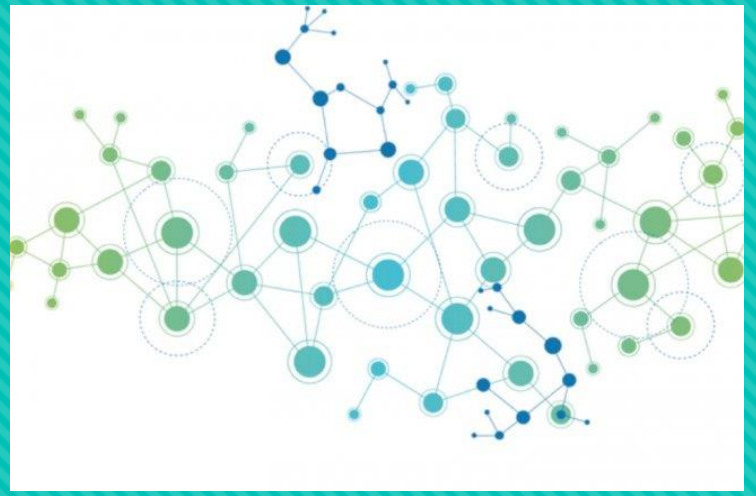
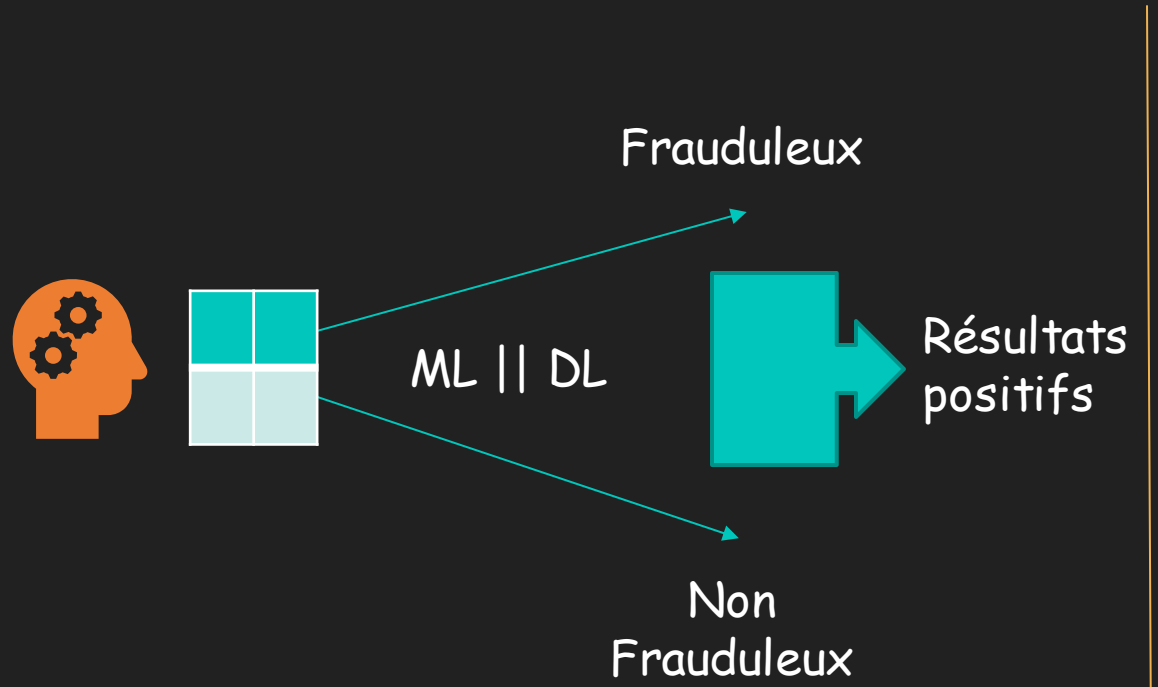


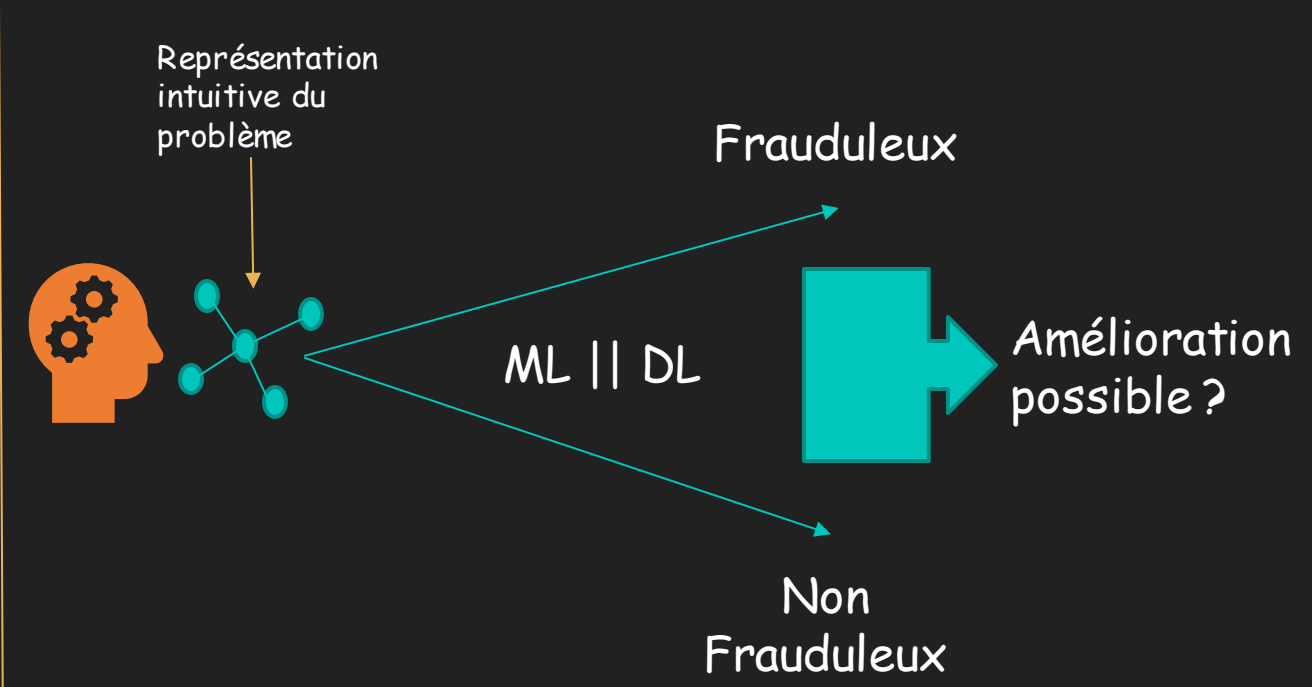
Fraude 2:
Fraud detection based on Graphs

