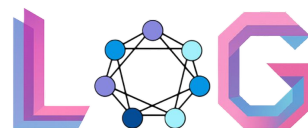




LoG Conference

Introdução às *Graph Neural Networks*

Marcos Gôlo



CONFERENCE

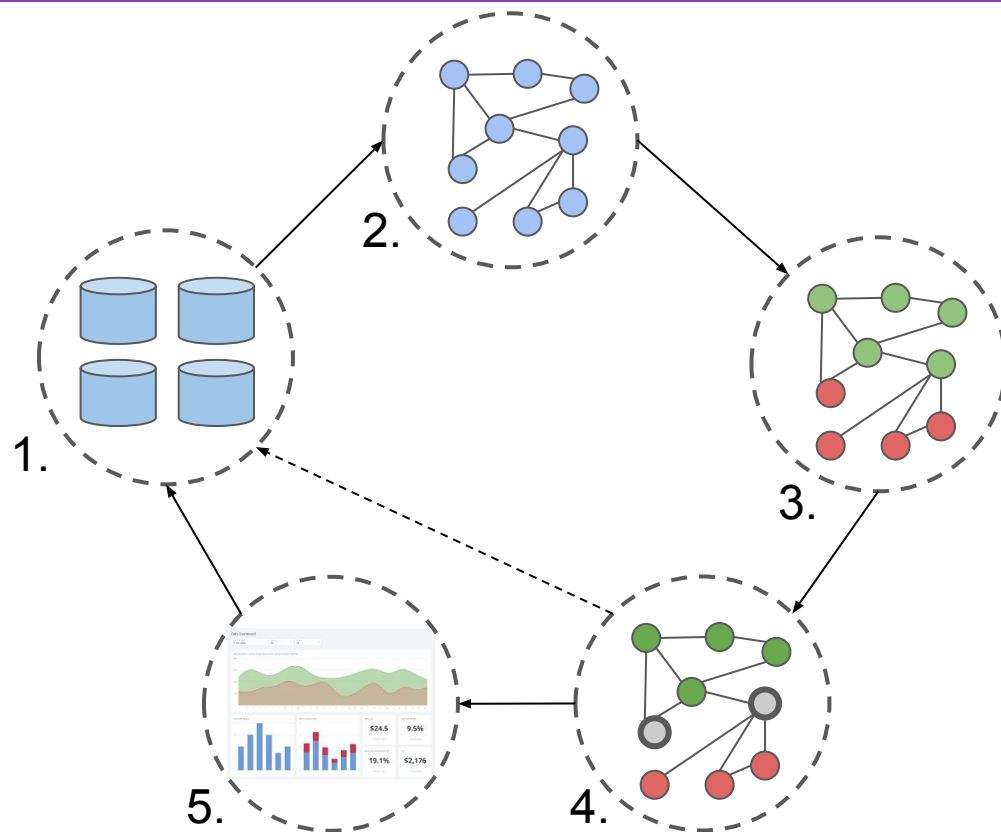


Sumário

1. Introdução
2. Grafos
3. Graph Neural Networks
4. Contextos de exploração (Aplicações)
5. Prática: **Unsupervised** Graph neural networks
6. Código extra: **Supervised** Graph neural networks

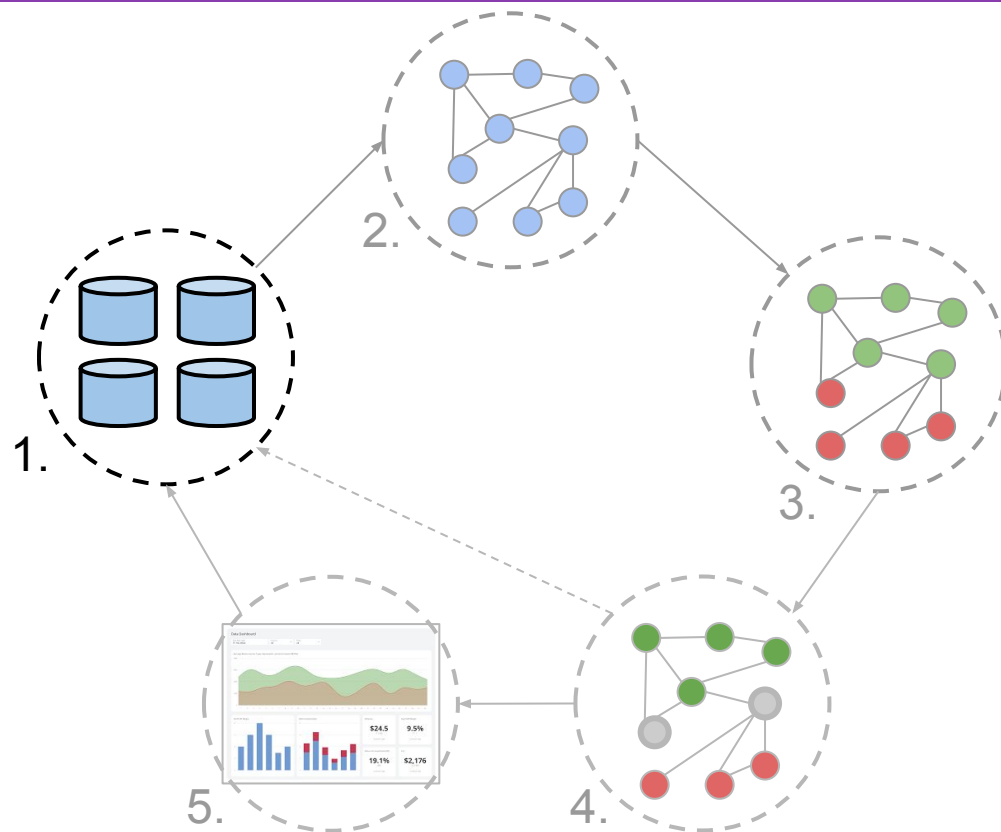
Mineração de dados

1. Dados
2. Pré-processamento
3. Aprendizado de padrões
4. Pós-processamento
5. Uso do conhecimento



Mineração de dados

1. Dados
2. Pré-processamento
3. Aprendizado de padrões
4. Pós-processamento
5. Uso do conhecimento



Dados

Estruturado

Nome	Altura	Peso	IMC
José	1.80	75	23.15
Maria	1.65	69	25.34
Paula	1.71	68	23.26

Tabela com dados pessoais

Não estruturado



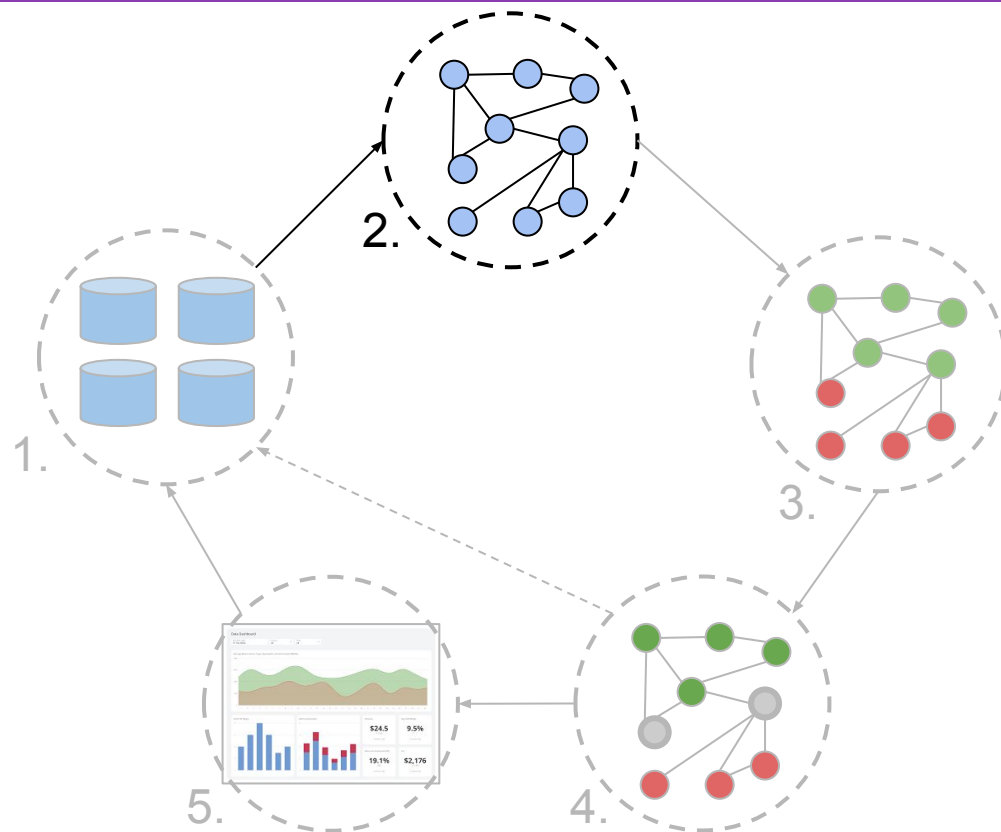
Flamengo campeão de tudo em 2025!

O malvadão voltou!!!!



Mineração de dados

1. Dados
2. Pré-processamento
3. Aprendizado de padrões
4. Pós-processamento
5. Uso do conhecimento obtido

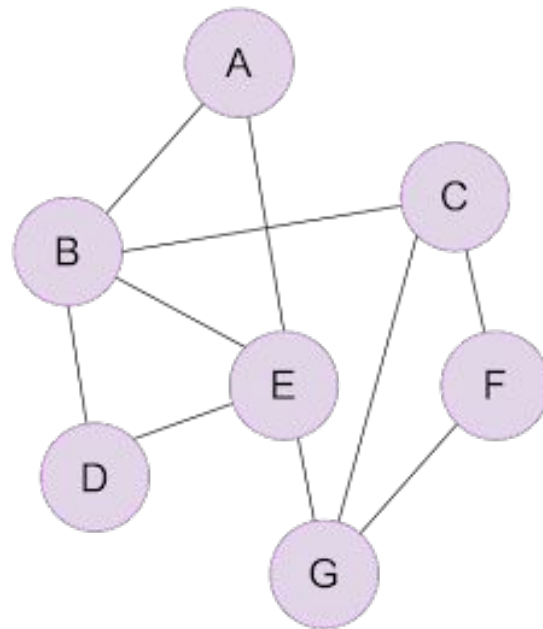


Grafos

Definição de grafos $G = (V, A)$

$V = \{A, B, C, D, E, F, G\}$

$A = \{(A, B), (B, C), (B, D), \dots, (F, G)\}$



Grafos

Definição de grafos $G = (V, A)$

$V = \{A, B, C, D, E, F, G\}$

$A = \{(A, B), (B, C), (B, D), \dots, (F, G)\}$

	A	B	C	D	E	F	G
A	0	1	0	0	1	0	0
B	1	0	1	1	1	0	0
C	0	1	0	0	0	1	1
D	0	1	0	0	1	0	0
E	1	1	0	1	0	0	1
F	0	0	1	0	0	0	1
G	0	0	1	0	1	1	0

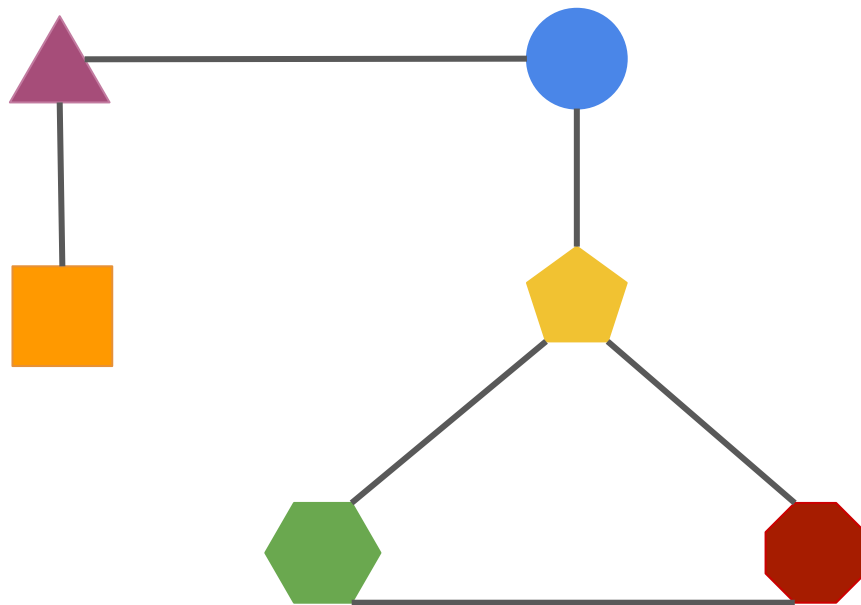
A = Matriz de adjacência

Modelagem em grafos

Dados: Pesquisadores

Nós:







- Marcos 
- Solange 
- Ricardo 
- Gilberto 
- Diego 
- Rafael 



Relações: Orientação/Pesquisa relacionada

Modelagem em grafos

Nós:

- A 
- B 
- C 
- D 
- E 
- F 







Relações: ?

O que fazer?

Modelagem em grafos

Dados: Não estruturados

Nós:

- A 
- B 
- C 
- D 
- G 
- H 

Relações: ?

**Podemos inferir as
relações por meio de
SIMILARIDADE**

Dados

Estruturado




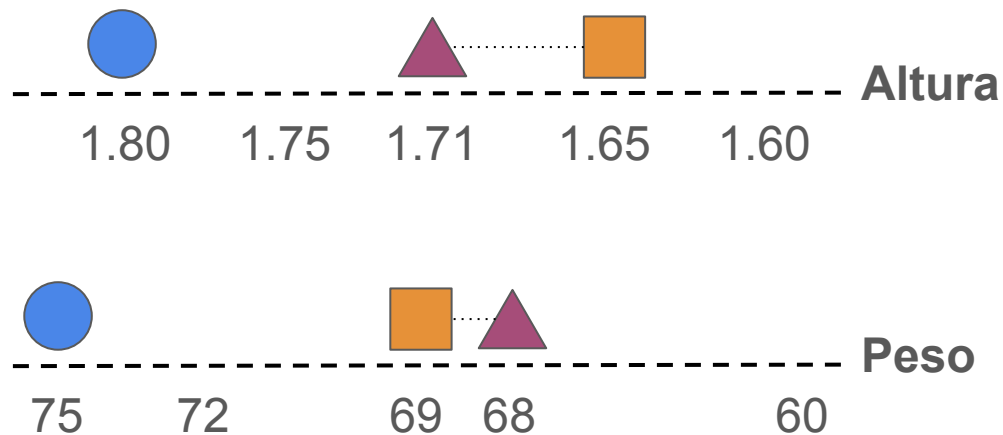
	Nome	Altura	Peso	IMC
	José	1.80	75	23.15
	Maria	1.65	69	25.34
	Julia	1.71	68	23.26

Tabela com dados pessoais



Dados

Não estruturado



Flamengo campeão de
tudo em 2025!
O malvadão voltou!!!!



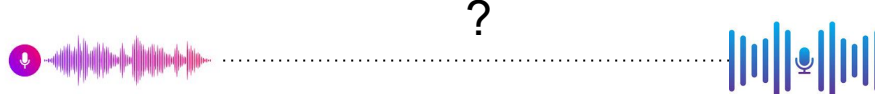
?



Flamengo campeão de
tudo em 2025!
O malvadão voltou!!!!

?

Flamengo campeão da
Supercopa do Br!

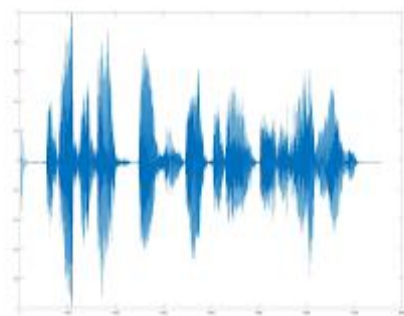


?

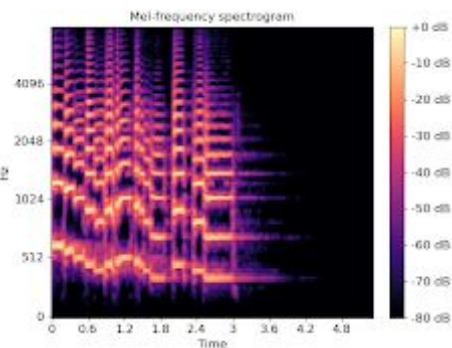


Áudio?

- características que englobam informações de tempo e frequência
- conseguimos enfatizar diferentes componentes do áudio



Sinal de áudio



Mel espectrograma

librosa. Disponível em: <https://librosa.org/doc>

Texto?

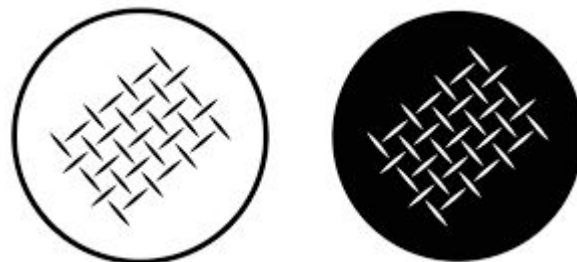
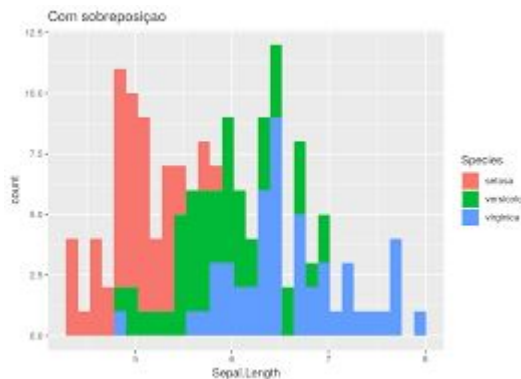
- construção de vetores a partir de textos
- características que enfatizam informação de contexto
- características que possibilitam computar a similaridade entre textos

1. O curso foi ótimo
2. Aprendi sobre GNN
3. Ótimo curso sobre GNN

0	curso	foi	ótimo	aprendi	sobre	GNN
1	1	1	1	0	0	0
0	0	0	0	1	1	1
0	1	0	1	0	1	1

Imagem?

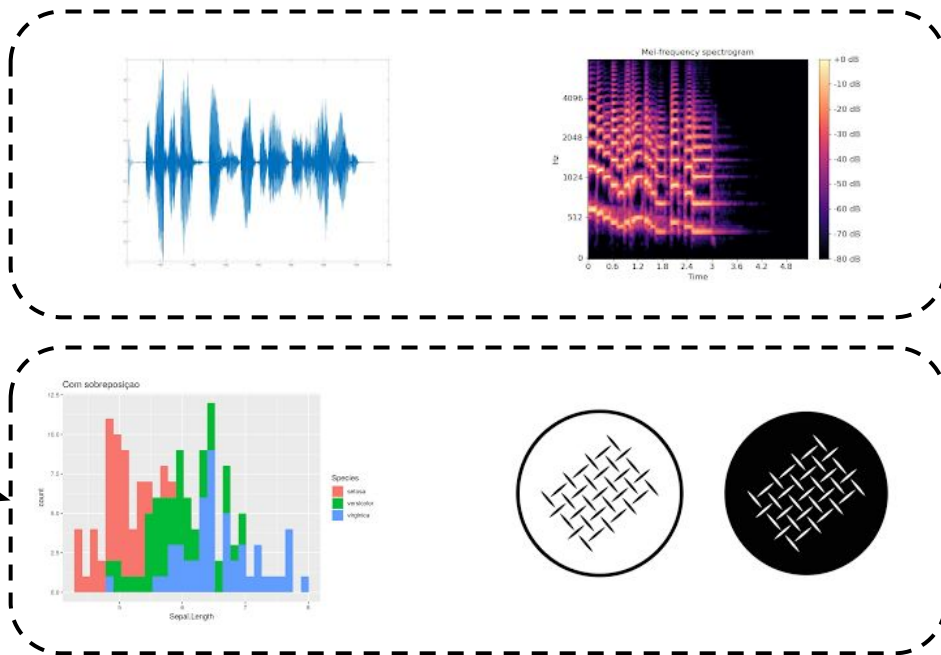
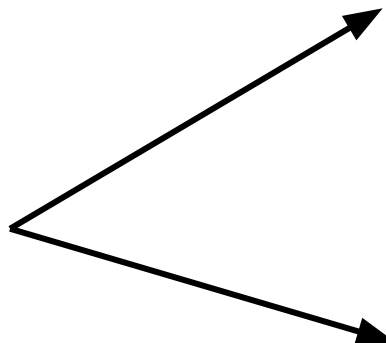
- construção de vetores a partir de histograma de cores
- características que enfatizam a textura
- características que possibilitam computar a similaridade entre imagem



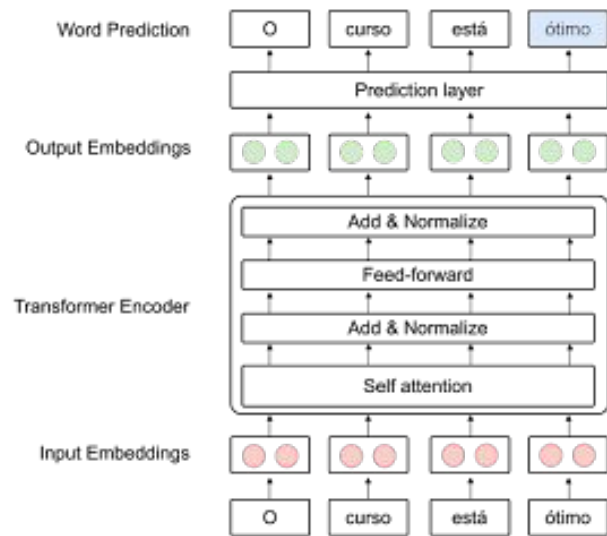
Csurka, G. et al. "Visual categorization with bags of keypoints." *Workshop on statistical learning in computer vision*. 2004.

