Un cadre semi-supervisé résilient pour la détection d'anomalie sur graphe attribué

Bastien Giles[*,**], Baptiste Jeudy[*], Christine Largeron[*], Damien Saboul[**]

[*] Laboratoire Hubert Curien UMR5516, UJM-Saint-Etienne, CNRS, IOGS,
Université de Lyon, F-42023 St-Etienne, France
prenom.nom@univ-st-etienne.fr
[**] be-ys research
prenom.nom@be-ys-research.com

Résumé. La détection d'anomalies dans des graphes est une tâche importante dans de nombreux domaines. Même si les modèles semi-supervisés existants se sont avérés efficaces pour identifier les anomalies, ils supposent cependant qu'un échantillon étiqueté du graphe est disponible mais sans prendre en compte le problème du manque de fiabilité d'un tel échantillon. Dans cet article, nous considérons des graphes attribués et nous proposons un nouveau cadre méthodologique basé sur deux auto-encodeurs à convolution de graphe entraînés selon un mécanisme de suspicion. Le premier est entraîné sur un échantillon censé être composé d'entités normales tandis que le second sur un échantillon supposé contenir des anomalies. La classification finale se fait en couplant le résultat des deux auto-encodeurs. Nous démontrons expérimentalement que notre approche obtient des performances au moins équivalentes aux méthodes de l'état de l'art dans le cas d'échantillons parfaits tout en étant plus résiliente aux erreurs d'étiquetage.

1 Introduction

L'identification d'anomalies est un problème important étudié depuis longtemps (Grubbs, 1969; Akoglu, 2021; Aggarwal, 2017; Chalapathy et al., 2018; Pang et al., 2021). Elle est utilisée dans divers domaines tels que la santé (Esteva et al., 2017), la détection des fraudes (Zhang et al., 2019; Lu et Li, 2020), ou encore la finance (Wang et al., 2019). Selon les applications, les données peuvent être représentées sous des formats variés requérant des méthodes appropriées. Les graphes attribués sont l'une de ces représentations (Ma et al., 2021). Alors que les graphes permettent de représenter par des liens des interactions entre des entités correspondant aux sommets, les graphes attribués fournissent en plus une matrice d'attributs qui contient les informations caractéristiques de chaque nœud. Par exemple, dans le cas d'un réseau social, le graphe attribué décrit les interactions entre les utilisateurs, mais également le profil de chaque utilisateur (âge, sexe, centre d'intérêt, etc.) (Interdonato et al., 2019).

Dans la pratique, les approches supervisées de détection d'anomalies dans des graphes attribués sont souvent difficilement applicables, car elles nécessitent un jeu de données complètement étiqueté, difficile et coûteux à obtenir manuellement. Pour cette raison, la plupart

des études expérimentales de détection d'anomalies privilégient des méthodes dites non supervisées, qui ne nécessitent aucun étiquetage et utilisent la rareté des anomalies pour guider la conception des modèles de classification. Si ces méthodes identifient efficacement les éléments aberrants, c'est-à-dire ceux qui diffèrent significativement du reste de l'ensemble de données, elles échouent à trouver les anomalies lorsqu'elles sont plus semblables à la classe majoritaire.

C'est pourquoi, des méthodes semi-supervisées, ne nécessitant qu'un petit échantillon des données sont également explorées. À l'aide de cet échantillon, la frontière entre entités normales et anomalies peut être mieux détectée. Dans la pratique, l'étiquetage de cet échantillon nécessite cependant une expertise humaine. Or, les méthodes semi-supervisées actuelles ne prennent pas en compte la possibilité d'erreurs humaines dans cet étiquetage qui se traduisent par l'existence d'entités normales étiquetées comme anormales et réciproquement, d'anomalies considérées comme normales. Dans la pratique, de telles erreurs d'étiquetage dans l'échantillon d'apprentissage ne sont pas rares et, en général, elles ne sont pas sans impact sur le résultat final.

Pour surmonter ces limites, nous proposons un nouveau cadre méthodologique générique pour détecter des anomalies dans les graphes attribués. Ce cadre, appelé Suspicious, est fondé sur des modèles à la pointe de l'état de l'art, les "Graph Convolutional Networks" (GCN). Suspicious, utilise simultanément le déséquilibre des classes dans l'ensemble de données et un mécanisme de suspicion pour produire un modèle de détection d'anomalies efficace et résistant à l'erreur humaine.

Plus précisément, Suspicious utilise deux auto-encodeurs qui calculent une représentation des nœuds du graphe attribué dans un espace vectoriel de dimension réduite. Ensuite, une approximation du graphe original est générée à partir de cette représentation. Le premier auto-encodeur reconstruit le graphe de telle façon que les point normaux soient bien reconstruits et le second auto-encodeur de sorte que les points anormaux soient bien reconstruits. Enfin, les erreurs de reconstruction des deux auto-encodeurs sont calculées puis combinées pour obtenir un score final permettant l'identification des anomalies.