Roméo Leylekian

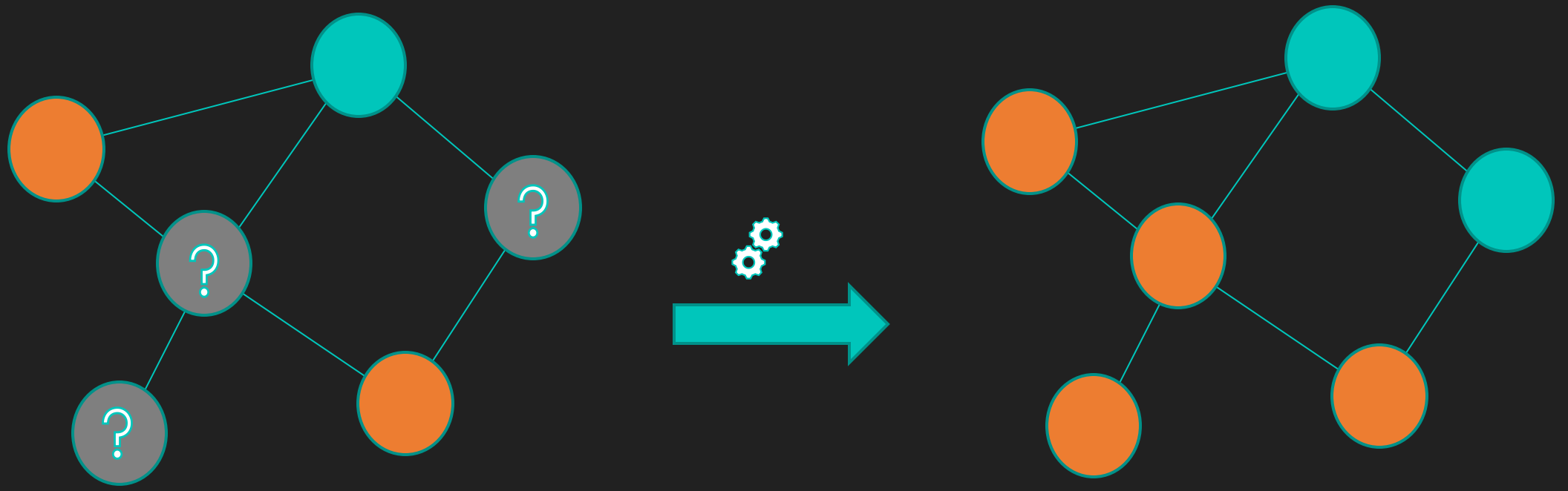
Nicolas Nerrienet

Contexte : IA et fraudes

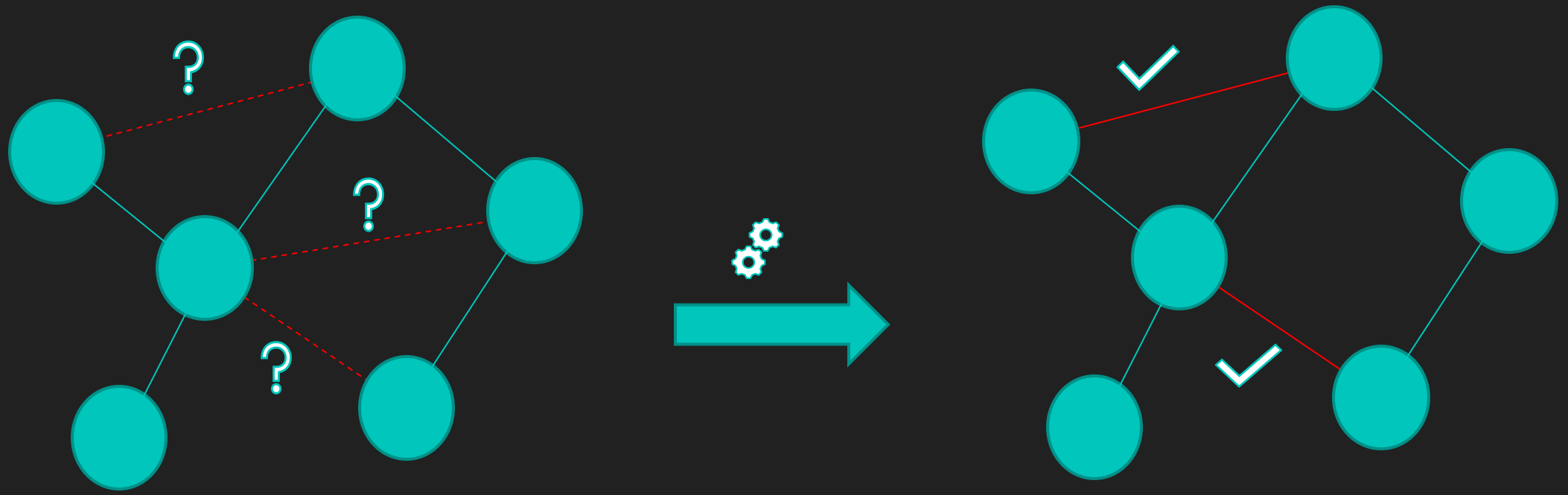




Tâches usuelles de ML sur graphes

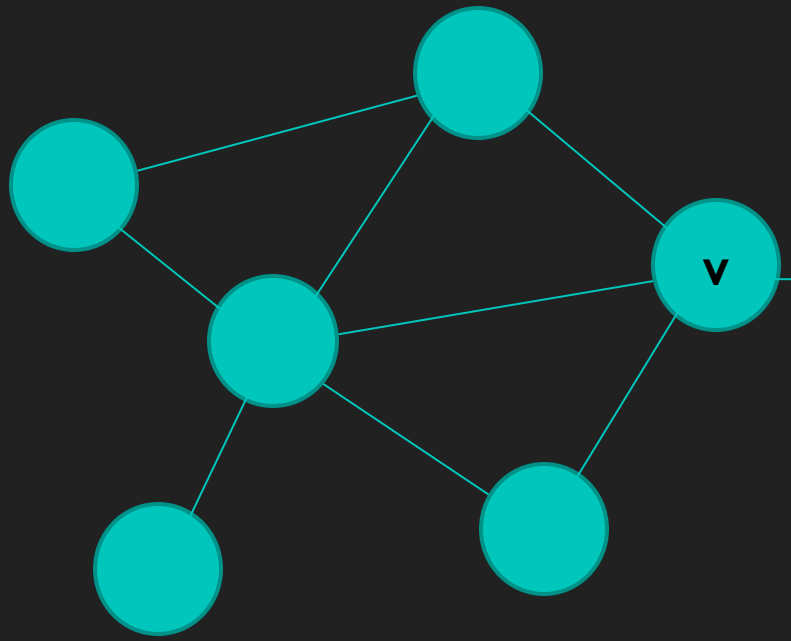


Node Classification



Prédiction de liens

Node2Vec

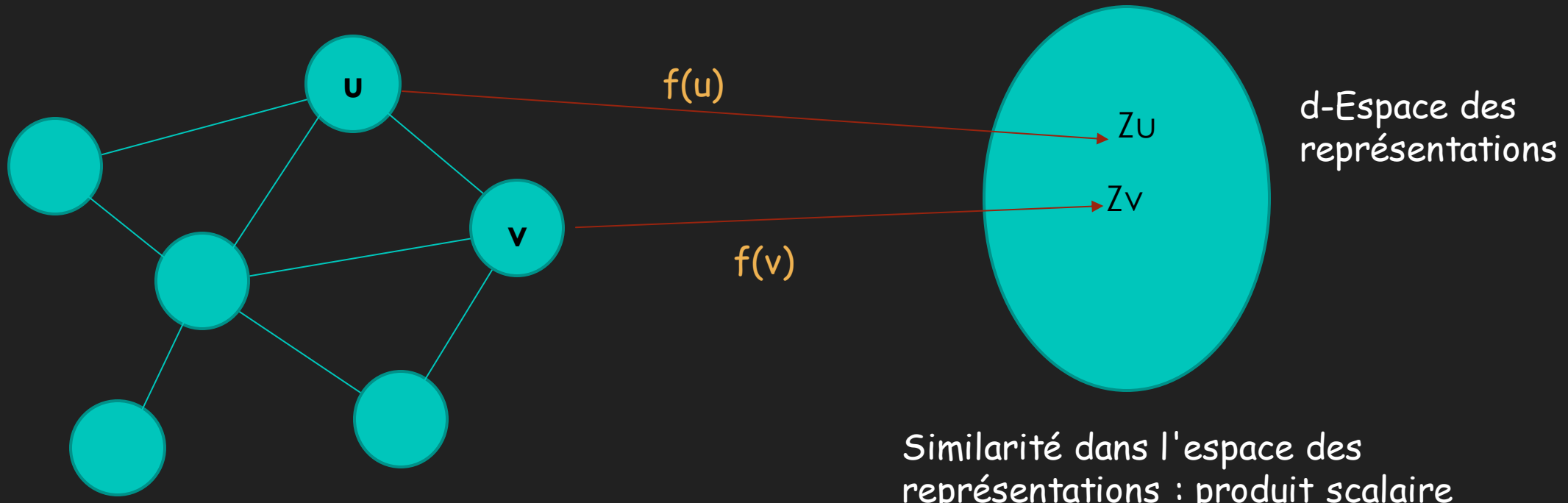


Fonction d'encodage vers un espace de dimension d

d: hyperparamètre

d-Vecteur : représentation des attributs d'un noeud