Video?

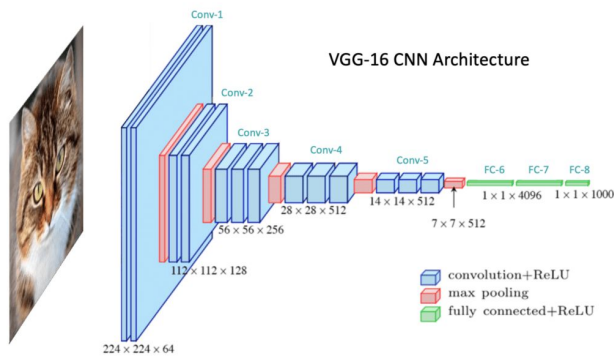
- Combinação entre vetores de características de áudio e imagem
- Fusão
- Multimodalidade



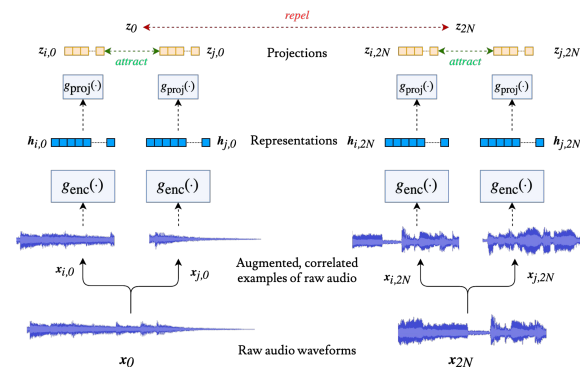
Embeddings



Texto



Imagem












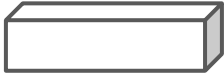


Áudio

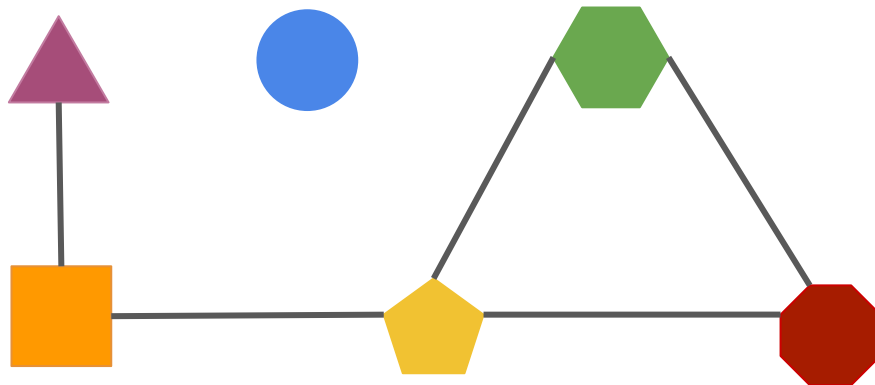
Jacob Devlin et al.. **BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.**
 Karen Simonyan, Andrew Zisserman. **Very Deep Convolutional Networks for Large-Scale Image Recognition.**
 Jordi Pons et al.. **End-to-end learning for music audio tagging scale.**

Modelagem em grafos

Nós:

- A  
- B  
- C  
- D  
- G  
- H  

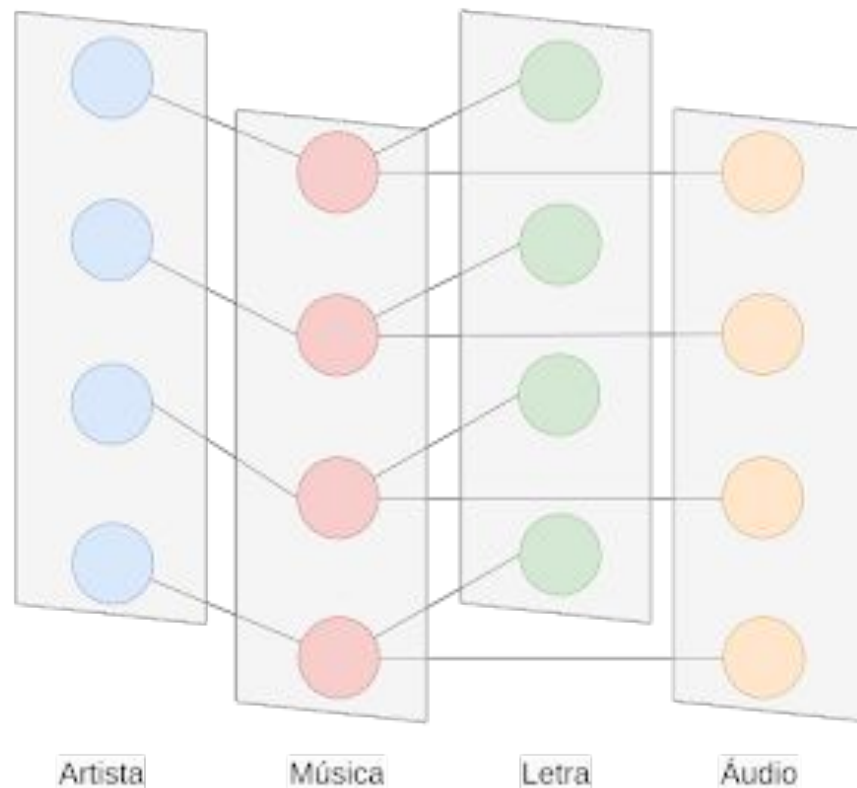
Relações: Inferidas por similaridade



Modelagem em grafos

Grafo desconexo

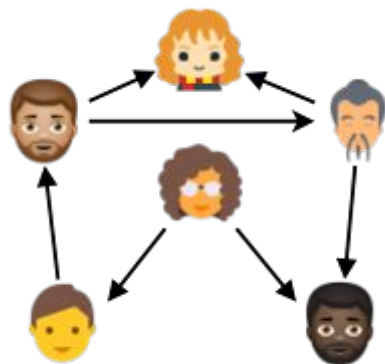
- Não há interação entre músicas distintas
- Não exploramos o potencial da modelagem dos dados em grafos
- Podemos enriquecer esse grafo!



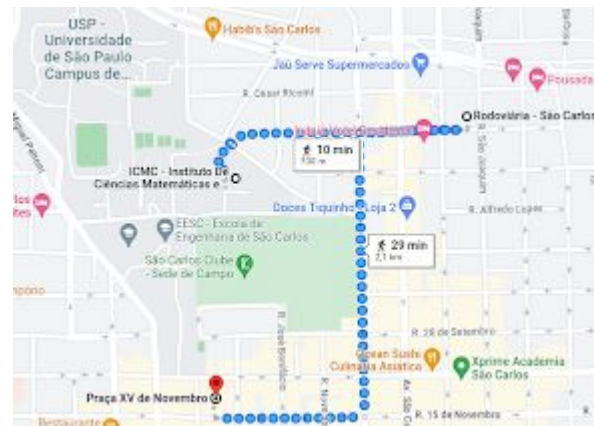
Grafos



Grafo não
direcionado

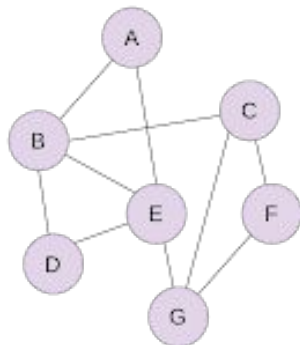


Grafo
direcionado

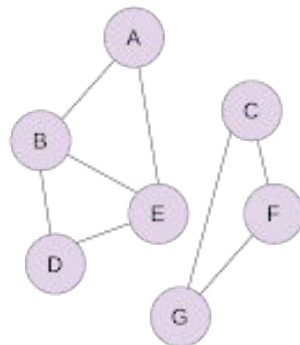


Grafo
ponderado

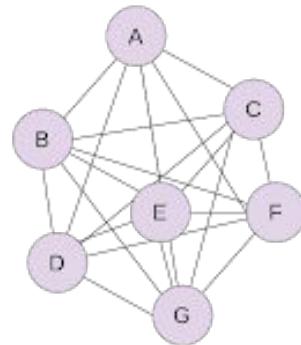
Alguns tipos de grafos



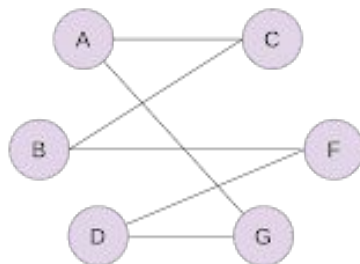
Conexo



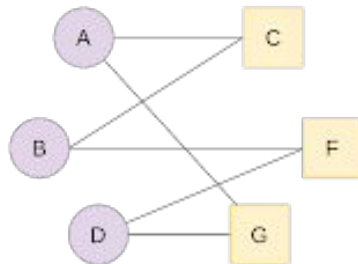
Desconexo



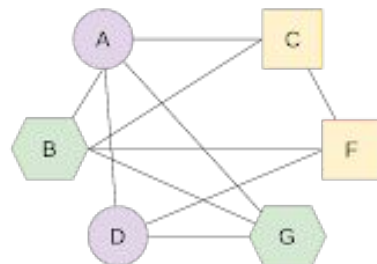
Completo



Homogêneo

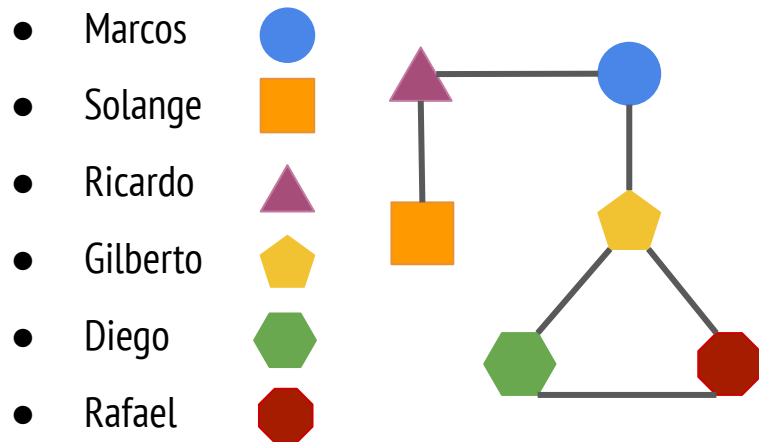


Heterogêneo e bipartido

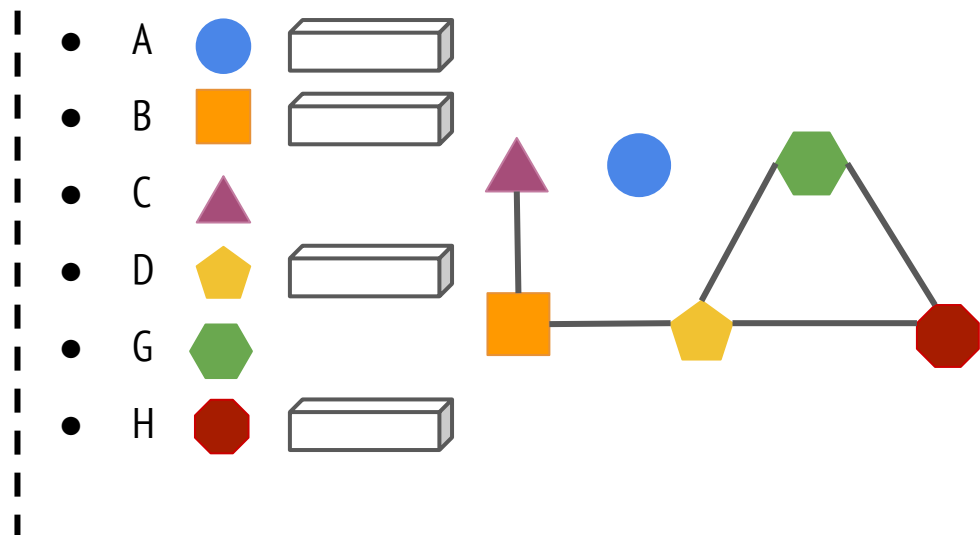


Heterogêneo

Como preparar seu grafo para a GNN?



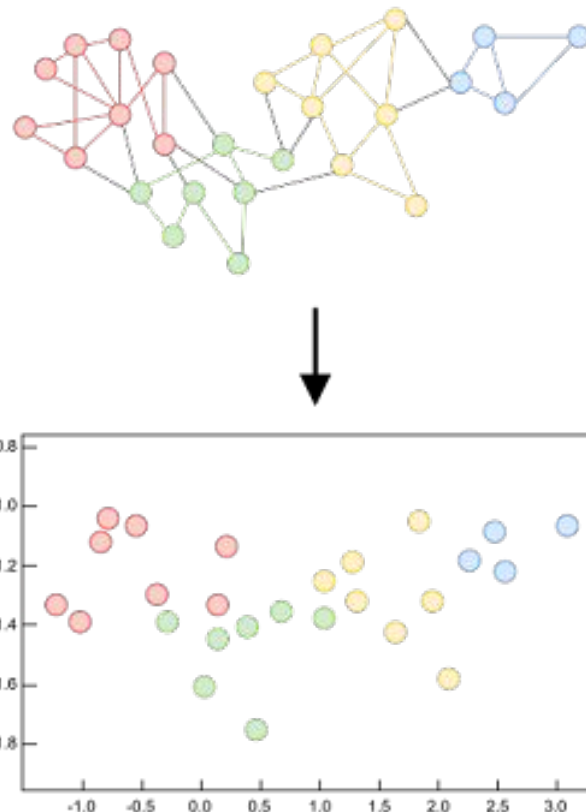
**Representações
iniciais?**



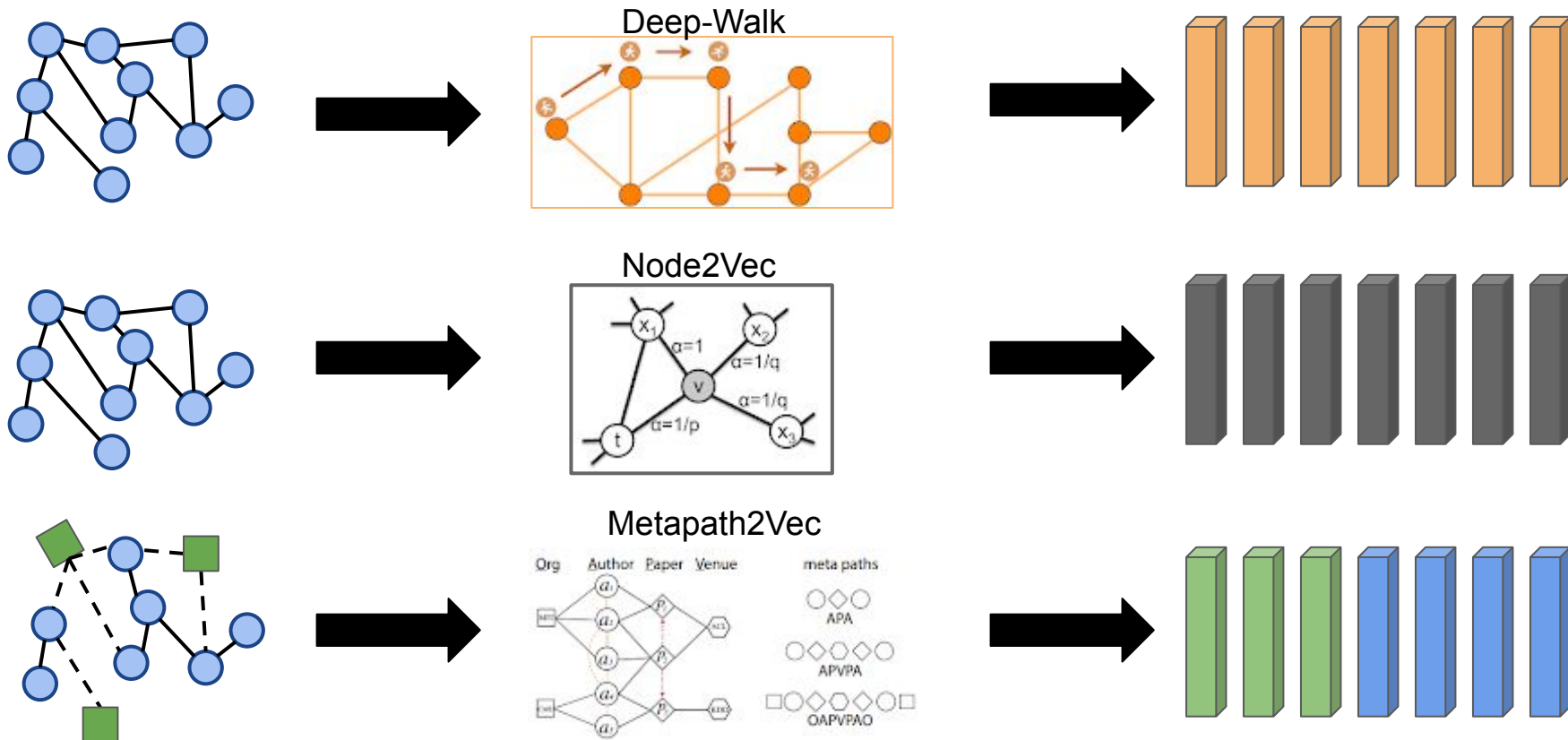
**Representações
faltantes!**

Graph Embedding

- **Vértices sem características**
- **Objetivo:** Encontrar embeddings para os vértices em um espaço d-dimensional que preserve as noções de similaridade
- **Ideia:** Aprender uma representação que aproxime no espaço construído vértices que sejam vizinhos no grafo vizinhos



Graph Embedding



Regularização de grafos

Características ausentes em alguns vértices

Objetivo da regularização:

- propagar informações entre os nós

Premissas:

- objetos que estão relacionados devem ter representações similares
- objetos que já possuem características, devem ter representação final similar à inicial

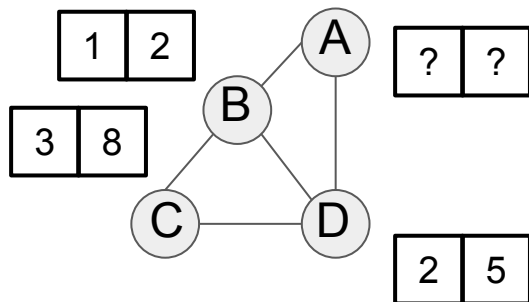
Resultado:

