



HAL
open science

ScoreCAM GNN: une explication optimale des réseaux profonds sur graphes

Adrien Raison, Pascal Bourdon, David Helbert

► To cite this version:

Adrien Raison, Pascal Bourdon, David Helbert. ScoreCAM GNN: une explication optimale des réseaux profonds sur graphes. XXVIIIe Colloque GRETSI - Traitement du Signal et des Images, Sep 2022, Nancy, France. hal-03686265

HAL Id: hal-03686265

<https://hal.science/hal-03686265>

Submitted on 25 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ScoreCAM GNN : une explication optimale des réseaux profonds sur graphes

Adrien RAISON^{1,2}, Pascal BOURDON^{1,2}, David HELBERT^{1,2}

¹XLIM-Asali, UMR CNRS 7252, Université de Poitiers

²I3M Laboratoire Commun CNRS-Siemens, Université de Poitiers
prénom.nom@univ-poitiers.fr

Résumé – L’explicabilité des réseaux profonds devient un problème central dans la communauté de l’apprentissage profond. Il en est de même pour l’apprentissage sur graphes, structure de données présente dans bon nombre de problématiques du monde réel. Nous proposons dans cette contribution, une méthode plus optimale, plus légère, consistante et exploitant mieux la topologie des graphes évalués que les méthodes de l’état de l’art.

Abstract – The explainability of deep networks is becoming a central issue in the deep learning community. It is the same for learning on graphs, data structure present in many problems of the real world. In this paper, we propose a method that is more optimal, lighter, consistent and exploits the of the evaluated graph that the methods of the state of the art.

1 Introduction

Les réseaux profonds sont largement utilisés aujourd’hui pour résoudre des problèmes complexes de nature académiques, industriels ou encore sociaux, dus à leur performance. À cause de la dimension de l’espace des paramètres ou encore de l’architecture des réseaux, les canaux d’informations constitués au cours de l’apprentissage sont compliqués à appréhender humainement. Il y a une difficulté à rendre intelligible le lien entre données d’entrée et de sortie, qui eux, sont parfaitement intelligibles, interprétables et compréhensibles. Ces architectures sont souvent qualifiées de “boîtes noires” dans la littérature. Le domaine de l’apprentissage profond sur graphe subit aussi ces inconvénients. Des méthodes s’intéressant à résoudre ces problèmes d’explicabilité des réseaux profonds sur graphes ont été proposées dernièrement. Cependant, l’interprétation des cartes de saillance est délicate car dépendante du contexte. Ces cartes doivent donc être soumises à expertise. Pour s’affranchir de cela et aussi du caractère subjectif des expertises, des métriques objectives ont été introduites [10][1]. La construction des explications d’une instance revient souvent à filtrer une partie de celle-ci pour extraire que l’information pertinente par rapport à la tâche d’apprentissage. Dans le cas d’une structure de donnée comme les graphes, il s’agit d’échantillonner intelligemment des sous-graphes, un problème d’optimisation souvent de type combinatoire. Dans cette étude, nous proposons une méthode plus légère, plus optimale, et plus consistante que celles proposées par l’état de l’art. Nous caractériserons d’une manière objective et subjective la

pertinence de notre méthode par rapport à l’état de l’art. Dans le contexte d’apprentissage profond sur graphes, il est commun de considérer un graphe G à N noeuds comme étant un couple (\mathbf{X}, \mathbf{A}) où $\mathbf{X} \in \mathbb{R}^{N \times F}$ est la matrice des descripteurs (de dimension F) des noeuds et où $\mathbf{A} \in \{0, 1\}^{N \times N}$ est la matrice d’adjacence de G . Les couches d’apprentissage profond sur graphes sont structurées de manière récursive. Pour un module de profondeur L , on a pour tout $l \in \{1, \dots, L\}$, $\mathbf{H}^{(l)} = \sigma(\mathbf{A}\mathbf{H}^{(l-1)}\mathbf{W}^{(l-1)})$ avec $\mathbf{H}^{(0)} = \mathbf{X}$, $\mathbf{W}^{(l-1)}$ est la matrice des paramètres de la $(l-1)$ -ème couche et où σ est une fonction d’activation. Pour tenter de répondre aux problématiques d’explicabilité, plusieurs méthodes ont été proposées. XGNN [12] construit ses cartes de saillance par un processus décisionnel de Markov apprenant une politique optimale. Cette méthode, sensible aux conditions d’initiales, nécessite l’entraînement d’un modèle supplémentaire (i.e en plus du classifieur) assurant une solution optimale asymptotiquement, donc pas atteignable en pratique. SubgraphX [13] sous échantillonne le graphe et optimise son procédé de construction des cartes de saillance au travers un arbre de recherche Monte-Carlo. L’importance du sous-graphe est donnée par la valeur *Shapely*. Cette méthode est elle aussi par essence sensible aux conditions initiales et ne propose qu’une solution optimale asymptotiquement. Des méthodes comme GNNExplainer [11] ou encore PGExplainer [5] s’intéressent à optimiser l’information mutuelle comprise entre des sous-graphes échantillonnés et la sortie du classifieur. GraphSVX [2] est une méthode basé sur l’échantillonnage de sous-graphes. Elle optimise un modèle linéaire dont les paramètres convergent vers les valeurs

