# Détection de communautés multi-relationnelles dans les réseaux sociaux hétérogènes

Soumaya Guesmi, Chiraz Trabelsi, Catherine Berrut, Chiraz Latiri

# Détection de communautés multi-relationnelles dans les réseaux sociaux hétérogènes

**Soumaya Guesmi*** — **Chiraz Trabelsi*** — **Catherine Berrut**** — **Chiraz Latiri***

* *Université Tunis El-Manar, LIPAH, Faculté des Sciences de Tunis, Tunisie*
** *Université Grenoble Alpes, LIG, Équipe MRIM, Grenoble, France*

RÉSUMÉ. *L'explosion des réseaux sociaux a rendu indispensable leur analyse et leur exploration, notamment pour la détection des communautés. Plusieurs méthodes ont été proposées afin de détecter des composantes possédant des propriétés structurelles spécifiques en termes de graphe au détriment de l'aspect sémantique régissant les différents liens entres les entités du réseau. Dans cet article, nous présentons une nouvelle approche pour la détection de communautés dans les réseaux sociaux dont la principale originalité est la prise en considération aussi bien des liens structurels entre les entités que des attributs sémantiques les décrivant. Nous proposons tout d'abord d'exploiter les techniques de l'Analyse Relationnelle de Concepts pour modéliser les différentes interactions et nous introduisons par la suite une nouvelle méthode d'exploration, appelée 'Req_Navigation', pour l'identification des communautés multi-relationnelles. L'étude expérimentale menée sur une collection de données réelles a montré des résultats prometteurs et ouvre plusieurs perspectives.*

ABSTRACT. *The explosion of social networks has made their analysis and exploration indispensable. Especially for the detection of communities. Several community detection methods were therefore proposed in order to detect clusters with specific structural properties the semantic aspect of the different links. This paper present a new approach for the detection of communities in social networks that consider the structural links between the entities as well as the semantic attributes describing them. We first propose to exploit the techniques of the Relational Concept Analysis to model the different interactions, thereafter we introduce a new method of exploration, called 'Req_Navigation', for the identification of multi-relational communities. Carried out experiments on real dataset showed promising results and opens several issues.*

MOTS-CLÉS : *Détection de communautés, Réseaux sociaux, Analyse Relationnelle de Concept.*

KEYWORDS: *Community Detection, Social Networks, Relational Concept Analysis.*

## 1. Introduction

L'objectif principal de la détection de communautés à partir des réseaux sociaux est de créer une partition d'individus (sommets), en prenant en compte des relations qui existent entre eux, de telle sorte que les communautés soient composées de sommets fortement connectés. Dans la littérature, la plupart des travaux de détection de communautés se concentrent sur la structure des liens, en ignorant les propriétés des sommets. Or dans de nombreuses applications, les réseaux sociaux peuvent être représentés par des graphes qui contiennent des sommets qui ont des attributs, et des liens qui décrivent les relations entre ces sommets, par exemple comme présenté dans la figure1 les individus peuvent être des amis, peuvent être intéressés par les mêmes films, et ils peuvent annoter les mêmes films. Par conséquent, ces informations peuvent être prises en compte pour détecter plus efficacement les communautés. Dans ce contexte, un nouveau défi en matière de détection de communautés consiste à combiner ces données relationnelles (les liens) avec les attributs qui décrivent les sommets (les individus). Plusieurs recherches ont tenté, récemment, de s'attaquer à ce problème de classification hybride. En revanche, la combinaison de plusieurs types de données soulève le problème du sens de la classification. En effet, l'utilisation des méthodes traditionnelles, telles que le calcul de distance et la comparaison ne pas être compatible et, par conséquent, conduire à des résultats contradictoires.

Pour cette raison, dans ce travail, nous proposons une nouvelle approche de détection de communautés basée sur les techniques de l'analyse relationnelle de concepts ($ARC$). L'$ARC$ permet de modéliser le réseau social, en regroupant un ensemble d'individus qui partage un ensemble d'attributs (décrivant les sommets) binaires et relationnels (c.à.d les liens). Ainsi, nous proposons un nouvel algorithme de navigation de requête appelé $Req_{Navigation}$ conçu au sein d'une base de données à plusieurs dimensions afin d'explorer le modèle généré et de détecter l'ensemble de communautés répondant aux besoins des utilisateurs. Le reste de cet article est organisé comme suit.

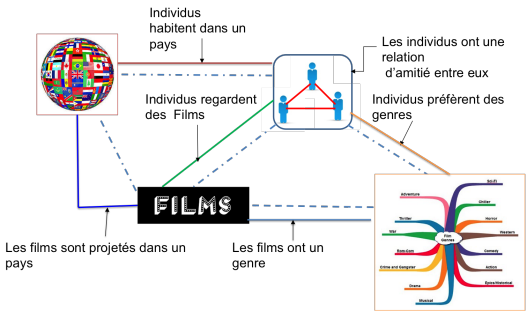


**Figure 1.** *Exemple illustratif d'un réseau hétérogène multi-relationnel.*

Dans la section 2, nous passons en revue les principales approches dédiées à la détection de communautés dans les réseaux hétérogènes. Nous présentons dans la section 3,

les définitions nécessaires à la compréhension de notre approche. Nous consacrons la section 4 pour la description de notre nouvelle approche de détection de communautés multi-relationnelles dans les réseaux hétérogènes. Les résultats des expérimentations menées sur un jeu de données réelles sont présentés dans la section 5. Enfin nous concluons dans la section 6.