- todos os objetos possuem informações

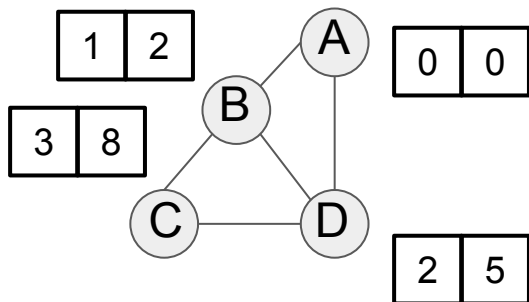
Algoritmos de regularização: <https://github.com/BruceNeves/GraphTLP/tree/main>

Regularização de grafos

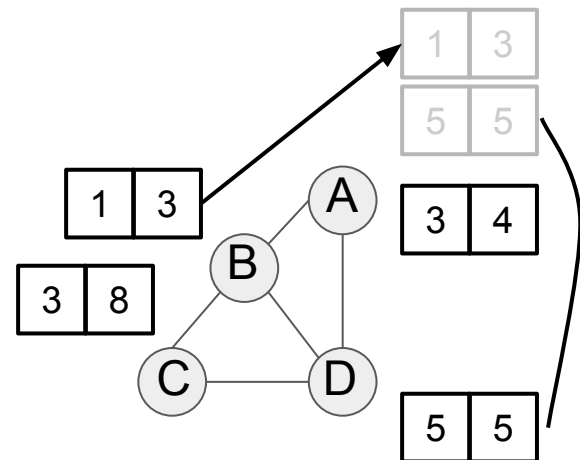
Características ausentes em alguns nós



Estado inicial do grafo. Vértice A não tem características



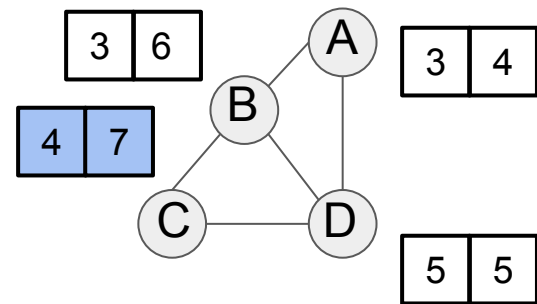
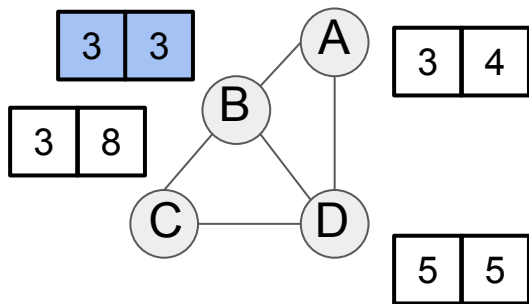
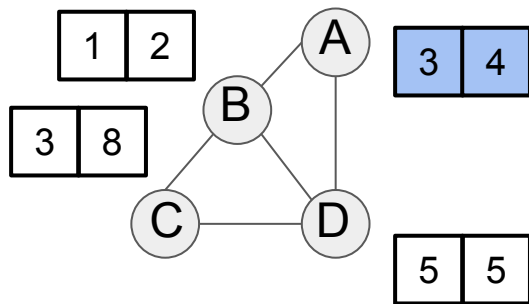
Inicialização das características dos vértice A



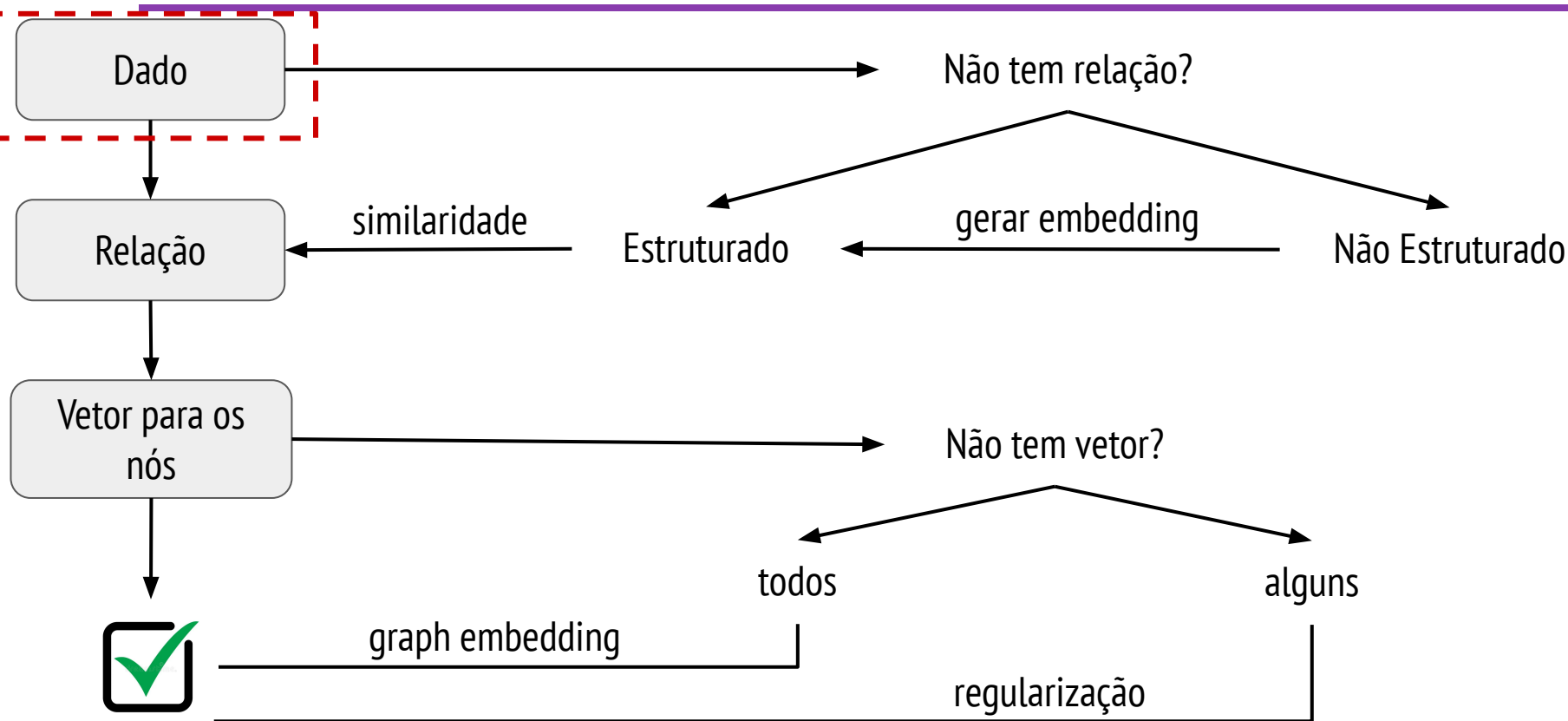
Propagação das características entre vértices vizinhos

Regularização de grafos

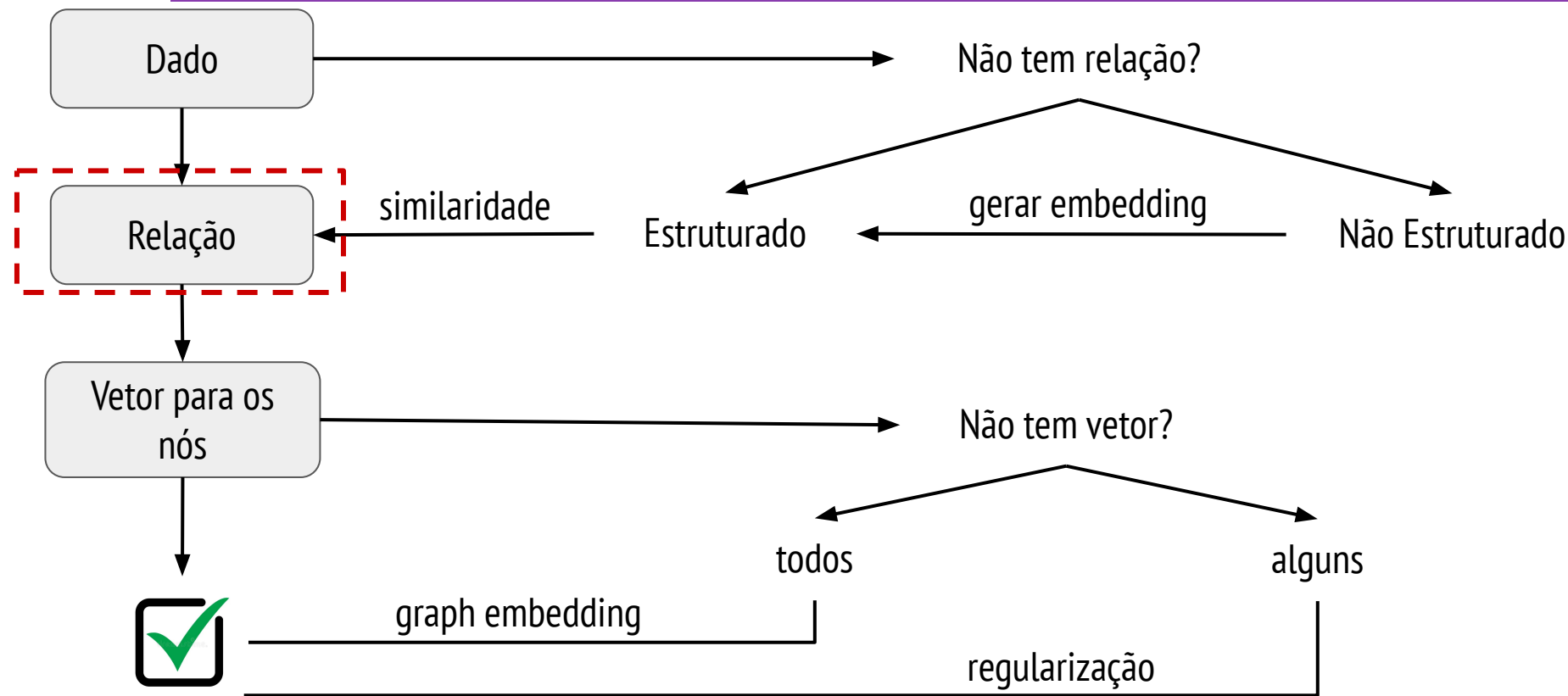
Características ausentes em alguns vértices