Shapely de chaque sous-graphe.

L'inconvénient de recourir à l'échantillonnage de sous graphes est de construire des cartes de saillances basées sur un socle variable. Cela produit donc des explications différentes pour un même modèle et une même instance. Or les fondements d'une explication faite à propos d'un unique phénomène ne doivent pas varier. De plus, la plupart des méthodes propose des approximations de cartes de saillance théoriquement atteignables et il a un réel intérêt à avoir accès aux cartes optimales lorsque l'on est sur des problématiques sensibles, comme celles du domaine médical ou judiciaire. Dans cette étude, nous proposons une méthode ne nécessitant pas d'échantillonnage ni de post-entraînement et fournissant des explications de meilleure qualité, concises, non redondantes et concentrée sur la topologie des graphes.

2 Score CAM GNN

La méthode ScoreCAM [9] initialement développée pour les CNN, est dérivée de la méthode *Class Activation Mapping* (CAM) [14]. La méthode construit ses cartes de saillance en combinant linéairement les poids de la couche de moyennage d'agrégation globale aux couches de descripteurs de plus hauts niveaux. Un inconvénient majeur de cette méthode est d'imposer une architecture particulière au classifieur. La méthode GradCAM [13] est une approche gradient basée sur le même principe qui s'affranchit de la nécessité d'avoir une architecture particulière, et donc qui est adaptable n'importe laquelle d'entre-elles. Elle souffre cependant du phénomène d'évaporation du gradient ou du principe de fausse attribution lui empêchant une totale expressivité. La méthode ScoreCAM est, elle, adaptable à n'importe quelle architecture et évite les problématiques liées au gradient. Par un procédé d'évaluation de carte de descripteurs masquée par l'entrée initiale, la méthode combine linéairement les scores de ces évaluations par rapport à la tâche de classification avec les couches de descripteurs de plus haut niveau pour construire ses cartes de saillance. Cette étude est porter sur l'extension de la méthode ScoreCAM à des données type graphe.

Définition 2.1. Pour un graphe $G = (\mathbf{X}, \mathbf{A})$ et une paire de masques $(\mathbf{M}_X, \mathbf{M}_A)$, on définit :

$$G \star (\mathbf{M}_F, \mathbf{M}_A) = (\mathbf{X} \circ \mathbf{M}_X, \mathbf{A} \circ \mathbf{M}_A)$$

où \circ est le produit d'Hadamard.

Définition 2.2. Pour $\mathbf{x} \in \mathbb{R}^n$ tel que pour tout $\lambda \in \mathbb{R}$, $\mathbf{x} \neq \lambda \mathbf{1}$, où $\mathbf{1} = (1, \dots, 1) \in \mathbb{R}^n$. On définit :

$$\begin{aligned} N : \mathbb{R}^n &\rightarrow \mathbb{R}^n \\ \mathbf{x} &\mapsto \frac{\mathbf{x} - \min_i \mathbf{x}_i}{\max_i \mathbf{x}_i - \min_i \mathbf{x}_i} \end{aligned}$$

Le graphe expliqué par la méthode ScoreCAM GNN est défini par :