## 2. État de l'art

Dans la littérature les approches de détection de communautés sont essentiellement concentrées sur l'analyse des réseaux multi-relationnels homogènes qui contiennent différents types d'arêtes et un seul type de nœuds. Cependant, de nombreux réseaux dans le monde réel sont décrits comme des hypergraphes hétérogènes multi-relationnels qui contiennent différents types de noeuds et de liens. Récemment, de nombreuses recherches ont porté sur la détection de communautés dans des réseaux hétérogènes multi-relationnels. Lin et *al.,* (Lin *et al.*, 2009) ont proposé une méthode basée sur la factorisation des tenseurs, appelée MetaFac (MetaGraph Factorization). Les auteurs supposent qu'une communauté contient des nœuds de différents types, et ils divisent les nœuds en se basant sur ces types. Si on partitionne les nœuds de différents types séparément, cela signifie que les nœuds de différents types ont le même nombre de communautés. Or, cette situation est très rare dans les faits. Dans Sun et *al.,* (Sun *et al.*, 2012), les auteurs ont proposé un algorithme de classification à partir des réseaux hétérogènes général appelé GenClus afin d'intégrer les informations d'attributs incomplètes et les informations de structure de réseau. Zhang et *al.,* (Zhang *et al.*, 2013) ont proposé une méthode basée sur la factorisation matricielle qui combine les contenus générés par l'utilisateur et les réseaux d'amitié pour découvrir des communautés d'utilisateurs partageant les mêmes intérêts. Cependant, l'inconvénient de ces deux approches est que le nombre de communautés, $k$, doit être connu à l'avance. Cependant, dans plusieurs applications, la valeur de $k$ est inconnue. Cela limite leur utilisation dans un système réel. Les auteurs de (Liu *et al.*, 2014) ont proposé une nouvelle approche pour détecter les communautés dans un réseau hétérogène multi-relationnel en se basant sur l'optimisation de la modularité. Cependant, dans (Fortunato et Barthélemy, 2007), les auteurs ont montré que les algorithmes fondés sur l'optimisation de la modularité souffrent d'une limite de résolution. Ces derniers ne peuvent pas distinguer des communautés plus petites d'une certaine taille limite. D'autre part, les auteurs dans (Chen *et al.*, 2015) ont montré que la maximisation de la modularité n'a pas seulement tendance à fusionner les petits groupes, mais aussi à éclater des grandes communautés, et il semble impossible d'éviter simultanément les deux problèmes.

Plusieurs approches ont ainsi été proposées dans la littérature pour la détection des communautés dans les réseaux hétérogènes multi-relationnels. Cependant, elles se sont principalement focalisées sur les propriétés topologiques de ces réseaux tout en ignorant l'information sémantique intégrée. Pour remédier à cette limitation, plusieurs travaux ont utilisé les techniques de l'analyse formelle de concepts (AFC) dans la classification conceptuelle. En effet, l'utilisation de l'AFC vise à extraire des communautés qui préservent les connaissances partagées dans chaque communauté. Dans

ces approches, les entrées sont des graphes bipartis et la sortie est un treillis de Galois qui révèle des communautés sémantiquement définies avec leurs connaissances partagées ou leurs attributs communs (Crampes et Plantié, 2012). Les sommets sont conçus comme des extensions du treillis et les liens sont étiquetés par des intentions du treillis (*i.e.,* connaissances partagées). Cependant, le treillis de Galois n'est pas suffisant dans la mesure où on peut obtenir un nombre exponentiel de communautés. Par conséquent, des méthodes de réduction doivent être introduites. Très peu de recherches se sont concentrées sur cette difficulté (Plantié et Crampes, 2013). Les auteurs dans (Roth *et al.*, 2008) ont utilisé la méthode iceberg ainsi que la méthode de stabilité comme méthodes de réduction de treillis de Galois. Les auteurs dans (Jay *et al.*, 2008) identifient des concepts avec des intentions fréquentes au-dessus d'un seuil fixé. La principale limite de cette approche est que certains concepts importants peuvent être négligés. Brandes et al. (Brandes *et al.*, 2008) combinent les méthodes de l'iceberg et de la stabilité, ils ont montré que cette approche donne de bons résultats pour extraire des communautés pertinentes basées sur des concepts.

Comme il est décrit dans la revue menée par Planti et Crampes (Plantié et Crampes, 2013), la détection de communautés basée sur les techniques de l'$AFC$ est plus efficace du point de vue sémantique, car elle extrait les communautés en utilisant leur sémantique précise. Toutefois, l'$AFC$, ne prend pas en considération les relations qui peuvent exister entre les objets dans le processus de construction des treillis de concepts. Les auteurs dans (Huchard *et al.*, 2007), ont proposé l'Analyse Relationnelle de Concepts ($ARC$) comme une extension relationnelle de l'$AFC$. L'$ARC$ permet ainsi d'injecter des liens inter-objets dans le processus de construction des treillis de façon à ce que les descriptions des concepts renferment une partie relationnelle inférée à partir du partage des liens. Dans ce contexte, nous introduisons dans cet article une nouvelle approche pour la détection de communautés multi-relationnelles basée sur l'$ARC$. Afin de connaître la structure de la communauté multi-relationnelle à travers les différentes dimensions du réseau, nous analysons les informations à partir de toutes les dimensions. En particulier, nous proposons d'abord d'utiliser les techniques de l'$ARC$ (Huchard *et al.*, 2007 ; Rouane-Hacene *et al.*, 2013) pour modéliser les différentes relations et entités intégrées dans les réseaux multi-relationnels. Ensuite, nous introduisons un nouvel algorithme, appelé $Req_{Navigation}$, basé sur l'exploration des treillis de Galois générés pour extraire les communautés multi-relationnelles.

## 3. Définitions préliminaires

Nous introduisons dans cette section les principales définitions utilisées dans la suite de cet article.

**A. Analyse formelle de concepts (AFC) :** L'AFC est une approche mathématique qui permet d'obtenir des concepts hiérarchiquement structurés à partir d'un ensemble d'entités en regroupant un ensemble d'objets partageant le même ensemble d'attributs. La hiérarchie résultant de l'AFC est appelée treillis de Galois ou treillis de concepts (Ganter et Wille, 1999). Nous rappelons dans ce qui suit les notions de base de l'AFC.

**Définition 1 (CONTEXTE FORMEL)** *Un contexte formel est un triplet $\mathcal{K} = (O, A, I)$ où $O$ est un ensemble d'objets, $A$ est un ensemble d'attributs et $I$ est une relation binaire entre $O$ et $A$ appelée relation d'incidence de $\mathcal{K}$ et vérifiant $I \subseteq O \times A$. Un couple $(o, a) \in I$ (noté aussi $oIa$) signifie que l'objet $o \in O$ possède l'attribut $a \in A$.*

Le contexte formel est représenté par un tableau où les lignes correspondent aux objets et les colonnes aux attributs. Le signe « $\times$ » dans un contexte formel signifie que l'objet $o$ est en relation $I$ avec l'attribut $a$.

**Définition 2 (OPÉRATEUR DE DÉRIVATION DE GALOIS)** *: Soit $\mathcal{K} = (O, A, I)$ un contexte formel. Pour tout $X \subseteq O$ et $Y \subseteq A$, on définit :*

$$X' = \{a \in A \,\forall\, o \in O, oIa\}, \quad Y' = \{o \in O \,\forall\, a \in I, oIa\}$$

$X^{'}$ correspond à l'ensemble des attributs communs entre tous les objets de $X$, alors que $Y^{'}$ correspond à l'ensemble des objets communs entre tous les attributs de $Y$. Cet opérateur de dérivation, introduit dans (Ganter et Wille, 1999), permet d'extraire les concepts formels.

