Extraction de motifs pour la détection d'anomalies dans des graphes : application à la fraude dans les marchés publics

Lucas Potin*, Rosa Figueiredo*, Vincent Labatut*, Christine Largeron**

* Laboratoire Informatique d'Avignon, F-84911, Avignon, France
{prénom.nom}@univ-avignon.fr,
** Laboratoire Hubert Curien UMR 5516, F-42023, Saint-Etienne, France
christine.largeron@univ-st-etienne.fr,

Résumé. Dans le cadre des marchés publics, il existe plusieurs indicateurs, appelés red flags, permettant d'estimer le risque de fraude. Ces red flags sont calculés en fonction des attributs spécifiques de chaque contrat et sont ainsi dépendants du bon remplissage des notices d'attributions. Nous proposons une méthode basée sur l'extraction de motifs pour la détection d'anomalies dans des graphes. Cette approche générique vise à identifier les sous-graphes associés à la présence de red flags, afin de construire un ensemble de nouveaux indicateurs. Ces motifs peuvent ensuite être utilisés dans les cas où les red flags sont manquants.

1 Introduction

Les marchés publics désignent l'achat de biens et de services par une autorité publique (le client), auprès d'une personne morale de droit public ou privé (le fournisseur). Lorsqu'un marché dépasse les seuils européens, l'appel d'offres ainsi que l'avis d'attribution de ce marché doivent être publiés au *Journal Officiel de l'Union Européenne* (JOUE). La version en ligne de ce journal, appelée *Tenders Electronic Daily*[1] (TED), publie plus de 650 000 avis de marché par an[2]. Par conséquent, le secteur des marchés publics fournit une énorme quantité de données accessibles au public.

Historiquement, les anomalies dans les marchés publics, qui font référence à des comportements suspicieux, sont liées à des caractéristiques spécifiques des contrats, appelées *red flags*, et utilisées comme indicateurs de fraude potentielle (Fazekas et Tóth, 2014; Ferwerda et al., 2017), par exemple, la modification du prix du contrat en cours de procédure, ou la réception d'une seule offre pour un appel d'offre donné (National Fraud Authority, 2016). Mais les informations nécessaires au calcul de ces red flags ne sont pas toujours disponibles. Dans les données françaises du TED, certains attributs sont largement absents (Potin et al., 2022), tel le nombre de réponses à un appel d'offres (vide pour 30% des lots), permettant uniquement le calcul de red flags *partiels*.

1. https://ted.europa.eu/
2. https://ted.europa.eu/TED/main/HomePage.do

Les graphes sont couramment utilisés pour des tâches de détection d'anomalies (Ma et al., 2021; Pourhabibi et al., 2020). Toutefois, dans le domaine de la détection de fraude dans les marchés publics, la plupart des études sont basées sur des données *tabulaires* (Carvalho et al., 2013; Carneiro et al., 2020), c'est-à-dire que chaque contrat est considéré séparément. Seuls quelques rares auteurs tentent de tirer parti des *relations* entre les contrats en adoptant une approche basée sur les graphes. Fazekas et Tóth (2016) proposent le CRI, un score composite de plusieurs red flags, mais n'utilisent les graphes que pour visualiser la distribution de ce score. Wachs et Kertész (2019) considèrent des graphes afin d'estimer la proportion de red flags chez les agents avec les relations les plus fréquentes. Cependant, à notre connaissance, il n'existe pas de méthode dans la littérature, basée sur les graphes pour créer des modèles prédictifs.

Ceci nous amène à proposer une méthode basée sur les graphes pour détecter les anomalies dans les marchés publics. Notre contribution est triple. Premièrement, nous formulons le problème de détection de graphes anormaux comme un problème de classification. Deuxièmement, nous proposons la méthode *PANG* (Pattern-Based Anomaly Detection in Graphs), qui tire parti de l'exploration de motifs pour résoudre ce problème. Troisièmement, nous appliquons cette méthode au domaine de la fraude dans les marchés publics.

2 Formulation du problème

Nous adoptons une approche inspirée de la recherche d'information. De la même manière qu'un document peut être modélisé comme un sac-de-mots, nous proposons de représenter un graphe comme un sac de sous-graphes, appelés *motifs*. Les motifs ont déjà été employés pour représenter un graphe (Acosta-Mendoza et al., 2016). Étant donné un ensemble de graphes, nous construisons un dictionnaire, composé des motifs apparaissant dans ces graphes. Sur la base de ce dictionnaire, chaque graphe peut être représenté comme un vecteur de longueur fixe, utilisable ensuite en entrée de n'importe quel algorithme classique d'apprentissage automatique.

