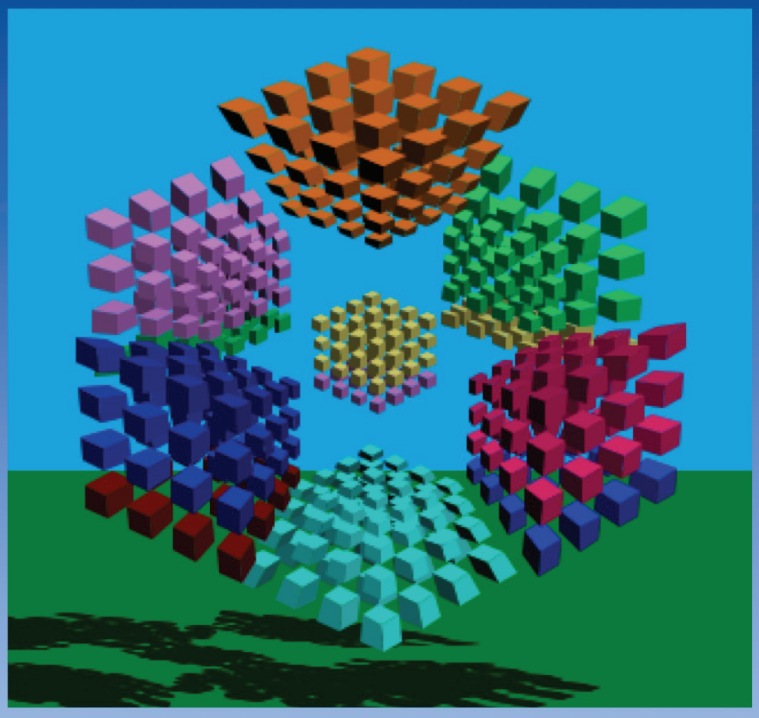
EGC'2013

13e Conférence Francophone sur l'Extraction et la Gestion de Connaissances

Université Paul Sabatier – IRIT – Toulouse

29 janvier 2013 – Journée Ateliers/Tutoriels



Fouille de Grand Graphes (FGG)

Université de Toulouse

Université Paul Sabatier TOULOUSE III

IRIT CNRS - INPT - UPS - UT1 - UTM

EGC

13e Conférence Francophone sur l'Extraction et la Gestion des Connaissances
29 janvier - 01 février 2013, Toulouse, France

1er Atelier sur la Fouille de Grands Graphes

Responsables

Lydia BOUDJELOUD-ASSALA (LITA – Université de Lorraine)
Bénédicte LE GRAND (CRI – Université Sorbonne Paris1)
Rushed KANAWATI (LIPN – Université Paris 13)

1er Atelier sur la Fouille de Grands Graphes

Lydia Boudjeloud-Assala*, Bénédicte Le Grand**, Rushed Kanawati***

*Université de Lorraine,
Laboratoire d'Informatique Théorique et Appliquée, LITA-EA 3097,
Metz, F-57045, France
lydia.boudjeloud-assala@univ-Lorraine.fr
**Université Paris 1 Panthéon-Sorbonne
Centre de Recherche en Informatique
90, rue de Tolbiac, F-75013 Paris
Benedicte.Le-Grand@univ-paris1.fr
***Institut Galilée - Université Paris Nord
LIPN CNRS UMR 7030
99 AV. J.B. Clément93430 Villetaneuse - France
Rushed.Kanawati@lipn.univ-paris13.fr

1 Le groupe de travail Fouille de Grands Graphes

Le groupe de travail Fouille de Grands Graphes a été créé en 2010, il s'intéresse à une thématique émergente et d'actualité autour de l'analyse et de l'étude de la dynamique dans les grands graphes. L'objectif du groupe est de proposer une structure d'animation scientifique pour des chercheurs venant de plusieurs disciplines et s'intéressant à la fouille de grands graphes. La recherche en modélisation, en analyse et en fouille de grands graphes a connu un net regain d'intérêt qui peut se justifier par les deux points suivants : dans plusieurs domaines les données se présentent souvent sous forme structurée : graphes ou objets reliés. Nous pouvons citer par exemple les systèmes biologiques, le web, les réseaux sociaux (facebook, twitter, ...) ou encore les réseaux bibliographiques. Les graphes issus de ces domaines ont des propriétés spécifiques qui les différencient des graphes aléatoires (graphes sans échelle, faible densité, faible degré de séparation, ...). Les techniques actuelles permettent d'observer de très grands réseaux qui évoluent dans le temps et nous posent ainsi le défi du passage à l'échelle et la nécessité de disposer d'outils de visualisation et de fouille pouvant s'adapter à ce nouveau type de données.

2 Objectif et thématique de l'atelier

Suite au succès des 3 journées thématiques Fouille de Grands Graphes du groupe de travail à Toulouse'2010, Grenoble'2011 et Villetaneuse'2012, le groupe de travail EGC Fouille de Grands Graphes (EGC-FGG) propose un atelier AFGG-EGC'2013, dans le cadre de la conférence Extraction et Gestion de Connaissances (EGC'2013). Cet atelier a pour but de réunir les

chercheurs intéressés par le traitement, la fouille et la visualisation de grands graphes. Notre ambition est de permettre aux participants d'aborder tous les thèmes de la fouille de grands graphes. L'atelier concerne aussi bien les chercheurs du monde académique que ceux du secteur industriel, et autant les notions conceptuelles que les applications.

Le thème principal pour cette édition est la détection de communautés , abordée notamment dans une présentation invitée. Cet atelier s'intéressera aussi aux réseaux temporels et aux structures communautaires egodentrées.

Deux applications seront également présentées, exploitant conjointement les graphes avec d'autres informations, par exemple le contenu sémantique d'un ensemble de documents.

3 Comité de programme

4 Programme

9h00 – 9h15 Ouverture de l'atelier

9h15 – 10h05 *Présentation invitée :* Détection de communautés sur des réseaux temporels, *Rémy Cazabet*

10h05 – 10h30 Déplier les structures communautaires egocentrées - une approche à base de similarité, *Maximilien Danisch, Jean-Loup Guillaume and Bénédicte Le Grand*

10h30 – 11h00 Pause

11h00 – 11h25 Mesurer l'intrication sémantique dans une collection de documents, *Benjamin Renoust, Guy Mélançon and Marie Luce Viaud*

11h25 - 11h50 Carte auto-organisatrice pour graphes étiquetés, *Nathalie Villa-Vialaneix, Madalina Olteanu and Christine Cierco-Ayrolles*

11h50 – 12h15 Primates : A Privacy Management System for Social Networks, *Imen Ben Dhia, Talel Abdessalem and Mauro Sozio*

12h15 – 12h25 Discussion du groupe de travail GT-FGG

12h25 – 12h30 Clôture de l'atelier

5 Remerciments

Nous tenons à remercier les auteurs et les membres du comité de programme pour la qualité de leurs contributions.

Nous remercions également les responsables des ateliers pour EGC 2013. Enfin nous remercions vivement les présidents : Christel Vrain, présidente du comité de programme, Florence Sèdes et André Péninou co-présidents du comité d'organisation d'EGC 2013.

Summary

Following the success of three Mining Large Graphs days : Toulouse'2010, Grenoble'2011 and Villetaneuse'2012. The Working Group Mining Large Graphs EGC (EGC-FGG) offers a workshop AFGG-EGC'2013 as part of the conference Extraction and Knowledge Management (EGC'2013). This workshop aims to bring together researchers interested in the processing, search and visualization of large graphs. Our goal is to enable participants to address all the issues of mining large graphs. The workshop covers both academic researchers as the industrial sector, and both the conceptual notions applications.

Détection de communautés sur des réseaux temporels

Rémy Cazabet

IRIT, Toulouse, France
remy.cazabet@gmail.com
http://www.cazabetremy.fr

1 Résumé

La détection de communautés dans les réseaux est un problème ayant donné lieu à de très nombreux travaux ces dernières années. Cependant, la grande majorité de ces approches sont statiques. Or, lorsque l'on s'intéresse à des réseaux réels, et en particulier aux grands graphes issus du Web 2.0, les données sont dynamiques. On parle alors de réseaux temporels.