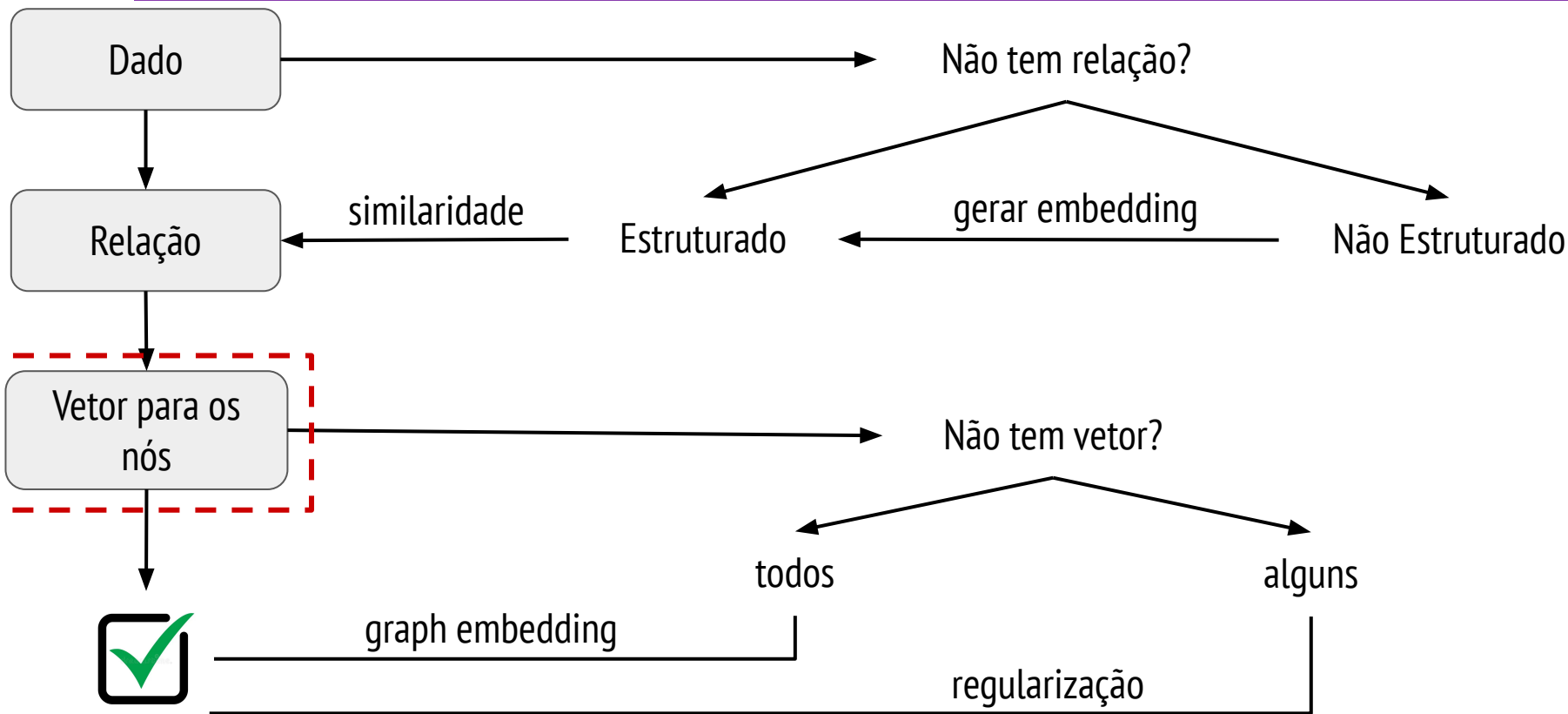
Grafos



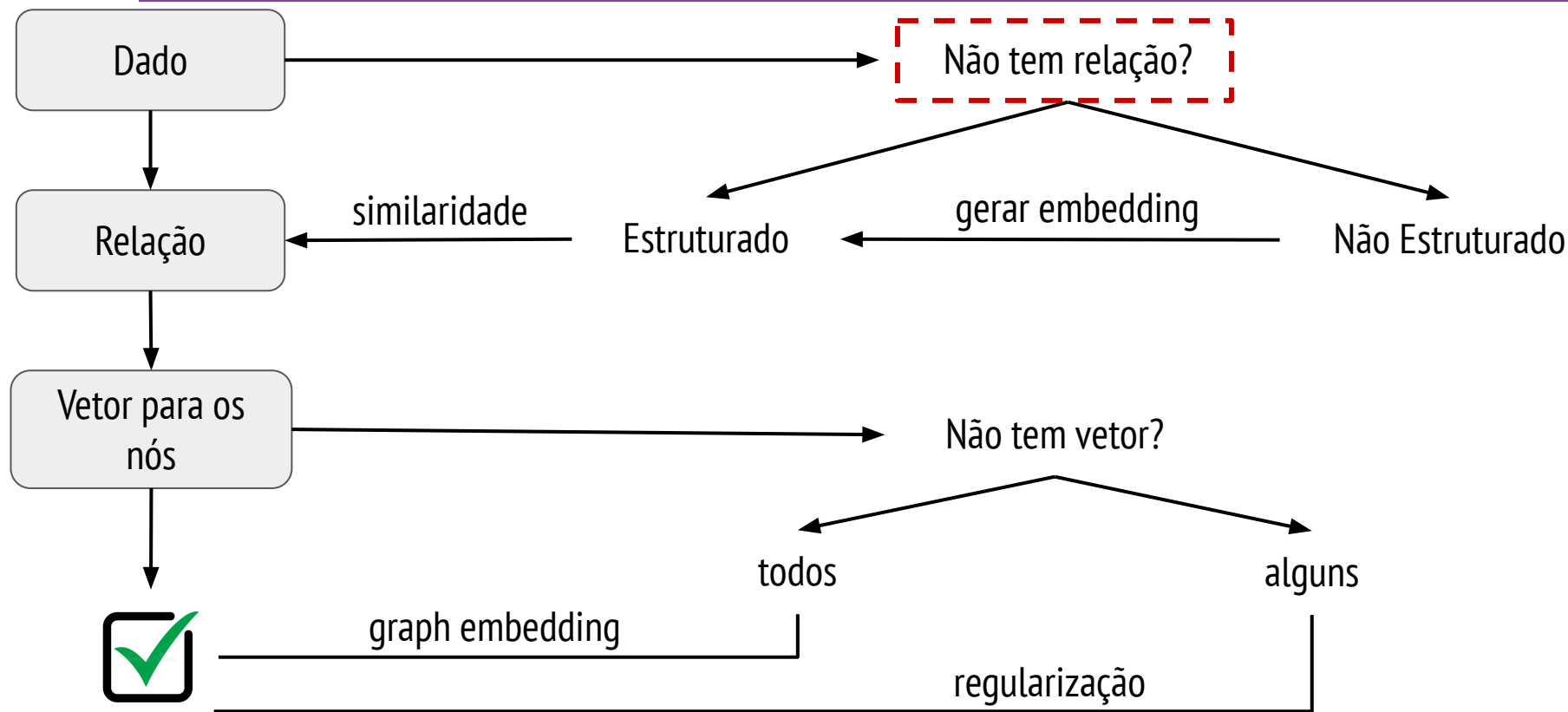
Grafos



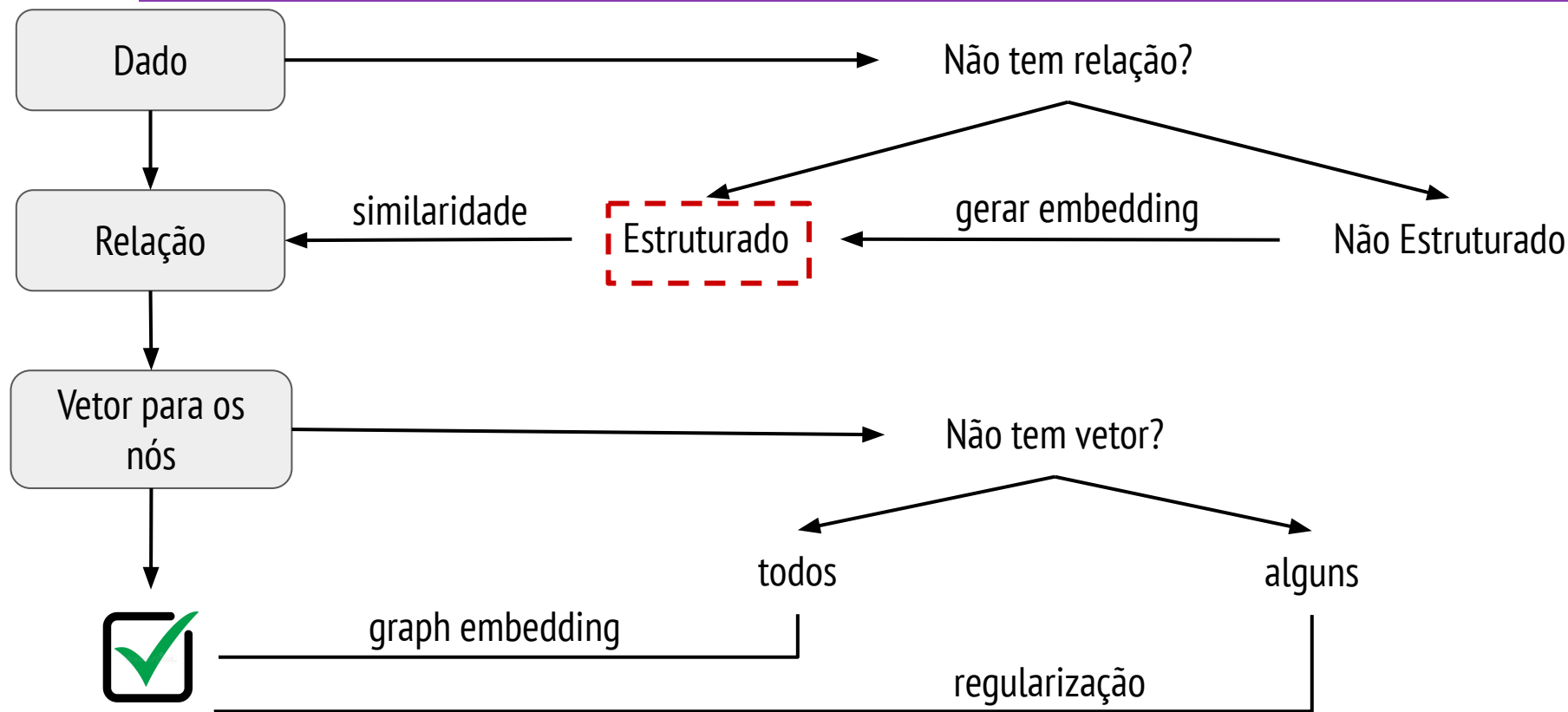
Grafos



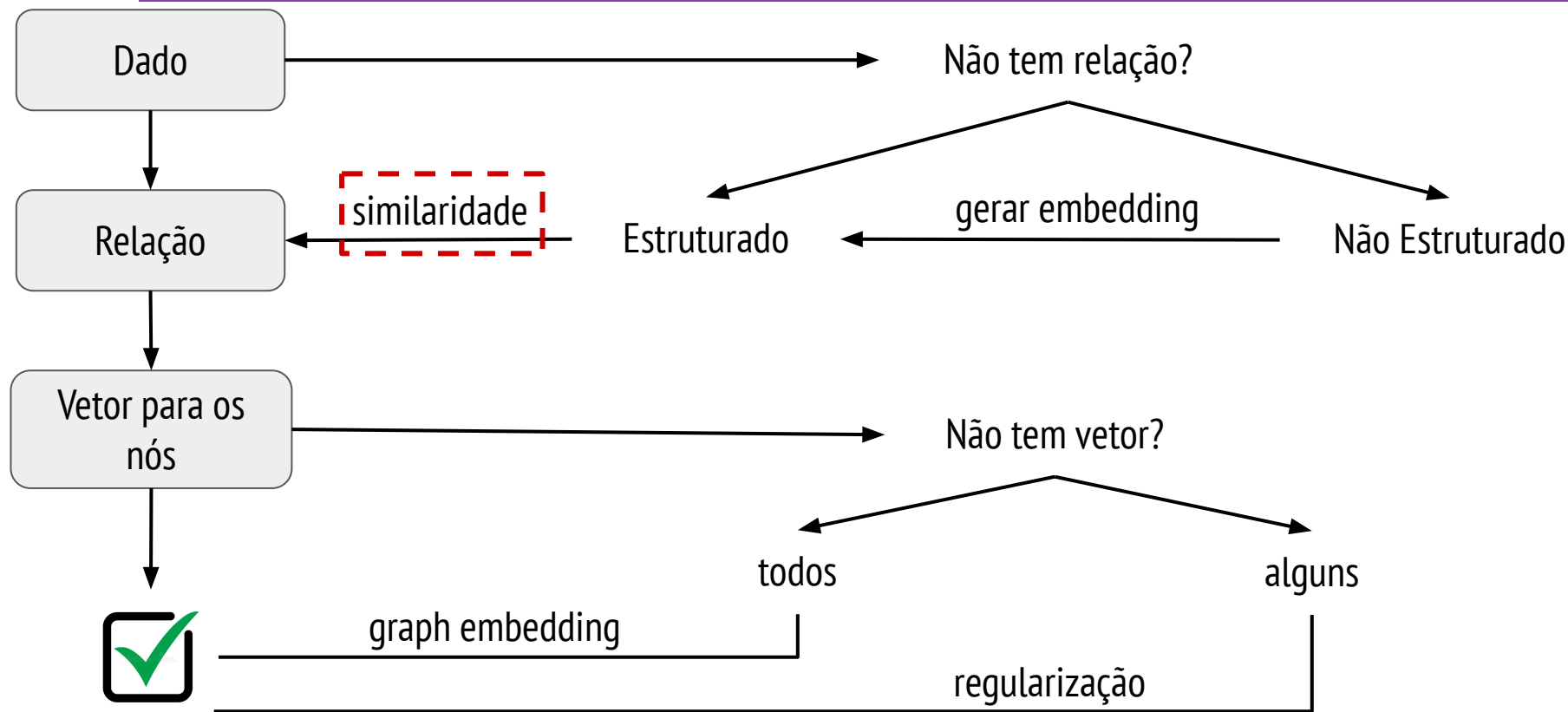
Grafos



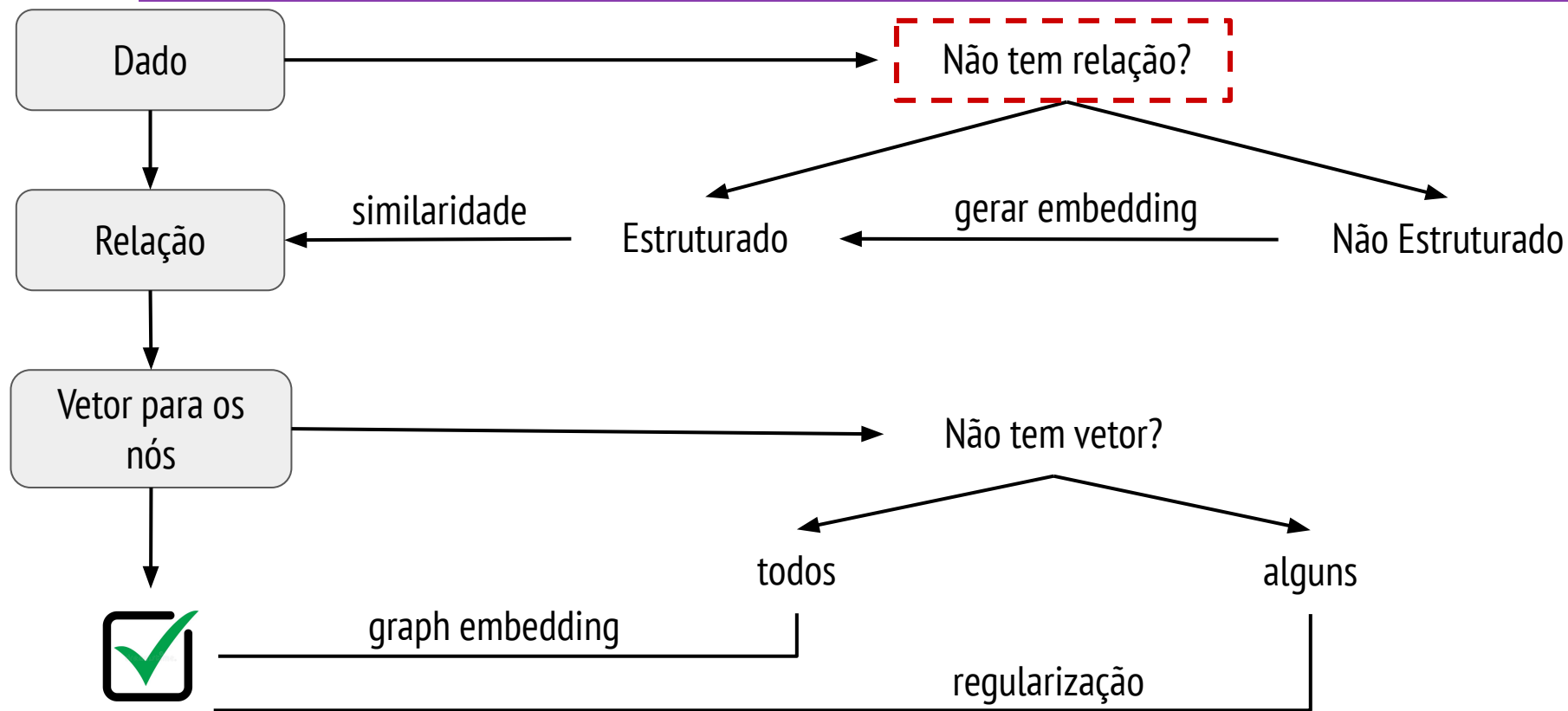
Grafos



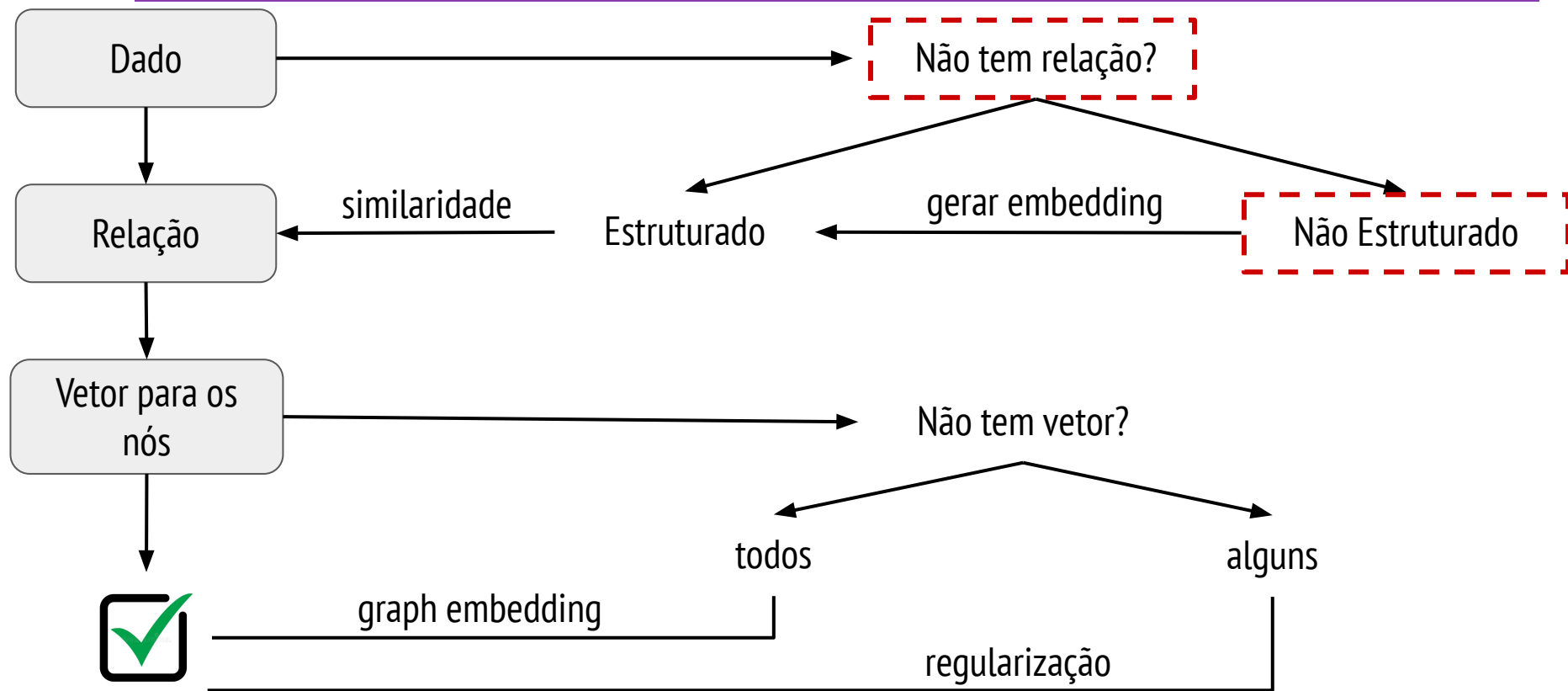
Grafos



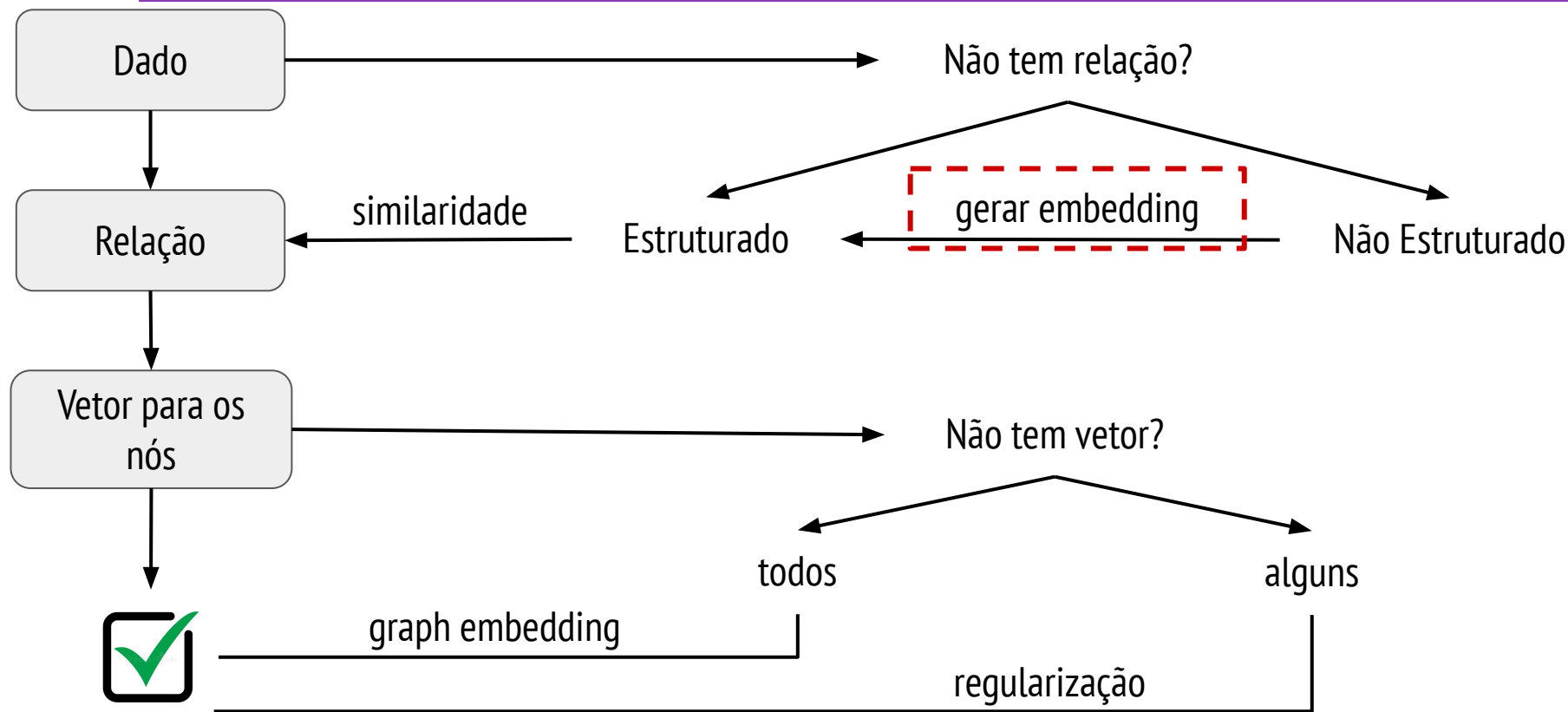
Grafos



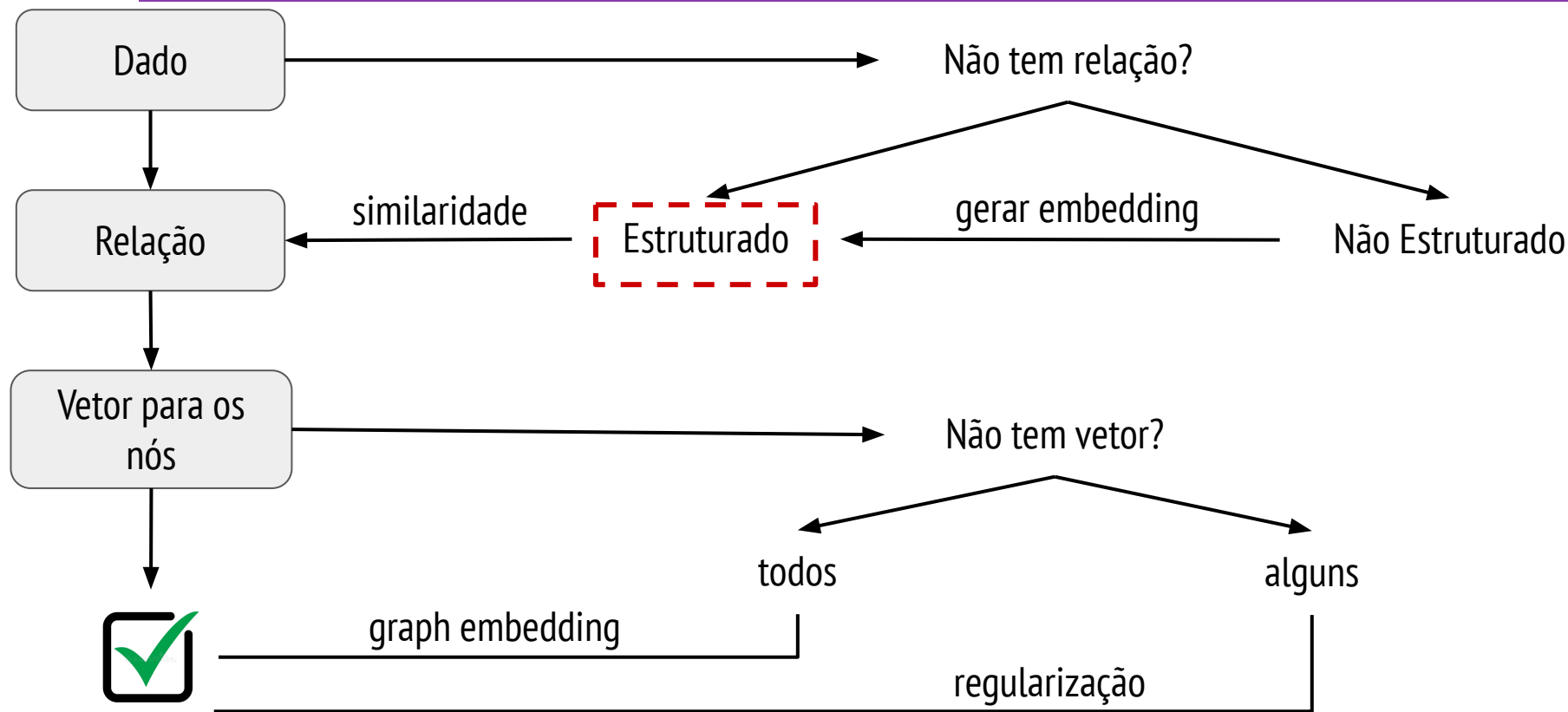
Grafos



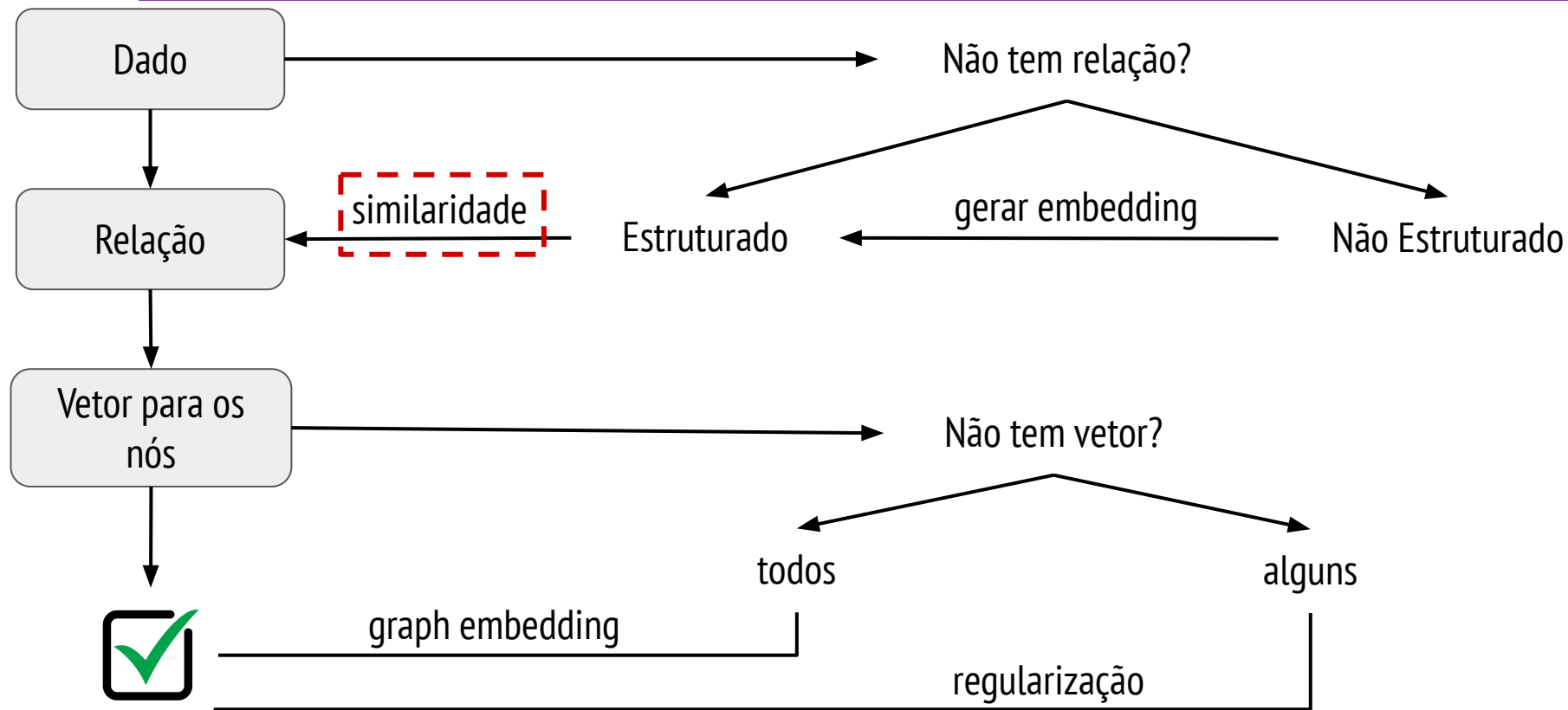
Grafos



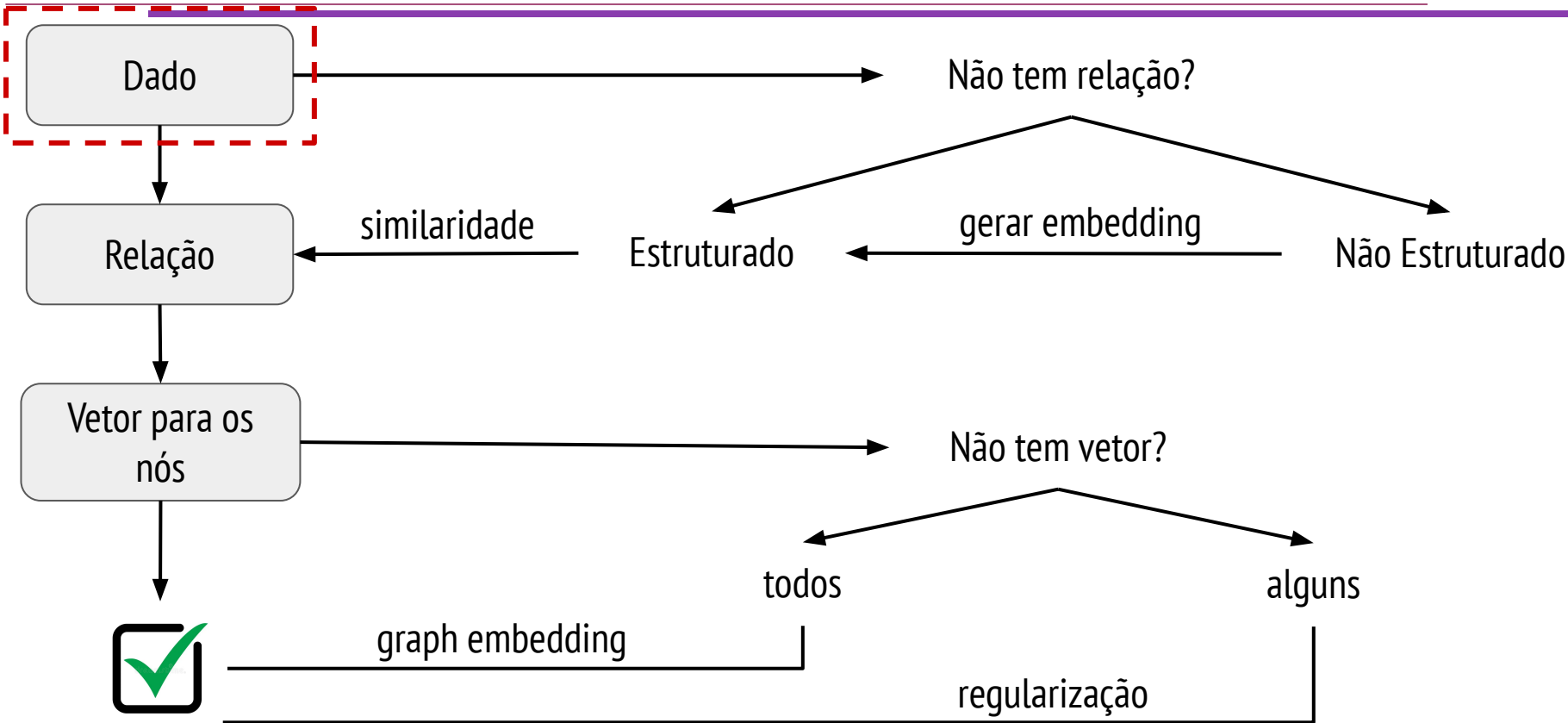
Grafos



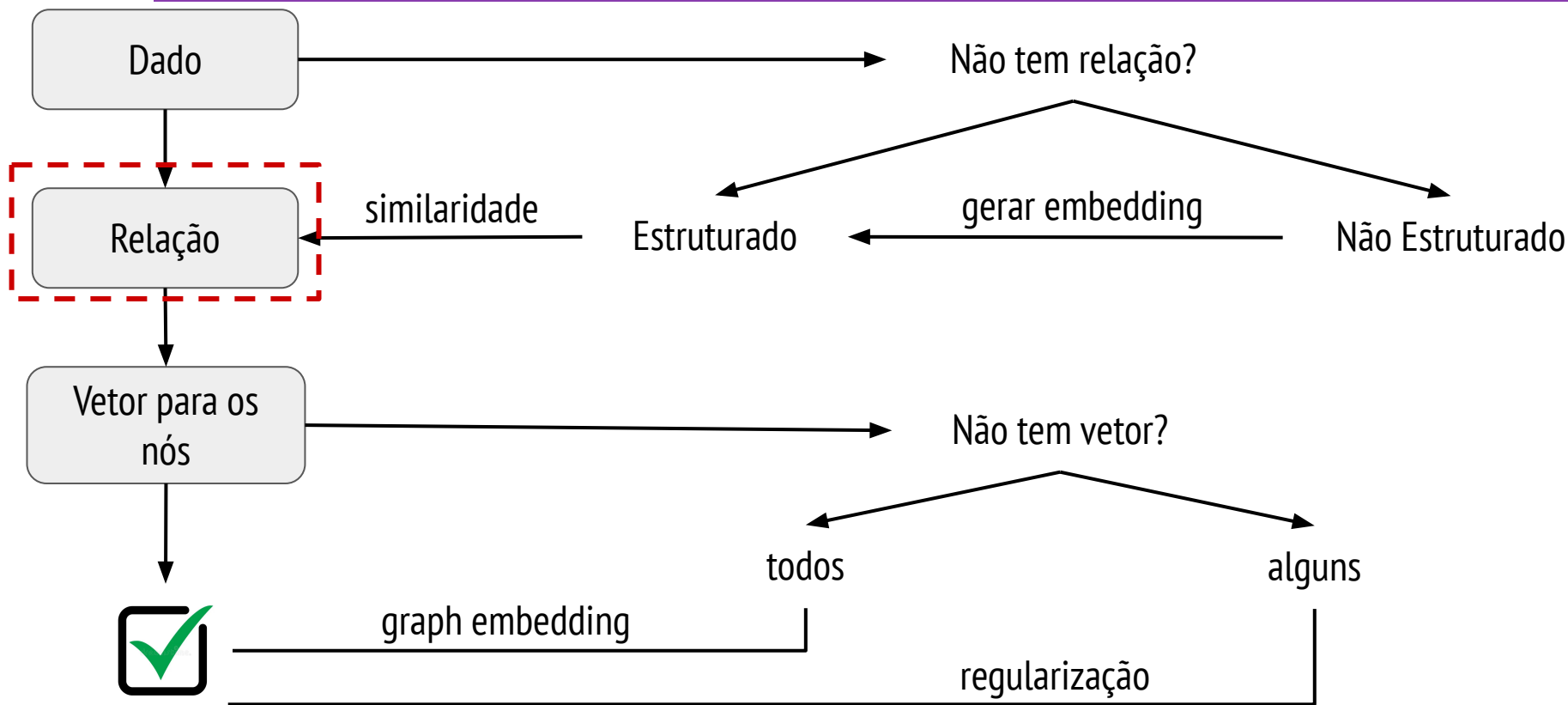
Grafos



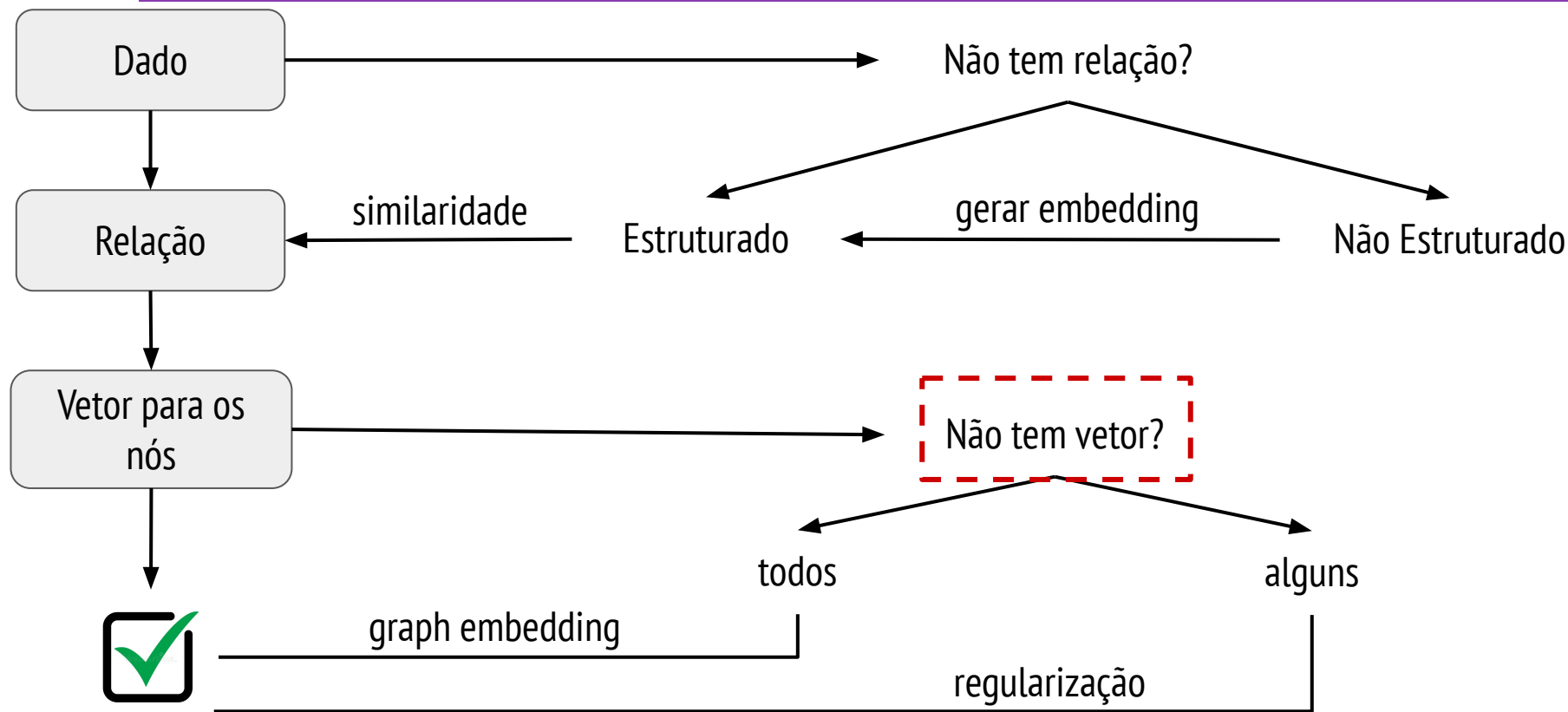
Grafos



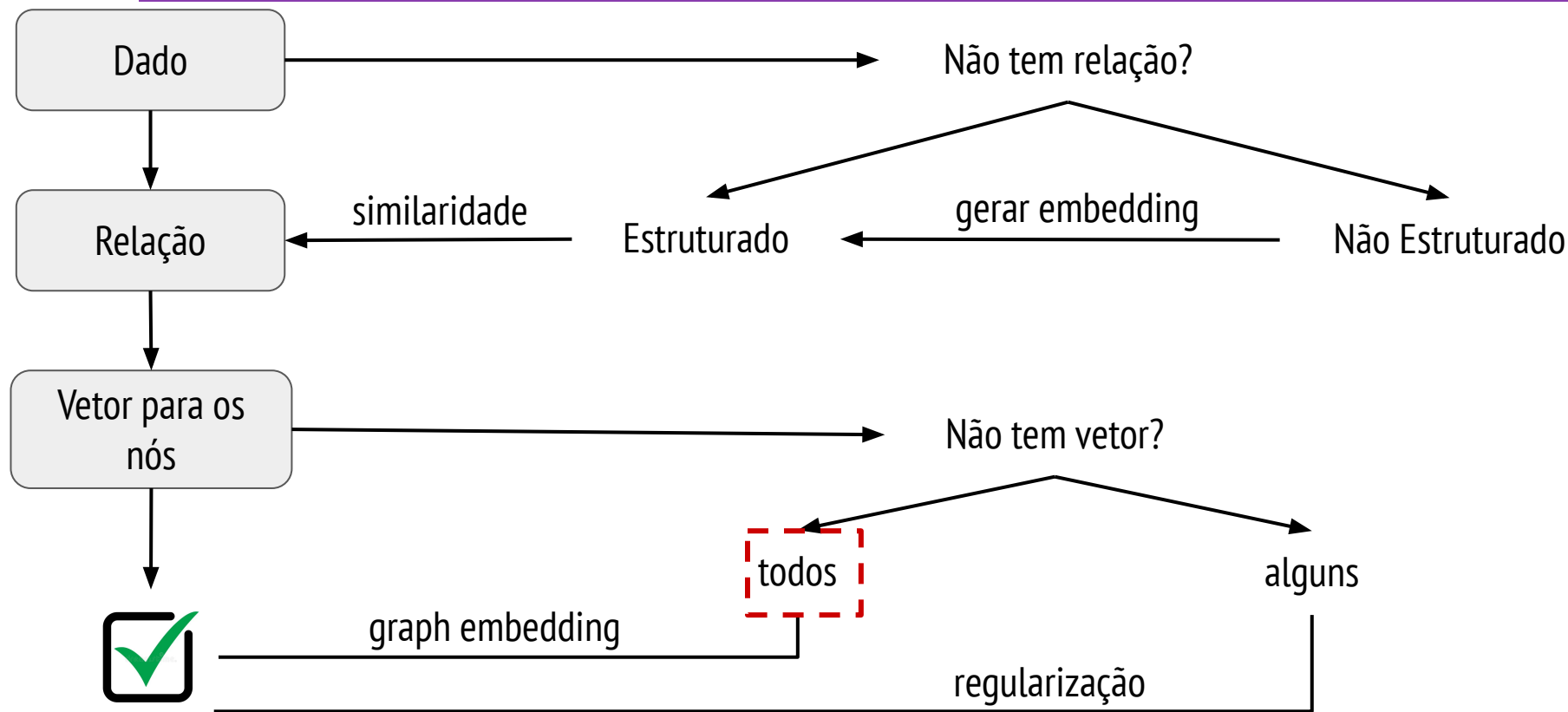
Grafos



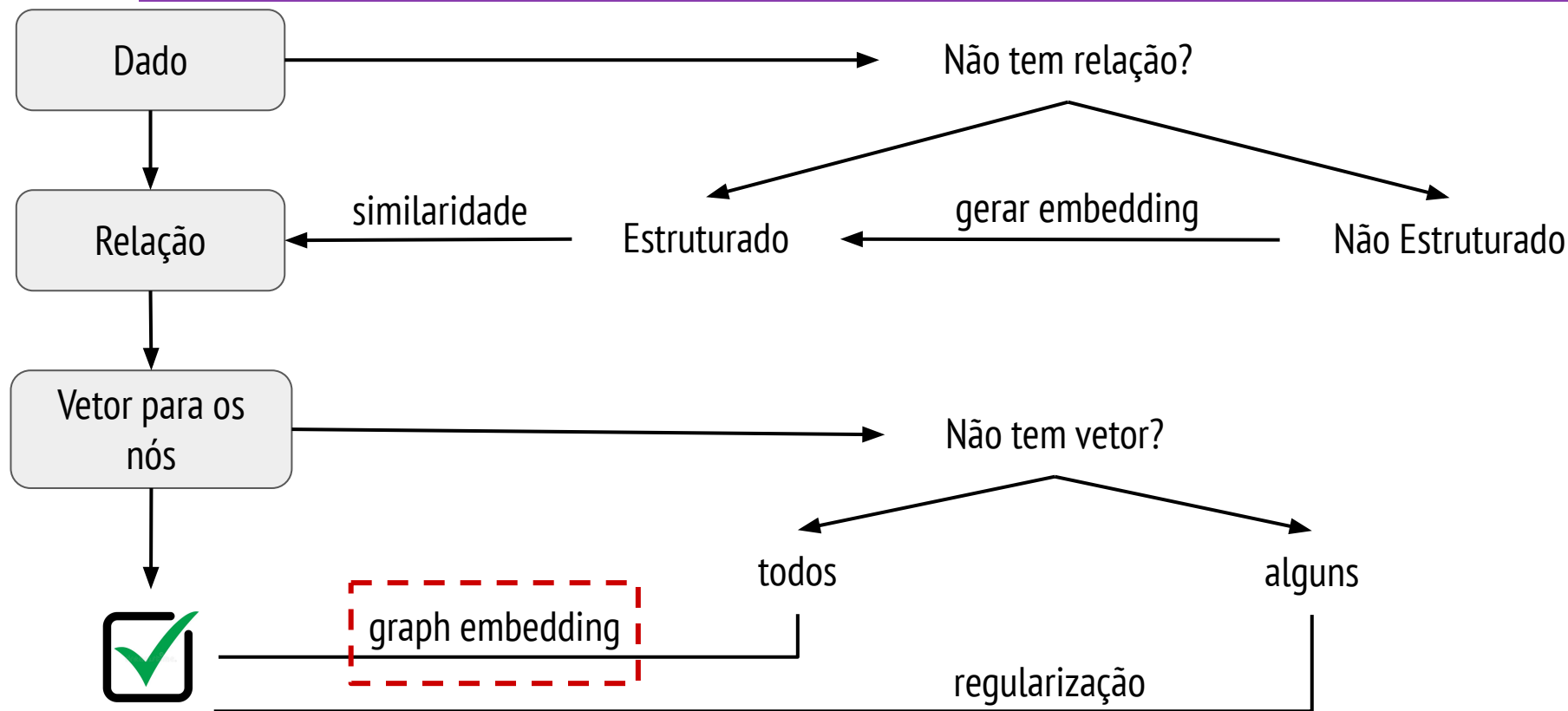
Grafos



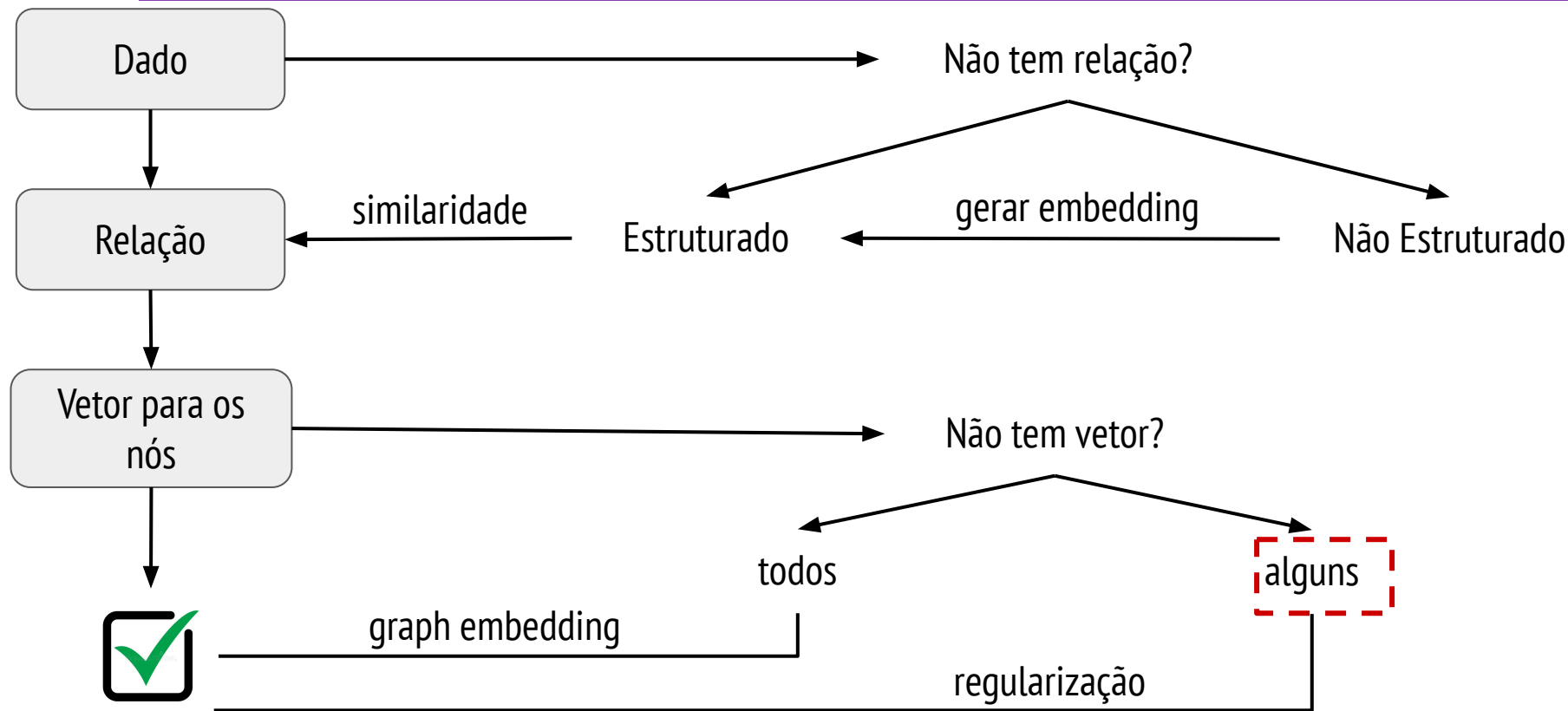
Grafos



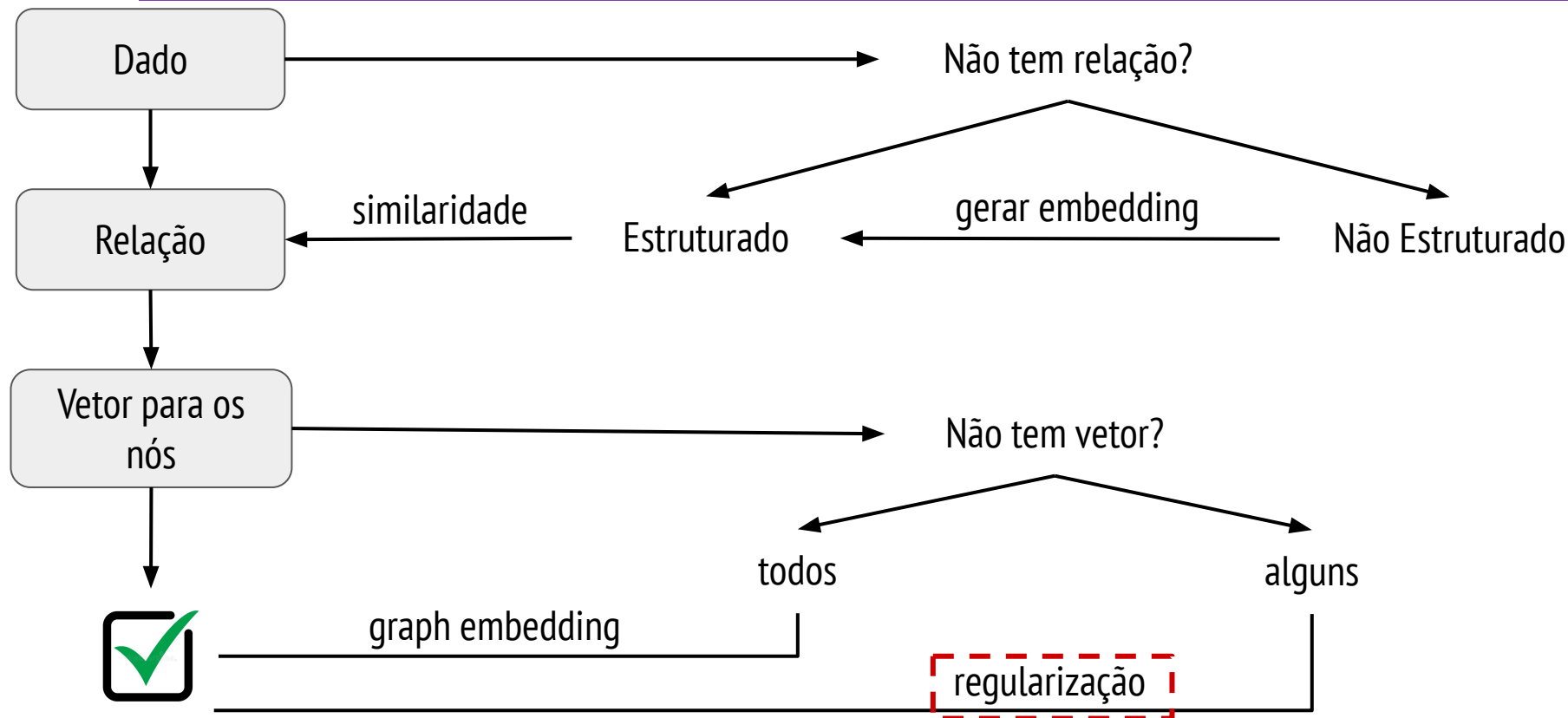
Grafos



Grafos



Grafos



Grafos

Vantagens

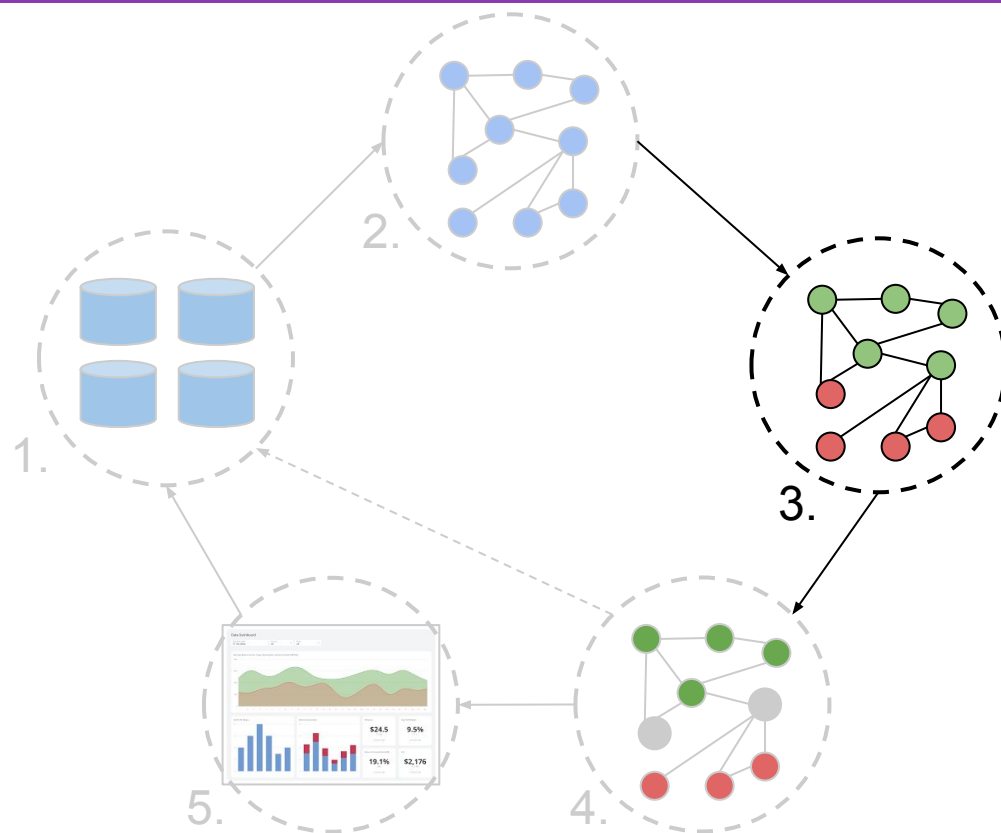
- Suporta dados multimodais