**Définition 3 (CONCEPT FORMEL)** *: Soit $\mathcal{K} = (O, A, I)$ un contexte formel. Un concept formel est un couple $(X, Y)$ tel que $X \subseteq O, Y \subseteq A, X' = Y$ et $Y' = X$. $X$ et $Y$ sont respectivement appelées extension ($extent$) et intension ($intent$) du concept formel $(X, Y)$. L'ensemble des concepts formels associés au contexte formel $\mathcal{K} = (O, A, I)$ est noté par $\mathcal{C}(O, A, I)$ ou simplement $\mathcal{C}_\mathcal{K}$.*

Un concept formel est également appelé rectangle maximal dans la mesure où tous les objets de l'extension d'un concept formel possèdent les attributs de son intention.

**Définition 4 (RELATION DE SUBSOMPTION)** *: Soient $(X1, Y1)$ et $(X2, Y2)$ deux concepts formels de $\mathcal{C}_\mathcal{K}$. $(X1, Y1) \leq (X2, Y2)$ si et seulement si $X1 \subseteq X2$ (ou de façon duale $X2 \subseteq X1$). $(X2, Y2)$ est dit super-concept de $(X1, Y1)$ et $(X1, Y1)$ est dit sous-concept de $(X2, Y2)$. La relation "$\leq$" est dite relation de $subsomption$.*

La relation de subsomption « $\leq$ » correspond soit à une généralisation, soit à une spécialisation. Un super-concept est ainsi considéré comme étant un concept général par rapport à ses sous-concepts, car son extension inclut les extensions de ces sous-concepts. D'une façon duale, un sous-concept est considéré comme un concept spécifique par rapport à ses super-concepts. Cette relation permet de construire une hiérarchie de concepts formels appelée, treillis de concepts.

**Définition 5 (TREILLIS DE CONCEPTS)** *: La relation "$\leq$" permet d'organiser les concepts formels en un treillis complet $(\mathcal{C}_\mathcal{K}, \leq)$ appelé treillis de concepts ou encore treillis de Galois et noté par $\mathcal{L}(\mathcal{C}_\mathcal{K})$ ou $\mathcal{L}_\mathcal{K}$. L'infimum et le supremum dans $\mathcal{L}_\mathcal{K}$ sont donnés par :*

$$\bigwedge_{j \in J} (X_j, Y_j) = (\bigcap_{j \in J} X_j, (\bigcup_{j \in J} Y_j)'')$$

$$\bigvee_{j \in J} (X_j, Y_j) = ((\bigcup_{j \in J} X_j)'', \bigcap_{j \in J} Y_j)$$

L'AFC, définie à la base pour représenter et explorer des données du monde réel, ne prend pas en considération les relations qui peuvent exister entre les objets dans le processus de construction des treillis de concepts. L'Analyse Relationnelle de Concepts a été ainsi introduite pour pallier cette principale limite.

**B. L'Analyse Relationnelle de Concepts (ARC) :** L'ARC est une extension relationnelle de l'AFC (Rouane-Hacene *et al.*, 2013). Elle traite de la modélisation des relations entre des ensembles d'objets décrits par leurs attributs. En effet, l'ARC introduit deux grandes étapes complémentaires, à savoir : une étape de **représentation des contextes** et une étape d'**échelonnage relationnel des contextes**.

**1- Représentation des contextes** : L'ARC permet d'intégrer les liens inter-objets partagés entre les concepts dans le processus de construction des treillis de concepts. Partant de cette notion de partage de liens, les concepts formels extraits par l'AFC sont enrichis par des relations reliant d'autres concepts formels. L'ARC génère ainsi une Famille de Contextes Relationnels ($FCR$), à partir d'un ou de plusieurs contextes binaires ($objets \times attributs$), *i.e.*, $\mathbb{K} = \{\mathcal{K}_i\}_{i=1..n}$, et d'un ensemble de relations ($objets \times objets$), *i.e.*, $\mathbb{R} = \{r_k\}_{k=1..m}$, représentant les relations d'incidence entre les ensembles d'objets de $\mathbb{K}$. Cette $FCR$ constitue le point de départ du processus itératif de construction des structures conceptuelles correspondantes appelées Famille de Treillis Relationnels ($FTR$). Formellement, une $FCR$ est définie comme suit :

**Définition 6 (FAMILLE DE CONTEXTES RELATIONNELS)** *: Une $FCR$ est une paire $(\mathbb{K}, \mathbb{R})$ avec :*

*– $\mathbb{K}$ est un ensemble de contextes formels $\mathcal{K}_i = (O_i, A_i, I_i)$,*

*– $\mathbb{R}$ est un ensemble de relations $r_k \subseteq O_i \times O_j$ où $O_i$ et $O_j$ sont des ensembles d'objets de certains contextes de $\mathbb{K}$.*

Pour chaque relation $r_k \subseteq O_i \times O_j$ de $\mathbb{R}$, $r_k$ est une relation orientée tel que $O_i$ représente le domaine de $r_k$, *i.e.*, $dom(r)$ ; et $O_j$ représente le co-domaine, *i.e.*, $ran(r)$. Les objets de $O_i$ appartiennent à $\mathcal{K}_i$ alors que les objets de $O_j$ appartiennent à $\mathcal{K}_j$. La fonction $rel$ permet ainsi de définir l'ensemble des relations qui ont pour source le contexte $\mathcal{K}_i$.

**Définition 7 (FONCTION DE CONTEXTE REL($\mathcal{K}$))** *: La famille des relations qui ont pour domaine un contexte $\mathcal{K}$ est définie par :*

$$rel : \mathbb{K} \rightarrow \mathcal{B}(\mathbb{R}),\ \ rel(\mathcal{K} = (O, A, I)) = \{r \in \mathbb{R} \mid dom(r) = O\}$$

La construction de l'ensemble des treillis de concepts relationnels associés à une $FCR$ est un processus itératif avec une condition d'arrêt qui alterne la construction de treillis et l'enrichissement des contextes formels à travers l'échelonnage relationnel.