But : apprendre cette fonction



Similarité dans l'espace des représentations : produit scalaire

Similarité dans le réseau: similarité() ?

f doit préserver la **similarité** des noeuds

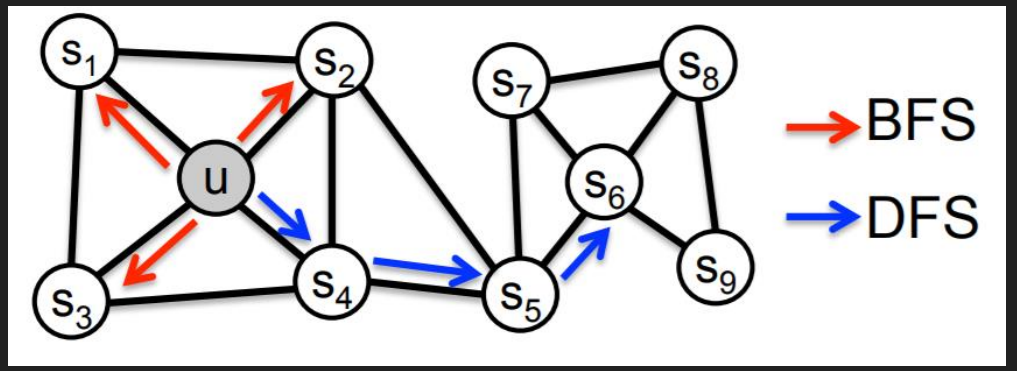
Les RW

Quelles stratégies pour RW ?

DeepWalk : non biaisé	Node2Vec : second-ordre biaisé

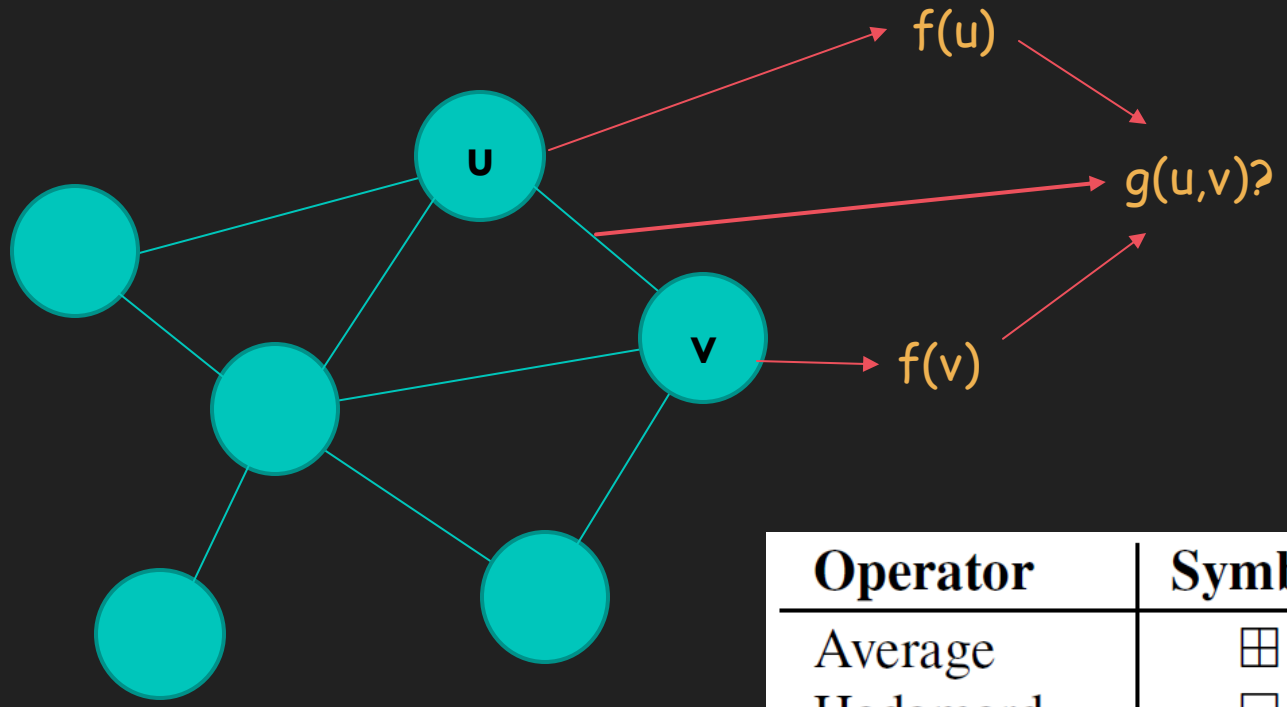
Rigide

Flexible



- P : retour
- Q : intérieur vs extérieur

Edge2vec?