Nous utilisons dans notre travail des graphes attribués, avec la définition suivante :

Définition 1 (Graphe attribué) *Un graphe attribué est défini comme un tuple* $G = (V, E, \mathbf{X}, \mathbf{Y})$ *dans lequel* V *est l'ensemble des* n *sommets,* E *l'ensemble des* m *arêtes de* G*,* $\mathbf{X}$ *une matrice de taille* $n \times d_v$ *dont la ligne* $\mathbf{x}_i$ *est le vecteur de taille* d_v *associé aux attributs du sommet* $v_i \in V$*, et* $\mathbf{Y}$ *une matrice de taille* $m \times d_e$ *dont la ligne* $\mathbf{y}_i$ *est le vecteur de taille* d_e *associé aux attributs de l'arête* $e_i \in E$*.*

Nous disposons d'une collection de tels graphes, comme illustré dans la Figure 1. Chaque sommet possède un attribut (bordeaux ou violet) de même que chaque arête (vert ou rouge). Chaque graphe G possède un label l_G choisi dans $\mathcal{L} = \{A, N\}$, indiquant respectivement un graphe anormal ou normal. Ce label n'est pas connu pour tous les graphes à notre disposition. Soit $\mathcal{G}$ l'ensemble des graphes dont le label est connu. Cet ensemble peut être divisé en deux sous-ensembles disjoints : $\mathcal{G} = \mathcal{G}_A \cup \mathcal{G}_N$ ($\mathcal{G}_A \cap \mathcal{G}_N = \emptyset$). Le sous-ensemble $\mathcal{G}_A$ contient les graphes anormaux tandis que $\mathcal{G}_N$ contient les graphes normaux. Notre objectif est d'entraîner un classifieur en utilisant les labels connus, afin de prédire les labels inconnus pour les graphes sans label.

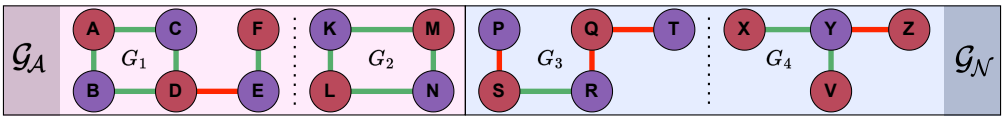

FIG. 1 – Exemple d'une collection de graphes $\mathcal{G}$, incluant les ensembles des graphes anormaux ($\mathcal{G}_A$) et normaux ($\mathcal{G}_N$).

Définition 2 (Motif d'un graphe) *Soit G un graphe attribué. Un graphe P est un motif de G s'il est isomorphe à un sous-graphe P' de G, i.e. $\exists P' \subset G : P \cong P'$*

Comme nous ne considérons que des graphes attribués, nous utilisons la définition de l'isomorphisme de graphe proposée par Hsieh et al. (2006), c'est-à-dire qu'un isomorphisme doit préserver non seulement les arêtes, mais aussi les attributs des sommets et des arêtes. Nous considérons que P est un motif d'un ensemble de graphes $\mathcal{G}$ lorsque P est un motif d'au moins un de ses graphes. La Figure 2 représente trois exemples de motifs de l'ensemble de graphes de la Figure 1.

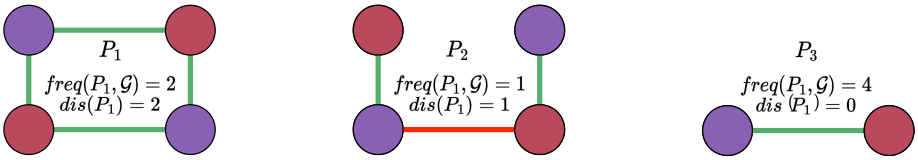

FIG. 2 – Exemple de motifs présents dans le graphe G_1 de la Figure 1.

Nous définissons $\mathcal{P}_A$ et $\mathcal{P}_N$ comme les ensembles de motifs de $\mathcal{G}_A$ et $\mathcal{G}_N$. De plus, nous notons $\mathcal{P} = \mathcal{P}_A \cup \mathcal{P}_N$ l'ensemble de tous les motifs de $\mathcal{G}$. Tous les motifs ne sont pas équivalents. Ainsi, parmi ceux de la Figure 2, P_3 est beaucoup plus présent que P_1 et P_2, dans l'ensemble des graphes de la Figure 1. La principale approche utilisée dans la littérature dans ce cas de figure est d'identifier les motifs *émergents* (Poezevara et al., 2011), c'est-à-dire caractéristiques d'une classe par rapport au reste des données. Nous adoptons une approche similaire, également basée sur un score mesurant le pouvoir discriminant des motifs, mais exploitant les motifs propres à chacune des deux classes. Nous définissons alors la fréquence et le score discriminant d'un motif.