**2- Échelonnage relationnel des contextes** : cette étape consiste à ajouter les concepts du treillis comme nouveaux attributs à un contexte formel. L'ajout d'un

concept à un contexte est effectué lorsqu'un objet du contexte est en relation avec l'un des objets appartenant à l'extension du concept considéré. Il existe deux types d'échelonnage relationnel : existentiel ($\exists$) et universel ($\forall$). Dans l'approche que nous proposons dans cet article, nous utilisons l'échelonnage existentiel défini comme suit (Rouane-Hacene *et al.*, 2013) :

**Définition 8 (ÉCHELONNAGE EXISTENTIEL)** *: Soit une relation $r \in rel(\mathcal{K})$ et un treillis $\mathcal{L}_j$ correspondant à $\mathcal{K}_j = (O_j, A_j, I_j)$ cible de $r$, l'opérateur de codage existentiel $\mathbb{S}_{(r,\exists),\mathcal{L}_j}$ fait correspondre à $\mathcal{K}$ le contexte $\mathcal{K}^+ = (O^+, A^+, I^+)$ tel que :*

– $O^+ = O$,

– $A^+ = \{\exists r : c \mid c \in \mathcal{L}_j\}$ *où tous les* $\exists r : c$ *sont des attributs relationnels,*

– $I^+ = \{(o, \exists r : c) \mid o \in O, c \in \mathcal{L}_j, r(o) \cap extension(c) \neq \varnothing\}$.

Le processus général de construction des treillis de concepts relationnels (Rouane-Hacene *et al.*, 2013) prend en entrée une famille de contextes relationnels $FCR = (\mathbb{K}, \mathbb{R})$ et fournit en sortie une famille de treillis relationnels ($FTR$). Le processus commence dans un premier temps par une étape d'initialisation qui permet de construire des treillis initiaux ${\mathcal{L}_i}^0$ des contextes formels ${\mathcal{K}_i}^0$ de la $FCR$ en considérant les objets formels avec leurs attributs binaires et en ignorant toute information relationnelle. Dans un deuxième temps et durant l'étape d'échelonnage relationnel, il traduit les liens entre les objets dans chaque contexte ${\mathcal{K}_i}^{p-1}$, en des attributs classiques de l'AFC en partant des treillis construits à l'étape précédente ${\mathcal{L}_i}^{p-1}$ ainsi que de l'ensemble de ses relations $rel(\mathcal{K}_i)$ (décrites par les contextes relationnels). Les contextes ${\mathcal{K}_i}^p$ sont produits par l'ajout de ces attributs aux contextes ${\mathcal{K}_i}^{p-1}$ de l'étape précédente. Par la suite, les treillis enrichis ${\mathcal{L}_i}^p$ sont construits à partir des contextes ${\mathcal{K}_i}^p$.

## 4. Nouvelle approche de détection de communautés multi-relationnelles

Nous introduisons dans cette section notre nouvelle approche pour la détection de communautés multi-relationnelles dans les réseaux hétérogènes. L'architecture générale de notre approche est illustrée par la figure 2. Elle se compose de deux phases à savoir : (*i*) **une phase de modélisation** du réseau hétérogène basée sur l'exploitation de l'ARC ; et (*ii*) **une phase de navigation** entre les treillis de concepts pour la détection de communautés multi-relationnelles. Nous introduisons durant cette phase un nouvel algorithme, appelé $Req_{Navigation}$, pour la détection de communautés dans le réseau hétérogène précédemment modélisé.

### 4.1. *Phase de modélisation*

Durant la phase de modélisation nous nous concentrons sur trois principales notions : le contexte objet, le contexte relation et la famille de treillis relationnels ($FTR$). En effet, en se basant par exemple sur le réseau hétérogène multi-relationnel, illustré par la figure 1, représentant une partie de la collection de films (exploitée pour notre étude expérimentale), nous pouvons relever les relations suivantes : un ensemble

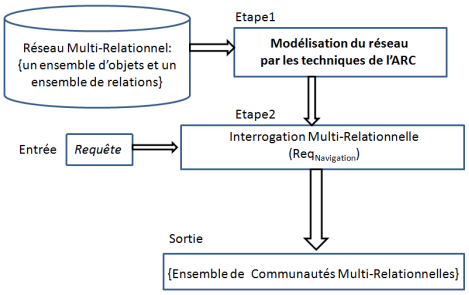


**Figure 2.** *Architecture générale de l'approche proposée.*

de personnes U= $\{u_1, u_2, \ldots, u_n\}$, peut habiter dans un pays P=$\{p_1, p_2, \ldots, p_m\}$ et peut également regarder les mêmes films F=$\{f_1, f_2, \ldots, f_k\}$. Par ailleurs, les films peuvent partager les mêmes genres G=$\{g_1, g_2, \ldots, g_l\}$) et peuvent également être projetés dans les mêmes pays P (ou encore peuvent être annotés par les mêmes Tag T=$\{t_1, t_2, \ldots, t_h\}$).

Afin de modéliser les différentes relations enfouies dans le réseau hétérogène multi-relationnel, nous représentons un contexte objet comme étant un ensemble d'objets de même type, *e.g.*, le contexte personne ($\mathcal{K}_{Personne}$) et le contexte film ($\mathcal{K}_{Film}$). Le contexte relation, modélisera ainsi les interactions entre les contextes objets, *e.g.*, la relation $A\_Regardé$ ($r_{A_Regardé}$) entre personne. A partir de ces contextes objets et relations, nous construisons la $FTR$ pour représenter le réseau multi-relationnel. Nous considérons ainsi 4 contextes objets, *i.e.*, $\mathcal{K}_{Utilisateur}$, $\mathcal{K}_{Film}$, $\mathcal{K}_{Genre}$ et $\mathcal{K}_{Pays}$. Chaque contexte objet est représenté par une matrice unité. Par ailleurs, la figure 3 illustre, les 3 contextes relations, *i.e.*, $r_{A\_Regardé}$, $r_{A\_Genre}$ et $r_{Habite}$, construits à partir de l'exemple considéré. Reprenons les exemples des 4 contextes objets et des 3 contextes relations, la $FTR$ construite à partir de ces contextes est composée de quatre treillis : les treillis modélisant le contexte Personne et le contexte Pays, illustrés par la figure 4, et les treillis représentant le contexte Film et le contexte Genre, illustrés par la figure 5.

### 4.2. *Phase de navigation*

La $FTR$ que nous obtenons après la première phase de modélisation représente une structure riche qui permet de prendre en compte les relations inter-concepts induites à partir des liens inter-objets. Ainsi, suite à la soumission d'une requête utilisateur, la $FTR$ nous permet d'identifier des communautés multi-relationnelles à travers l'interrogation des treillis. La phase de navigation se déroule en deux étapes : une première étape de transformation de requête et une seconde étape d'interrogation.