Définition 2.3. Pour un graphe $G = (\mathbf{X}, \mathbf{A})$ et un classifieur ϕ_θ de profondeur L avec $\mathbf{H}^L = (\mathbf{h}_k^L)_k$ la dernière carte de descripteurs de ϕ_θ on a :

$$\psi_{SCGNN}(G, \phi_\theta) = (ReLU(\sum_k u_k \mathbf{h}_k^L), \mathbf{A})$$

où pour tout k ,

$$u_k = \frac{e^{\phi_\theta(G \star (N(\mathbf{h}_k^L), \mathbf{1}))_c)}}{\sum_i e^{\phi_\theta(G \star (N(\mathbf{h}_i^L), \mathbf{1}))_c}}$$

et où $c = \arg \max \phi_\theta(G)$

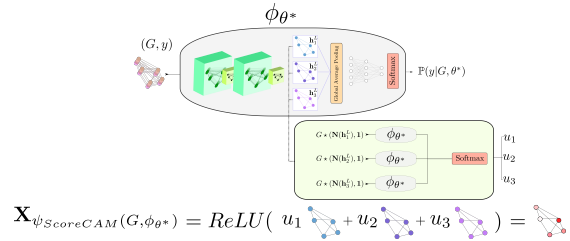


FIGURE 1 – Configuration de la méthode ScoreCAM GNN

3 Résultats

Nous avons évalué ScoreCAM GNN sur l'ensemble de données MNISTSuperpixels. Cet ensemble introduit par [6] est basé sur l'ensemble de données MNIST [4]. Il contient 70 000 exemples répartis sur 10 classes. Pour chaque exemple est fourni un graphe où chaque nœud est un superpixel de l'instance image associée dans MNIST ; il possède aussi la même étiquette. Les nœuds sont reliés en fonction de leurs relations spatiales. Pour classifier les instances de MNISTSuperpixels, nous avons utilisé dans deux réseaux différents, les deux modules d'apprentissage profonds les plus utilisés par l'état de l'art pour traiter des données type graphes : *Graph Convolutional Network* (GCN) [3] et *Graph Attention Network* (GAT) [8]. Ils seront indiqués par "Module Graphe" ou (MG) dans la description suivante :

$$\begin{aligned} (\mathbf{X}, \mathbf{A}) &\rightarrow \text{MG}(1,64) \rightarrow \text{TanH} \rightarrow (\text{MG}(64,64) \rightarrow \text{TanH}) \times 3 \\ &\rightarrow [\text{Global Average Pooling}, \text{Global Max Pooling}] \rightarrow \\ &\quad \text{Linéaire}(128,10) \end{aligned}$$

Nous avons utilisé l'optimiseur Adam avec un taux d'apprentissage de 7×10^{-4} . Nous avons obtenu 65% de précisions au bout de 500 épochs.

Évaluation objective Expliquer les prises de décisions effectuées par un classifieur profond est souvent lié à la nécessité d'évaluer la pertinence des cartes de saillance obtenue relativement au contexte. Cela requiert donc l'évaluation des ces dernières par un expert, elles sont par essence

subjectives et peuvent être biaisées. Dans [10], une métrique d’évaluation de la pertinence des cartes de saillances de manière objective est proposée. Elle quantifie la fidélité d’une méthode explicative par rapport à une instance et son classifieur. Une métrique complémentaire utile pour évaluer la qualité de cartes de saillance est d’évaluer sa parcimonie. En effet, une bonne carte de saillance doit se concentrer avec minutie uniquement sur les points d’intérêt et ne pas attribuer d’importance aux autres. Ces deux métriques sont de bons moyens pour évaluer la justesse globale d’une méthode d’explicabilité. Pour le modèle basé sur GCN, ScoreCAM GNN est 89% moins infidèle que le taux d’infidélité moyen observé par rapport aux autres méthodes. Elle atteint un taux de parcimonie 3.3 fois plus important que le taux moyen observé sur les autres méthodes. Concernant le modèle GAT, ScoreCAM GNN est respectivement 93% moins infidèle que le taux d’infidélité moyen observé par rapport aux autres méthodes. Sa parcimonie est 2.2 fois plus importante que la parcimonie moyenne observée sur les autres méthodes de l’état de l’art.