Dans cet exposé, je vais commencer par présenter les intérets de faire de la détection de communautés sur des réseaux temporels, ainsi que les difficultés que cela entraîne, notamment l'explosion de la complexité par rapport aux approches statiques, ainsi que le problème de l'instabilité de la détection de communautés.

Je poursuivrai en présentant quelques méthodes déjà proposées, qui adoptent des démarches différentes : certaines découpent l'évolution du réseau en plusieurs instantanés étudiés indépendamment, tandis que d'autres proposent de les étudier simultanément. D'autres méthodes ne découpent pas l'évolution du réseau en instantanés, mais travaillent directement sur le réseau temporel.

Dans une dernière partie, je présenterai quelques exemples d'applications à des grands graphes réels, tels que des réseaux bibliographiques, des réseaux sociaux et des plateformes web 2.0.

2 Biographie

Rémy Cazabet va soutenir en Mars 2013 son doctorat, qu'il a effettué à l'IRIT (Institut de Recherche en Informatique de Toulouse), sous la direction de Frédéric Amblard et de Chihab Hanachi. Il a également effectué des stages de recherche au National Institute of Infomatics, à Tokyo, sous la direction d'Hideaki Takeda, ainsi qu'à l'Université Aalto, en Finlande, dans l'équipe de Santo Fortunato.

Il travaille principalement dans le domaine de la détection de communautés sur des grands graphes, et en particulier sur la question de la dynamique de ces communautés dans des réseaux temporels (réseaux évoluant au cours du temps), ainsi que sur les problématiques liées à cette question. (visualisation de la dynamique, application sur des grands graphes de terrains, etc.)

Déplier les structures communautaires egocentrées - une approche à base de similarité

Maximilien Danisch*, Jean-Loup Guillaume*, Bénédicte Le Grand**

*ComplexNetworks.fr, Laboratoire d'Informatique de Paris 6, Université Pierre et Marie Curie, 4 Place Jussieu, 75005 Paris, France
**Centre de Recherche en Informatique. Université Paris 1 Panthéon-Sorbonne. 90 rue de Tolbiac, 75013 Paris, France

Résumé. Nous proposons ici une approche performante pour déplier la structure communautaire egocentrée d'un noeud. Nous montrons que, bien que chaque nœud d'un réseau appartienne potentiellement à plusieurs communautés, il est généralement possible d'identifier une communauté unique si l'on considère deux nœuds bien choisis. La méthodologie que nous proposons repose sur cette notion de communauté multi-egocentrée ainsi que sur l'utilisation d'une mesure de similarité dérivée de techniques de dynamique d'opinion, appelée l'opinion propagée. Cette approche pallie les limites des fonctions de qualité utilisées traditionnellement pour la détection de communautés egocentrées, et consiste à étudier les irrégularités dans la décroissance de cette mesure de similarité.

1 Introduction

De nombreux réseaux réels, tels que des réseaux sociaux ou des réseaux informatiques, peuvent être modélisés par des grands graphes, souvent appelés graphes de terrain. Ces réseaux réels ont été fortement étudiés ces dernières années, du fait de l'augmentation du nombre de jeux de données disponibles et du besoin de compréhension de tels systèmes dans de très nombreux contextes. La notion de communauté, intuitivement définie comme un groupe de sommets fortement connectés entre eux mais peu liés au reste du réseau, est centrale dans ce contexte, du fait de ses nombreuses applications. La recherche d'algorithmes efficaces pour identifier automatiquement de telles communautés constitue un défi, comme l'atteste l'article de synthèse de Fortunato (2010).

Dans des systèmes réels, les communautés se chevauchent naturellement et sont qualifiées de recouvrantes. Par exemple dans un réseau social chaque individu appartient à plusieurs communautés : famille, collègues, divers groupes d'amis, etc. La découverte de tous ces groupes dans de grands graphes est complexe : dans un graphe à n sommets, il peut exister 2^n communautés distinctes et il y a 2^{2^n} structures communautaires potentielles. Outre le problème de l'efficacité du calcul, l'interprétation d'une structure en communautés recouvrantes peut être difficile. Certains travaux ont cependant abordé ce problème, par exemple Palla et al. (2005); Evans et Lambiotte (2009).

La complexité de la détection et de l'interprétation de ces communautés recouvrantes ont conduit la plupart des chercheurs à se limiter à un partitionnement, dans lequel chaque sommet appartient à une et une seule communauté. La détection de communautés disjointes est également complexe et il n'y a pas de solution parfaite, même si de nombreuses définitions et algorithmes associés existent. La définition la plus populaire à l'heure actuelle est la modularité Girvan et Newman (2002), qui privilégie la recherche de communautés plus denses que ce que l'on pourrait attendre et la méthode de Louvain Blondel et al. (2008) fait partie des quelques méthodes très efficaces pour trouver des partitions de bonne modularité.

Une approche intermédiaire pour conserver le réalisme des communautés recouvrantes tout en simplifiant un peu le problème consiste à se concentrer sur un seul sommet et à chercher à identifier l'ensemble des communautés auxquelles il appartient ; ces communautés sont généralement appelées communautés ego-centrées. Ce problème est le plus souvent abordé via des fonctions de qualité : en partant d'un groupe de sommets qui ne contient initialement que le sommet considéré, des sommets sont ajoutés ou supprimés de ce groupe pas-à-pas pour optimiser une fonction de qualité donnée. En général on distingue les sommets appartenant à la communauté, les sommets voisins de la communautés qui pourraient la rejoindre et les sommets restants. Par exemple dans Clauset (2005), la qualité d'une communauté dépend des degrés interne et externe des sommets à la frontière de la communauté. Dans Luo et al. (2008) la fonction de qualité dépend des degrés interne et externe des sommets au coeur de la communauté. Dans Chen et al. (2009) la fonction de qualité dépend de la densité intra-communautaire et de la densité de liens sortant de la communauté ; une amélioration de cette approche est proposée dans Ngonmang et al. (2012). Enfin, dans Friggeri et al. (2011) la fonction de qualité utilisée, appelée cohésion, dépend de la densité de triangles internes et externes.

L'approche à base de fonction de qualité souffre généralement de deux défauts majeurs : (i) la mise en oeuvre de fonctions de qualité est difficile du fait de l'existence de paramètres d'échelle cachés. Par exemple dans Friggeri et al. (2011), la cohésion mesure la densité de triangles, qui décroit en $O(s^3)$ (où s est le nombre de sommets sélectionnés) dans les graphes peu denses. Cela a tendance à privilégier de très petites communautés. Cela peut être atténué en prenant la puissance a ($a \leq 1$) de ce terme de densité pour privilégier des communautés plus grandes. Dans la cohésion ce terme prend par défaut la valeur 1. (ii) L'optimisation de la fonction de qualité est aussi très complexe du fait de la nature non-convexe de l'espace d'optimisation. En conséquence, comme l'optimisation est en général gloutonne il faut s'autoriser à repasser par des zones de moindre qualité pour trouver les meilleures zones.

Dans cet article nous proposons une approche orthogonale pour identifier les communautés ego-centrées d'un sommet : étant donné un sommet u, nous mesurons la similarité de ce sommet à tous les autres sommets du graphe pour chercher des irrégularités dans la décroissance des similarités. De tels irrégularités peuvent refléter la présence d'une ou de plusieurs communautés. Plus précisément, si un groupe de sommets sont également similaires au sommet u alors ce groupe est une communauté de u. Nous utilisons la notion de communautés multi-ego-centrées, introduite dans Danisch et al. (2012), pour répondre aux questions suivantes : étant donné un sommet, à quelles communautés appartient-il ? Comment décrire ou nommer ces communautés ? Nous explicitons les résultats de notre approche en l'appliquant à un réseau réel contenant l'ensemble des pages de wikipedia (2 millions de pages) et les liens hypertextes entre ces pages (40 millions de liens). Ce réseau date de 2008 et provient de Palla et al. (2008).