Définition 3 (Fréquence d'un motif) *Soit $\mathcal{G}$ un ensemble arbitraire de graphes attribués. La fréquence $freq(P, \mathcal{G})$ d'un motif P dans $\mathcal{G}$ est le nombre de graphes dans $\mathcal{G}$ ayant P comme motif : $freq(P, \mathcal{G}) = |\{G \in \mathcal{G} : \exists P' \subset G \text{ t.q. } P \cong P'\}|$.*

Définition 4 (Score discriminant) *Étant donné un motif P de $\mathcal{G}$, le score discriminant de P est défini par $dis(P) = |freq(P, \mathcal{G}_A) - freq(P, \mathcal{G}_N)|$.*

Un score proche de 0 indique un motif aussi fréquent dans $\mathcal{G}_A$ que dans $\mathcal{G}_N$, alors qu'un score élevé signifie que le motif est plus fréquent dans l'un des deux sous-ensembles.

Nous utilisons ce score pour classer les motifs de $\mathcal{P}$, et sélectionner les s ($1 \leq s \leq |\mathcal{P}|$) plus discriminants afin de construire l'ensemble des motifs discriminants, noté $\mathcal{P}_s$. Le

paramètre s permet de contrôler la taille de la représentation vectorielle des graphes. Sur la base de $\mathcal{P}_s$, nous construisons une représentation vectorielle $\mathbf{h}_i \in \mathbb{R}^s$ de chaque graphe $G_i \in \mathcal{G}$. Chaque valeur de $\mathbf{h}_i$ mesure l'importance du motif correspondant dans ce graphe spécifique. Nous discutons du calcul de ces valeurs dans la Section 3. Nous obtenons ainsi une matrice $\mathbf{H} \in \mathbb{R}^{|\mathcal{G}| \times s}$ dont la ligne i représente $\mathbf{h}_i^T$. Par conséquent, H_{ij} correspond à la valeur associée au motif P_j dans le graphe G_i.

Sur la base de cette représentation de nos données, notre problème de détection d'anomalies revient à classer des graphes avec des labels inconnus comme anormaux ou normaux. Plus formellement, étant donné, pour chaque graphe $G \in \mathcal{G}$, le label l_G et la représentation vectorielle $\mathbf{h}$, il s'agit d'apprendre une fonction $f : \mathbb{R}^s \rightarrow \{A, N\}$, qui associe un label anormal ou normal à la représentation vectorielle du graphe.

3 Algorithme PANG

Pour résoudre ce problème de classification, nous proposons la méthode PANG [3] (Pattern-Based Anomaly Detection in Graphs), décrite dans la Figure 4.

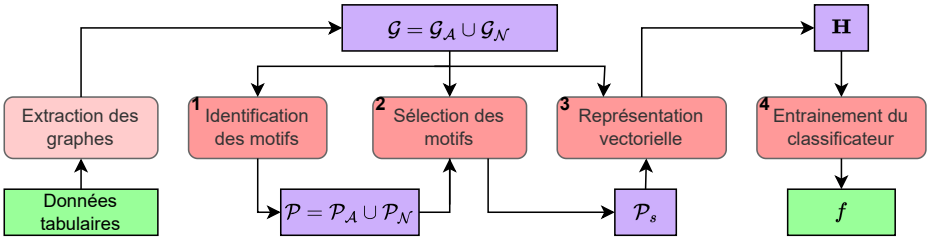

FIG. 3 – Étapes de l'algorithme PANG proposé.

Identification des motifs : Afin de créer $\mathcal{P}$, nous utilisons un outil existant d'extraction de motifs de graphes. Plusieurs outils de ce type sont disponibles (Yan et Han, 2002; Fournier-Viger et al., 2019). Nous choisissons TKG (Fournier-Viger et al., 2019), disponible dans la librairie SPMF [4] (Fournier-Viger et Lin, 2016).

