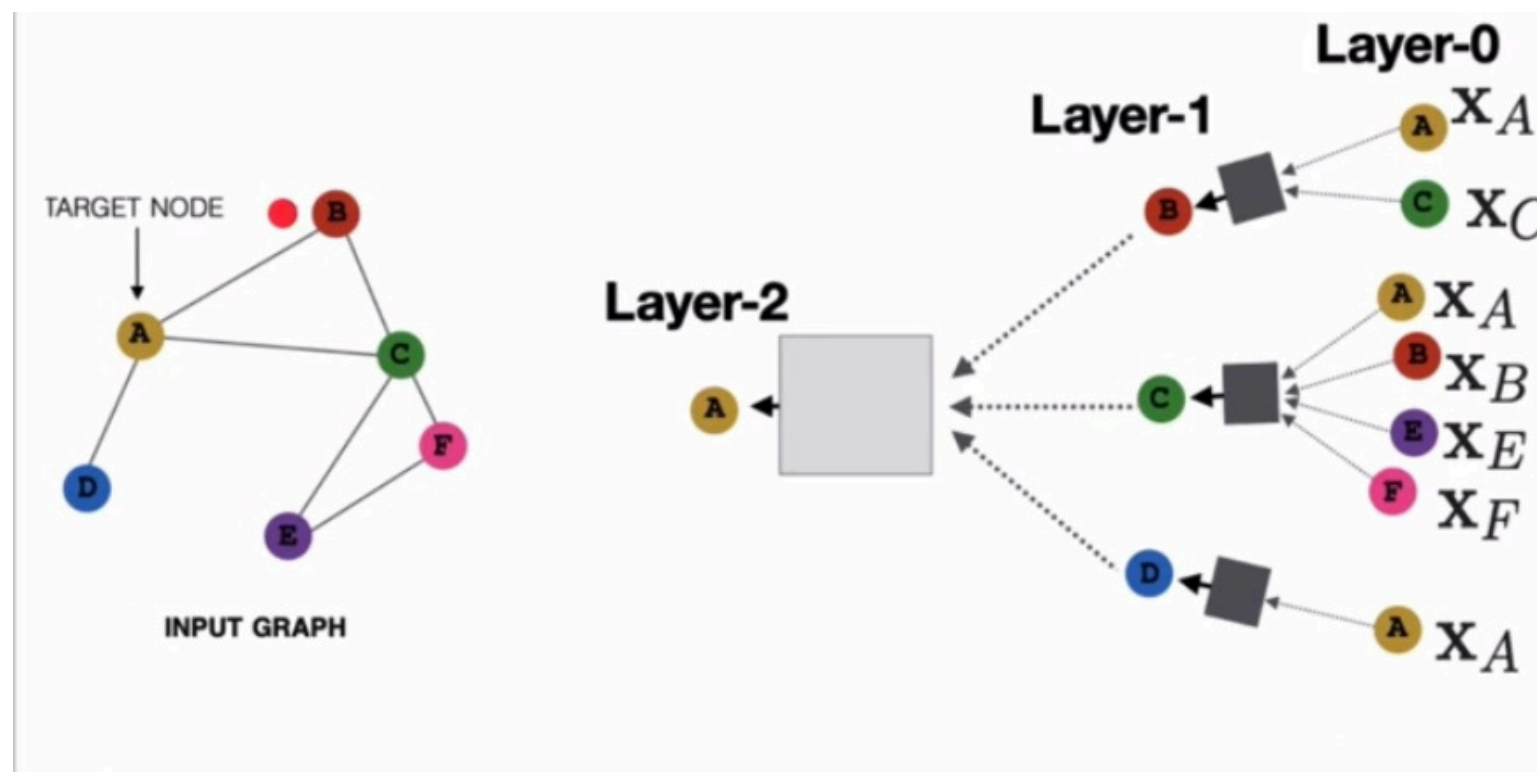
Graphic Convolutional Network (GCN)

GCN - Graph Convolutional Networks

1

Ideia principal

- Propagar as “características” (embeddings) dos nós vizinhos por camadas



2

Operação de Convolução

$$h_v^0 = x_v \quad \leftarrow \text{Embedding inicial do nó } v$$

$$h_v^K = \sigma \left(W_K \cdot \left(\sum_{u \in N(v) \cup v} \frac{h_u^{K-1}}{\sqrt{|N(u)| \cdot |N(v)|}} \right) \right)$$