Dans le cadre de notre évaluation expérimentale, nous utilisons une architecture d'auto-encodeur semblable à celle présentée dans Dominant (Ding et al., 2019). Mais, Suspicious, du fait de son caractère générique, supporte n'importe quelle autre architecture d'auto-encodeur de graphe, notamment en changeant la méthode de plongement.

Notre contribution est la suivante :

1. Nous proposons *Suspicious*, un cadre semi-supervisé général fondé sur la reconstruction, pour la détection d'anomalies dans les graphes attribués. Il présente en outre l'avantage d'être résilient aux jeux de données mal étiquetés.

2. Nous démontrons expérimentalement sur cinq jeux de données réels que notre cadre est aussi performant que les meilleures méthodes actuelles sur des jeux de données parfaitement étiquetés, tout en les surpassant de façon systématique lorsqu'il y a des erreurs d'étiquetage dans le jeu d'entraînement.

Après une présentation de l'état de l'art permettant de mieux positionner cette proposition dans la Section 2, nous décrivons Suspicious dans la Section 3 et son évaluation expérimentale dans la Section 4 avant de conclure.

2 État de l'art

Lorsqu'on considère des données relationnelles représentées par un graphe, un élément anormal peut être un sous-graphe, un lien entre deux nœuds ou le nœud lui-même. Dans cet article, nous nous concentrons sur ce dernier type d'anomalies. Parmi les méthodes conçues pour les identifier, nous pouvons mentionner l'approche basée sur la proximité (Jeh et Widom, 2002; Antonellis et al., 2008) qui mesure la proximité des objets à travers la structure du graphe et considère que les objets proches dans le graphe sont susceptibles d'appartenir à la même classe (anormale ou normale). Une autre famille de méthodes dédiées aux réseaux ayant une structure communautaire (Xu et al., 2007) composée de groupes de nœuds fortement connectés, considère comme anormaux les nœuds (ou arêtes) qui relient deux communautés. Cependant, nous pouvons remarquer que les méthodes appartenant à ces familles ne réussissent qu'à trouver un type très spécifique d'anomalies "structurelles".

À l'opposé, une troisième famille de méthodes, conçues pour les graphes attribués, ne considère que la matrice d'attributs et réduit le problème à de la détection d'anomalies dans des données tabulaires. La littérature dans ce domaine est assez vaste (Akoglu, 2021; Chandola et al., 2009), avec divers modèles tels que ceux basés sur la distance, la densité, le regroupement d'éléments similaires, la profondeur, et bien d'autres. Cependant, ces méthodes ignorent toutes les informations relationnelles contenues dans la structure du graphe.

Pour exploiter à la fois les informations relationnelles et les attributs des nœuds contenus dans un graphe attribué, l'état de l'art actuel utilise les plongements (Kipf et Welling, 2017; Veličković et al., 2018). Ces plongements sont produits par un encodeur qui crée une représentation vectorielle de faible dimension des nœuds en utilisant les deux types d'informations : la matrice d'attributs et la matrice d'adjacence. Les techniques classiques de détection d'anomalies sur les vecteurs peuvent alors être appliquées sur ces plongements. Par exemple, dans le modèle semi-supervisé présenté dans Kumagai et al. (2021), la distance entre la représentation vectorielle des nœuds dans le plongement et le centre d'une hypersphère apprise est calculée et elle définit un score d'anomalie. L'hypersphère est entraînée par un GCN sur des nœuds étiquetés afin qu'elle n'englobe que des nœuds normaux. Cependant, en tentant d'exploiter les rares données étiquetées, ces méthodes deviennent extrêmement sensibles aux erreurs présentes dans celles-ci.

D'autre part, les méthodes basées sur la reconstruction visent également à exploiter la structure du graphe et les attributs des nœuds pour détecter les anomalies (Li et al., 2017; Peng et al., 2018). Elles utilisent la factorisation matricielle pour créer une approximation du graphe original, avant de calculer la distance de chaque nœud à sa reconstruction en tant que score d'anomalie.

Certaines approches actuelles (Ding et al., 2019; Akcay et al., 2019; Fan et al., 2020) combinent les plongements et la reconstruction. Dans ce cas, après le plongement, un décodeur est ajouté pour essayer de recréer le graphe et les attributs originaux à partir du plongement. Un score d'anomalie est calculé à partir de l'erreur de reconstruction.

Parmi ces méthodes, Dominant (Ding et al., 2019) est probablement la plus proche de notre travail. Elle utilise un GCN (Kipf et Welling, 2017) pour créer un auto-encodeur qui compresse le graphe dans un plongement puis elle décode les représentations précédemment obtenues pour obtenir une approximation de la matrice d'adjacence et de la matrice d'attributs. Finalement, les nœuds mal reconstruits sont considérés comme anormaux. Cependant, les ano-

malies détectées par Dominant semblent limitées à un certain type. Il s'agit des nœuds dont les valeurs d'attributs varient de façon significative par rapport à celles de leurs voisins.