TKG permet de trouver les K motifs les plus *fréquents* dans un ensemble de graphes. Cependant, nous voulons *tous* les motifs, donc nous fixons K de manière à exploiter chaque motif dans notre ensemble de données. TKG s'appuie sur un algorithme itératif, qui part d'un motif fréquent et cherche exhaustivement à l'étendre en ajoutant une arête. Les nouveaux motifs ne sont alors stockés que s'ils sont fréquents.

Sélections des motifs : Après avoir obtenu tous les motifs de $\mathcal{G}$ à l'étape précédente, nous calculons leurs scores discriminants comme expliqué dans la Section 2. Nous conservons les s motifs les plus discriminants afin de construire $\mathcal{P}_s$. Ce paramètre a pour but de limiter la taille de l'espace de représentation.

Représentation vectorielle : Après la création de $\mathcal{P}_s$, nous construisons la représentation vectorielle de chaque graphe dans $\mathcal{G}$. Ici, plusieurs approches sont possibles.

3. `https://github.com/CompNet/Pang`
4. `https://www.philippe-fournier-viger.com/spmf/`

Dans ce travail, nous choisissons de construire un vecteur binaire indiquant la présence ou l'absence de chaque motif dans le graphe considéré. Nous définissons la matrice $\mathbf{H}$ comme suit : pour chaque graphe $G_i \in \mathcal{G}$ et chaque motif $P_j \in \mathcal{P}$, nous attribuons 1 à H_{ij} si ce motif P_j est présent dans G et 0 s'il est absent. Par conséquent, notre représentation actuelle ne permet pas de distinguer les cas où un motif apparaît une ou plusieurs fois dans un graphe. Cependant, elle peut être étendue pour intégrer d'autres pondérations telles que TF-IDF ou BM25 (Amini et Gaussier, 2013). En reprenant les graphes contenus dans la Figure 1 et en utilisant les motifs de la Figure 2 comme $\mathcal{P}_s$, nous obtenons alors $h_1 = (1, 1, 1)$ pour G_1, $h_2 = (1, 0, 1)$ pour G_2, $h_3 = (0, 0, 1)$ pour G_3 et $h_4 = (0, 0, 1)$ pour G_4. Notons qu'avec notre méthode, deux graphes différents, G_3 et G_4, peuvent avoir la même représentation vectorielle.

Apprentissage du classifieur : Après l'étape précédente, chaque graphe est représenté par un vecteur de taille fixe, quel que soit son nombre de sommets ou d'arêtes. Nous utilisons cette représentation pour entraîner un classifieur qui prédit les labels des graphes. Notre méthode supporte tout classifieur, mais nous nous concentrons sur Random Forest (Ho, 1995), qui a donné les meilleurs résultats expérimentaux.

4 Application aux marchés publics

Nous utilisons des données de la base FOPPA (Potin et al., 2022), extraite du site *Tenders Electronic Daily*[5]. Nous nous intéressons au sous-ensemble des contrats publiés en France au cours de la période 2015–2019, contenant 417 809 lots.

Pour chaque municipalité présente dans l'ensemble de données, nous extrayons un sous-ensemble de contrats et construisons un graphe G. Les sommets représentent les agents, avec un attribut pour distinguer un client d'un fournisseur, et les arêtes représentent les contrats entre agents, avec un attribut lié au nombre de lots associés. Cet attribut peut prendre 3 valeurs : exactement un lot, entre 2 et 5 lots, et 6 lots ou plus.

Nous attribuons un red flag à un contrat si son nombre d'offres reçues est strictement égal à 1, ce qui révèle un manque de concurrence. Nous considérons qu'une arête est anormale si elle contient *au moins* un tel contrat. Le label d'un graphe dépend de son nombre d'arêtes anormales : normal (label N) si inférieur à 2, anormal (label A) sinon.

Notre méthode d'extraction produit 389 graphes normaux et 330 anormaux. Afin d'obtenir des classes équilibrées pour calculer la fréquence des motifs sans biais, nous sous-échantillonnons pour garder un nombre égal de graphes normaux et anormaux. Les statistiques des graphes obtenus sont représentées dans la Table 1.

Label du graphe	**Nombre de graphes**	**Nombre moyen de sommets (e-t)**	**Nombre moyen d'arêtes (e-t)**
Anormal	330	15.76 (5.56)	17.09 (7.86)
Normal	330	12.54 (5.41)	12.59 (6.90)

TAB. 1 – Caractéristiques du dataset.