Normalização por vizinhos dos embeddings da camada anterior
Não-linearidade (ex.: ReLU)

Embedding do nó v na K -ésima camada

Teste de Weisfeiler-Lehman (WL) como Limite

1

Teste 1-WL (Discreto)

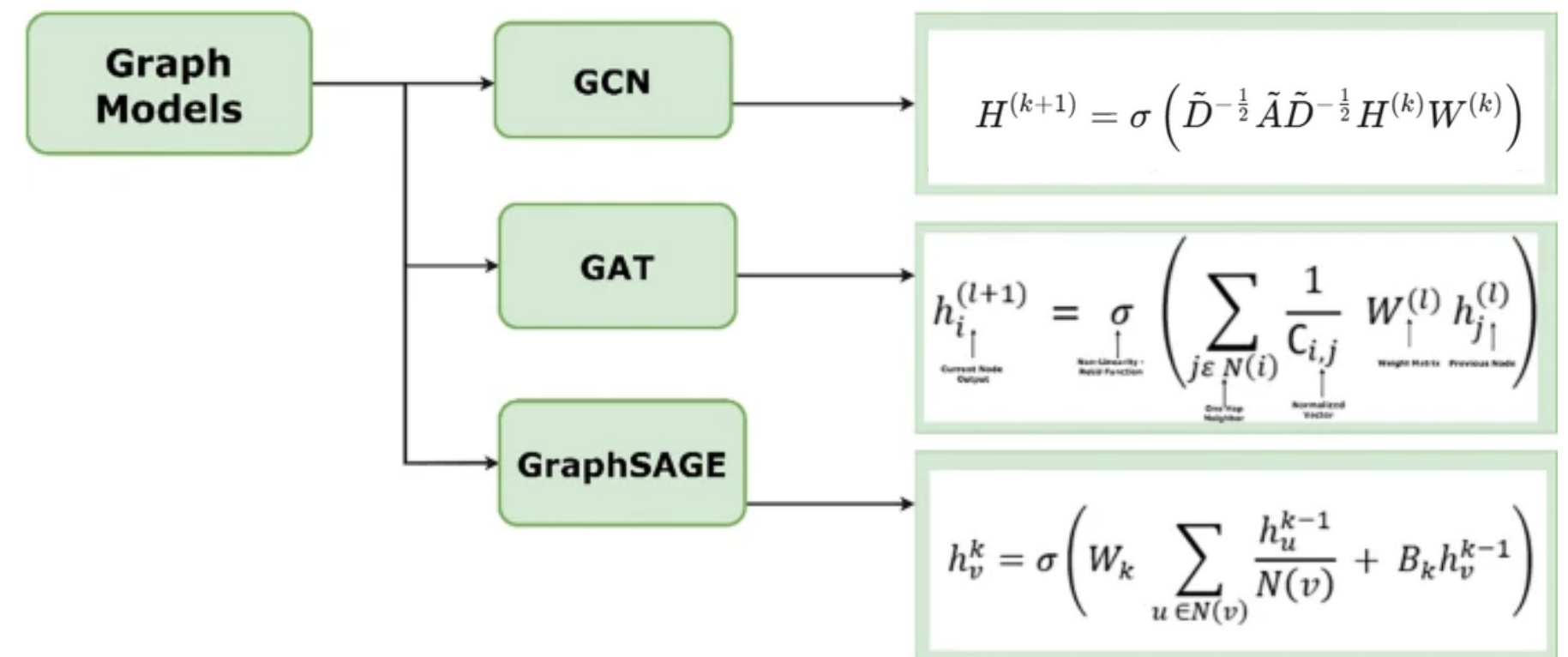
$$l_v^{(k+1)} = \text{HASH}(l_v^{(k)}, \{\{l_u^{(k)}\}_{u \in \mathcal{N}(v)}\})$$