XAI Method	Model type	Mean Infidelity (\downarrow)	Mean Sparsity (\uparrow)
DL	GAT	307.08 (\pm 469.43)	0.10 (\pm 0.05)
GCAM	GAT	85.84 (\pm 62.94)	0.48 (\pm 0.15)
GNNEX	GAT	201.67 (\pm 106.18)	0.51 (\pm 0.38)
GSVX	GAT	851.77 (\pm 132.90)	0.01 (\pm 0.34)
SCGNN	GAT	23.89 (\pm 0.37)	0.62 (\pm 0.08)
DL	GCN	862.51 (\pm 26.31)	0.14 (\pm 0.06)
GCAM	GCN	340.88 (\pm 92.17)	0.46 (\pm 0.15)
GNNEX	GCN	592.14 (\pm 31.33)	0.27 (\pm 0.19)
GSVX	GCN	583.58 (\pm 23.15)	0.01 (\pm 0.03)
SCGNN	GCN	68.67 (\pm 7.01)	0.74 (\pm 0.04)

TABLE 1 – Métriques objectives sur MNISTSuperpixels

Évaluation subjective Pour établir visuellement la pertinence de la méthode ScoreCAM GNN, nous avons superposé, pour une instance donnée, son image et son graphe de superpixels associé. Classifier des chiffres non égaux entre eux demande de se concentrer sur leur topologie de manière à pouvoir établir des critères de différenciation pertinents. Expliquer comment classifier tel ou tel chiffre doit donc être basé sur des arguments structurels. De par sa définition, la méthode ScoreCAM GNN est proche de l’architecture du classifieur qu’elle explique. Son avantage par rapport à l’état de l’art est de fournir des explications non redondantes et prendre en compte pleinement la topologie du graphe. On observe que notre méthode se concentre sur les descripteurs clés pour fournir ses cartes de saillance.

Pour le chiffre 1 : Dans la Figure 2c, la carte est distribuée spatialement de manière uniforme sur l’ensemble des nœuds superposant spatialement le chiffre considéré. En effet, la structure spatiale du chiffre est ici régulière, par conséquent aucun superpixel saillant n’a de quantité d’informations plus importante, d’un point de vue de la tâche de classification, qu’un autre. Contrairement à la méthode DeepLIFT (Figure 2a), l’instance produite par

ScoreCAM n’introduit pas de redondance, ce qui permet une meilleure expressivité, une explication contenue et concentrée, mais conservant malgré tout ses qualités explicatives. L’instance 478 appartient à la même classe que l’instance 346. Dans la Figure 2f, la structure du chiffre est plus complexe, pour ScoreCAM GNN, la redistribution de l’importance a été effectué majoritairement sur les points de la courbure du chiffre les plus remarquables, ceux qui qualifient le mieux la structure de celui-ci. Dans la Figure 2d, DeepLIFT ne parvient pas à rendre compte de l’augmentation de la complexité de la structure du chiffre et fournit une distribution équivalente à celle fournie pour l’instance 346. Elle fournit ainsi une explication équivalente alors qu’il y a bien une différence notable entre ces deux instances. Par ailleurs, DeepLIFT conserve son caractère redondant. La méthode GradCAM fournit, sur un régime constant, des cartes de saillance difficilement qualifiables et ne rendant manifestement pas compte de manière humainement intelligibles de la tâche de classification.

Pour le chiffre 8 : L’instance 444 appartient à la classe du chiffre huit. Ce chiffre est singularisé structurellement par son point de croisement interne. Notre méthode (Figure 2i concentre toute sa distribution de saillance sur ce point structurel remarquable qui qualifie majoritairement ce qu’est le chiffre huit. Les autres méthodes (Figure 2g, 2h) n’accordent pas prioritairement d’importance à cette caractéristique singulière.

Pour le chiffre 3 : L’instance 969 a une courbure importante, la carte de saillance est distribuée de manière prépondérante sur cette région, qui est objectivement caractérisante. De par sa structure, ScoreCAM construit ses cartes de saillance en respectant la construction algébrique des classifieurs sur graphes. En effet, on observe notamment Figure 2l un centre d’importance majeur au nœud voisin des nœuds caractérisant la structure du chiffre considéré, conséquence de la phase d’agrégation additive [7]. Cette agrégation permet une synthétisation de l’interprétation de plus haut niveau de la carte de saillance.

4 Conclusion

Dans cet étude, nous proposons une méthode d’explicabilité de réseaux profond sur graphes plus optimale, en étant plus fidèle au classifieur et en fournissant des cartes de saillance avec moins de redondance et plus de parcimonie tout en conservant une qualité d’explication importante. En évitant les écueils liés au conditions d’initialisation et en proposant des solutions optimales, non approximatives et ne nécessitant pas l’optimisation d’un modèle supplémentaire, la méthode ScoreCAM surpasse les méthodes de l’art de manière objective et subjective.