- Complementaridade das informações
- Versatilidade para domínios complexos
- Propagação de informação
- Construir características apenas com a topologia do grafo

Pontos de atenção

- grafo desconexo -> criar links
- nós com informações faltantes -> regularização (embeddings propagation)
- over smoothing - embeddings muitos similares
- cenário indutivo

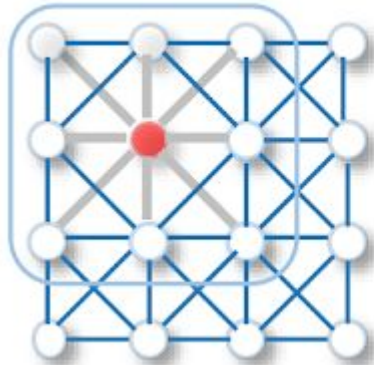
Mineração de dados

1. Dados
2. Pré-processamento
3. Aprendizado de padrões
4. Pós-processamento
5. Uso do conhecimento obtido

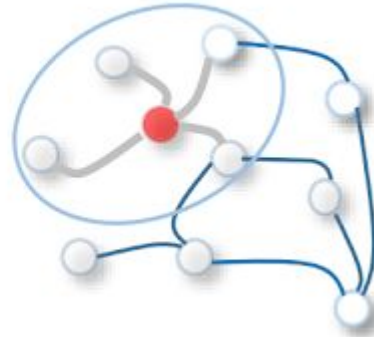


Justificativa para GNNs

- Avanços das redes neurais em diferentes cenários de dados euclidianos
- E para dados não euclidianos, como os **grafos**?
- Propuseram as Redes Neurais para Grafos, ou as *Graph Neural Networks*



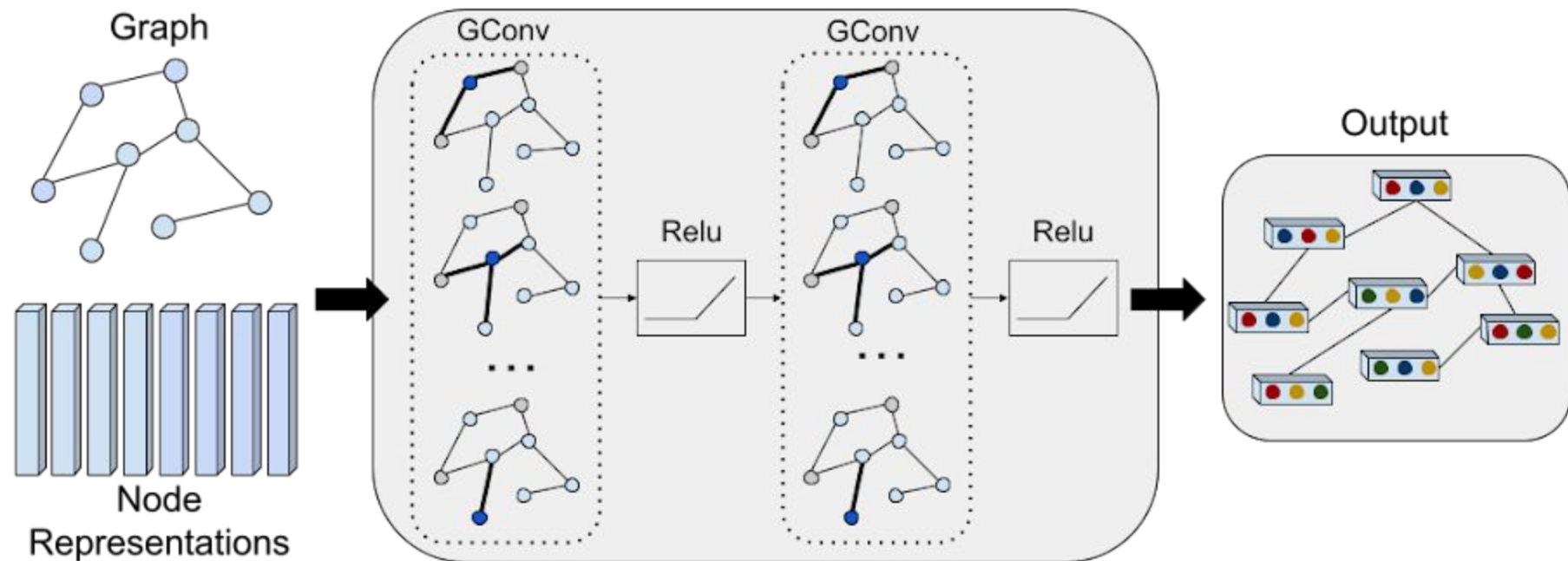
Rede Neural para imagem



Rede Neural para Grafo