Notre modèle contourne cette faiblesse de Dominant en ajoutant de l'apprentissage semi-supervisé. De plus, il pallie l'absence de résilience des modèles semi-supervisés en ayant recours à deux auto-encodeurs.

3 Suspicious

Après avoir défini plus formellement la problématique à résoudre et introduit les notations utilisées dans la suite, cette section présente le cadre méthodologique Suspicious proposé pour identifier des anomalies dans un graphe attribué, de façon résiliente aux erreurs d'étiquetage de l'échantillon d'apprentissage.

3.1 Problématique

Soit $G = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ un réseau attribué défini par l'ensemble des nœuds $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$; l'ensemble des arêtes $\mathcal{E}$, représenté par une matrice d'adjacence symétrique $\mathbf{A}$ de dimension $n \times n$ où $a_{k,j} = 1$ s'il existe une arête entre les nœuds k et j et $a_{k,j} = 0$ sinon ; et la matrice d'attributs $\mathbf{X} \in \mathbb{R}^{(n \times d)}$ où $\mathbf{x}_i$ représente le vecteur d'attributs du i-ème nœud.

On suppose que l'on dispose d'un sous-ensemble $\mathcal{V}_l$ de $\mathcal{V}$, lui-même composé de deux sous ensembles disjoints $\mathcal{V}_s$ et $\mathcal{V}_n$, contenant respectivement des nœuds déjà identifiés comme anormaux et normaux : $\mathcal{V}_l = \mathcal{V}_s \cup \mathcal{V}_n$ tel que $\mathcal{V}_s \cap \mathcal{V}_n = \emptyset$.

Le problème que nous cherchons à résoudre peut être exprimé de la façon suivante :

Étant donné le graphe attribué $G = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ et les deux sous-ensembles de nœuds $\mathcal{V}_s$ et $\mathcal{V}_n$ déjà identifiés, le but est d'estimer un score d'anomalie des nœuds non étiquetés de telle sorte que les nœuds anormaux aient un score plus élevé que les nœuds normaux.

3.2 Modèle

3.2.1 Principe sous-jacent à Suspicious

Pour résoudre cette tâche, notre modèle Suspicious [1] utilise deux auto-encodeurs de graphes, *Susp* et *Norm*, comme illustré sur la Figure 2. L'auto-encodeur *Norm* est appris de façon à ce que les nœuds de $\mathcal{V}_n$ soient correctement reconstruits et ceux de $\mathcal{V}_s$ ne le soient pas. Pour sa part, l'auto-encodeur *Susp* fait l'inverse : il essaye de mieux reconstruire les nœuds de $\mathcal{V}_s$ que ceux de $\mathcal{V}_n$.

Pour un nœud non étiqueté, on peut ensuite obtenir ainsi une erreur de reconstruction (*i.e.* score) pour *Norm* et une pour *Susp* . Ces deux scores sont utilisés pour catégoriser les nœuds de la façon suivante :

— Des scores élevés en sortie des deux auto-encodeurs pour un nœud signifie que le modèle est incapable de reconstruire le nœud peu importe les données sur lesquelles il a été entraîné. Il s'agit donc d'une valeur aberrante et non pas d'une anomalie recherchée. Il faut donc que le score final attribué à ce nœud soit faible.

1. implémentation disponible à l'adresse https ://github.com/GILESBastien/Suspicious_EGC

— Des scores faibles en sortie des deux modèles signifient que le nœud est facilement reconstruit peu importe les données sur lesquels le modèle a été entraîné. Ce n'est donc pas l'anomalie recherchée et il faut donc aussi que le score final attribué à ce nœud soit faible.
— Un score faible attribué par *Norm* et un score élevé avec *Susp* signifie qu'il s'agit plutôt d'un nœud normal ; ce qui doit aboutir aussi à l'attribution d'un score final faible.
— Un score élevé dans *Norm* et un score faible dans *Susp* signifie que les deux modèles sont en accord sur l'anormalité du nœud. Ce qui signifie que le nœud correspond bien à une anomalie pertinente et, par conséquent, il doit obtenir un score final parmi les plus hauts.