2 Contexte

2.1 L'opinion propagée

L'opinion propagée (carryover opinion en anglais) est une mesure de similarité introduite dans Danisch et al. (2012) qui se base sur la dynamique d'opinions. La similarité entre un sommet u et tous les autres sommets du graphe peut être calculée efficacement (en temps linéaire de manière empirique) avec un calcul de point fixe. Ce calcul consiste à répéter de manière itérative les trois étapes suivantes :

$$\begin{array}{lcll} C^t(u) & = & MC^{t-1}(u) & \text{MOYENNE} \\ C^t(u) & = & \frac{C^t(u)-min(C^t(u))}{1-min(C^t(u))} & \text{MISE A L'ECHELLE} \\ C^t_u(u) & = & 1 & \text{REINITIALISATION} \end{array}$$

où :

- $C^t(u)$ est le vecteur de similarités après t itérations pour le sommet de départ u. La composante j du vecteur $C^t(u)$ est notée $C^t_j(u)$;
- $C^0(u)$ est le vecteur nul sauf pour le sommet u qui est mis à 1 : $C^0_u(u) = 1$;
- M est la matrice de moyennage, qui est la transposée de la matrice stochastique de transition : $M_{kl} = \frac{l_{kl}}{d_k}$, où l_{kl} est le poids du lien (k, l) et d_k le degré du sommet k.

Le vecteur $C^\infty(u)$ est appelé vecteur de l'opinion propagée du sommet u. Le nom vient du fait que la répétition des étapes sans mise à l'échelle est un simple processus de diffusion d'opinion qui mène à une valeur de 1 pour tous les sommets. L'étape de mise à l'échelle permet de capturer la proximité des sommets à u, qui est propagée durant le processus.

2.2 Communautés ego-centrées

Une application directe de l'opinion propagée est la détection de communautés ego-centrées via la recherche d'irrégularités dans la décroissance de valeurs de l'opinion propagée. Comme le montre la figure 1a, en triant les valeurs de $C^\infty(u)$ par ordre décroissant, on observe parfois un ou plusieurs plateaux entre les phases de décroissance de ces valeurs. On considère que les sommets d'un plateau constituent une communauté du sommet de départ.

Cette méthodologie pour trouver les communautés ego-centrées ne nécessite pas d'autre paramètre que le graphe lui-même. L'opinion propagée est calculable très rapidement, ce qui la rend utilisable sur des graphes de grande taille. Cependant, un sommet donné appartient en général à plusieurs communautés et l'alternance claire de plateaux et de décroissances brutales n'est que rarement observée. La figure 1b illustre d'autres situations : les transitions claires indiquent que les communautés sont bien marquées et les transitions lentes indiquent qu'elles sont plus floues. Des lois puissance déformées sont observées quand plusieurs communautés se recouvrent et des lois puissance nettes indiquent qu'aucune échelle nette n'apparait ce qui est le cas si de nombreuses communautés de taille variable se recouvrent ou s'il n'y a pas de structure communautaire.

Nous allons maintenant présenter une méthodologie permettant de traiter les cas où les communautés ne sont pas détectables directement, notamment dans les cas de loi puissance.

3 Méthodologie

3.1 Communautés multi-ego-centrées

En tant que telle, l'opinion propagée ne permet pas de traiter les cas où l'on observe une loi puissance et, afin de résoudre ce problème, une notion utile est celle de communautés multi-ego-centrées : bien qu'un sommet puisse appartenir à plusieurs communautés, deux sommets judicieusement choisis peuvent définir une unique communauté. Par exemple, bien que deux collègues appartiennent chacun à différentes communautés (familles, collègues, amis, etc.), l'intersection de leurs communautés peut caractériser leur communauté professionnelle.

On peut utiliser l'opinion propagée pour détecter de telles communautés multi-ego-centrées en se basant sur l'idée que si un sommet appartient à la fois à une communauté d'un sommet $s1$ et à une communauté d'un sommet $s2$ alors il doit être similaire à $s1$ ET à $s2$. La figure 2 montre comment procéder en deux étapes : (i) évaluer pour chaque sommet du graphe sa similarité à $s1$ et à $s2$ et (ii) la similarité du sommet à l'ensemble $\{s1, s2\}$ peut être calculée comme le minimum des deux similarités [1]. A nouveau, si un plateau suivi d'une décroissance brusque est observé alors les sommets avant la décroissance forment une communauté.

Il faut noter que cette procédure peut parfois détecter des communautés qui ne contiennent pas $s1$ ou $s2$, contrairement au cas observé sur la figure 2. On souhaite généralement que $s1$ et $s2$ appartienne à la communauté car on cherche à identifier ses communautés et si l'on souhaite vraiment que $s1$ et $s2$ soient dans la communauté alors on peut exclure toutes les communautés ne contenant pas ces deux sommets. Cependant, on peut s'intéresser uniquement aux communautés contenant au moins $s1$ et considérer que $s2$ est utilisé uniquement pour calculer des communautés mais n'a pas plus d'utilité.

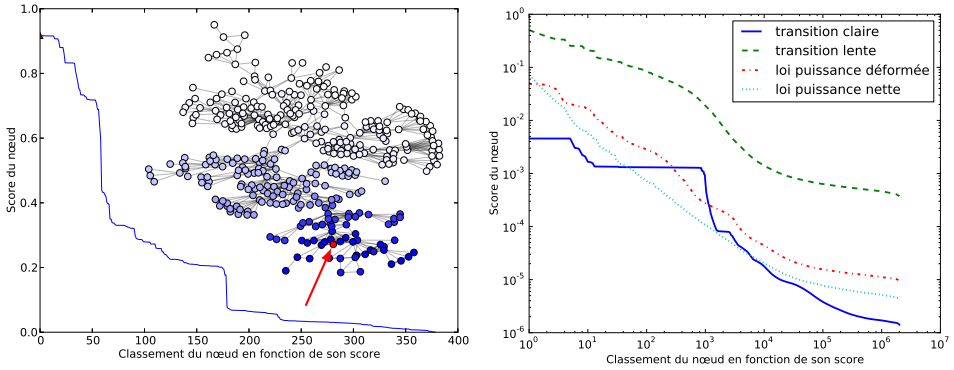
Nous proposons maintenant une méthodologie utilisant le concept de communautés multi-ego-centrées pour, étant donné un graphe et un sommet d'intérêt $s1$, trouver les communautés ego-centrées sur $s1$ et les étiqueter. Si la courbe de l'opinion propagée de $s1$ conduit à une courbe comportant une ou plusieurs transitions nettes alors les communautés sont déjà identifiées, comme sur 1a et 1b (courbe avec transition brutale). Malgré tout, la plupart du temps on observe une loi puissance indiquant la présence de plusieurs communautés recouvrantes et de taille différente. Dans ce cas nous proposons de choisir un ensemble de sommets candidats et de voir, pour chacun d'eux, s'il existe une communauté bi-ego-centrée sur $s1$ et le candidat, c'est-à-dire s'il existe une décroissance brutale dans la courbe de l'opinion propagée du minimum de $s1$ et du candidat.

3.2 Choix des candidats

Tout d'abord il faut calculer la courbe de l'opinion propagée de $s1$, ce qui donne une valeur pour chaque sommet du graphe. Le tri de ces valeurs par ordre décroissant donne la courbe de l'opinion propagée et si le résultat est une loi puissance alors il n'y a pas d'échelle caractéristique et $s1$ appartient à plusieurs communautés de taille variable.

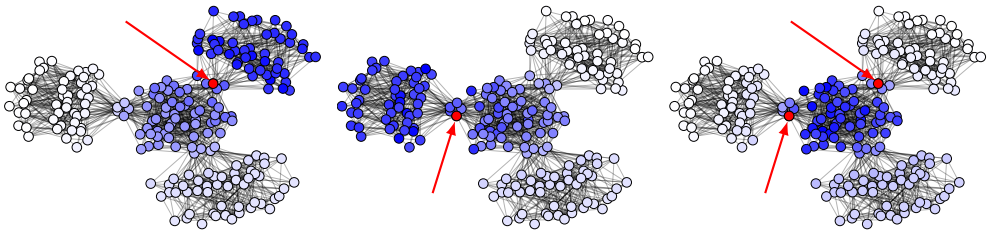
On souhaite ensuite choisir $s2$ de sorte que $s1$ et $s2$ partagent une seule communauté. Si $s2$ est très dissimilaire à $s1$ alors il est peu probable que les deux sommets partagent une même communauté : calculer le minimum des scores obtenus à partir de $s1$ et $s2$ donnera des

1. Cette quantité mesure la similarité à $s1$ ET $s2$. Calculer le maximum permettrait d'identifier des sommets appartenant à une communauté de $s1$ OU $s2$, ce qui n'est pas pertinent dans notre contexte.