La première étape, consiste à transformer chaque requête en un chemin de requête $CQ$ qui dirige la navigation dans la $FTR$. En effet, un chemin de requête $CQ$ est

r_A_Regardé

| A_Regardé | Film 1 | Film 2 | Film 3 | Film 4 | Film 5 | Film 6 | Film 7 |
|---|---|---|---|---|---|---|---|
| Jack | x | x | | | | | |
| Alexender | x | x | | | | | |
| Jakline | x | x | | | | | |
| John | | | x | x | | | |
| Mariya | | | x | x | | | |
| Robert | | | x | x | | | |
| Henrique | | | | | x | x | x |
| Sofia | | | | | x | x | x |
| Peter M.Maurer | | | | | x | x | x |

r_Habite

| Habite | Chine | Amérique | Comédie | France |
|---|---|---|---|---|
| Jack | x | | | |
| Alexender | x | | | |
| Jakline | x | | | |
| John | | x | | |
| Mariya | | x | | |
| Robert | | | x | |
| Henrique | | | x | |
| Sofia | | | | x |
| Peter M.Maurer | | | | x |

r_A_Genre

| A_Genre | Action | Romantique | Science Fiction | Comédie |
|---|---|---|---|---|
| Film1 | x | | | |
| Film2 | x | | | |
| Film3 | | x | | |
| Film4 | | x | | |
| Film5 | | | | x |
| Film6 | | | x | |
| Film7 | | | X | |

**Figure 3.** *Les contextes relationnels extraits à partir d'un réseau hétérogène multi-relationnel.*

l'ordre inverse de la requête relationnelle $RR$ qui est composée de plusieurs requêtes simples $RS$ définies comme suit :

**Définition 9 (REQUÊTE SIMPLE($RS$))** *Soit un contexte $K = (A, O, R)$, une Requête Simple ($RS$), notée $RS$=$\{o_{req}\}$, est un ensemble d'objets avec $o_{req} \subset O$.*

Une requête relationnelle se présente alors comme un ensemble de requêtes simples et un ensemble de relations entre les requêtes simples.

**Définition 10 (REQUÊTE RELATIONNELLE($RR$))** *Une Requête Relationnelle $RR$=$\{req_0, req_1, \ldots, req_m\}$ appliquée à une $FTR$ est un triplet $RR$=$(req'_s, r_{sc}, req'_c)$ tel que :*

*– $req'_s$ et $req'_c$, requête source et requête cible respectivement, sont des requêtes simples $RS$.*

*– $r_{sc}$ est la relation entre $req'_s$ et $req'_c$. Elle permet un mapping un-à-un entre $req'_s$ et $req'_c$.*

Afin d'explorer la $FTR$, nous devons construire le chemin de requête $CR$ correspondant. En effet, $CR$ permet de connaître le chemin à suivre à travers la spécification des treillis sources et cibles.

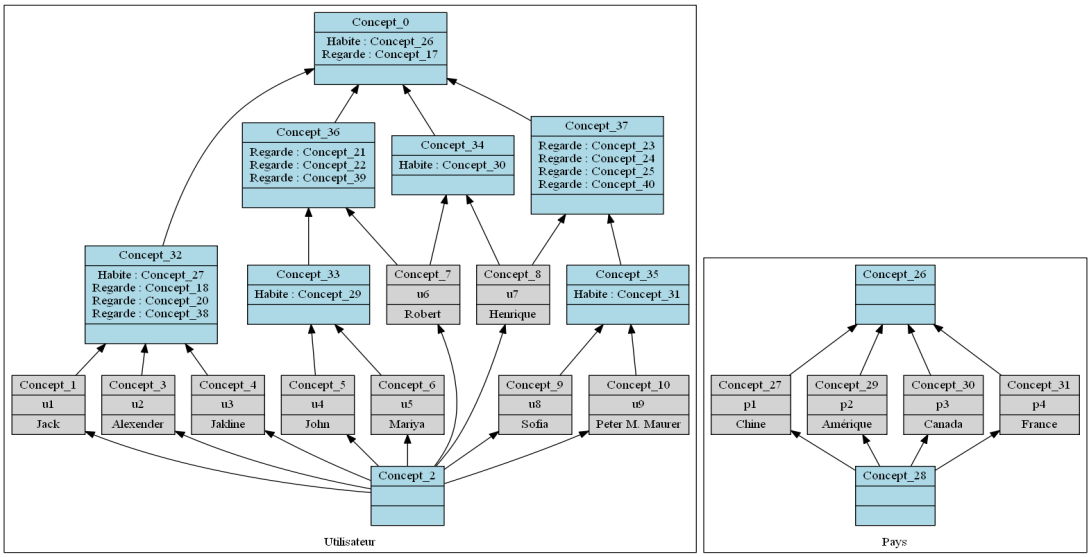


**Figure 4.** *Treillis d'Utilisateur et de Pays.*

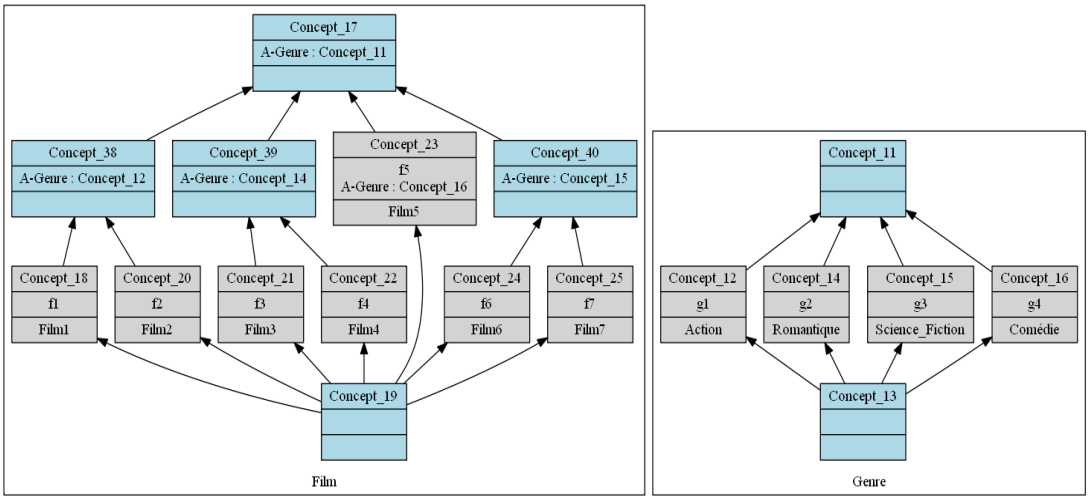


**Figure 5.** *Treillis de Film et de Genre.*

**Définition 11 (CHEMIN DE REQUÊTE)** *Soit* $CR$=$\{cr_0, cr_1, \ldots, cr_n\}$ *et* $cr_i$ *est une paire* $((req_s,T_s),(req_c,T_c))$ *avec* $T_s$ *et* $T_c \subset FTR$, *les treillis sources et cibles respectivement. Le chemin de requête* $CR$ *est l'ordre inverse de la requête relationnelle. C'est à dire* $cr_0 = rr_m$ *et* $cr_n = rr_0$ *; avec* $req_{s0} = req'_{cm}$ *et* $req_{c0} = req'_{sm}$.