Susp	***Norm***	**Suspicious**
Erreur faible	**Erreur élevée**	**Anomalie**
Erreur élevée	Erreur élevée	Nœud aberrant
Erreur faible	Erreur faible	Nœud normal
Erreur élevée	Erreur faible	Nœud normal

TAB. 1 – *Classification des nœuds en fonction des erreurs de reconstruction de Susp et Norm .*

3.2.2 Architecture de Suspicious

L'architecture de ces auto-encodeurs peut être fondée sur celle de Dominant (Ding et al., 2019) qui utilise des GCN (Kipf et Welling, 2017) comme encodeur pour créer un plongement de nœuds $\mathbf{Z}$ de G, puis deux autres GCN, un décodeur d'attributs et un décodeur de la matrice d'adjacence afin de recréer respectivement à partir de $\mathbf{Z}$ des approximations $\tilde{X}$ et $\tilde{A}$ de X et A, comme illustré dans la Figure 1. Il convient cependant de noter que notre méthode peut utiliser n'importe quel auto-encodeur de graphe et par conséquent d'autres méthodes de plongement de graphes comme GraphSAGE (Hamilton et al., 2017) ou GAT (Veličković et al., 2018) ; ce qui lui confère un caractère générique.

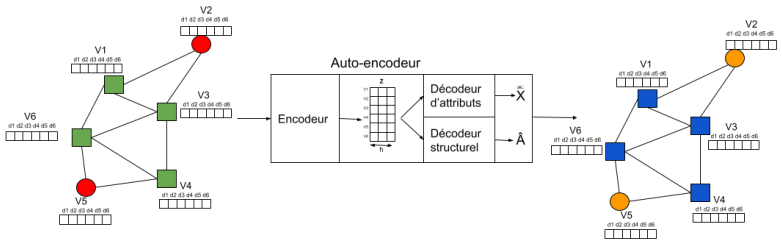

FIG. 1 – *Architecture d'un auto-encodeur. Les nœuds verts sont des nœuds normaux, les nœuds rouges sont des nœuds anomalies, les nœuds orange sont des nœuds mal reconstruits par l'auto-encodeur, et les nœuds bleus sont les nœuds bien reconstruits.*

Les GCN créent les plongements de nœuds de sorte que $\mathbf{H}^{(l+1)}$, le plongement de G après $l+1$ couches est obtenu à partir du plongement précédent $\mathbf{H}^{(l)}$ à l'aide de la règle de propaga-

tion suivante :

$$\mathbf{H}^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}\mathbf{H}^{(l)}W^{(l)} + b^{l}), \tag{1}$$

où $\tilde{A} = A+I$ et $\tilde{D} \in \mathbb{R}^{n\times n}$ est la matrice de degré de $\tilde{A}$, $\mathbf{W}^{(l)} \in \mathbb{R}^{h\times h}$, $b^{(l)} \in \mathbb{R}^{h}$ représentent respectivement la matrice des poids et le vecteur de biais de leurs couches respectives et $\sigma(..)$ est une fonction d'activation, ReLU pour notre expérience. Pour la première couche, nous utilisons la matrice d'attributs comme plongement :

$$\mathbf{H}^{(0)} = X. \tag{2}$$

L'encodeur de notre modèle utilise un GCN à k couches pour créer un plongement de nœuds à h dimensions $\mathbf{Z} \in \mathbb{R}^{n\times h}$ de G, tel que $z_i \in \mathbb{R}^{h}$ est le plongement du nœud v_i :

$$\mathbf{Z} = \mathbf{H}^{(k)}. \tag{3}$$

Le décodeur d'attributs reconstruit la matrice d'attributs originale à partir des attributs compressés dans $\mathbf{Z}$. Il utilise un GCN à une seule couche sur le plongement $\mathbf{Z}$ pour construire $\hat{\mathbf{X}}$ une approximation de $\mathbf{X}$:

$$\hat{\mathbf{X}} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}ZW^{(k)} + b^{k}), \tag{4}$$

$\mathbf{W}^{(k)} \in \mathbb{R}^{h\times d}$ et $b^{(k)} \in \mathbb{R}^{d}$ représentent respectivement la matrice de poids et le vecteur de biais de leurs couches respectives et h le paramètre dimension cachée.

Le décodeur de structure reconstruit l'information relationnelle, c'est à dire la matrice d'adjacence, à partir de $\mathbf{Z}$. Il utilise une couche de prédiction de liens pour construire $\hat{\mathbf{A}}$ une approximation de $\mathbf{A}$:

$$\hat{\mathbf{A}} = \sigma(\mathbf{Z}\mathbf{Z}^{T}), \tag{5}$$

3.2.3 Calcul des erreurs d'approximation

Pour chaque nœud v_i et chaque auto-encodeur AE, qui peut être *Susp* ou *Norm*, nous définissons l'erreur de reconstruction du nœud :

$$error_{AE}(v_i) = \|\mathbf{a}_i - \hat{\mathbf{a}}_i\|_2^2 + \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2 \tag{6}$$

On en déduit l'erreur associée à un sous-ensemble de nœuds V_{set} comme la somme des erreurs des nœuds qui le composent :

$$Error_{AE}(V_{set}) = \sum_{\mathbf{v}_i \in V_{set}} error_{AE}(v_i) \tag{7}$$