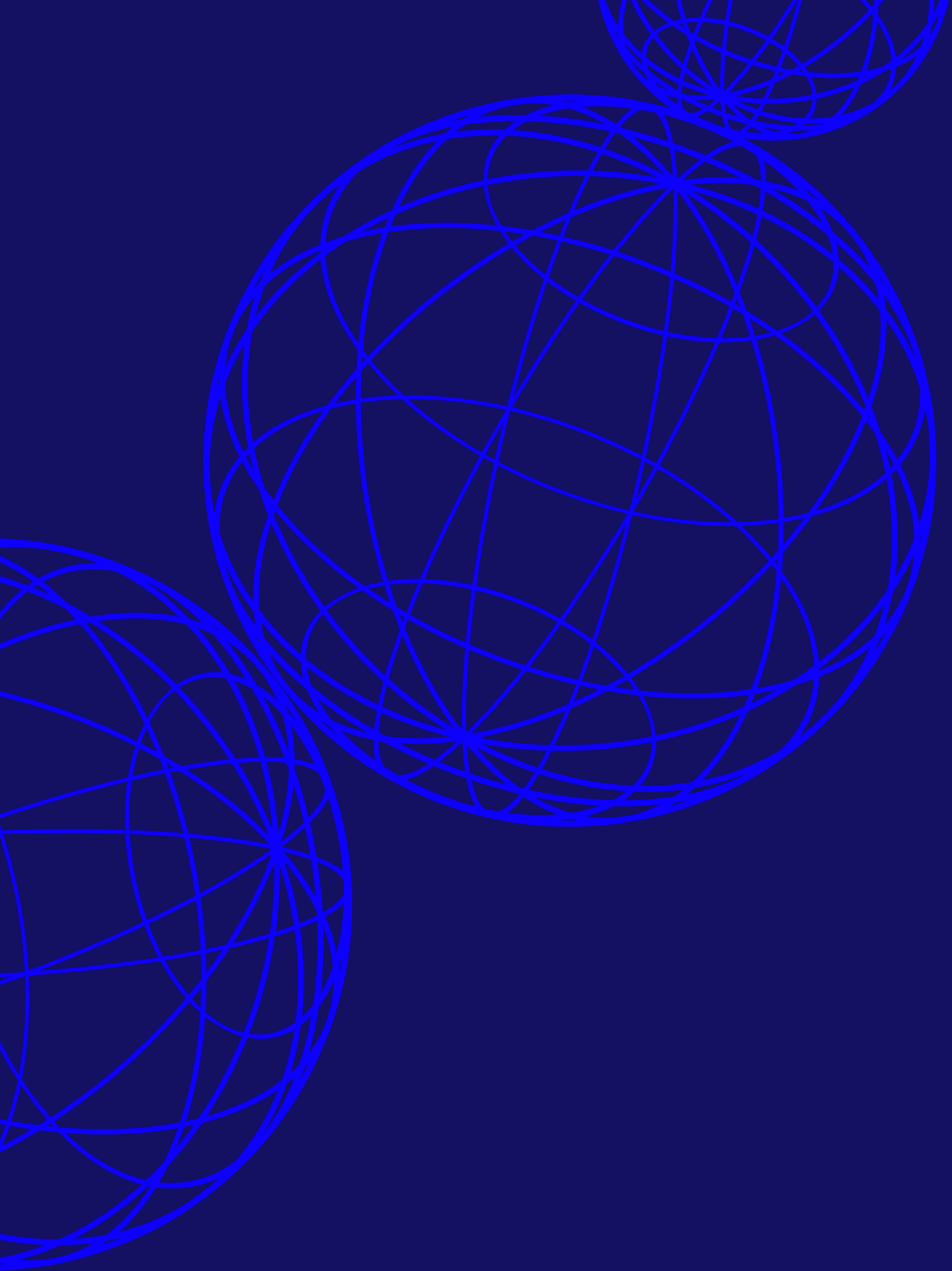
2

k-GNNs (Contínuo)

- Custo Operacional: $O(n \wedge k)$

$$h_s^{(k+1)} = \text{UPDATE}(h_s^{(k)}, \text{AGGREGATE}(\{h_{s'}^{(k)}\}_{s' \in \mathcal{N}(s)}))$$



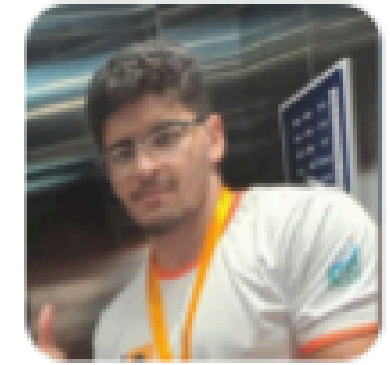


Exemplo de GNN

Graph Neural Network to detect fake news messages using the dataset

'User Preference-aware Fake News Detection'.

lucasdb05/gnn-fake-news-detection



Graph Neural Network to detect fake news messages using the dataset 'User Preference-aware Fake News Detection'.