Fonte: Wu, Zonghan, et al. "A comprehensive survey on graph neural networks." IEEE transactions on neural networks and learning systems 32.1 (2020):

Graph Neural Networks



Graph Neural Networks

- GNN: $g(\mathbf{O}, \mathbf{A}; \mathbf{W})$
- \mathbf{O} : Representações dos objetos
- \mathbf{A} : Matriz de adjacência
- \mathbf{W} : Pesos da rede neural
- Uma GNN genérica pode ser definida com dois passos:
 - **Agregação**: agrega informações dos vizinhos de uma nó
 - **Combinação**: combina a representação aprendida pelo neurônio com a representação agregada

Graph Neural Networks

$g(\mathbf{O}, \mathbf{A}; \mathbf{W})$



$$\mathbf{H}^{(l+1)} = g(\mathbf{H}^{(l)}, \mathbf{A}; \mathbf{W}^{(l)})$$

$$\boxed{\mathbf{H}^{(l+1)}} = g(\boxed{\mathbf{H}^{(l)}}, \boxed{\mathbf{A}}, \boxed{\mathbf{W}^{(l)}})$$

representações de saída

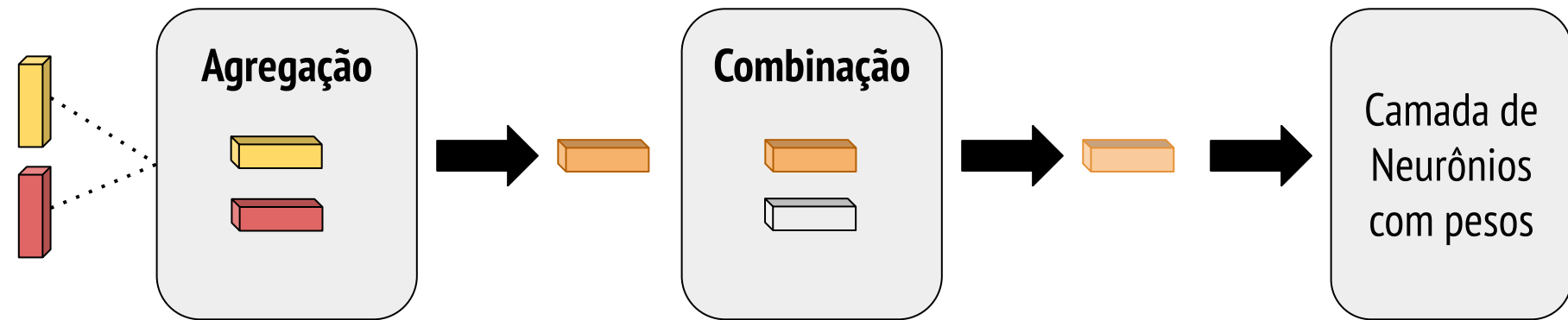
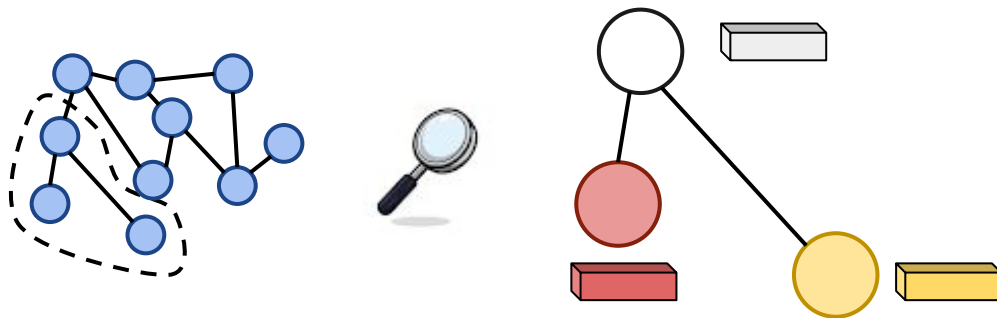
Pesos da rede neural