Notre modèle essaie de minimiser l'erreur de reconstruction de chaque auto-encodeur sur son propre échantillon tout en maximisant l'erreur sur le reste des données étiquetées. Ce qui conduit à la définition de deux fonctions de perte. La fonction de perte pour *Norm* est :

$$Loss_{Norm} = \frac{Error_{Norm}(\mathcal{V}_n)}{Error_{Norm}(\mathcal{V}_s)}. \tag{8}$$

Celle pour *Susp* est :

$$Loss_{Susp} = \frac{Error_{Susp}(\mathcal{V}_s)}{Error_{Susp}(\mathcal{V}_n)}. \tag{9}$$

De cette façon, les nœuds qui sont semblables aux nœuds majoritaires dans leur échantillon respectif obtiennent des scores plus faibles, tandis que les nœuds moins représentés dans l'échantillon obtiennent des scores plus élevés.

Ainsi, la différence majeure de Suspicious par rapport à Dominant réside, non seulement dans son architecture basée sur deux auto-encodeurs, comme illustré sur la Figure 2, alors que Dominant n'en a qu'un mais aussi sur le calcul de la fonction de perte. Alors que Dominant essaye de reconstruire le graphe entier avec une erreur minimale, dans notre modèle, chaque encodeur essaie de minimiser l'erreur de reconstruction par rapport à son propre échantillon.

3.2.4 Score final et critère de décision

Nous normalisons ensuite l'erreur calculée pour chaque nœud par chaque auto-encodeur AE afin que les scores soient tous deux dans l'intervalle $[0, 1]$ et sur une échelle similaire :

$$En_{AE}(\mathbf{v}_i) = \frac{error_{AE}(\mathbf{v}_i) - Min_{\mathbf{v}_j \in \mathcal{V}}(error_{AE}(\mathbf{v}_j))}{Max_{\mathbf{v}_j \in \mathcal{V}}(error_{AE}(\mathbf{v}_j)) - Min_{\mathbf{v}_j \in \mathcal{V}}(error_{AE}(\mathbf{v}_j))}. \tag{10}$$

Nous utilisons ensuite ces erreurs normalisées produites par les deux auto-encodeurs pour calculer un score final qui permettra d'ordonner les nœuds :

$$Ranking_{score}(v_i) = \frac{En_{Norm}(\mathbf{v}_i)}{En_{Susp}(\mathbf{v}_i)}. \tag{11}$$

Grâce à cette opération, nous obtenons un critère de décision où des scores équivalents dans En_{Norm} et En_{Susp} donnent un score final moyen. Des scores faibles dans En_{Norm} et élevés dans En_{Susp} aboutissent à un score final faible. Des scores élevés dans En_{Norm} avec des scores faibles dans En_{Susp} produisent un haut score final. Le score final produit donc bien une liste dans laquelle les nœuds ayant les plus hauts scores correspondent aux anomalies recherchées.

4 Expérimentation

Afin d'évaluer notre cadre méthodologique, nous suivons le protocole expérimental introduit par Kumagai et al. (2021).

4.1 Jeux de données

Les expérimentations sont réalisées sur 5 jeux de données réelles dont les caractéristiques sont résumées dans le Tableau 2

Réseaux de citation : Cora, Citeseer et PubMed sont des réseaux de citation publics très populaires (Sen et al., 2008). Dans ces graphes, chaque nœud est une publication scientifique, et l'arête représente la citation d'une autre publication. Leurs matrices d'attributs correspondent au contenu des publications représenté sous forme vectorielle de sac de mots.

Graphe d'achat en commun : Amazon Photo et Amazon Computers sont également des jeux de données très populaires. Dans ces graphes, chaque nœud est un produit, une arête existe si deux produits sont souvent achetés ensemble ; les matrices d'attributs sont également composées de vecteurs d'attributs de type sac de mots.

Suspicious

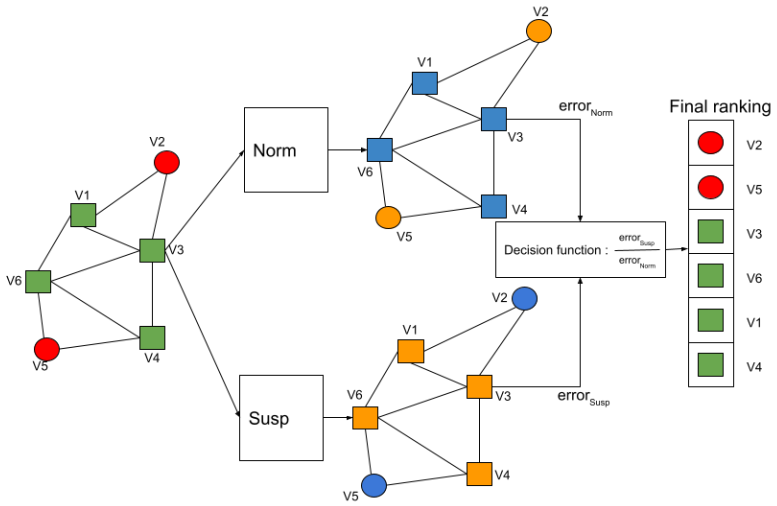

FIG. 2 – *Architecture de Suspicious. Norm et Susp sont des auto-encodeurs semblables à ceux présentés dans la Figure 1. Les nœuds verts sont des nœuds normaux, les nœuds rouges sont des nœuds anomalies, les nœuds orange sont des nœuds mal reconstruits par l'auto-encodeur, et les nœuds bleus sont les nœuds bien reconstruits.*