(a) (b)

FIG. 1: La figure 1a illustre le vecteur de l'opinion propagée sur un réseau de co-écriture d'articles à 379 sommets et 914 liens Newman (2006). Le sommet considéré est pointé par une flèche et les sommets les plus sombres sont ceux dont le score est le plus élevé. La courbe montre trois plateaux séparés par des décroissances nettes. Les sommets des deux premiers plateaux correspondent à deux communautés imbriquées du sommet considéré.
La figure 1b montre le même calcul sur le réseau wikipedia Palla et al. (2008) en échelle logarithmique (nécessaire pour distinguer les plateaux). 4 courbes correspondant à 4 sommets différents montrent les différents comportement observables.



(a) (b) (c)

FIG. 2: Résultat pour 4 graphes Erdos-Renyi se chevauchant sur 5 sommets chacun. Chaque graphe est généré avec une probabilité d'existence de lien de $0,2$. Plus un sommet est sombre, plus son score est élevé. Sur chaque figure la flèche pointe vers le sommet considéré. (2c) présente le minimum (remis à l'échelle) des scores des expériences présentées sur les figures (2a) et (2b). La communauté centrale émerge alors.

valeurs très faibles car, si les deux sommets ne sont pas dans une même communauté alors pour chaque sommet $s3$, au moins un des deux scores sera faible. Inversement, si $s2$ est très similaire à $s1$ alors les deux sommets partageront certainement de nombreuses communautés : pour tout sommet $s3$ du graphe, le score de $s3$ sera le même vu de $s1$ et de $s2$, les trois courbes ($s1$, $s2$ et minimum) seront très similaires et ne feront pas émerger de communauté.

En conséquence, $s2$ doit être similaire à $s1$ afin qu'ils partagent une communauté mais pas

trop. Il faut donc que le score de $s2$ pour l'opinion propagée de $s1$ ne soit ni trop élevé ni trop bas. Il est possible de fixer des seuils haut et bas pour choisir des sommets à la bonne distance qui serviront de candidats pour le calcul de communautés bi-ego-centrées.

Il est probable que de nombreux sommets mènent à l'identification de la même communauté, il n'est donc pas forcément utile de tous les sélectionner comme candidats. Il est par exemple possible d'en sélectionner un sous-ensemble de manière aléatoire si le temps de calcul est un critère important, avec le risque de manquer quelques communautés. Des procédures plus fines de sélection peuvent être imaginées et nous discuterons ce point en conclusion.

3.3 Identification de communautés bi-ego-centrées

Afin d'identifier les communautés potentielles centrées sur $s1$ et $s2$ il faut également calculer l'opinion propagée de $s2$ puis, pour chaque sommet du graphe, calculer le minimum des deux scores. Pour un sommet $s3$ donné, le minimum est une mesure de l'appartenance de $s3$ aux communautés de $s1$ et de $s2$. Il faut ensuite trier ces minima et représenter la courbe de l'opinion propagée minimale. Comme précédemment, une irrégularité dans la courbe, c'est-à-dire un plateau suivi d'une décroissance, indique que les sommets du plateau constituent une communauté.

La détection d'un plateau suivi d'une décroissance forte peut se faire de manière automatique, par exemple, si la pente maximale de la courbe est supérieure à un seuil fixé. Ce seuil doit être fixé manuellement. Pour l'instant on se contente de chercher la pente maximale mais il peut y avoir plusieurs pentes supérieures au seuil indiquant une imbrication de communautés.

De plus, si $s1$ est situé avant la décroissance alors il appartient à la communauté détectée et l'on peut dire que cette communauté fait partie de la structure communautaire de $s1$. Actuellement on ne demande pas que $s2$ fasse partie de la communauté détectée car on cherche des communautés de $s1$ et $s2$ est uniquement utilisé pour parvenir à cet objectif.

Telle quelle, cette méthode n'est pas performante lorsqu'elle est utilisée depuis un sommet $s1$ de fort degré, ces derniers étant en général connectés à de nombreuses communautés. Dans ce cas, l'opinion propagée a tendance à attribuer des scores élevés à de nombreux sommets du graphe et le minimum de ces scores avec ceux d'un sommet $s2$ moins populaire va simplement donner les scores de $s2$. Une mise à l'échelle avant de calculer le minimum peut résoudre le problème. Plus précisément, cela peut-être fait en mettant à l'échelle (en échelle logarithmique) les valeurs telles que les valeurs les plus basses (généralement résultant en un plateau final) se trouvent au même niveau.

3.4 Nettoyage des communautés et étiquetage

Les deux premières étapes fournissent un ensemble de communautés avec des scores pour chaque sommet de chaque communauté. Ces différentes communautés doivent subir un traitement a posteriori car nombre d'entre elles sont similaires.

Nous proposons dans un premier temps de supprimer les communautés similaires. En utilisant la similarité de Jaccard [2] (ou toute autre similarité pertinente) entre deux communautés on peut savoir si deux communautés sont semblables ou pas. Si la valeur obtenue est élevée alors, à quelques sommets près, les communautés sont identiques. Dans ce cas on ne conserve

2. Pour deux ensembles A et B, la similarité de Jaccard est donnée par $Jac(A, B) = \frac{|A \cap B|}{|A \cup B|}$.

que l'intersection des deux communautés. À chaque sommet conservé on associe un nouveau score qui est la somme des scores qu'il avait obtenu dans chacune des communautés.

Dans un second temps, nous proposons de supprimer toute communauté qui n'a pas été intersectée avec une autre, en effet une communauté pertinente devrait être identifiée via plusieurs candidats. Une validation manuelle de la pertinence de ces communautés uniques montre qu'en général on a détecté une décroissance forte inexistante. Ceci peut arriver pour plusieurs raisons, notamment si le seuil est mal fixé. Cette étape est optionnelle mais nos études montrent que les résultats obtenus sont plus pertinents avec cette seconde phase de nettoyage.

Enfin, nous attribuons une étiquette à chaque communauté, qui est celle du sommet le mieux classé dans la communauté. Comme nous avions calculé la somme des scores à chaque intersection, cela revient à prendre l'étiquette du sommet qui est le mieux classé en moyenne. Il est possible que plusieurs communautés distinctes obtiennent la même étiquette, ce qui peut correspondre à des communautés imbriquées.

Cette suite d'étape aboutit finalement à un ensemble de communautés étiquetées et pas trop similaires, contenant chacune un ensemble de sommets associés à un score.

4 Résultats et validation

Nous présentons ici les résultats pour un unique sommet, la page wikipedia intitulée *Chess Boxing* [3]. Les résultats obtenus sur cette page sont très pertinents, facilement interprétables et validables manuellement.

Pour le sommet « Chess Boxing », nous avons utilisé l'algorithme présenté précédemment sur 3000 sommets candidats choisis aléatoirement compris entre le centième et le dix-millième dans l'ordre de l'opinion propagée calculée pour « Chess Boxing ». Sur les 3000 tentatives, 770 ont abouti à l'identification de plateaux et de décroissances nettes, fournissant autant de communautés. La figure 3 montre trois exemples de candidats fournissant une communauté claire et un exemple d'essai infructueux.

La figure 4a présente la matrice de similarité de Jaccard des 770 communautés identifiées avant la phase de nettoyage. Les colonnes et les lignes ont été réordonnées de sorte que des groupes similaires apparaissent proches les uns des autres. On identifie aisément un groupe de 716 communautés très similaires mais peu similaires aux autres. L'intersection de ces communautés donne une communauté dont l'étiquette est « Queen's Gambit » (nous expliquerons plus loin les étiquettes des groupes obtenus, voir tableau 1).

Un zoom sur le reste de la matrice, figure 4b, montre la présence de 4 plus petits groupes de communautés. Les communautés au sein de chaque groupe sont similaires mais peu similaires aux communautés d'autres groupes. Ces groupes correspondent, après intersection, aux communautés étiquetées « Enki Bilal », « Uuno Turhapuro », « Da Mystery of Chessboxin' » et « Gloria » (voir tableau 1). On observe également 6 groupes ne contenant chacun qu'une unique communauté peu similaire aux autres. Ces groupes sont liés à de mauvaises identifications de plateaux et décroissances et sont supprimés dans la seconde phase de nettoyage.