**Algorithme 1 :** $Req_{Navigation}$

**Entrées** : **-Un chemin de requête** $CR$= {$cr_i$} **avec cr=(**($req_s$,$T_s$)**,**($req_c$,$T_c$)**)**
**Sorties** : **-Réponse à la requête utilisateur**

```
début                                                                    1
    pour chaque cr dans CR faire                                         2
        si i=0 alors                                                     3
            o_s := req_s;                                                4
            pour chaque Concept dans T_s faire                           5
                si o_s ⊆ Extension(Concept) alors                        6
                    req_c := req_c ⋃ C;                                  7
        pour chaque Element dans req_c faire                             8
            pour chaque Concept dans T_c faire                           9
                si Element ⊆ Intension(Concept) alors                    10
                    Obj := Obj ⋃ Extension(Concept);                     11
                    req_c := req_c ⋃ Concept;                            12
    retourner (Obj);                                                     13
```

La seconde étape de la phase de navigation consister à trouver tous les objets qui sont pertinents par rapport à la requête transformée. Pour cela nous introduisons un nouvel algorithme appelé $Req_{Navigation}$ qui permet de naviguer entre les différents treillis en se basant sur les chemins de requête $CR$. Le pseudo-code de $Req_{Navigation}$ est donné par l'algorithme 1. $Req_{Navigation}$ prend comme entrée tous les chemins de requête $CR$=$\{cr_i\}$ avec $cr$=(($req_s$, $T_s$), ($req_c$, $T_c$)) et renvoie la communauté identifiée qui répond à la requête utilisateur $Req$.

Le processus d'interrogation $Req_{Navigation}$ commence par explorer tous les concepts du treillis source $T_s$, afin d'extraire les concepts correspondants ($Concept_i$) qui répondent au chemin de requête initial $cr_0$. Tout d'abord, il exploite les extensions des concepts du treillis $T_s$ puis il extrait les concepts qui contiennent une extension liée à la requête $req_s$. Le résultat de la phase initiale est un ensemble de concepts $Concept_i$ qui répond à la requête $req_s$.

La deuxième phase de $Req_{Navigation}$ consiste à générer itérativement un ensemble de concepts contenant l'ensemble de concepts ($Concept_i$) extraits dans la phase initiale. Il consiste à explorer l'intention du concept du treillis $T_c$ afin d'extraire l'ensemble de concepts ($Concept_{i+1}$) contenant le concept $Concept_i$.

Finalement, s'il n'y a plus de chemin de requête à explorer, $Req_{Navigation}$ effectue l'intersection des derniers concepts sélectionnés ($Concept_k$) et extrait ses extensions. Ce résultat représente l'ensemble des individus qui forme la communauté

multi-relationnelle retournée à l'utilisateur.

**Exemple :** Prenons l'exemple de la requête suivante : $Req$ = *Je cherche la communauté d'utilisateurs qui habitent en* $Amérique$ *et regardent un film du genre* $Romantique$. Durant la première étape, $q$ est transformée en une requête relationnelle comme suit :
$RR$=(($req_{Utilisateur}$, $'Habite'$, $req_{Pays}$); ($req_{Utilisateur}$, $'Regarde'$, $req_{Film}$); ($req_{Film}$, $'A-Genre'$, $req_{Genre}$)) avec $req_{Pays}$='Amérique' et $req_{Genre}$='Romantique'. Les chemins de requête correspondant sont :
$CQ_1 = (req_{Pays}, T_{Pays}), (req_{Utilisateur}, T_{Utilisateurs})$;
$CQ_2$=(($req_{Genre}$, $T_{Genre}$), ($req_{Film}$, $T_{Film}$); ($req_{Film}$, $T_{Film}$), ($req_{Utilisateur}$, $T_{Utilisateur}$)).

Dans ce cas, l'algorithme $Req_{Navigation}$ commence par répondre à la requête $CQ_1$ en explorant dans un premier temps toutes les extensions des concepts dans le treillis source $T_{Genre}$, afin d'extraire les concepts correspondants et qui permettent de répondre à la requête $req_{Pays}$. En considérant les figures 4 et 5, le résultat de la première requête est le $Concept\_29$ qui contient 'Amérique' comme extension. La deuxième étape consiste à explorer le treillis $T_{Utilisateur}$ afin d'extraire l'ensemble de concepts contenant le $Concept\_29$ dans leurs intentions. Le résultat de cette requête est l'ensemble $E_1$ constitué des deux concepts : $E_1$={$Concept\_5$, $Concept\_6$}. $Req_{Navigation}$ applique le même processus d'exploration afin d'explorer les treillis et d'extraire les concepts qui répondent à la deuxième requête $CQ_2$. Dans ce cas le résultat de cette requête est l'ensemble $E_2$ constitué des deux concepts : $E_2$={$Concept\_5$,$Concept\_6$}. Une fois que $Req_{Navigation}$ a considéré tous les chemins de requêtes, il effectue l'intersection de tous les résultats et extrait les extensions des concepts obtenus pour répondre finalement à la requête de l'utilisateur. Le résultat final ainsi obtenu est donné par : Extension($E_1 \bigcap E_2$)={$Concept\_5$,$Concept\_6$}={John, Mariya}.

## 5. Étude expérimentale

### 5.1. *Description du jeu de données*

Nous avons utilisé deux jeux de données issues de la collection de films Movielens : $Movielens1$ [1] et Movielens2 [2]. Nous avons utilisé $Movielens1$ pour extraire les informations associées aux films ('User', 'Movie', 'Genre', 'Director', 'Country', 'Tag'). Nous avons par la suite effectué un matching avec la collection Movielens2 afin d'extraire le profil utilisateur ('User', 'Movie', 'Âge', 'Sexe', 'Occupation', 'Zipcode'). Le jeu de données final ainsi obtenu contient : 500 utilisateurs, 7 intervalles d'âges, 21 métiers, 1070 films publiés entre les dates 1992 et 1995, 27 descripteurs (tags), 14 pays et 18 genres.