Jeu de données	Type	Nœuds	Arrêtes	Attributs	Classes	Taux d'anomalies
Cora	Citation	2708	5278	1433	7	0.066
Citeseer	Citation	3327	4732	3703	6	0.079
PubMed	Citation	19717	44338	500	3	0.208
Amazon Photo	Achats groupés	7487	119043	745	8	0.043
Amazon Comp.	Achats groupés	13381	245778	767	10	0.020

TAB. 2 – *Caractéristiques des jeux de données.*

Protocole : Pour adapter ces jeux au problème de détection d'anomalies, nous suivons le protocole de Kumagai et al. (2021) qui consiste à changer les étiquettes de sorte que dans chaque jeu de données, les éléments de la plus petite classe, en effectif, soient ré-étiquetés comme anormaux, tandis que tous les autres sont classés comme normaux. De cette façon, nous disposons d'un jeu adapté à un problème de classification binaire avec déséquilibre de classes. Pour chaque jeu de données, l'échantillon d'apprentissage est construit en choisissant aléatoirement 10 % des nœuds. Afin d'évaluer la résilience des méthodes de détection d'anomalies, nous introduisons ensuite des erreurs d'étiquetage des nœuds appartenant au jeu d'apprentissage en inversant les étiquettes (anomalies/normaux) des nœuds avec une certaine probabilité (le taux d'erreur). Ainsi, nous créons pour chaque jeu, et chaque taux d'erreur de 0 (pas d'erreur d'étiquetage), 10 % (erreur moyenne d'étiquetage), 20 % (forte erreur d'étiquetage) à 30 % (très forte erreur d'étiquetage), 10 échantillons d'apprentissage.

4.2 Méthodes concurrentes

Nous comparons notre cadre méthodologique Suspicious avec les méthodes de l'état de l'art suivantes :

— Méthode Kumagai et al. : une méthode d'intégration de graphe semi-supervisée qui utilise le régularisateur AUC comme support pour minimiser le volume d'une hypersphère qui englobe les nœuds étiquetés normaux (Kumagai et al. (2021)).

— Dominant : une méthode non supervisée de reconstruction de graphe, basée sur des auto-encodeurs, qui calcule les scores d'anomalie comme la somme des erreurs de reconstruction commises sur les attributs et la structure de graphe (Ding et al. (2019)).

4.3 Paramètres et mesure d'évaluation

La méthode Kumagai est implémentée avec les paramètres publiés dans l'article : un plongement de dimension 32, un maximum de répétitions avec un mécanisme d'arrêt précoce, le centre C défini comme la moyenne du plongement des nœuds étiquetés comme normaux après la première couche du modèle. Pour Dominant, nous utilisons l'implémentation de pygod (Liu et al. (2022)) et les paramètres de la publication avec un GCN à deux couches comme encodeur et un GCN à une couche comme décodeur d'attributs, une dimension de plongement de 32, 500 répétitions et α=0.5. Pour notre framework, nous utilisons aussi les mêmes paramètres : une dimension de plongement de 32, un dropout égal à 0.5 pour les deux auto-encodeurs, 500 répétitions et un pas d'apprentissage de 0.005.

Nous mesurons ensuite la performance des trois approches en calculant les scores moyens d'AUC (et l'écart type) obtenus lors du classement des nœuds non étiquetés correspondant aux 10 échantillons d'apprentissage construits pour chaque ensemble de données.

4.4 Résultats

	Suspicious	Dominant	Kumagai
Cora	**95.92(3.4)**	49.47(0.66)	93.59(4.3)
Citeseer	**72.30(1.63)**	40.18(0.09)	67.31(3.9)
PubMed	92.41(0.61)	50.93(0.27)	**94.36(0.35)**
Computers	**99.74(0.0)**	46.45(0.09)	97.21(5.46)
Photo	**96.95(0.3)**	51.89(0.07)	60.86 [2](10.61)
Moy	**91.46(1.33)**	47.78(0.24)	82.67(4.92)

TAB. 3 – *Résultats sur des jeux de données parfaitement étiquetés (0% d'erreur).*

Les Tableaux 3 à 6 correspondent chacun à un taux d'erreur d'étiquetage : 0, 10, 20 et 30% et les résultats sont moyennés sur les nœuds non étiquetés correspondant aux 10 échantillons d'apprentissage.