Cette décomposition en 5 groupes principaux est facilement obtenue en calculant l'intersection des communautés (avec un seuil de similarité fixé à $0,7$). Les 6 groupes ne contenant qu'une communauté sont supprimés. Le tableau 1 présente pour chacun des 5 groupes, l'éti-

3. Le ChessBoxing est un sport mêlant échecs et boxe avec des rounds alternés.

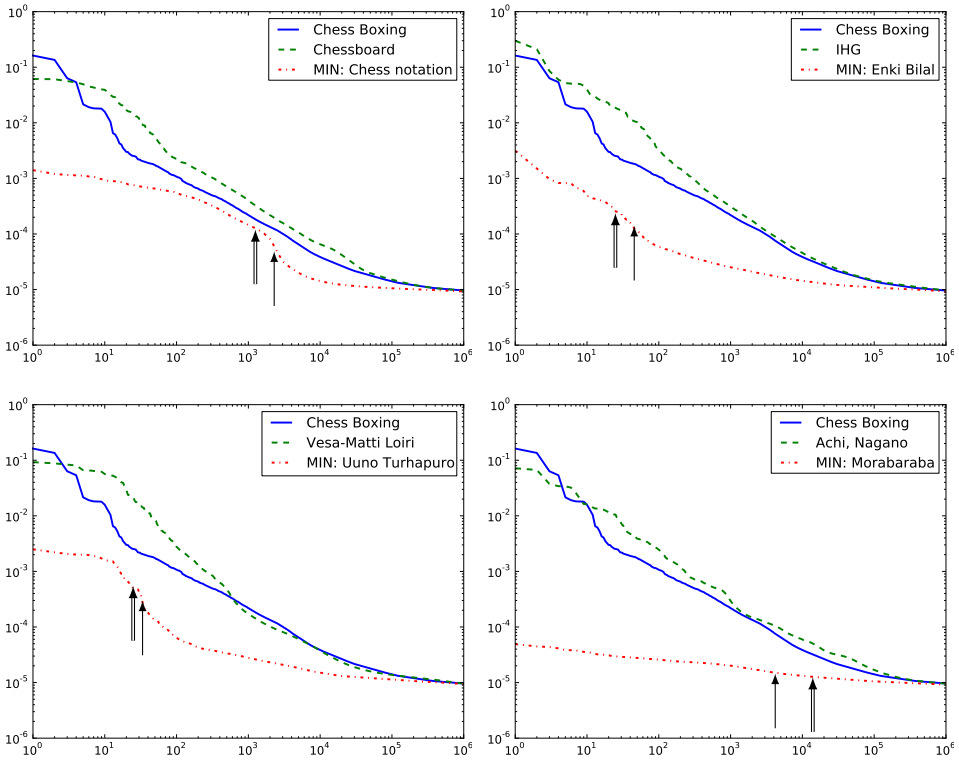
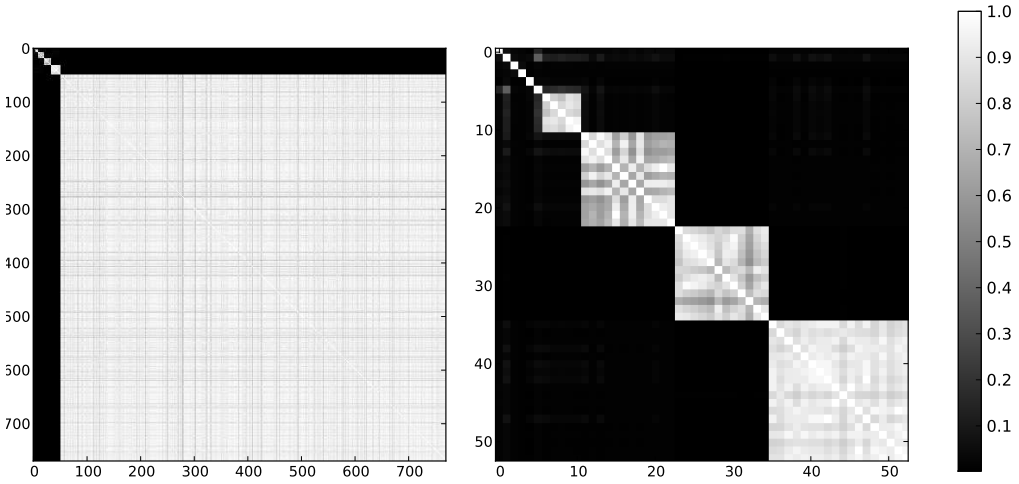


FIG. 3: Chaque figure présente les courbes de l'opinion propagée pour le sommet « Chess Boxing », pour un sommet candidat et pour le minimum des valeurs de l'opinion propagée. L'étiquette du sommet le mieux classé est indiquée dans la légende (MIN). La flèche simple indique l'endroit où la pente maximale est détectée et la flèche double indique le classement de « Chess Boxing » dans la courbe du minimum. Seules les trois premières courbes permettent d'identifier une pente suffisante et donc une communauté.

quette choisie (correspondant au sommet le mieux classé en moyenne), la taille de la communauté finale ainsi que des exemples de sommets bien classés. Comme on peut le voir, l'algorithme parvient à identifier des groupes de taille très variée, de 26 à 1619 sommets sur cet exemple. Ceci est une propriété très intéressante car d'autres approches ont une tendance naturelle à trouver des communautés de taille limitée.

Il est possible de vérifier que les communautés et leurs labels ont du sens via les pages wikipedia associées :

– Enki Bilal est un dessinateur et scénariste de bande dessinée français. Il a en particulier écrit et illustré « Froid Équateur » en 1992 qui est cité par l'inventeur du « Chess Boxing » comme sa source d'inspiration. Les sommets de la communauté sont majoritairement d'autres oeuvres de Bilal.
– Uuno Turhapuro est un personnage fictif d'une comédie finlandaise et, de même que



(a) (b)

FIG. 4: La figure 4a présente la matrice de la similarité de Jaccard des 770 communautés, réordonnée pour que les groupes similaires soient proches. Un groupe de 716 communautés émerge. La figure 4b présente la matrice privée des 716 communautés afin que le reste soit plus compréhensible.

Bilal, ce personnage a aussi servi de source d'inspiration dans une scène de « Uuno Turhapuro—herra Helsingin herra » en 1991 où le héros joue aux échecs à l'aveugle en téléphonant et en boxant avec une autre personne.

– « Da Mystery of Chessboxin' » est une chanson du groupe de rap américain « The Wu-Tang Clan ». Les sommets de la communauté sont liés au groupe ou au rap en général.
– « Gloria » est une page d'homonymie de wikipedia liée à de nombreuses pages contenant différents sens de Gloria. Il n'y *a priori* aucun lien entre « Gloria » et « Chess Boxing » et la seule référence à Gloria dans la page « Chess Boxing » est la phrase « On April 21, 2006, 400 spectators paid to watch two chess boxing matches in the Gloria Theatre, Cologne ». Nous avons trouvé dans l'historique qu'un lien a été ajouté le 3 mai 2006 vers la page « Gloria » et supprimé le 31 janvier 2008. Le fait que le jeu de données ait été construit pendant cette période explique que l'on trouve cette communauté qui n'existe plus maintenant.
– Enfin, « Queen's Gambit » est une ouverture classique aux échecs et la communauté est composée de pages liées en échecs. On aurait sans doute préféré que la communauté soit étiquetée « Chess » mais « Queens' Gambit » est un terme très spécifique aux échecs et caractérise donc parfaitement la communauté.