1. http ://grouplens.org/datasets/hetrec-2011/
2. http ://grouplens.org/datasets/movielens/

### 5.2. *Évaluation de notre approche*

Afin de valider notre approche, nous avons dans un premier temps tester ses performances pour l'extraction des relations cachées entre les utilisateurs. Nous avons par la suite évaluer l'efficacité de notre approche en considérant 3 catégories de requêtes, à savoir :
$Q1$ : concerne $4$ entités, *i.e.*, User, Movie, Date et Genre ; 3 relations, *i.e.*, Watch, has_Date et has_Genre.
$Q2$ : concerne 3 entités, *i.e.*, User, Movie et Country ; 2 relations, *i.e.*, Watch et has_Country.
$Q3$ : concerne $4$ entités, *i.e.*, User, Movie et Occupation, Genre ; 3 relations, *i.e.*, has_Occupation, Watch et has_Genre.
Nous nous concentrons dans cette partie, uniquement sur les deux premières requêtes relationnelles $Q1$ et $Q2$. Comme illustré dans la figure 6, nous avons calculé les tailles des communautés extraites par notre approche, c.à.d le nombre d'individus par communauté, et les attributs (labels) partagés par chaque communauté pour les requêtes $Q1$ et $Q2$. Nous pouvons remarquer, que les communautés détectées ont des tailles différentes : la taille des communautés extraites à partir de la requête $Q1$ varient entre 3 et $78$ individus par communauté et entre 3 et $150$ individus par communautés pour la requête $Q2$. Ainsi, notre approche ne souffre pas du problème de résolution puisqu'elle n'élague pas les communautés de petite taille au profit des grandes. Les expériences montrent que notre approche soulève le défi de détection les communautés multi-relationnelles riches sémantiquement. En effet, $Q1$ permet d'extraire $27$ communautés, dont chaque communauté est étiquetée de $1$ attribut à $52$ attributs et $Q2$ permet d'extraire $14$ communautés, dont chaque communauté est étiquetée de $1$ attribut à $104$ attributs.

**Détection de communautés avec une vérité terrain :** Nous avons utilisé la vérité terrain $VT$ définie dans (Yang et Leskovec, 2012). $VT$ se base sur l'hypothèse suivante : le descripteur explicite de chaque utilisateur est une communauté appartenant à la vérité terrain $VT$, c.à.d $VT$ contient l'ensemble des utilisateurs qui annotant la même ressource par les mêmes descripteurs. Afin d'évaluer l'efficacité de notre approche, nous avons choisi la structure de base de référence la plus populaire qui suggère qu'un ensemble d'utilisateurs qui ont le même comportement d'annotation forme une communauté. La performance est évaluée par les mesures de $Rappel$, $Pr\acute{e}cision$ et F$\beta$_mesure qui sont calculées sur tous les sommets (Song *et al.*, 2014). Ces mesures estiment si la prédiction de ces sommets dans la même communauté est correcte. Étant donné un ensemble de communautés algorithmiques $C$ et un ensemble de communautés de vérité terrain $S$, la précision indique combien de sommets sont réellement dans la même communauté de la vérité terrain :

$$Pr\acute{e}cision = \frac{|C \cap S|}{|C|}; Rappel = \frac{|C \cap S|}{|S|}$$

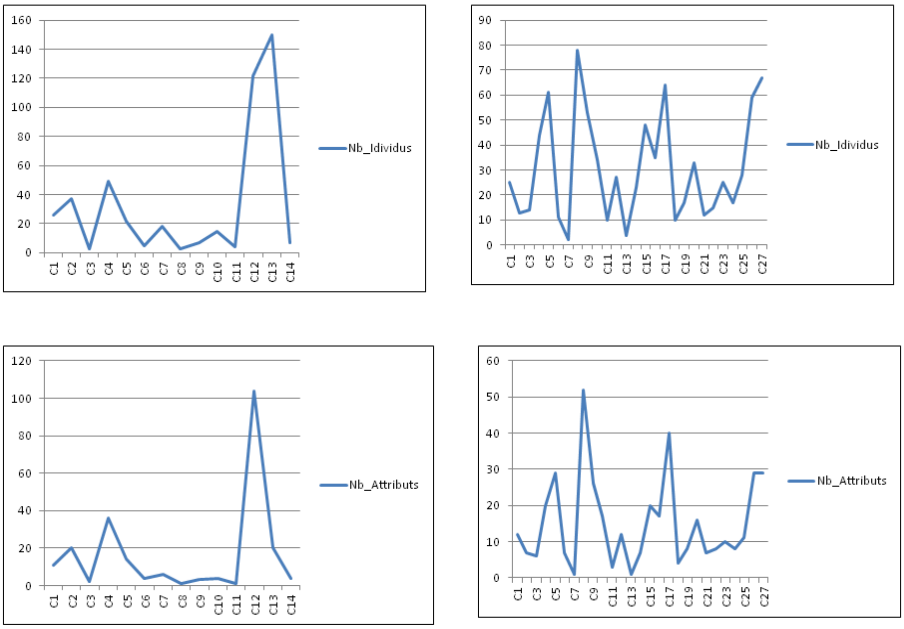


**Figure 6.** ***En haut*** *: Nombre d'individus par communauté.* ***En bas*** *: Nombre d'attributs (labels) par communauté.*

La mesure de $Rappel$ indique le nombre de sommets appartenant à la même communauté que celle de la communauté récupérée.

$$F\beta\_mesure = \beta \times \frac{Précision \times Rappel}{Précision + Rappel}; \beta \in \{1, 2\}$$

Pour comparer les performances de notre approche avec celles du modèle de référence (baseline), nous avons calculé le score moyen global des $4$ métriques : $Précision$, $Rappel$, $F1\_mesure$ et $F2\_mesure$. Les résultats sont représentés par la figure 7, les barres agrégées correspondent aux résultats des $4$ mesures en appliquant les requêtes $Q1$, $Q2$ et $Q3$, respectivement. Ainsi, selon l'histogramme présenté dans la figure 7, nous pouvons souligner que notre approche retourne de meilleurs résultats que ceux du modèle de référence. Comme illustré dans l'histogramme, le $Rappel$ moyen est $25.5\%$, $24.19\%$ et $23.96\%$ pour notre approche en appliquant les trois requêtes respectivement et $12,02\%$ pour le modèle de référence, c.à.d une amélioration d'environ $13.48\%$, $12.7\%$ et $11.94\%$, respectivement. Par ailleurs, nous remarquons, que les valeurs de rappel du modèle de référence sont beaucoup moins significatives

que celles obtenues par notre approche. On peut justifier la faiblesse des résultats de rappel $(12,02\%)$ précision $(6,4\%)$, F1_mesure$(8,36\%)$ et F2_mesure (10,23%), par le fait que les communautés du modèle de référence sont disjointes (une personne n'a annotée qu'une seule ressource) or les communautés de vérité terrain $VT$ sont chevauchantes. On peut également noter que les requêtes $Q1$ et $Q3$ contenant le plus grand