~~g définie sur E?~~
g définie sur V^2

Operator	Symbol	Definition
Average	$\boxplus$	$[f(u) \boxplus f(v)]_i = \frac{f_i(u)+f_i(v)}{2}$
Hadamard	$\boxdot$	$[f(u) \boxdot f(v)]_i = f_i(u) * f_i(v)$
Weighted-L1	$\|\cdot\|_{\bar{1}}$	$\|f(u) \cdot f(v)\|_{\bar{1}i} = \|f_i(u) - f_i(v)\|$
Weighted-L2	$\|\cdot\|_{\bar{2}}$	$\|f(u) \cdot f(v)\|_{\bar{2}i} = \|f_i(u) - f_i(v)\|^2$

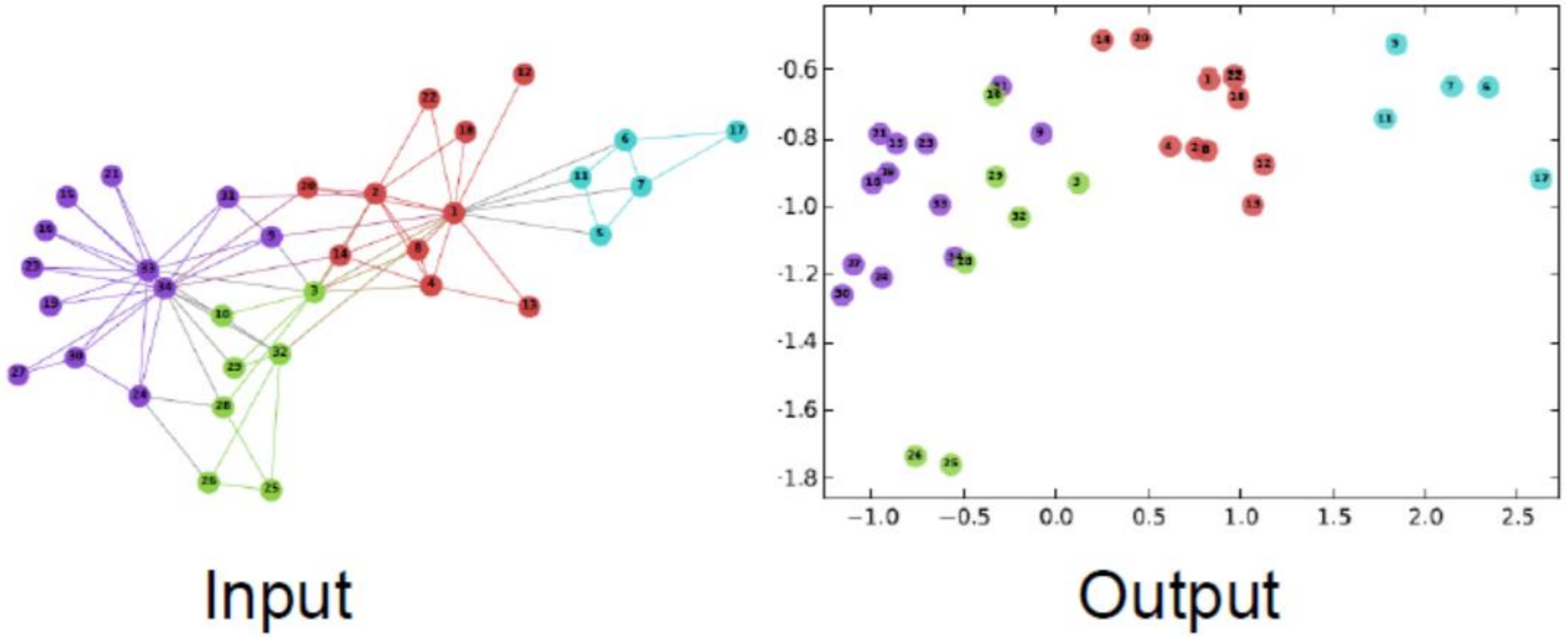
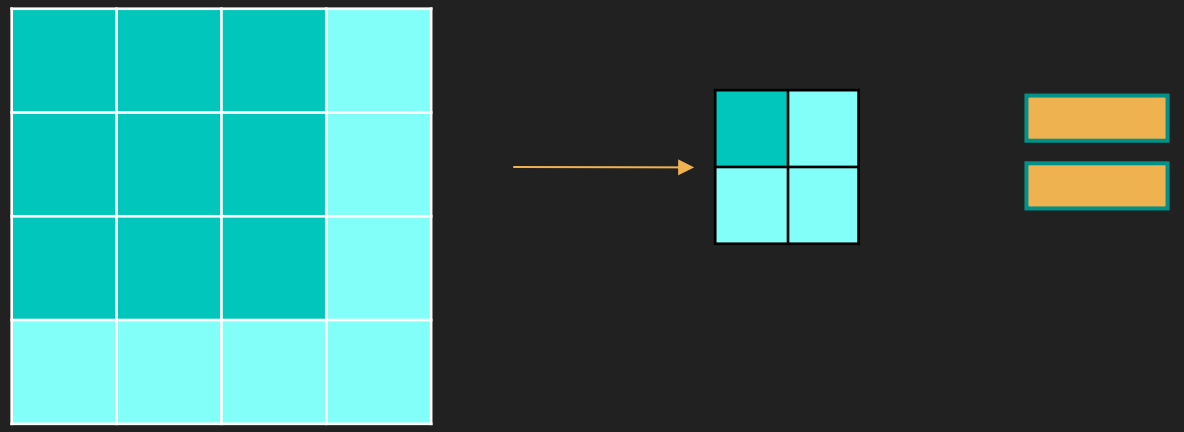


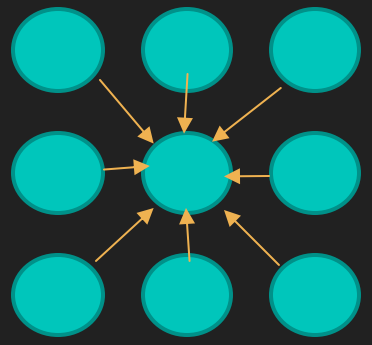
Image from: Perozzi et al. DeepWalk: Online Learning of Social Representations. *KDD 2014.*