De manière surprenante l'algorithme ne trouve aucune communauté liée à la boxe. Ceci pourrait être un problème de l'algorithme mais la page wikipedia de « Chess Boxing » indique que la plupart des chess boxers ont un passé dans les échecs et apprennent la boxe par la suite. Ils peuvent donc être plus importants dans le monde des échecs que dans celui de la boxe. Ceci

Label	Taille	Exemples de sommets
Enki Bilal	35	Rendez-vous à Paris, Exterminateur 17, Iepe Rubingh, Le Sommeil du monstre, White Birds, The Black Order Brigade, Froid-Équateur, La Foire aux immortels, Fabrice Giger, Goran Vejvoda
Uuno Turhapuro	26	Uuno Turhapuro kaksoisagentti, Uuno Turhapuro (film), Professori Uuno D.G. Turhapuro, Simo Salminen, Jori Olkkonen, Funny-Films Oy, Uuno Epsanjassa, Marjatta Raita, Chess boxing
Da Mystery of Chessboxin´	254	Wu-Tang Clan, Protect Ya Neck, Legend of the Wu-Tang Clan, Grandmasters (album), Gravel Pit, Wu-Tang Clan discography, Shame on a Nigga, Mr. Xcitement, U-God, Masta Killa, N-Tyce
Gloria	55	Gloria (Poulenc), Mass in E Flat Major, Gloria D. Miklowitz, Desiree Casado, Gloria (1999 film), Gloria (Disillusion album), Gloria, Oriental Mindoro, Gloria (singer), Chess boxing, Mass in F Minor
Queen's Gambit	1619	Checkmate, Fast chess, Baltic Defense, Symmetrical Defense, Closed Game, Marshall Defense, Tarrasch Defense, Torre Attack, Chigorin Defense, Chess handicap, Blindfold chess, Chess notation

TAB. 1: Label, taille et exemples de sommets dans les communautés pour la page wikipedia « Chess Boxing ». Il est possible de vérifier directement le sens des labels et exemples sur wikipedia.

pourrait expliquer que le sommet « Chess Boxing » est dans la communauté des échecs mais à la limite de celle de la boxe.

4.1 Comparaison

Comme expliqué précédemment, d'autres approches existent pour calculer des communautés ego-centrées, basées sur l'optimisation d'une fonction de qualité. Nous comparons maintenant de manière brève nos résultats à ceux obtenus avec la méthode de Ngonmang et al. (2012) qui, selon nous, est la plus avancée du fait qu'elle corrige de nombreux défauts des méthodes précédentes.

Sur la page « Chess Boxing », l'approche de Ngonmang et al. (2012) ne trouve que deux très petites communautés ce qui est un défaut classique des approches à base de fonction de qualité. Les deux communautés identifiées sont :

– une communauté de 7 sommets : Comic book, Enki Bilal, Cartoonist, La Foire aux immortels, La Femme Piège, Froid-Équateur et Chess boxing. Cette communauté est très similaire à celle que notre algorithme à étiqueté « Enki Bilal » et est parfaitement pertinente.

– une communauté de 5 sommets : Germany, Netherlands, 1991, International Arctic Science Committee et Chess boxing. Cette seconde communauté ne ressemble à aucune de celles trouvées par notre algorithme et nous n'arrivons pas à comprendre son origine ni à justifier sa pertinence.

5 Conclusion et perspectives

Nous avons proposé un algorithme qui permet d'identifier les communautés ego-centrées pour un sommet d'un graphe. Notre approche n'est pas basée sur l'optimisation d'une fonction de qualité mais sur la recherche d'irrégularités dans la décroissance des valeurs d'une mesure de similarité. L'algorithme est efficace en temps et permet de trouver les communautés d'un sommet sur des graphes contenant des millions de sommets.

En utilisant la notion de communautés multi-ego-centrées, l'algorithme identifie dans un premier temps des sommets candidats pouvant permettre à l'identification de communautés, puis cherche des communautés centrées sur notre cible et sur ces candidats, et procède enfin à une phase de nettoyage et d'étiquetage des communautés.

Bien que cet algorithme soit performant en l'état, de nombreuses pistes d'amélioration sont possibles. Tout d'abord, la détection de communautés se base sur la recherche de plateaux et de décroissances fortes. La méthode actuelle peut être améliorée, notamment par la recherche de plusieurs décroissances, ce qui permettrait de trouver plusieurs communautés à des échelles différentes pour un même candidat.

De plus, l'algorithme n'utilise pour l'instant qu'une notion de communautés bi-centrées et il est possible que certaines communautés n'apparaissent que centrées sur 3 sommets ou plus. Cette généralisation doit être validée sur des exemples de petites tailles car le temps de calcul sera fortement augmenté s'il faut tester toutes les paires de candidats au lieu de tester tous les candidats. Afin que cette méthode soit utilisable une piste serait notamment d'améliorer la méthode de sélection. Par exemple, si un candidat $s2$ a fourni de bons résultats alors il est probable que des sommets très similaires à $s2$ n'apporteront pas de nouvelle information et qu'ils pourraient donc être laissés de coté.

Cette notion de rapidité de l'algorithme est centrale car comme on l'a vu avec la disparition du lien de « Chess Boxing » vers « Gloria » les communautés vont évoluer. Si l'on souhaite donc suivre l'évolution des communautés sur plusieurs instants il est important que les calculs soient aussi efficaces que possible.

Enfin, nous avons vu que l'algorithme peut avoir des difficultés à identifier de petites communautés si elles sont proches de grosses communautés. Pour cette raison, tenter d'appliquer la méthodologie à des sommets très populaires tels que « Biology » ou « Europe » ne conduit qu'à une grosse communauté alors que l'on s'attendrait à trouver divers sous-domaines de la biologie ou différents pays européens. Ceci peut s'améliorer par exemple en relançant récursivement l'algorithme sur des communautés identifiées pour trouver des sous-communautés.

Remerciements

Ce travail est partiellement soutenu par l'Agence Nationale pour la Recherche, projet DynGraph ANR-10-JCJC-0202.

Références

Blondel, V., J. Guillaume, R. Lambiotte, et E. Lefebvre (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics : Theory and Experiment 2008*(10), P10008.

Chen, J., O. Zaïane, et R. Goebel (2009). Local community identification in social networks. In *Social Network Analysis and Mining, 2009. ASONAM'09. International Conference on Advances in*, pp. 237–242. IEEE.

Clauset, A. (2005). Finding local community structure in networks. *Physical Review E 72*(2), 026132.

Danisch, M., J. Guillaume, et B. Le Grand (2012). Towards multi-ego-centered communities : a node similarity approach. *Int. J. of Web Based Communities*.

Evans, T. et R. Lambiotte (2009). Line graphs, link partitions, and overlapping communities. *Physical Review E 80*(1), 016105.

Fortunato, S. (2010). Community detection in graphs. *Physics Reports 486*(3), 75–174.

Friggeri, A., G. Chelius, et E. Fleury (2011). Triangles to capture social cohesion. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pp. 258–265. IEEE.

Girvan, M. et M. Newman (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences 99*(12), 7821–7826.

Luo, F., J. Wang, et E. Promislow (2008). Exploring local community structures in large networks. *Web Intelligence and Agent Systems 6*(4), 387–400.

Newman, M. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical review E 74*(3), 036104.

Ngonmang, B., M. Tchuente, et E. Viennet (2012). Local community identification in social networks. *Parallel Processing Letters 22*(01).

Palla, G., I. Derényi, I. Farkas, et T. Vicsek (2005). Uncovering the overlapping community structure of complex networks in nature and society. *Nature 435*(7043), 814–818.

Palla, G., I. Farkas, P. Pollner, I. Derényi, et T. Vicsek (2008). Fundamental statistical features and self-similar properties of tagged networks. *New Journal of Physics 10*(12), 123026.

Summary

We propose here a framework to unfold the ego-centered community structure of a given node in a network. The framework is not based on the optimization of a quality function, but on the study of the irregularity of the decrease of a similarity measure. It is a practical use of the notion of multi-ego-centered community and we validate the pertinence of the approach on a real-world network of wikipedia pages.

Mesurer l'intrication sémantique dans une collection de documents

Benjamin Renoust[*,**], Guy Mélançon[**], Marie-Luce Viaud[*]

[*]Institut National de l'Audiovisuel
{brenoust, mlviaud}@ina.fr,
[**]CNRS UMR 5800 LaBRI & INRIA Bordeaux Sud-Ouest
{benjamin.renoust, guy.melancon}@labri.fr

Résumé. Avec l'explosion des ressources en ligne, l'exploration efficace de collections de documents est une tâche quotidienne pour bon nombre de professions, mais paradoxalement reste un enjeu pour la recherche. Nous proposons trois outils pour aider à percevoir le contenu sémantique d'un ensemble de documents : un réseau pondéré d'interaction des termes associés au jeu, une mesure d'intensité et une mesure d'homogénéité d'intrication sémantique reflétant les caractéristiques de partage des termes par les différents items de la collection. De plus, ces mesures sont utilisées pour catégoriser les collections.