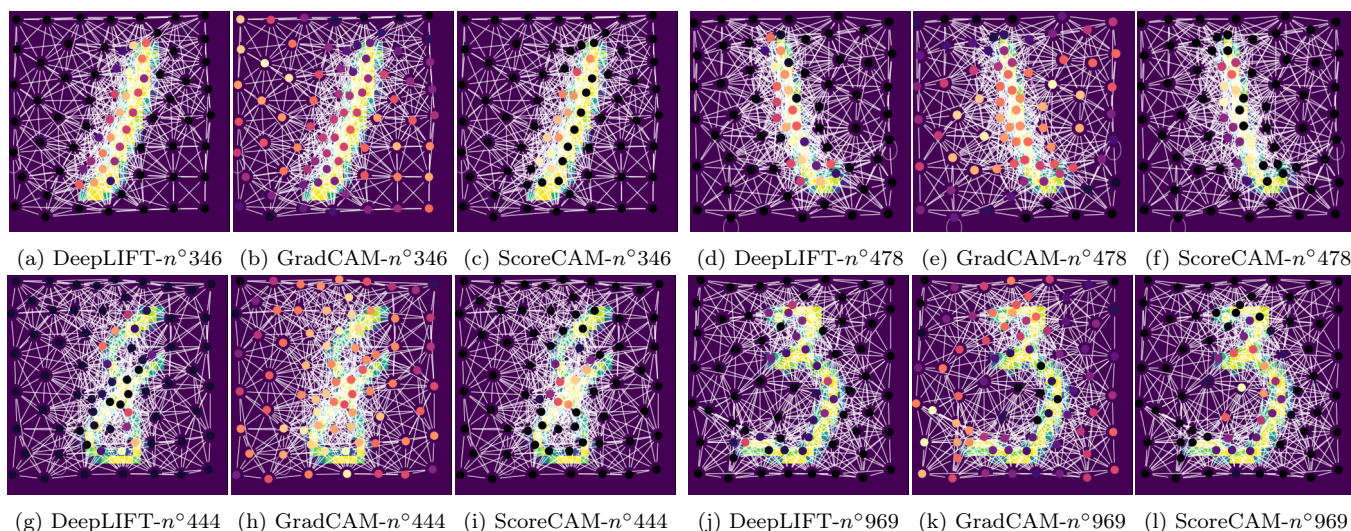


FIGURE 2 – Différentes explications de MNISTSuperpixels

Références

- [1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity Checks for Saliency Maps. page 11.
- [2] Alexandre Duval and Fragkiskos Malliaros. Graphsvx : Shapley value explanations for graph neural networks. In *European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, 2021.
- [3] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv :1609.02907 [cs, stat]*, February 2017. arXiv : 1609.02907.
- [4] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, November 1998. Conference Name : Proceedings of the IEEE.
- [5] Dongsheng Luo, Wei Cheng, Dongkuan Xu, Wenchao Yu, Haifeng Chen, Xiang Zhang, and Bo Zong. Parameterized Explainer for Graph Neural Network. page 12.
- [6] Federico Monti, Davide Boscai, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M. Bronstein. Geometric Deep Learning on Graphs and Manifolds Using Mixture Model CNNs. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5425–5434, Honolulu, HI, July 2017. IEEE.
- [7] F. Scarselli, M. Gori, Ah Chung Tsoi, M. Hagenbuchner, and G. Monfardini. The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 20(1) :61–80, January 2009.
- [8] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. *arXiv :1710.10903 [cs, stat]*, February 2018. arXiv : 1710.10903.
- [9] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-CAM : Score-Weighted Visual Explanations for Convolutional Neural Networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 111–119, Seattle, WA, USA, June 2020. IEEE.
- [10] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Sai Suggala, David I Inouye, and Pradeep Ravikumar. On the (In)fidelity and Sensitivity of Explanations. page 12.
- [11] Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. GNNExplainer : Generating Explanations for Graph Neural Networks. page 12.
- [12] Hao Yuan, Jiliang Tang, Xia Hu, and Shuiwang Ji. XGNN : Towards Model-Level Explanations of Graph Neural Networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 430–438, Virtual Event CA USA, August 2020. ACM.
- [13] Hao Yuan, Haiyang Yu, Jie Wang, Kang Li, and Shuiwang Ji. On Explainability of Graph Neural Networks via Subgraph Explorations. page 12.
- [14] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning Deep Features for Discriminative Localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, Las Vegas, NV, USA, June 2016. IEEE.