Exemple d'encodage du réseau du Karaté Club de Zachary vers un espace 2D

GraphSAGE and DeepLearning on Graph

Généralisation de la notion de convolution :
Neighborhood aggregation



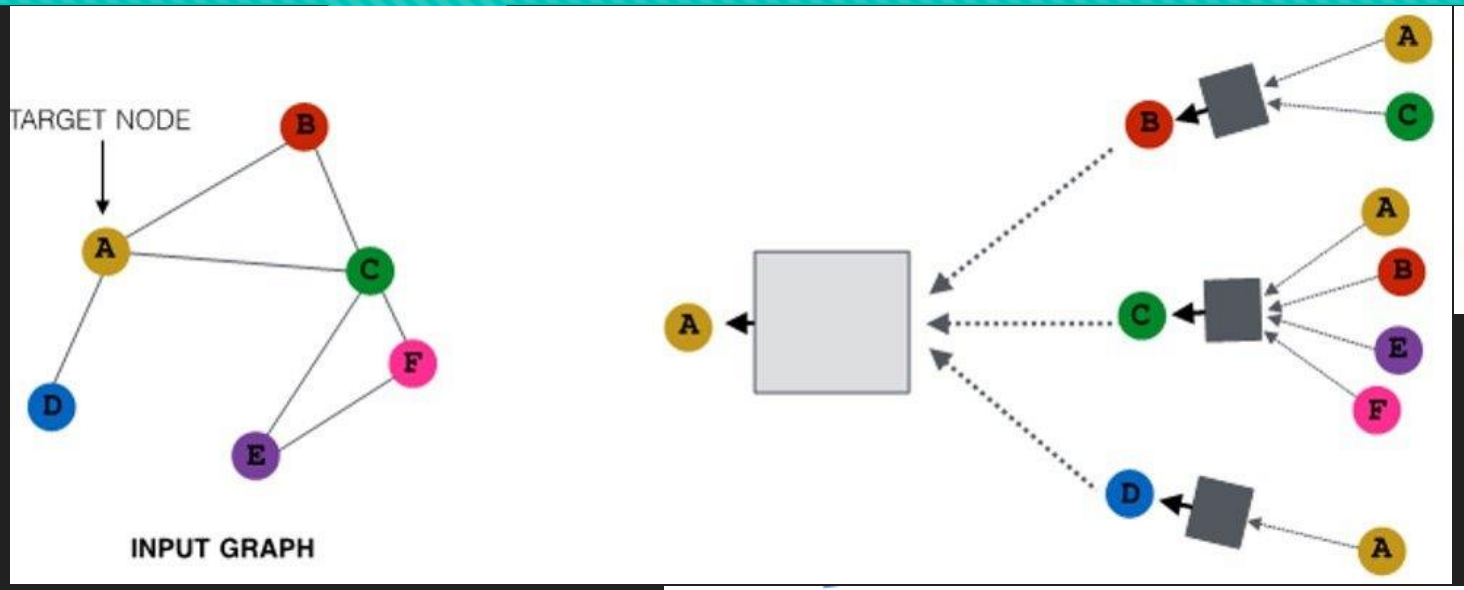


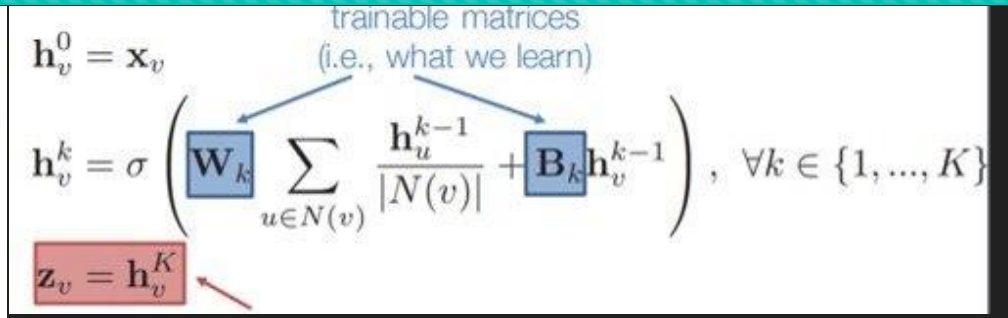
Transformer les informations provenant des voisins et les combiner :

- $\sum wi * hi$
- informations = attributs des noeuds

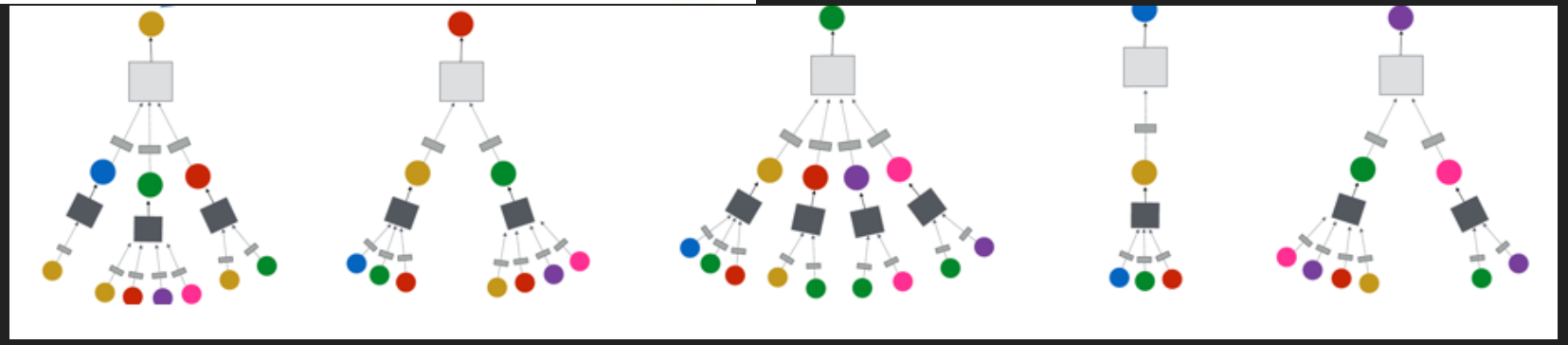
Problème : Structure des graphes réels plus complexes
Solution GraphSAGE