1 Introduction

La plupart des utilisateurs de système d'information explore quotidiennement des collections de documents. Les scientifiques parcourent aussi chaque jour des bases de données de publications. Les journalistes interrogent des bases de données à la recherche de documents publiés autour d'un évènement. Ainsi les sociétés ou instituts spécialisés (tels que l'AFP, Reuters, l'Ina, la BnF...) offrent des services de requête et de recherche dans leur archives documentaires. La réalisation de nouvelles techniques ou l'amélioration de techniques d'exploration et de recherche des collections de documents satisfait une réelle demande des utilisateurs tant grand public qu'acteurs industriels.

La plupart des moteurs de recherche classe les documents correspondant à une requête suivant un ordre de pertinence [Croft et al. (2009)], parmi ces indices, Pagerank [Brin et Page (1998)] est sûrement le plus connu et le plus utilisé. Cependant ce classement ne prend pas en compte directement la sémantique des documents retournés et renvoie un ordre souvent peu compréhensible à l'utilisateur.

Notre contribution vient en complément des approches d'indexation et d'identification classiques. Nous introduisons un *réseau pondéré d'interaction des termes* permettant d'introduire plusieurs mesures d'intrication sémantique d'une collection. Ce réseau est généré à partir de termes associés à la collection quelque soit leur provenance (annotations manuelles, statistiques, LSA, LDA...). Nous conduisons une analyse spectrale inspirée des analyses de réseaux sociaux [Burt et Scott (1985)] afin de définir un *indice d'intrication* associé à chaque terme et deux mesures globales d'intrication sémantique. Notre objectif est de permettre à l'utilisa-

teur d'évaluer sémantiquement la cohésion d'une collection ou d'un sous-ensemble désigné ou d'identifier des zones de forte intrication.

Ce papier présente brievement l'état de l'art avant d'introduire en section 2 le réseau d'interaction des termes et les indices d'intrication. Une étude de cas est ensuite discutée afin de montrer le potentiel d'utilisation de ces outils (section 3). Nous montrons ensuite le graphe d'interaction des termes et les indices d'intrication sur des données obtenues à l'Ina [1]. Nous rapportons enfin les résultats d'une étude utilisateurs conduite avec des documentalistes experts. Nous concluons avec une discussion en section 5.

1.1 État de l'art

L'analyse de collections de documents considère souvent une matrice de co-occurrence à partir de laquelle différents indices peuvent être dérivés. Un indice connu est le tf-idf [Salton (1983)] qui détermine un poids pour des termes. Les documents d, d' peuvent alors être considérés comme des vecteurs de poids indexés par terme, correspondant à une ligne dans la matrice de co-occurrence. Ces vecteurs peuvent être utilisés pour évaluer les (dis)similarités entre documents. La similarité cosinus $\cos(d, d') = \frac{\langle d, d' \rangle}{\|d\| \|d'\|}$ est une autre mesure connue et très utilisée.

Depuis les travaux pionniers de Salton (1983), les chercheurs ont proposé des améliorations au model "bag-of-words" (par exemple [Beaza-Yates et Ribeiro-Neto (1999)]). L'analyse sémantique latente (LSA) [Dumais et al (1988)] ou l'indexation sémantique latente (LSI) [Deerwester et al. (1990)] exploite l'idée que les mots ayant un sens similaire apparaissent souvent de manière proche dans un texte. Ces méthodes évaluent la proximité sémantique en appliquant une décomposition en valeurs singulières sur une matrice d'occurences de termes. L'indexation latente sémantique probabiliste (PSLI) [Thomas (1999)] s'appuie sur une décomposition mixte dérivée d'un modèle latent classique qui peut être ajusté par un algorithme d'esperance-maximisation (EM).

L'allocation de Dirichlet latente (LDA) [Blei et al. (2003)] est un modèle de sujets similaire à PLSI, où chaque document est vu comme un mélange de sujets variés. LDA suppose que chaque document est un mélange d'un petit nombre de sujets, où la présence des mots dans les documents est attribuable à l'un des sujets du document. LDA utilise une modélisation probabiliste des fréquences d'occurrence de termes dans les documents pour définir un sujet, ayant pour but de trouver les termes ou sujets les plus pertinents dans un jeu de documents. Les documents peuvent ainsi être décrits en utilisant un vecteur pondéré ou une distribution probabiliste par terme, ce qui permet à l'utilisateur de calculer des similarités entre documents pouvant ainsi nourrir différents algorithmes de recherche et/ou d'agrégation sur des collections de documents [Kohonen et al. (2000) ou Zhao et al. (2005)]. Il est necessaire ici de bien distinguer un sujet d'un terme. En pratique, LDA calcule une distribution de *sujets* en utilisant une collection de documents, ce qui correspond à une distribution de probabilités associée à un jeu de mots présents dans la collection de documents. Nous nous intéressons seulement aux *termes* qui correspondent aux mots trouvés dans des documents ou issus d'un vocabulaire contrôlé utilisé pour l'indexation des documents.

Notre approche diffère de ces techniques d'indexation sur plusieurs points. Nous considérons d'abord le réseau d'interaction commme un élement central duquel sont dérivés nos

1. www.ina.fr

indices d'intrication et à partir duquel nous pouvons tirer nos conclusions. Notre approche peut être semblable à celle de Arun et al (2010) en ce qu'elle considère un réseau termes-documents plutôt que la matrice termes-sujets utilisée par LDA. Arun et al (2010) utilisent une matrice de sujets-documents pour estimer le véritable nombre de sujets présents dans une collection de documents. Nos objectifs sont différents puisque nous visons à établir si un groupe de documents forme un groupe cohésif selon un jeu de termes d'indexation donné. Cependant la topologie du reseau d'interaction (composantes connexes, densités) permet de faire émerger les différents sujets (donc leur nombre) emmêlés dans la collection de documents (section 3)

L'indice d'intrication peut être déterminé sur *n'importe* quel groupe de documents et à partir de *n'importe* quel jeu de termes indexant ces documents. Notre technique apparaît alors comme une procédure a posteriori, offrant un retour sur une quelconque procédure d'agrégation ou d'indexation. Les indices d'intrication sont basés sur les interactions qui prennent place entre les termes (section 2) et qui exploitent totalement la topologie du réseau d'interaction.

2 Intrication sémantique

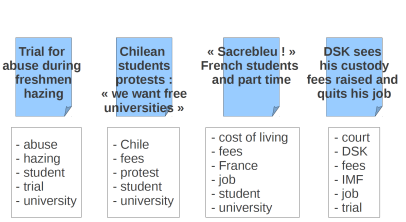
Nous allons maintenant définir un indice d'intrication sur la base d'une analyse spectrale d'un réseau d'interaction des termes. Soit D une collection de documents $d \in D$, chacun indexé par des termes $t \in T$, où T denote une collection de termes. Les *termes* ici *indexent* les documents et correspondent à des mots issus d'un vocabulaire contrôlé (thésaurus) ou extraits des documents. Nous supposons ici que les termes ont déjà étés identifiés et/ou déterminés, ainsi tous les documents sont associés à un jeu de termes. Soit $M = (m_{d,t})_{d \in D, t \in T}$ la matrice d'occurrence, avec $m_{d,t}$ le nombre d'occurrences du terme t dans le document d. Le document d peut alors être vu comme un vecteur de poids indexés par les termes $t \in T$, soit, $d = (m_{d,t})_{t \in T}$ correspond à une ligne dans la matrice d'occurrence M.

2.1 Réseau d'interaction des termes

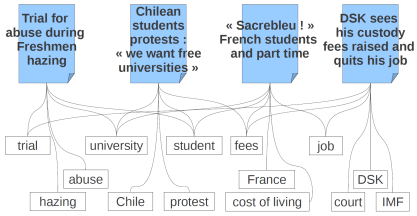
On peut aussi définir les relations documents-termes avec une représentation sous forme de graphes. La matrice d'occurrence correspond en effet à un graphe $G_{D,T} = (V, E)$, dont les noeuds sont soit des documents soit des termes, $V = D \cup T$ et les arêtes $e = \{d, t\} \in E$ connectent les documents aux termes. Ce graphe est de toute évidence *biparti*, puisque les arêtes ne connectent jamais deux documents ou deux termes directement. La figure 1 illustre cette construction depuis un jeu de 4 documents différents indexés par des termes (1 (a)). La figure 1 (b) correspond au graphe biparti defini à partir de ces documents et termes.