5. `https://ted.europa.eu/`

5 Résultats

Motifs discriminants : Lorsqu'il est appliqué à notre jeu de données, TKG renvoie un nombre total de 15 793 motifs distincts. La Figure 4.a représente la distribution des scores discriminants. Nous observons que la plupart des motifs (85%) ont un score compris dans $[0; 20]$, et peuvent donc être considérés comme non discriminants. La Figure 4.b présente deux exemples de motifs discriminants P_4 et P_5, de scores respectifs 91 et 64. présents dans les graphes de grande taille, plus souvent associés au label A.

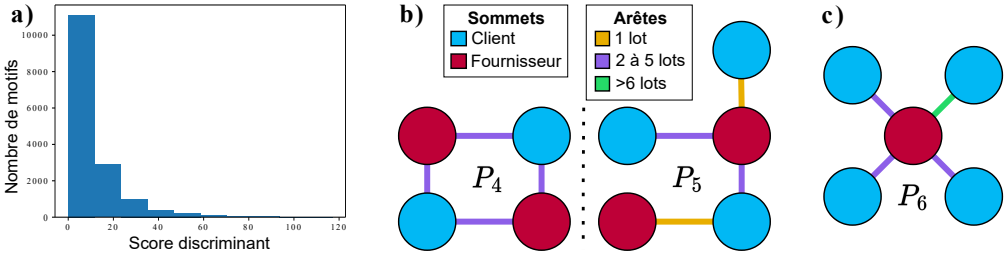

FIG. 4 – (a) Distribution des scores discriminants. (b) Deux exemples de motifs discriminants d'après notre score. (c) Motif lié à du favoritisme.

Nombre de motifs : Nous nous intéressons à la performance de Random Forest en fonction de s, i.e. la taille de la représentation vectorielle. Pour comparaison, nous appliquons également ce classifieur à une représentation à base de plongement de graphes entiers, apprise automatiquement avec Graph2Vec (Annamalai et al., 2017).

La Table 2 indique les résultats pour chaque métrique selon différentes valeurs de s. La ligne *Tous* signifie que tous les motifs disponibles sont considérés comme discriminants ($s = |\mathcal{P}|$). Nous observons que la construction d'une représentation vectorielle avec seulement 100 motifs, c'est-à-dire moins de 1% du nombre total de motifs, qui est égal à 15 793, est suffisante pour obtenir un résultat supérieur à 80% pour chaque métrique et classe. Cela représente plus de 90% du F-Score maximum obtenu avec tous les motifs. L'augmentation de s à 150 nous amène à 95% de cette performance. Nous obtenons alors des performances comparables à Graph2Vec, mais notre méthode a l'avantage d'être plus explicable et interprétable, en permettant l'analyse des motifs.

Nombre	**Graphes anormaux**				**Graphes normaux**			
de motifs	**Pre**	**Rec**	**FS**	**%Max**	**Pre**	**Rec**	**FS**	**%Max**
10	0.69	0.77	0.72	79	0.68	0.59	0.63	69
100	0.84	0.84	0.84	91	0.81	0.81	0.81	88
150	0.89	0.85	0.87	**95**	0.88	0.87	0.87	**95**
Tous	**0.94**	**0.90**	**0.92**	100	**0.89**	**0.93**	**0.91**	100
Graph2Vec	0,88	0.89	0.88	96	0.88	0.86	0.87	95

TAB. 2 – Résultats du classifieur selon la taille de $\mathcal{P}_s$.

Analyses des motifs : Notre méthode permet d'identifier directement les motifs les plus discriminants, et donc de tirer parti de l'expertise humaine pour comprendre

la signification de ces motifs, du point de vue économique. Dans la Figure 4, P_4 et P_6 sont deux exemples de motifs discriminants d'après Random Forest. P_4 est également discriminant selon notre propre score. Ce motif représente une relation entre deux clients et deux fournisseurs, avec quelques contrats (pas seulement un) entre chacun d'eux. Nous supposons que ces motifs se produisent plus fréquemment dans les graphes avec plus de contrats, ce qui est le cas en moyenne pour nos graphes anormaux. Dans le motif P_6, nous observons la présence d'une seule arête verte pour un fournisseur parmi plusieurs clients. Nous pouvons alors interpréter ce phénomène comme du favoritisme : un fournisseur travaille beaucoup plus avec une mairie qu'avec les autres : la mairie est alors plus susceptible de réaliser des appels d'offres sur mesure pour ce fournisseur.