GCN



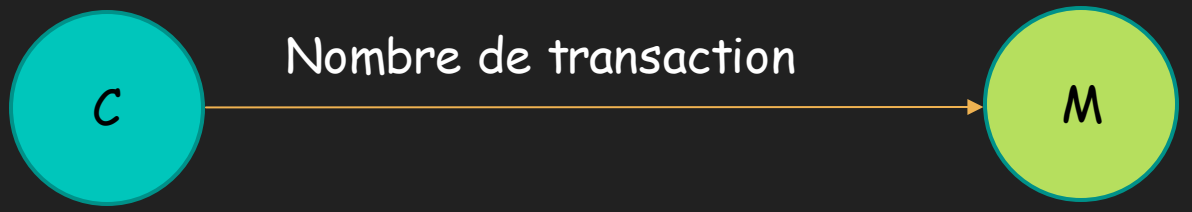


Chaque nœud définit son propre computation graph

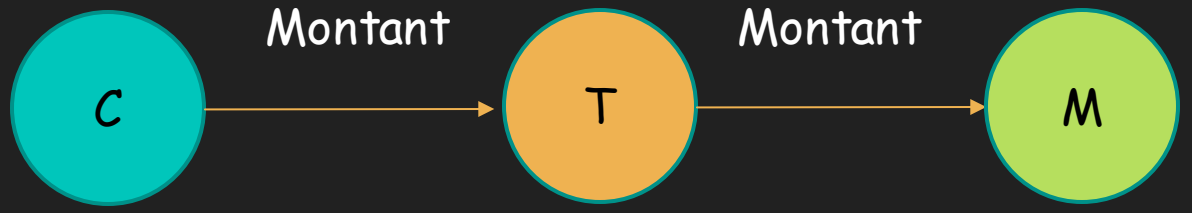


Modélisations du problème

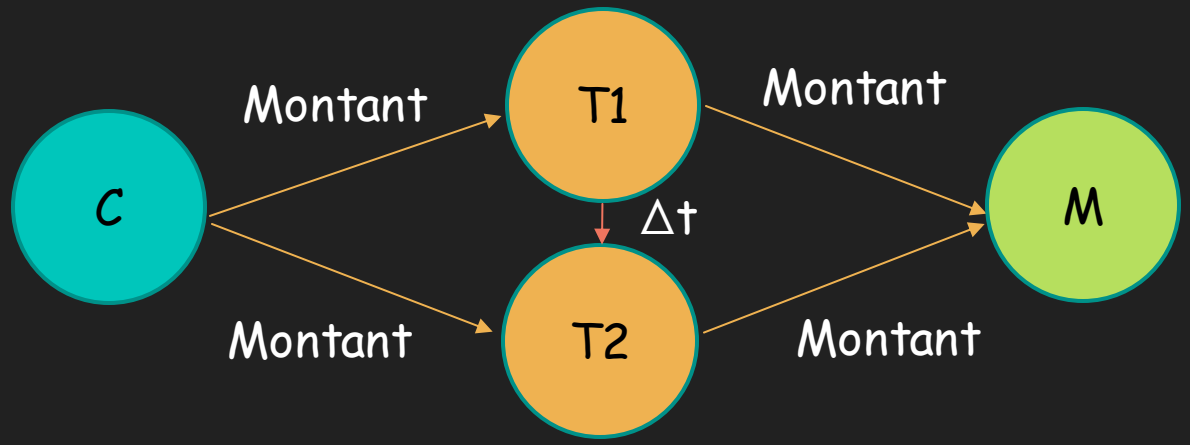
Simple



Bipartite (pondéré)



Bipartite-séquencé (pondéré)