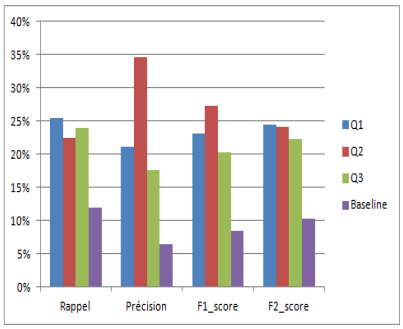


**Figure 7.** *Précision, Rappel, F1_mesure et F2_mesure de notre approche en appliquant les trois catégories requêtes $Q1$, $Q2$ et $Q3$,* Vs. *Ceux du modèle de référence.*

nombre d'entités (4 entités) et de relations (3 relations) atteignent un meilleur rappel $25,5\%$ pour $Q1$ et $23,96\%$ pour $Q3$ par rapport à la requête $Q2$ qui atteint $22,46\%$. Par conséquent, plus la communauté est riche de relations et entités partagées, plus le rappel augmente et donc plus l'adéquation entre les communautés de vérité de terrain $VT$ et les communautés extraites est importante. Finalement, nous constatons d'après l'étude expérimentale que nous avons menée, que l'approche proposée se distingue en 4 points essentiels : (1) elle détecte un ensemble de communautés multi-relationnelles sémantiquement riches. (2) elle surmonte les limites de résolution puisqu'elle arrive aussi bien à détecter des communautés de petite taille que celles de grande taille. (3) elle n'a pas besoin de connaissance a priori du nombre de communauté à extraire. (4) elle extrait différents types de communautés et répond à différents types de requêtes *i.e.*, indépendamment du nombre de relations ou d'entités, sans aucune obligation de changement du modèle par rapport à la structure de la requête.

## 6. Conclusion

Dans cet article, nous avons présenté une nouvelle approche pour la détection de communautés multi-relationnelles à partir des réseaux sociaux hétérogènes. Tout d'abord, nous avons proposé d'utiliser les techniques de l'analyse relationnelle de concepts $ARC$ pour modéliser les différentes entités et relations du réseau social. Ensuite, nous avons proposé une nouvelle méthode appelée : $Req_{Navigation}$ pour la détection des communautés multi-relationnelles, qui permet de naviguer entre les différents réseaux liés par des relations d'attributs. Notre future recherche se concentrera sur l'étude d'autres quantificateurs tels que $\forall$ pour considérer un ensemble plus diversifié de requêtes. Nous envisageons également d'évaluer et de tester notre approche sur

d'autres réseaux réels multi-relationnels tels que la collection de données génétiques pour le diagnostic médical.

## 7. Bibliographie


Brandes U., Delling D., Gaertler M., Gorke R., Hoefer M., Nikoloski Z., Wagner D., « On Modularity Clustering », *IEEE Trans. on Knowl. and Data Eng.*, vol. 20, p. 172-188, 2008.

Chen M., Kuzmin K., Szymanski B. K., « Community Detection via Maximization of Modularity and Its Variants », *CoRR*, 2015.

Crampes M., Plantié M., « Détection de communautés chevauchantes dans les graphes bipartis », *MARAMI 2012 : conférence sur les modèles et l'analyse des réseaux : Approches mathématiques et informatiques*, 2012.

Fortunato S., Barthélemy M., « Resolution limit in community detection », *The National Academy of Sciences*, 2007.

Ganter B., Wille R., *Formal Concept Analysis : Mathematical Foundations*, Springer, Berlin, 1999.

Huchard M., Hacene M. R., Roume C., Valtchev P., « Relational concept discovery in structured datasets », *Ann. Math. Artif. Intell.*, vol. 49, n° 1-4, p. 39-76, 2007.

Jay N., Kohler F., Napoli A., « Analysis of Social Communities with Iceberg and Stability-Based Concept Lattices », *in* R. Medina, S. Obiedkov (eds), *Formal Concept Analysis*, vol. 4933, Springer, p. 258-272, 2008.

Lin Y.-R., Sun J., Castro P., Konuru R., Sundaram H., Kelliher A., « MetaFac : Community Discovery via Relational Hypergraph Factorization », *Proceedings of the 15th ACM SIGKDD*, USA, p. 527-536, 2009.

Liu X., Liu W., Murata T., Wakita K., « A Framework for Community Detection in Heterogeneous Multi-Relational Networks », *Advances in Complex Systems*, 2014.

Plantié M., Crampes M., « Survey on Social Community Detection », *Book Chapter, Social Media Retrieval, Computer Communications and Networks*, p. 65-85, 2013.

Roth C., Obiedkov S. A., Kourie D. G., « On Succinct Representation of Knowledge Community Taxonomies with Formal Concept Analysis », *Int. J. Found. Comput. Sci.*, vol. 19, p. 383-404, 2008.

Rouane-Hacene M., Huchard M., Napoli A., Valtchev P., « Relational Concept Analysis : Mining Concept Lattices from Multi-relational Data », *Annals of Mathematics and Artificial Intelligence*, vol. 67, n° 1, p. 81-108, January, 2013.

Song S., Cheng H., Yu J. X., Chen L., « Repairing Vertex Labels under Neighborhood Constraints », *PVLDB*, vol. 7, p. 987-998, 2014.

Sun Y., Aggarwal C. C., Han J., « Relation Strength-Aware Clustering of Heterogeneous Information Networks with Incomplete Attributes », *PVLDB*, vol. 5, n° 5, p. 394-405, 2012.

Yang J., Leskovec J., « Defining and Evaluating Network Communities Based on Ground-Truth », *ICDM*, IEEE Computer Society, p. 745-754, 2012.

Zhang Z., Li Q., Zeng D., Gao H., « User Community Discovery from Multi-relational Networks », *Decis. Support Syst.*, vol. 54, n° 2, p. 870-879, January, 2013.