On peut observer dans le Tableau 3 qu'en cas d'absence d'erreur d'étiquetage, Suspicious obtient des résultats comparables à ceux de Kumagai dans la plupart des cas, voire meilleurs, et

2. Il convient de noter que les performances rapportées dans Kumagai et al. (2021) sont supérieures pour le jeu de données Photo, mais que nous n'avons pas été en mesure de reproduire bien que nous ayons retenu les mêmes paramétrages et protocole.

Suspicious

	Suspicious	Dominant	Kumagai
Cora	**90.42(3.02)**	49.66(0.53)	77.44(12.63)
Citeseer	**67.54(4.49)**	40.26(0.13)	61.55(2.56)
PubMed	91.30(0.10)	50.98(0.02)	**92.81(0.09)**
Computers	**97.56**(1.23)	46.37(0.09)	52.51(7.71)
Photo	**88.68(15.2)**	51.73(0.19)	51.64(8.12)
Moy	**87.1(4.81)**	47.91(0.21)	67.19(6.22)

TAB. 4 – *Résultats sur des jeux de données avec 10% d'erreur d'étiquetage.*

	Suspicious	Dominant	Kumagai
Cora	**82.10(7.9)**	49.46(0.36)	63.88(6.76)
Citeseer	**62.44(4.94)**	40.23(0.1)	59.30(3.2)
PubMed	**89.10(4.67)**	50.87(0.02)	87.59(6.80)
Computers	**93.60**(11.06)	46.35(0.1)	49.13(6.86)
Photo	**78,85(13.69)**	51.75(0.15)	48.83(4.21)
Moy	**81.22(8.45)**	47.73(0.20)	61.74(5.56)

TAB. 5 – *Résultats sur des jeux de données avec 20% d'erreur d'étiquetage.*

largement supérieurs à ceux de Dominant. Les mauvaises performances de Dominant ne sont pas surprenantes dans la mesure où le type d'anomalies recherchées ne correspond pas bien à celle pour laquelle la méthode s'avère la plus efficace, à savoir des nœuds ayant des valeurs d'attributs différentes de celles de leurs voisins.

De plus, lorsque des erreurs sont commises dans l'étiquetage des jeux d'apprentissage, ce qui est fréquent dans la pratique, comme on peut le voir dans le Tableau 4 pour un taux d'erreur de 10 %, dans le Tableau 5 pour un taux de 20 %, puis dans le Tableau 6 pour un taux de 30%, notre modèle souffre de la dégradation de l'ensemble d'entraînement, ce qui se traduit par une diminution des scores d'AUC par rapport à ceux présentés dans le Tableau 3. Mais, il obtient des résultats qui restent satisfaisants et qui sont constamment meilleurs et très supérieurs à ceux des autres méthodes ; ce qui prouve sa résilience à des défauts du jeu d'apprentissage.

4.5 Conclusion

Nous avons proposé un cadre général semi-supervisé pour la détection d'anomalies dans les graphes attribués qui est plus résistant à l'erreur humaine. Nous utilisons des auto-encodeurs

	Suspicious	Dominant	Kumagai
Cora	**69.82(7.06)**	50.07(0.7)	53.19(4.95)
Citeseer	**59.17(6.61)**	40.27(0.09)	57.03(4.28)
PubMed	**89.02(5.21)**	50.69(0.02)	67.55(4.66)
Computers	**79.51(18.25)**	46.34(0.09)	48.61(2.32)
Photo	**63.73(16.85)**	51.76(0.22)	48.69(4.9)
Moy	**72.25(10.79)**	47.82(0.22)	55.01(4.22)

TAB. 6 – *Résultats sur des jeux de données avec 30% d'erreur d'étiquetage.*

entraînés sur un sous-ensemble de nœuds étiquetés pour mieux identifier les anomalies. Nos expériences montrent que, dans le cas où aucune erreur d'étiquetage n'a été commise, les résultats de Suspicious sont comparables à ceux de l'état de l'art actuel et, qu'ils sont systématiquement meilleurs s'il y a eu des erreurs d'étiquetage. Ces résultats confirment la performance et la résilience de Suspicious.

Références

Aggarwal, C. C. (2017). *Outlier Analysis*. Springer International Publishing.

Akcay, S., A. Atapour-Abarghouei, et T. P. Breckon (2019). Ganomaly : Semi-supervised anomaly detection via adversarial training. In *Computer Vision – ACCV 2018*, pp. 622–637.

Akoglu, L. (2021). Anomaly mining - past, present and future. In *IJCAI*, pp. 4932–4936.

Antonellis, I., H. G. Molina, et C. C. Chang (2008). Simrank++ : query rewriting through link analysis of the click graph. *Proceedings of the VLDB Endowment 1*, 408–421.

Chalapathy, R., A. K. Menon, et S. Chawla (2018). Anomaly detection using one-class neural networks. *arXiv :1802.06360 [cs.LG]*.