representações de entrada

$$\mathbf{O} == \mathbf{H}^{(0)}$$

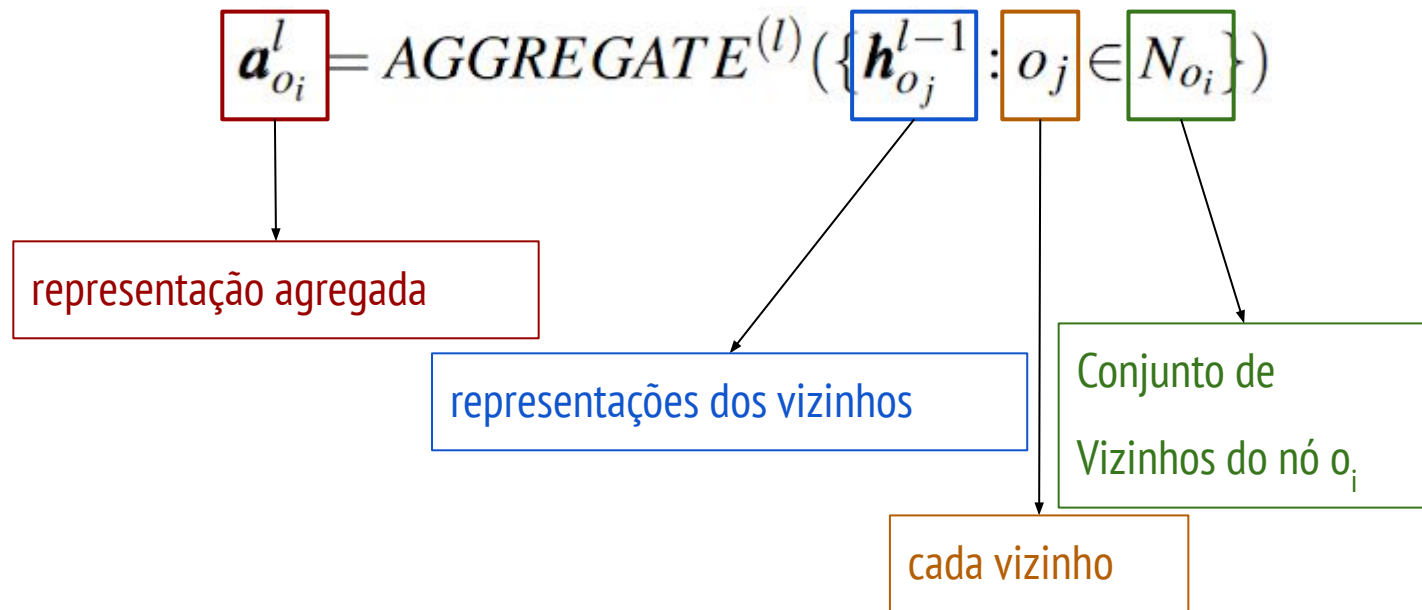
matriz de adjacência

Graph Neural Networks



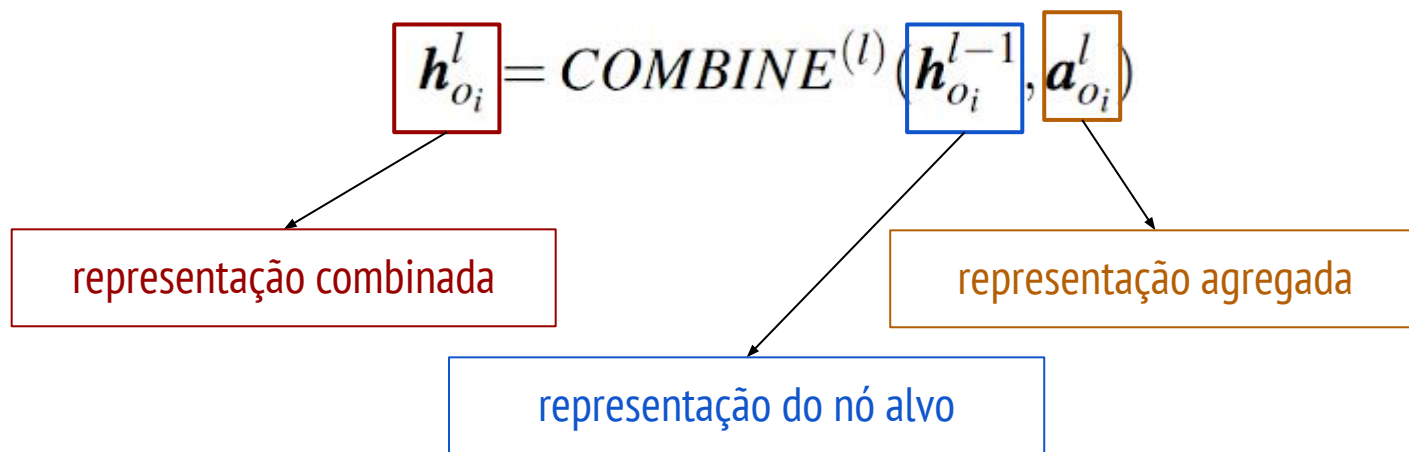
Graph Neural Networks

$$\mathbf{a}_{o_i}^l = \text{AGGREGATE}^{(l)}(\{\mathbf{h}_{o_j}^{l-1} : o_j \in N_{o_i}\})$$

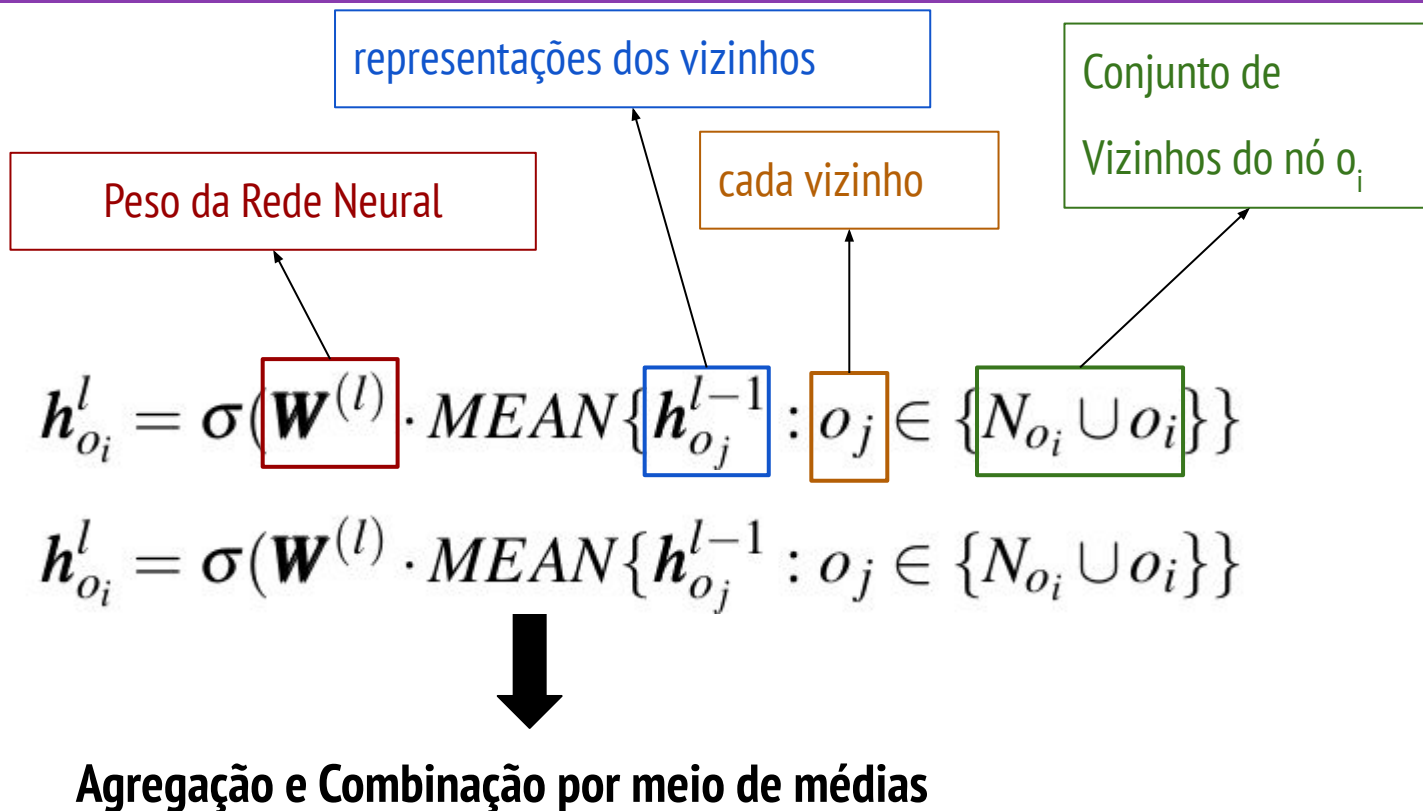


Graph Neural Networks

$$\mathbf{h}_{o_i}^l = \text{COMBINE}^{(l)}(\mathbf{h}_{o_i}^{l-1}, \mathbf{a}_{o_i}^l)$$



Graph Convolutional Networks



GAT e GraphSAGE

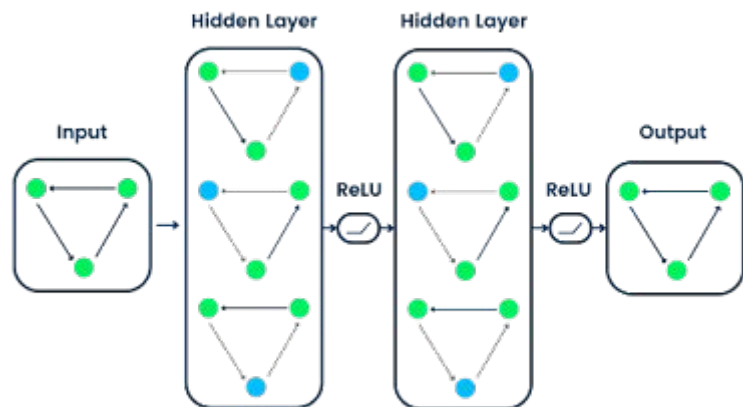
- **Graph Attention Network (GAT)**

- Na GCN, a importância dos nós vizinhos em é a mesma
- Ruídos podem influenciar negativamente no processo de aprendizagem
- A GAT foca as arestas mais importantes através do mecanismo de atenção
- A GAT tem atenção às principais relações do grafo, melhorando a agregação das informações

- **GraphSAGE**

- Novo método de combinação e diferentes agregadores.
- GraphSAGE realiza uma amostragem nos vizinhos
- Reduz o tempo e a complexidade da memória

Graph Autoencoder (GNN não supervisionada)



GNN vista anteriormente

$$GAE = \begin{cases} \text{Encoder: } \mathbf{H}^{(L)} = g(\mathbf{O}, \mathbf{A}; \mathbf{W}) \\ \text{Decoder: } \hat{\mathbf{A}} = \sigma(\mathbf{H}^{(L)} \cdot \mathbf{H}^{(L)\top}) \end{cases}$$

Matriz de adjacência
reconstruída

$\mathbf{H}^{(L)}$ transposto

Função de perda: Diferença entre \mathbf{A} e $\hat{\mathbf{A}}$

Vantagens e Desvantagens da GNN

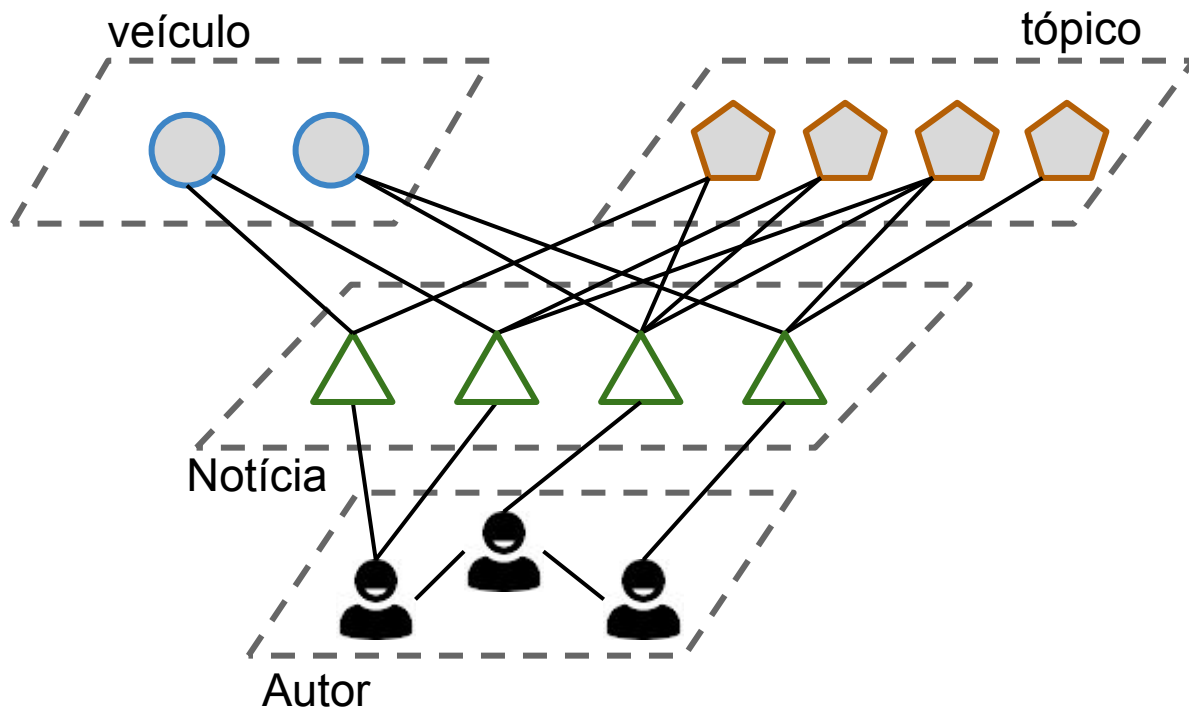
- **Vantagens**

- Resolve **diferentes tarefas** de forma **end-to-end**
 - classificação de nós
 - classificação de aresta
 - predição de aresta
 - classificação de grafos
- Adaptabilidade para diferentes tipos de dados e modelagens de grafo
- Resultados estado-da-arte

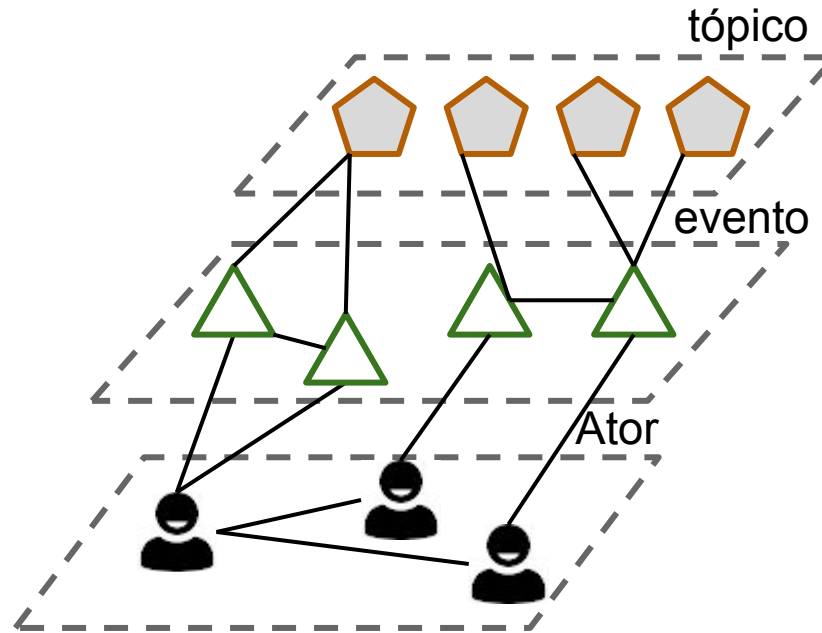
- **Desvantagens**

- Representações convergem para mesmo local com muitas camadas
- Black-box (explicabilidade)

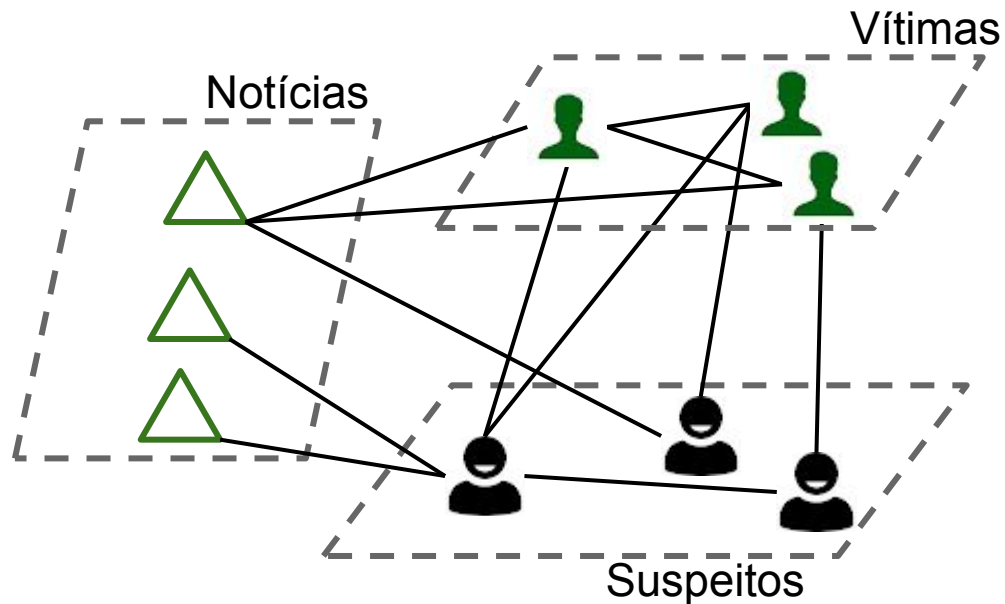
Detecção Notícias Falsas



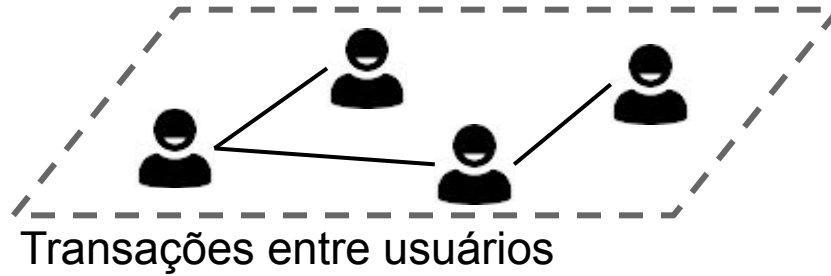
Detecção de Eventos



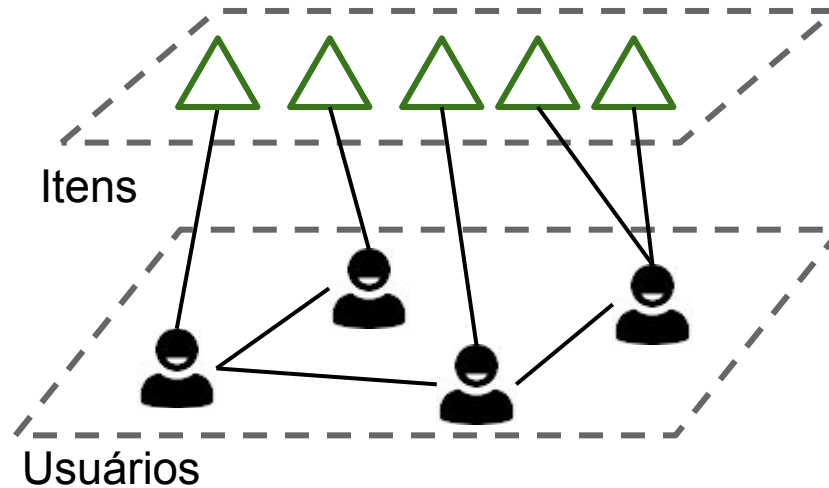
Detecção de Crimes



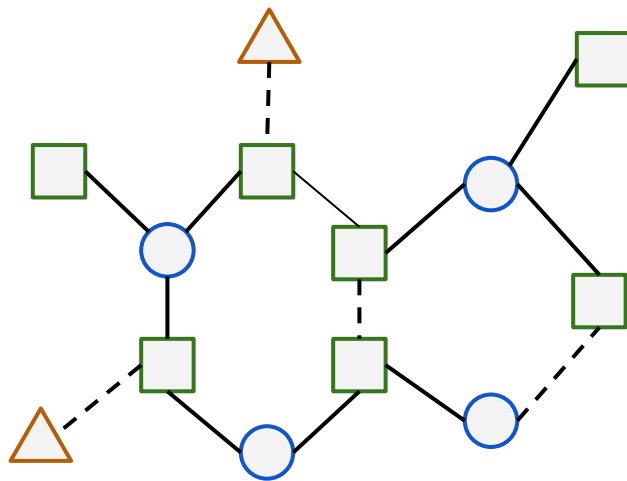
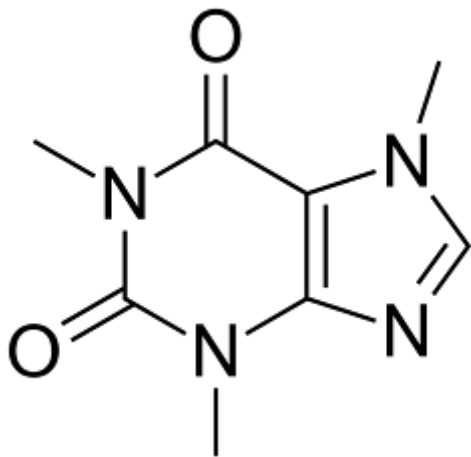
Detecção de Fraude