1

Contributor

0

Issues

0

Stars

0

Forks



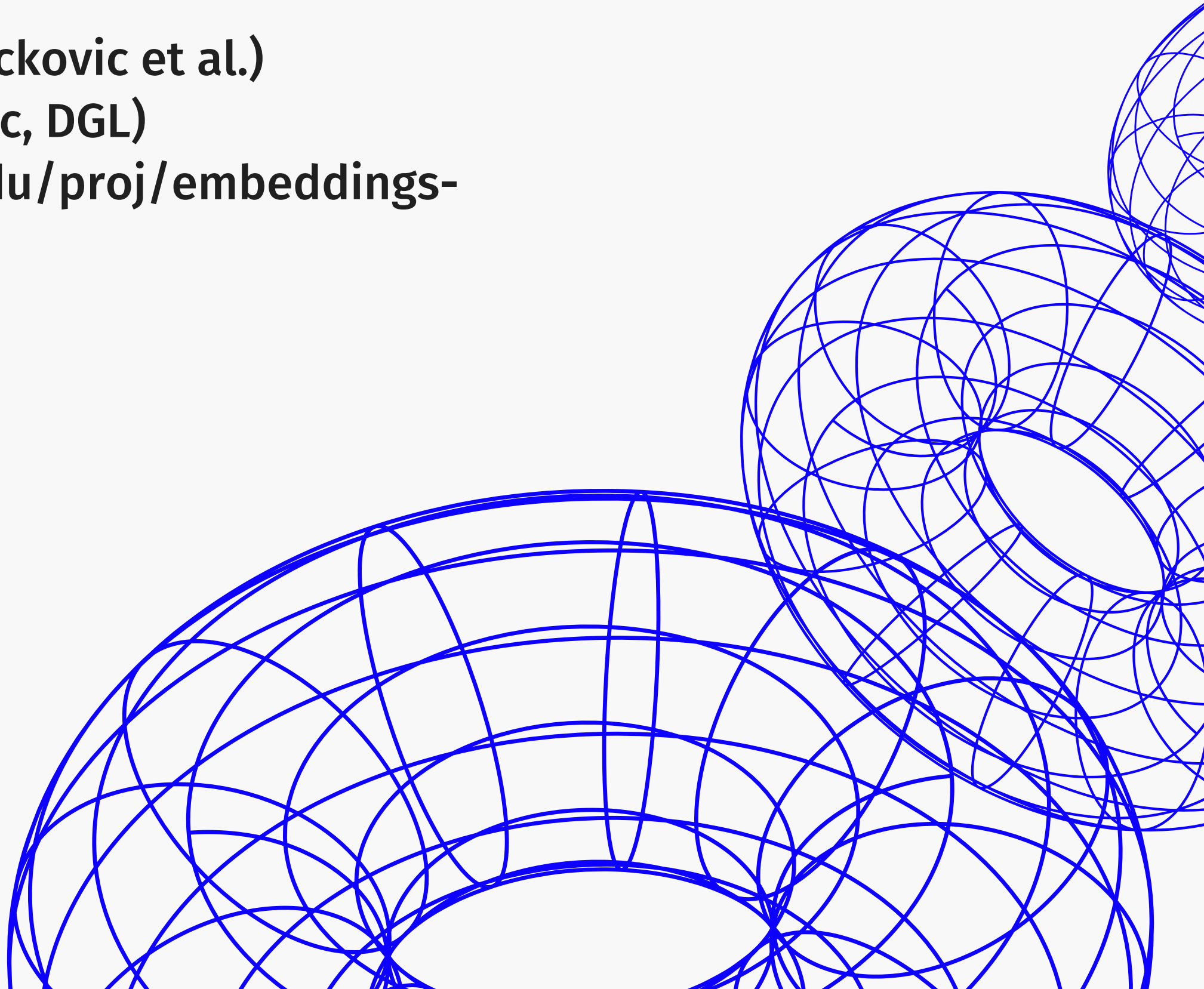
lucasdb05/gnn-fake-news-detection: Graph Neural Network to detect fake news messages using the datas...

Graph Neural Network to detect fake news messages using the dataset 'User Preference-aware Fake News Detection'. - lucasdb05/gnn-fake-news-detection

GitHub

Referências

- Papers importantes (Kipf & Welling, Velickovic et al.)
- Frameworks de GNNs (PyTorch Geometric, DGL)
- Snap Stanford (<https://snap.stanford.edu/proj/embeddings-www/files/nrltutorial-part2-gnns.pdf>)
- Tutoriais e benchmarks





Thank you!