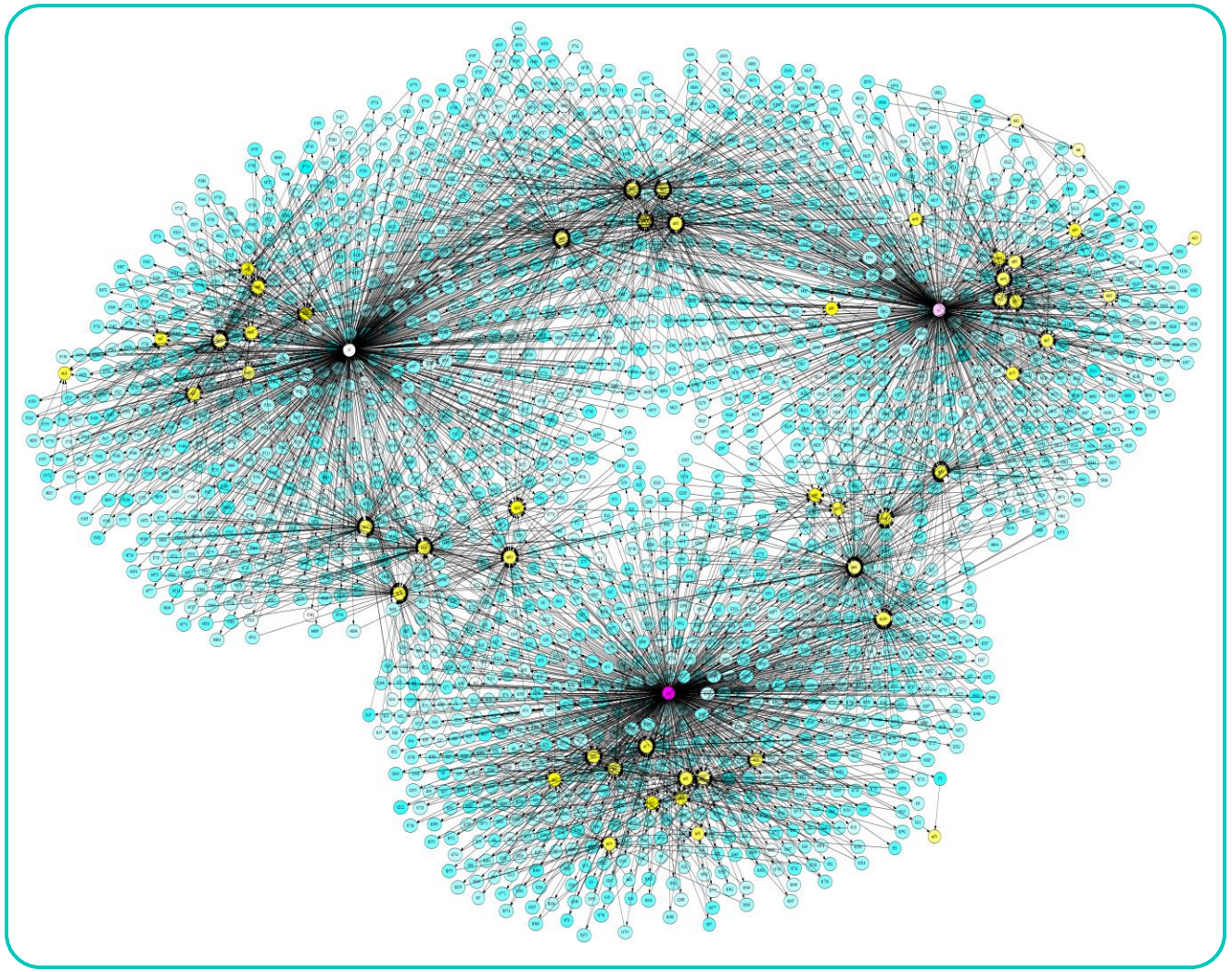


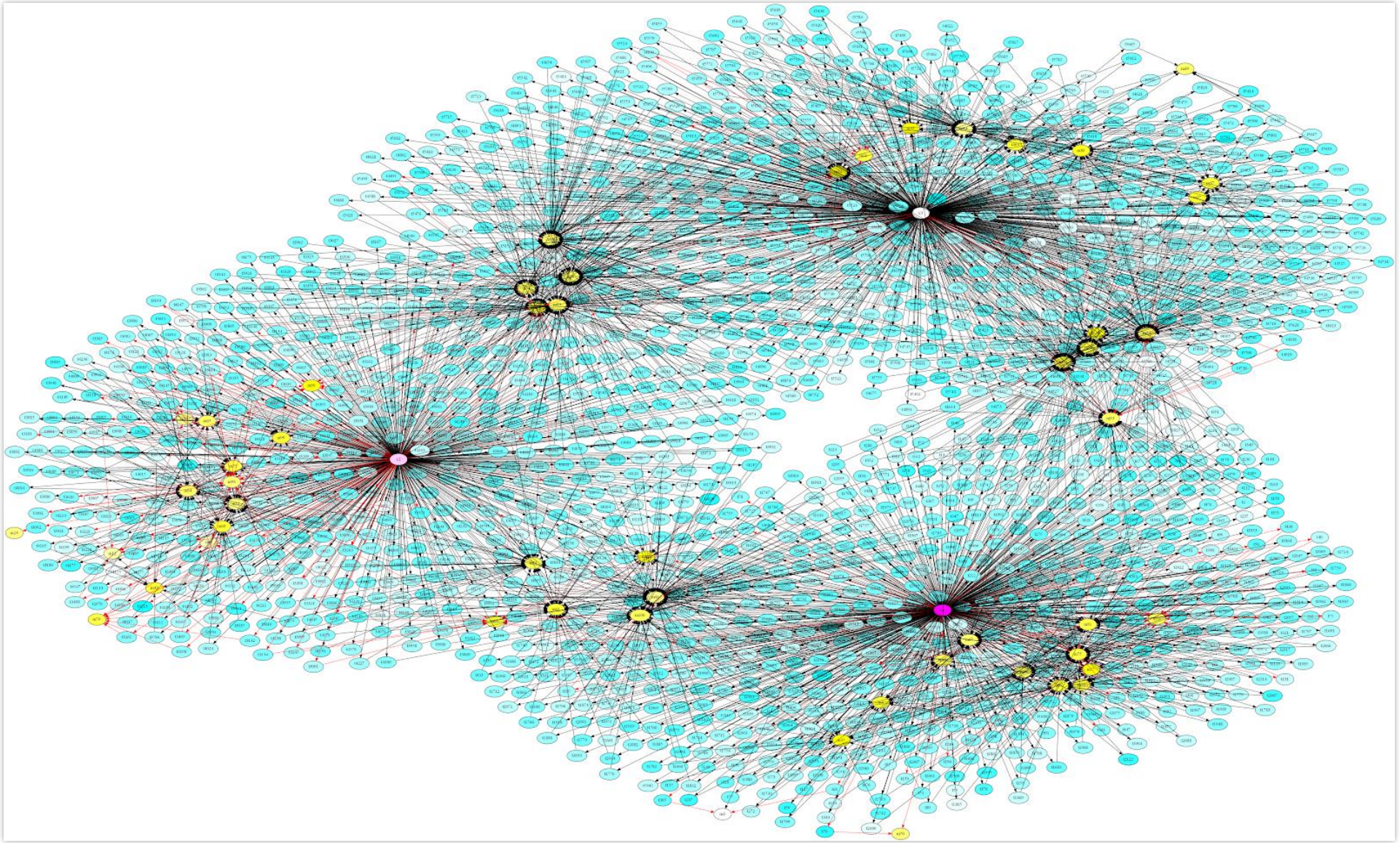
Visualisation des graphes

- Représentation graphique statique des graphes : GraphViz

- Automatisable
- Statique / non-interactif

GraphViz: Bipartite Weighted





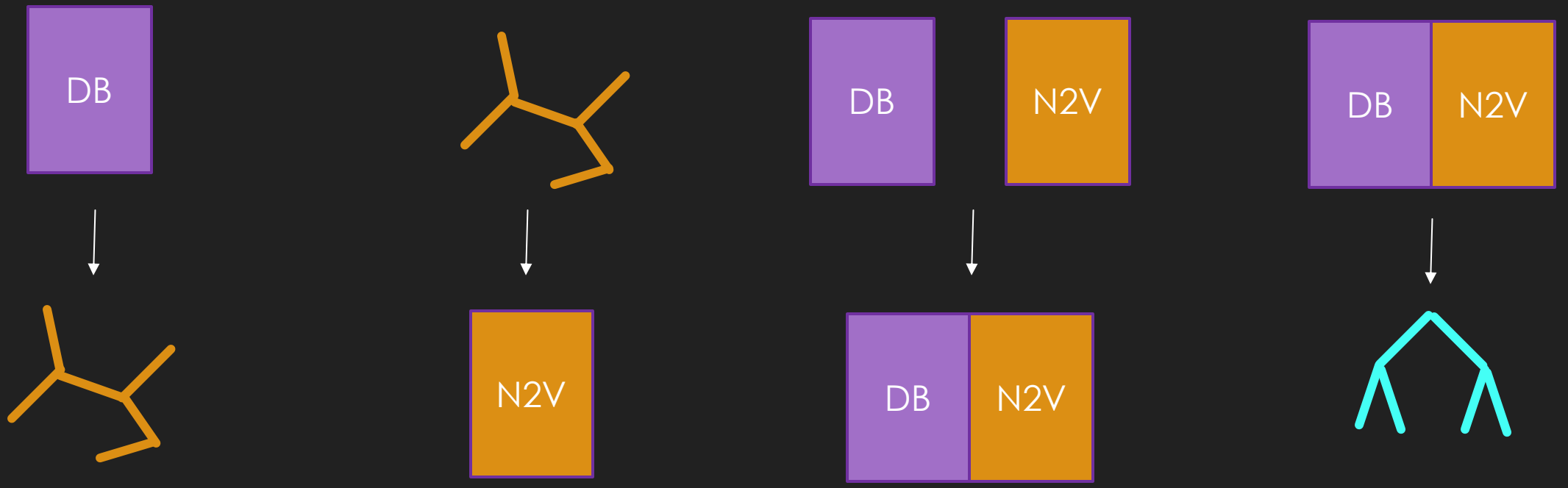
Procédure adoptée?