6 Conclusion

Dans cet article, nous proposons PANG, une méthode générique utilisant l'extraction de motifs pour représenter des graphes sous forme de vecteurs, et pour les classifier. Nous l'utilisons ensuite pour détecter des fraudes dans les marchés publics. Nos expériences montrent que PANG est capable d'identifier un ensemble de motifs qui peuvent être utilisés pour représenter chaque graphe sous forme de vecteur pour ensuite classer les graphes sans les red flags. Elle permet également d'associer des motifs à des comportements économiques connus des marchés.

Nous prévoyons d'étendre cette approche de plusieurs manières. Tout d'abord, nous voulons améliorer la représentation vectorielle en exploitant le nombre d'occurrences des différents motifs dans le graphe considéré, au lieu de simplement représenter leur présence/absence, similairement à $TF\text{-}IDF$ ou BM25. Nous prévoyons également de prendre en compte d'autres red flags identifiés dans la littérature.

Références

Acosta-Mendoza, N., A. Gago-Alonso, J. A. Carrasco-Ochoa, J. Francisco Martínez-Trinidad, et J. Eladio Medina-Pagola (2016). Improving graph-based image classification by using emerging patterns as attributes. *Eng Appl Artif Intell 50*, 215–225.

Amini, M. R. et E. Gaussier (2013). *Recherche d'Information - applications, modèles et algorithmes*. Algorithmes. Eyrolles.

Annamalai, N., C. Mahinthan, V. Rajasekar, C. Lihui, L. Yang, et J. Shantanu (2017). graph2vec : Learning distributed representations of graphs.

Carneiro, D., P. Veloso, et A. Ventura (2020). Network analysis for fraud detection in portuguese public procurement. In *IDEAL*, pp. 390–401. Springer.

Carvalho, R. N., S. Matsumoto, K. B. Laskey, P. C. G. Costa, M. Ladeira, et L. L. Santos (2013). Probabilistic ontology and knowledge fusion for procurement fraud detection in brazil. In *URSW II*, pp. 19–40. Springer.

Fazekas, M. et I. J. Tóth (2014). New ways to measure institutionalised grand corruption in public procurement. Technical report, U4 Anti-Corruption Resource Centre.

Fazekas, M. et I. J. Tóth (2016). From corruption to state capture : A new analytical framework with empirical applications from hungary. *PRQ 69*(2), 320–334.

Ferwerda, J., I. Deleanu, et B. Unger (2017). Corruption in public procurement : finding the right indicators. *Eur. J. Crim. Policy Res. 23*(2), 245–267.

Fournier-Viger, P., C. Cheng, L. Chun-Wei J., U. Yun, et R. U. Kiran (2019). TKG : Efficient mining of top-k frequent subgraphs. In *Big Data Analytics*, pp. 209–226.

Fournier-Viger, P. et J. C.-W. Lin (2016). The SPMF open-source data mining library version 2. In *Machine Learning and Knowledge Discovery in Databases*, pp. 36–40.

Ho, T. K. (1995). Random decision forests. In *3rd International Conference on Document Analysis and Recognition*, pp. 278–282.

Hsieh, S.-M., C.-C. Hsu, et L.-F. Hsu (2006). Efficient method to perform isomorphism testing of labeled graphs. In *ICCSA*, pp. 422–431. Springer.

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. *IEEE Transactions on Knowledge and Data Engineering in press*.

National Fraud Authority (2016). Red flags for integrity : Giving the green light to open data solutions. Technical report, Open Contracting Partnership.

Poezevara, G., B. Cuissart, et B. Crémilleux (2011). Extracting and summarizing the frequent emerging graph patterns from a dataset of graphs. *JIIS 37*, 333–353.

Potin, L., V. Labatut, R. Figueiredo, C. Largeron, et P.-H. Morand (2022). FOPPA : a database of french open public procurement award notices.

Pourhabibi, T., K.-L. Ong, B. H. Kam, et Y. L. Boo (2020). Fraud detection : A systematic literature review of graph-based anomaly detection approaches. *Decision Support Systems 133*, 113303.

Wachs, J. et J. Kertész (2019). A network approach to cartel detection in public auction markets. *Scientific Reports 9*, 10818.

Yan, X. et J. Han (2002). gspan : graph-based substructure pattern mining. In *2002 IEEE International Conference on Data Mining*, pp. 721–724.

Summary

In the context of public procurement, several indicators, called red flags, are used to estimate fraud risk. These red flags are calculated according to certain contract attributes and are therefore dependant on the proper filling of the award notices. In this paper, we propose a general framework based on pattern extraction to detect anomalous graphs. It aims to identify subgraph patterns associated with the presence of red flags, in order to construct a set of new red flag indicators. These patterns can then be used in cases where red flags information is missing.