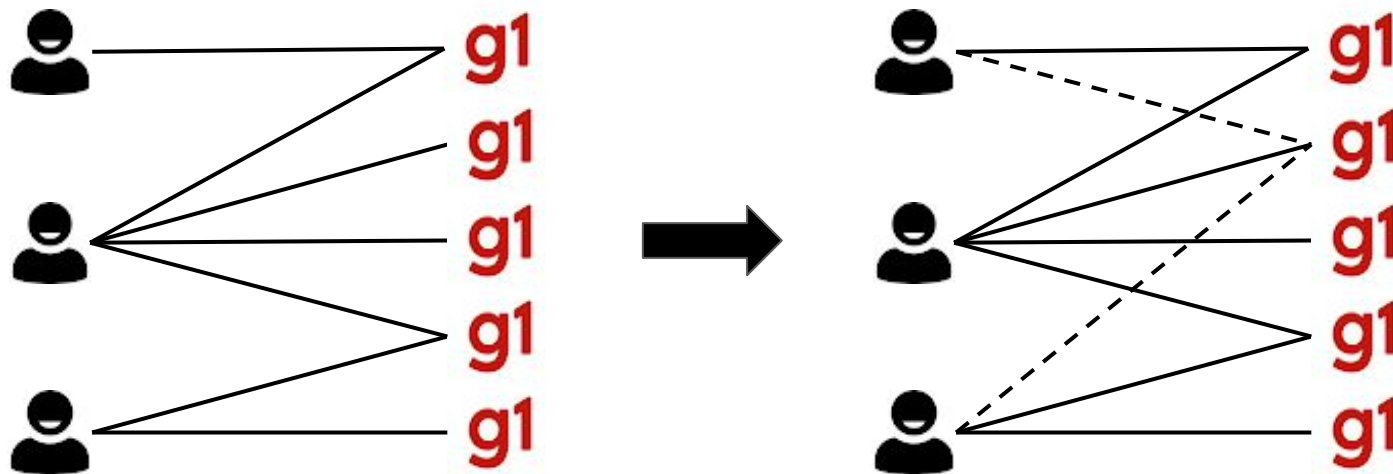
Recomendação



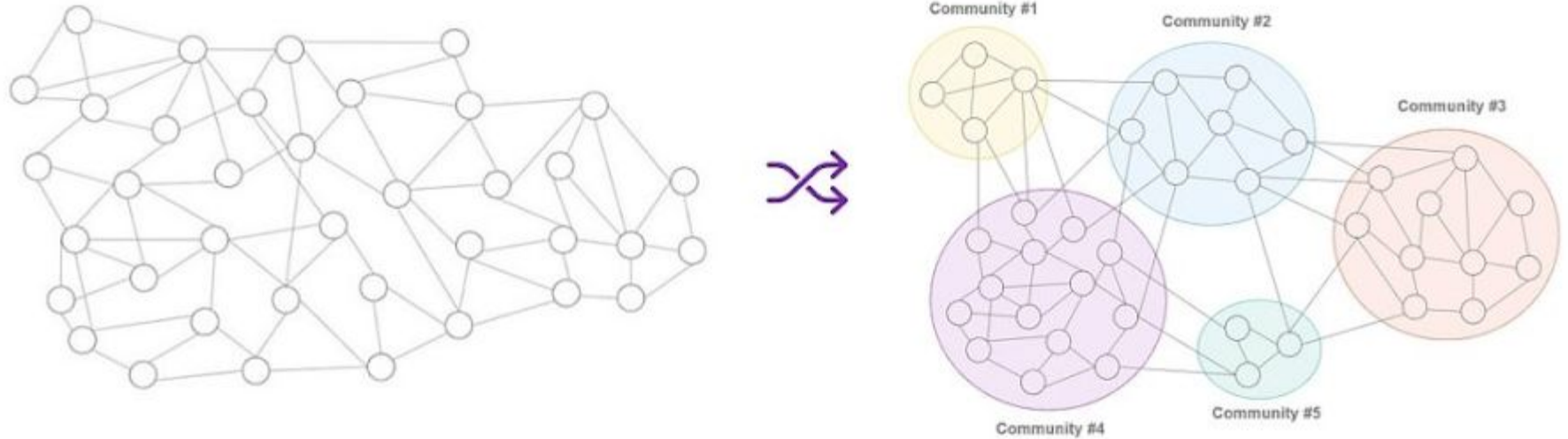
Classificação de Grafos



Recomendação de Notícias



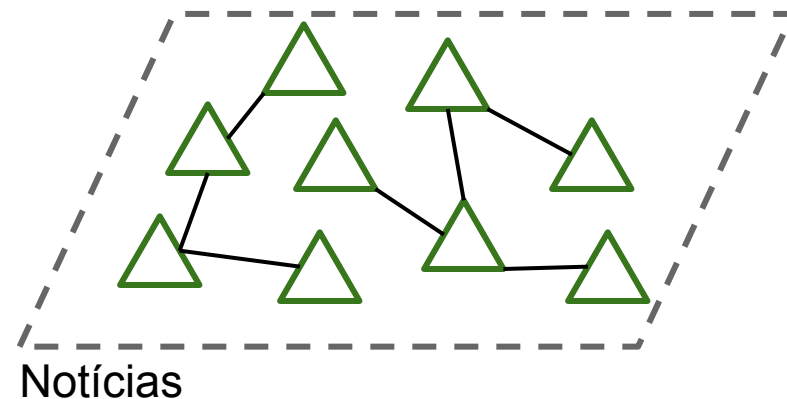
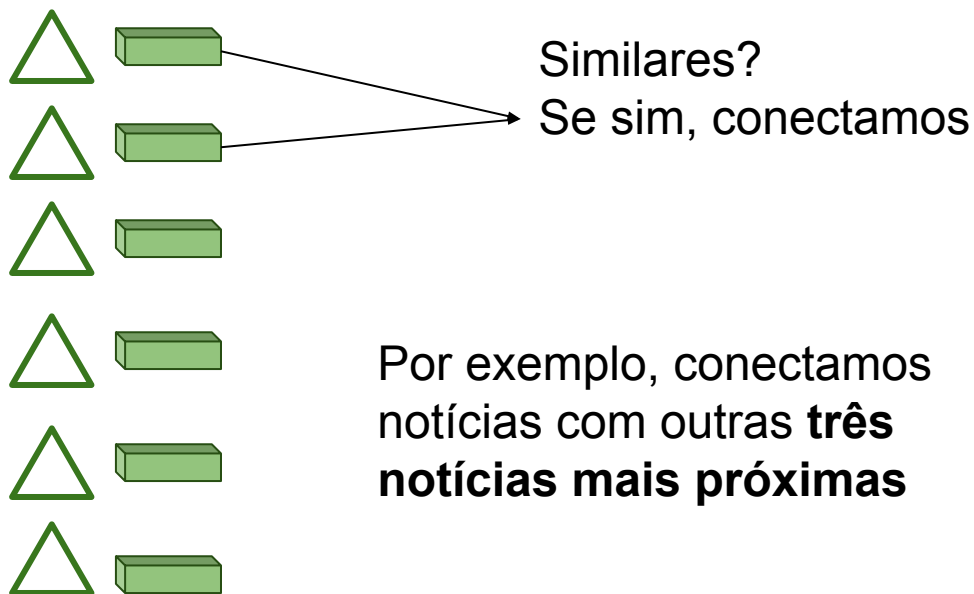
Detecção de Comunidades



Detecção de Notícia Falsa: Dataset

- Notícias Falsas sobre política
- Notícias Falsas de 2019
- Classe Real e Falsa
- Anotação por humanos
- Notícias: 2064
- Notícias Falsas: 1044
- Notícias Reais: 1020

Detecção de Notícia Falsa: Grafo

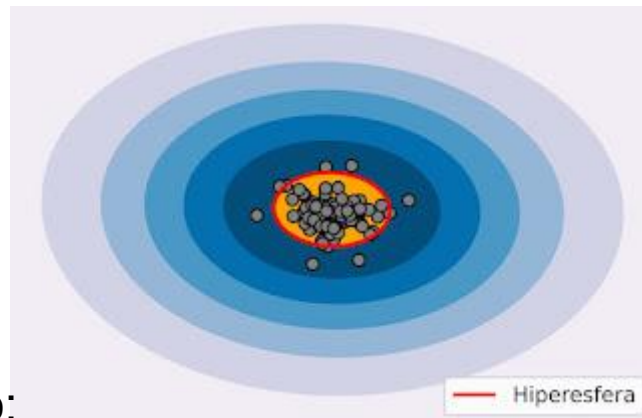


Detecção de Notícia Falsa

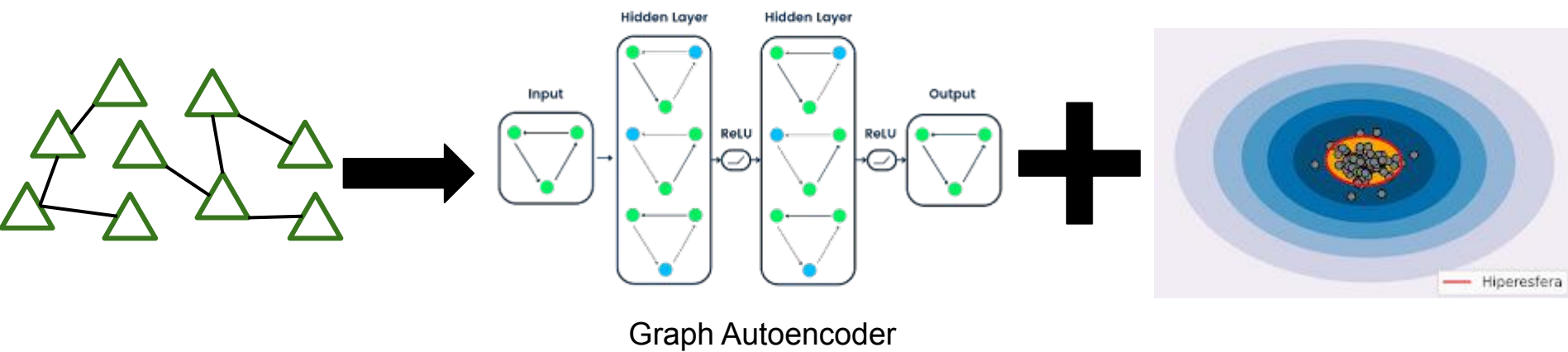
- GNN para duas classes?
- No mundo real, teríamos que **rotular tanto notícias falsas quanto reais**.
- Rotulação é **caro!!!!**
- Será que podemos utilizar algum **aprendizado que só precisamos rotular as notícias falsas** que são nosso foco?
- **SIM!!!**

One-Class Learning (OCL)

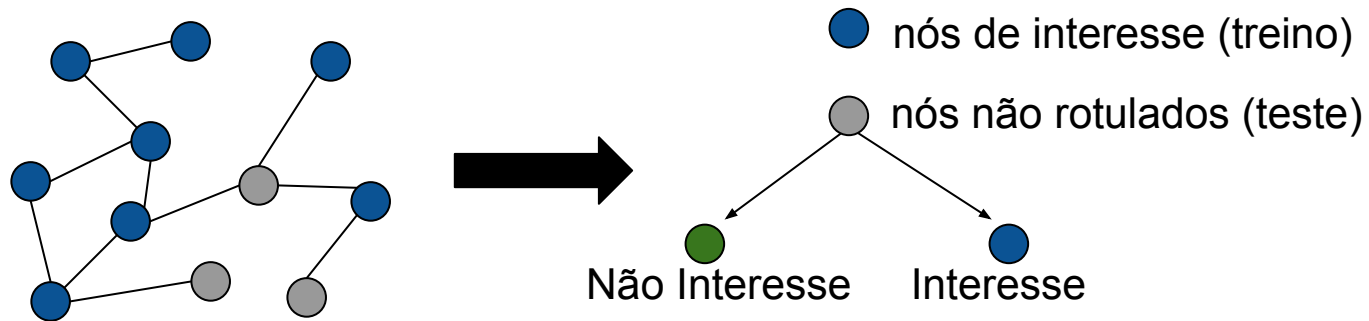
- No OCL o treinamento é apenas em amostras de **uma** classe (classe de interesse). **Ausência de contra-exemplos.**
- OCL **reduz esforços** de rotulação e **não exige cobertura** abrangente da classe de não interesse
- Aplicações de **domínio aberto** ou quando há **interesse em uma única classe** do problema
- Após treinar, funciona com um classificador binário:
 - Classe de interesse
 - Não classe de interesse



Pipeline

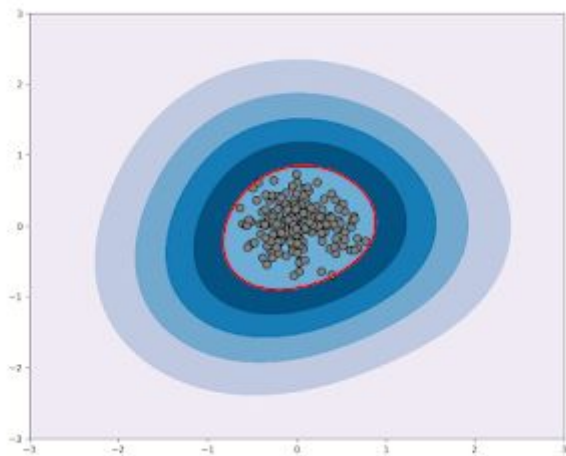


Tarefa

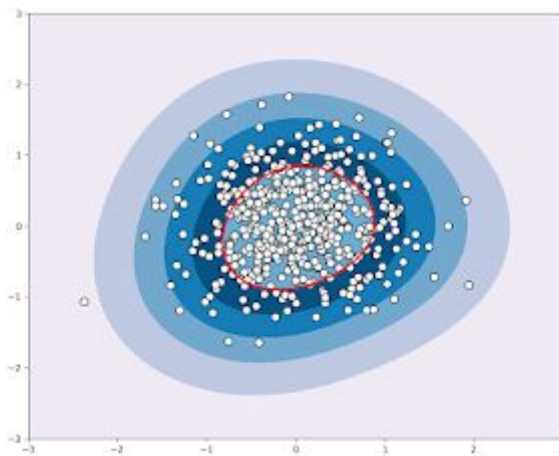


One-Class Learning

Treino



Teste



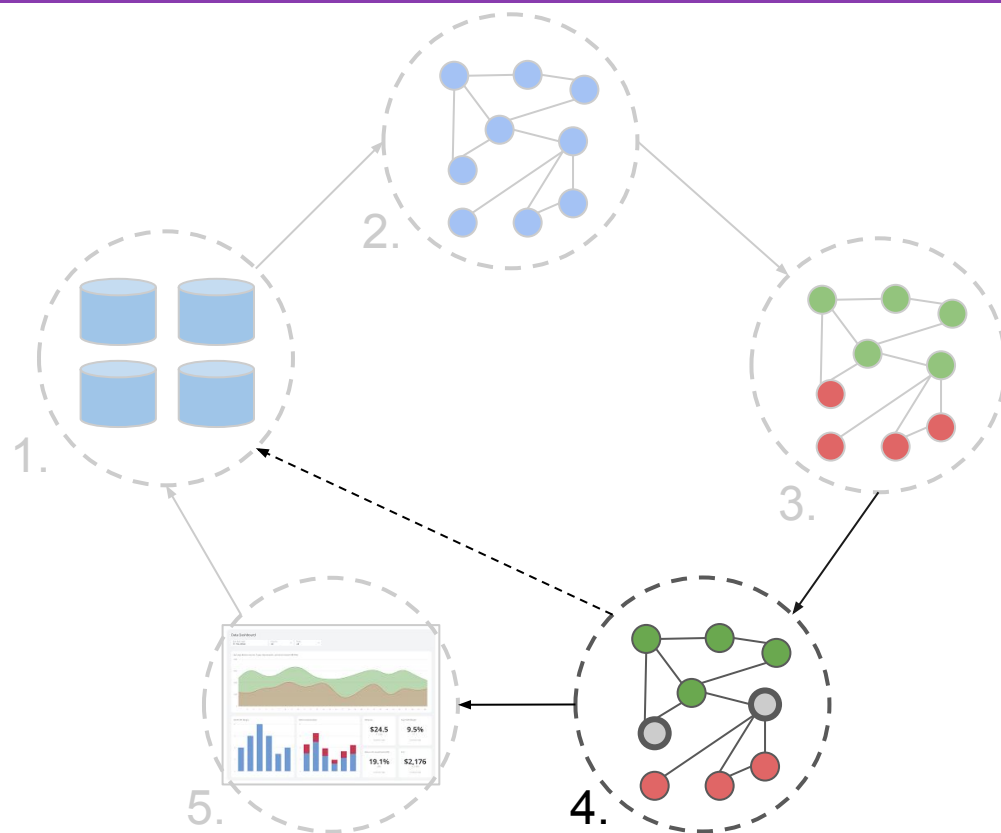
- notícias falsas
- notícias não rotuladas

Vamos praticar?

Prática - Fake News

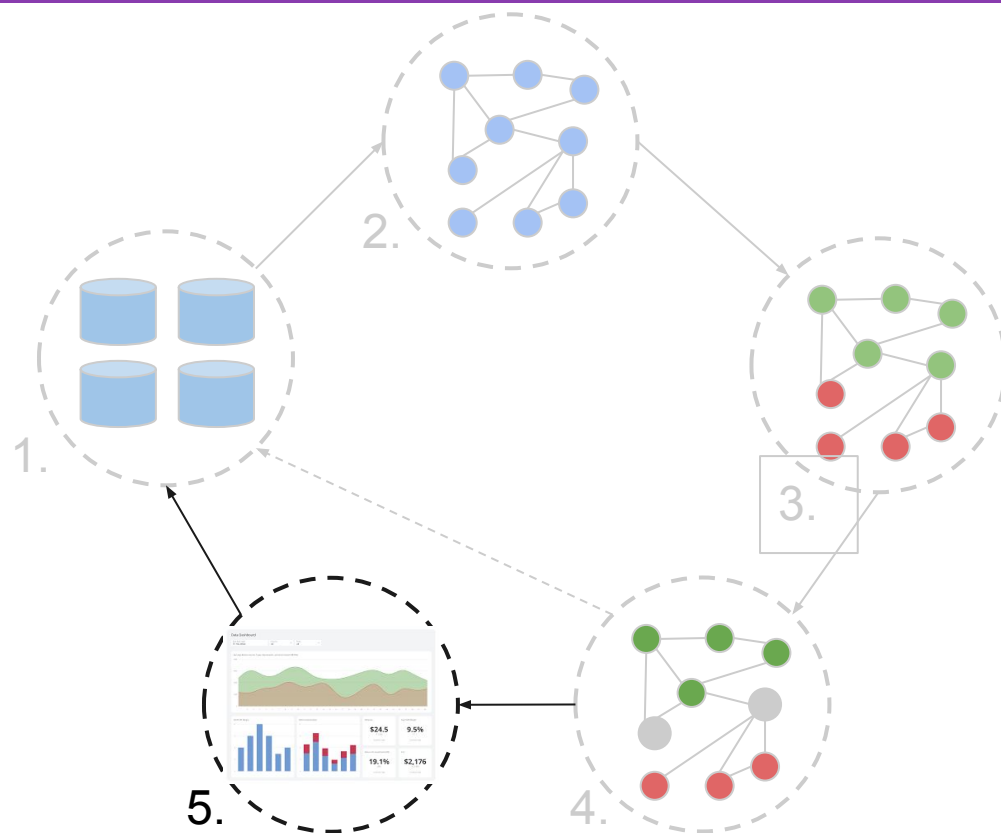
Mineração de dados

1. Dados
2. Pré-processamento
3. Aprendizado de padrões
4. Pós-processamento
5. Uso do conhecimento



Mineração de dados

1. Dados
2. Pré-processamento
3. Aprendizado de padrões
4. Pós-processamento
5. Uso do conhecimento





LoG Conference

Obrigado!

Agradecimento especial: **Angelo Cesar Mendes da Silva**

marcosgolo@usp.br