Extraire un graphe de transactions

Lancer node2vec

Joindre les features obtenues

Classification



Difficultés rencontrées et résultats

Premiers résultats non prometteurs:
- Résultats moins bon avec Node2Vec que sans Node2Vec (??)

Retrait des engineered features : le modèle était déjà bien trop performant pour voir l'apport de node2vec (99,9% d'accuracy)

	TN	FP	FN	TP
Sans	83781.8	1.8	1.4	8670.9
Avec	83742.4	3.1	3.3	8707.2

Difficultés rencontrées et résultats(2)

Retrait des features et nouvel essai → Node2Vec n'apporte aucune amélioration, ni détérioration → Remise en question des données et de la méthode

Explication → Données générées par un DT

Peu d'informations de type graphe?

Features de node2vec inutiles?

Difficultés rencontrées et résultats(3)

Obtention d'une nouvelle base de données, reflétant mieux des possibles habitudes de consommations

Mais...

	TN	FP	FN	TP
Sans Node2vec	59693.4 [59676.2, 59710.5]	49.7 [47.9, 51.6]	52.2 [50.1, 54.3]	6757.7 [6740.2, 6775.2]
Simple	59724.4 [59679.6, 59769.2]	48.9 [43.0, 54.8]	53.7 [48.3, 59.1]	6726.0 [6678.1, 6773.9]
BP	59651.4 [59605.5, 59697.3]	51.5 [47.4, 55.6]	57.0 [52.6, 61.4]	6793.1 [6748.9, 6837.3]
BP pondéré	59698.7 [59647.3, 59750.1]	54.7 [49.6, 59.8]	53.8 [49.1, 58.5]	6745.8 [6697.9, 6793.7]
BP-seq	59682.0 [59641.6, 59722.4]	52.2 [46.1, 58.3]	54.0 [49.9, 58.1]	6764.8 [6727.0, 6802.6]
BP-seq pondéré	59707.06 [59688.9, 59725.2]	49.54 [47.3, 51.8]	50.76 [48.6, 52.9]	6745.64 [6727.4, 6763.9]

Analyse détaillée de Node2Vec

Essais sur les features générées par node2vec uniquement

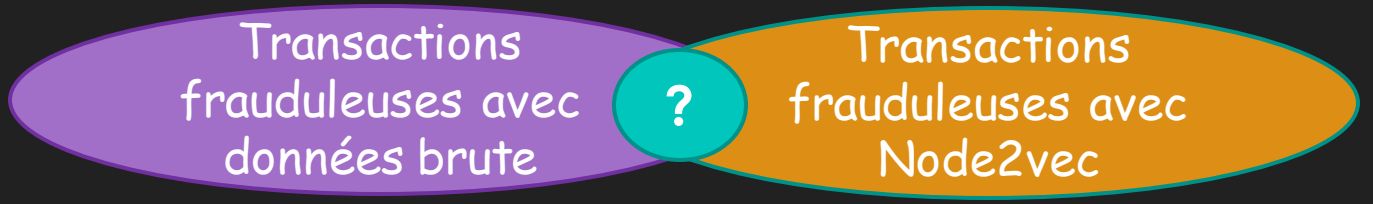
- Modélisation "Simple" est de loin la plus performante
- Donne de "bons" résultats sans les features de la base de données: Node2vec fonctionne-t-il?
- Un calcul de temps d'exécution montre que la classification se fait sans vraiment "regarder" les features générées par node2vec.

	TN	FP	FN	TP
Simple	59385.2	360.7	414.8	6392.3
BP	52845.1	6900.3	5811.2	996.4
BP pondéré	52885.6	6852.2	5813.4	1001.8
BP-seq	53773.4	6001.9	5225.2	1552:5
BP-seq pondéré	53097.8	6633.2	5688.6	1133.4
BP-seq pondéré (p = 2)	53583.3	6157.4	5379.0	1433.3

Simple > BP-seq > BP-seq pondéré > BP

Si node2vec seul fonctionne si bien, peut-on construire un **méta classifier**?

Méta-classification



Pour modèle simple:	1330	5497	55
Pour les autres modèles:	5900-600	800-900	7000-7100

Méta classification envisageable

	TN	FP	FN	TP	G_{FP}(%)	P_{FN}(%)	Eff
Sans Node2vec	59693.4 [59676.2, 59710.5]	**49.7 [47.9, 51.6]**	52.2 [50.1, 54.3]	6757.7 [6740.2, 6775.2]			
Simple	59699.0 [59621.1, 59776.9]	**31.4 [28.2, 34.6]**	1355.2 [1320.2, 1390.2]	5467.4 [5411.6, 5523.2]	36.8	2500	**67.9**
BP	59690.7 [59633.5, 59747.9]	**8.7 [6.2, 11.2]**	5994.1 [5939.1, 6049.1]	859.5 [838.3, 880.7]	82.5	11400	**138**
BP pondéré	59762.9 [59724.5, 59801.3]	**9.1 [6.7, 11.5]**	5915.7 [5878.5, 5952.9]	865.3 [854.2, 876.4]	81.7	11200	**137**
BP-seq	59776.0 [59711.7, 59840.3]	**5.9 [4.3, 7.5]**	5957.4 [5894.1, 6020.7]	813.7 [795.2, 832.2]	88.1	11300	**128**
BP-seq pondéré	59765.0 [59716.9, 59813.1]	**8.6 [6.8, 10.4]**	5923.5 [5874.1, 5972.9]	855.9 [847.5, 864.3]	82.7	11200	**135**

Est-ce meilleur qu'une sélection aléatoire?
Quel seuil d'efficacité choisir?

$$G_{\mathrm{FP}} = \frac{\mathrm{FP} - \mathrm{FP_M}}{\mathrm{FP}}$$

$$P_{\mathrm{FN}} = \frac{\mathrm{FN_M} - \mathrm{FN}}{\mathrm{FN}}$$

Conclusion

- ✓ Etude approfondie de 2 algorithmes: **Node2Vec** & **GraphSAGE**
- ✓ **Modélisation** d'un jeu de transactions sous forme de Graphe:
  - ✓ Simple
  - ✓ Bipartite
  - ✓ Bipartite-séquencé
- ✓ **Expérimentation** de Node2Vec sur les modèles de graphe et **amélioration partielle** du processus de classification des transactions
- ✓ **Mesure** de l'augmentation partielle de performance
- ✓ Travail sur la **visualisation** des graphes et de Node2Vec

Pistes pour le futur

Explorer GraphSAGE

Neo4J pour une visualisation des données dynamique

D'autres méthodes : analyse spectrale des graphes