Chandola, V., A. Banerjee, et V. Kumar (2009). Anomaly detection : A survey. *ACM Comput. Surv. 41*(3), 1–58.

Ding, K., J. Li, R. Bhanushali, et H. Liu (2019). Deep anomaly detection on attributed networks. In *SIAM International Conference on Data Mining*, pp. 594–602.

Esteva, A., B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, et S. Thrun (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 115–118.

Fan, H., F. Zhang, et Z. Li (2020). Anomalydae : Dual Autoencoder for Anomaly Detection on Attributed Networks. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5685–5689.

Grubbs, F. E. (1969). Procedures for detecting outlying observations in samples. *Technometrics 11*(1), 1–21.

Hamilton, W., Z. Ying, et J. Leskovec (2017). Inductive Representation Learning on Large Graphs. In *Advances in Neural Information Processing Systems*, pp. 1025–1035. Curran Associates, Inc.

Interdonato, R., M. Atzmueller, S. Gaito, R. Kanawati, C. Largeron, et A. Sala (2019). Feature-rich networks : going beyond complex network topologies. *Appl. Netw. Sci.*, 4 :1–4 :13.

Jeh, G. et J. Widom (2002). Simrank : A measure of structural-context similarity. In *ACM SIGKDD*, pp. 538–543.

Kipf, T. N. et M. Welling (2017). Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*.

Kumagai, A., T. Iwata, et Y. Fujiwara (2021). Semi-supervised Anomaly Detection on Attributed Graphs. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8.

Li, J., H. Dani, X. Hu, et H. Liu (2017). Radar : Residual Analysis for Anomaly Detection in Attributed Networks. In *Proceedings of the Twenty-Sixth International Joint Conferences on Artificial Intelligence, IJCAI-17*, pp. 2152–2158.

Liu, K., Y. Dou, Y. Zhao, X. Ding, X. Hu, R. Zhang, K. Ding, C. Chen, H. Peng, K. Shu, L. Sun, J. Li, G. H. Chen, Z. Jia, et P. S. Yu (2022). Benchmarking node outlier detection on graphs. *arXiv :2206.10071*.

Lu, Y.-J. et C.-T. Li (2020). GCAN : Graph-aware co-attention networks for explainable fake news detection on social media. In *Association for Computational Linguistics*, pp. 505–514.

Ma, X., J. Wu, S. Xue, J. Yang, C. Zhou, Q. Z. Sheng, H. Xiong, et L. Akoglu (2021). A Comprehensive Survey on Graph Anomaly Detection with Deep Learning. arXiv : 2106.07178.

Pang, G., C. Shen, L. Cao, et A. v. d. Hengel (2021). Deep Learning for Anomaly Detection : A Review. *ACM Computing Surveys*, 1–38.

Peng, Z., M. Luo, J. Li, H. Liu, et Q. Zheng (2018). ANOMALOUS : A Joint Modeling Approach for Anomaly Detection on Attributed Networks. In *IJCAI*, pp. 3513–3519.

Sen, P., G. Namata, M. Bilgic, L. Getoor, B. Galligher, et T. Eliassi-Rad (2008). Collective Classification in Network Data. *AI Magazine 29*(3), 93.

Veličković, P., G. Cucurull, A. Casanova, A. Romero, P. Liò, et Y. Bengio (2018). Graph Attention Networks. *International Conference on Learning Representations (Poster)*.

Wang, D., J. Lin, P. Cui, Q. Jia, Z. Wang, Y. Fang, Q. Yu, J. Zhou, S. Yang, et Y. Qi (2019). A Semi-supervised Graph Attentive Network for Financial Fraud Detection. In *ICDM*, pp. 598–607.

Xu, X., N. Yuruk, Z. Feng, et T. A. J. Schweiger (2007). SCAN : a structural clustering algorithm for networks. In *ACM SIGKDD*, pp. 824–833.

Zhang, C., D. Song, C. Huang, A. Swami, et N. V. Chawla (2019). Heterogeneous graph neural network. In *ACM SIGKDD*, pp. 793–803.

Summary

Graph based anomaly detection is an important task in many real-world domains such as health care, insurance, finance, and cyber-security. Even if existing semi-supervised models have proven to be efficient in identifying anomalies, they assume however that a labeled sample of the network is available but without taking into account the real-world problem of the unreliability of such a sample. In this paper we consider attributed networks and, we propose a new framework based on two graph convolutional (GCN) auto-encodeurs trained following a suspicion mechanism: the first GCN is trained on a sample suspected of being composed of normal entities while the second one on a sample suspected of containing anomalies. The final classification is done by coupling the result of both auto-encodeurs. We demonstrate that our approach obtains at least equivalent performances as state-of-the-art methods in the perfect sample case while being more resilient to the introduction of mistakes in these labeled samples.