Les arêtes $e \in E$ peuvent être pondérées, $\omega : E \to \mathbb{R}$. Une fonction de poids évidente serait $\omega(e) = m_{d,t}$. Si les documents sont indexés par des termes générés par LDA, le poids $\omega(e) = P(t|d)$ pourra être la probabilité donnée par le modèle. N'importe quelle autre fonction de poids résultant de l'indexation des documents peut aussi être utilisée. La littérature présente plusieurs techniques et algorithmes permettant d'exploiter ce graphe biparti afin de rechercher ou segmenter la collection de documents [Xu et al. (2003)], le jeu de termes [Slonim et Tishby (2000)] ou bien les deux en même temps [Dhillon (2001)].

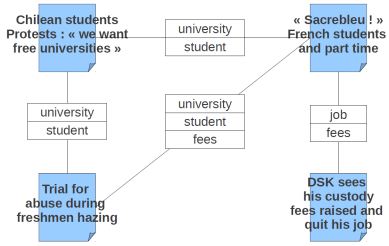
Ce graphe biparti est utilisé pour dériver le graphe $G_D = (D, E_D)$, reliant directement les documents. Le graphe est construit à partir de $G_{D,T}$ en projetant les chemins $d - t - d'$ sur les arêtes $e = \{d, d'\} \in E_D$. La figure 1 (c) illustre comment G_D est obtenue à partir de



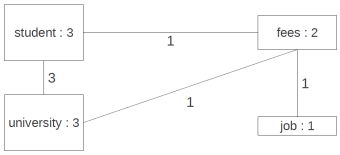
(a) Documents D indexés par les termes T



(b) Graphe biparti $G_{D,T} = (D \cup T, E)$



(c) Graphe document-document $G_D = (D, E_D)$



(d) Réseau d'interaction des termes $I_T = (T, F)$

FIG. 1 – *A partir d'une collection de documents indexés par des termes (a), nous construisons un graphe biparti liant documents et termes (b). Nous considérons ensuite le graphe document-document avec les termes en attributs des arêtes (c) et en dérivons le* réseau d'interaction des termes *(d) (ici pondéré par le nombre d'occurrences et co-occurrences des termes dans G_D).*

$G_{D,T}$ (il *n'inclue aucune boucle* reliant un document à lui-même). L'arête $e = \{d, d'\}$ entre les documents d et d' sur $G_{D,T}$ est induite s'ils partagent les termes distincts $t, t', \ldots$. Ces termes deviennent des attributs *associés* à l'arête e. Notre analyse portant sur l'interaction des termes, nous ne considérons dans le graphe G_D qu'uniquement les arêtes partageant *au moins 2 termes distincts*.

Nous construisons maintenant un *réseau d'interaction des termes*. Considerons le graphe $I_T = (T, F)$, avec pour noeuds les termes $t \in T$. L'arête $f \in F$ relie les termes $t, t' \in T$ lorsqu'ils sont *ensemble associés* à l'arête $e = \{d, d'\} \in E_D$, c'est à dire, lorque deux documents distincts $d, d' \in D$ sont tous les deux indexés par les termes t, t'.

Le graphe $I_T = (T, F)$ n'est pas tout à fait construit à partir de $G_{D,T}$ en projetant les chemins $t - d - t'$ sur les arêtes $e = \{t, t'\}$, mais plutôt à partir des arêtes de G_D. La figure 1 (d) illustre comment I_T est obtenu à partir du graphe G_D. Le *réseau d'interaction des termes*, défini *après* que les documents ont été indexés, est notre principal objet d'intéret pour explorer l'espace des documents.

L'idée de la construction d'un réseau d'interaction des termes est empruntée à l'analyse des réseaux sociaux [Burt et Scott (1985)]. Nous calculons *les matrices d'interaction* $N_I = (n_{t,t'})$ et $C_I = (c_{t,t'})$ (où les indices correspondent aux termes $t, t' \in T$). Soit $e \in E_D$ une arête de G_D et $\tau(e) \subset T$ le jeu de termes associés à e. De la même manière, soit $\tau^{-1}(t)$ le jeu d'arête $e \in E_D$ ayant pour terme associé $t \in T$. Nous écrivons $n_t = |\tau^{-1}(t)|$ la cardinalité de ce

jeu. Nous définissons alors $n_{t,t'}$ le nombre d'arêtes $e \in E_D$ avec $\{t, t'\} \subset \tau(e)$, c'est à dire, $n_{t,t'}$ est égale au nombre d'arêtes $e \in E_D$ portant à la fois les termes t et t'. En d'autres mots, $n_{t,t'} = |\tau^{-1}(t) \cap \tau^{-1}(t')|$.

Définissons $c_{t,t} = \frac{n_{t,t}}{|F|}$, et $c_{t,t'} = \frac{n_{t,t'}}{n_t}$. Si la matrice N_I est symétrique, la matrice C_I elle, ne l'est pas. Les entrées de la diagonale $c_{t,t}$ dans la matrice C_I peuvent être considérées de manière non formelle comme la probabilité qu'une arête de E_D soit associée au terme t. Les autres entrées $c_{t,t'}$ correspondraient alors aux probabilités conditionnelles qu'une arête soit associée au terme t sachant qu'elle est associée au terme t'.

Considérons les matrices N_I (à gauche) et C_I (à droite) ci-dessous. Ces matrices sont déterminées à partir de la clique de 5 termes de la figure 2, section 2.3, indexant une collection de 18 documents partageant 103 liens. En regardant la diagonale, $n_{1,1} = 71$ on voit que les liens sont associés avec le premier terme (sécurité routière), et $n_{2,2} = 48$ avec le second (prévention des accidents). Le nombre de liens associés à la fois avec le premier et le second terme est donc $n_{1,2} = n_{2,1} = 35$. En regardant la première entrée de C_I, le premier terme est associé à $c_{1,1} = 69\%$ pour tous les liens, et il y a $c_{1,2} = 73\%$ de chance de trouver un lien associé avec le second terme parmi tous ceux associés au premier terme, lorsque seuls $c_{2,1} = 49\%$ des liens qui sont associés avec le second terme le sont aussi avec le premier.

$$\begin{bmatrix} 71 & 35 & 61 & 46 & 28 \\ 35 & 48 & 35 & 41 & 15 \\ 61 & 35 & 78 & 42 & 45 \\ 46 & 41 & 42 & 67 & 21 \\ 28 & 15 & 45 & 21 & 45 \end{bmatrix} \quad \begin{bmatrix} 0.69 & 0.73 & 0.78 & 0.69 & 0.62 \\ 0.49 & 0.47 & 0.45 & 0.61 & 0.33 \\ 0.86 & 0.73 & 0.76 & 0.63 & 1 \\ 0.65 & 0.85 & 0.54 & 0.65 & 0.47 \\ 0.39 & 0.31 & 0.58 & 0.31 & 0.44 \end{bmatrix}$$

2.2 Indice d'intrication

Nous souhaitons maintenant mesurer *l'indice d'intrication* soit la participation d'un terme t dans l'intrication globale du groupe. Cette notion d'intrication est adaptée directement d'une notion similaire, l'ambigüité des relations sociales [Burt et Scott (1985)]. Posons λ l'indice d'intrication maximal parmi tous les termes et γ_t la fraction d'intrication correspondant au terme t. L'indice pour le terme t peut alors être déterminé comme $\gamma_t \cdot \lambda$. Comme l'intrication d'un terme est renforcée par ses interactions avec d'autres termes intriqués, et comme les entrées $c_{t,t'}$ de la matrice C_I offrent une interprétation probabiliste, nous pouvons poser l'équation suivante définissant les valeurs γ_t.

$$\gamma_{t'} \cdot \lambda = \sum_{t \in T} c_{t,t'} \gamma_t \tag{1}$$

Le vecteur $\gamma = (\gamma_t)_{t \in T}$, collecte alors les valeurs pour tous les termes t, formant ainsi le vecteur propre *à droite* de la matrice transposée C'_I. A partir de l'eq. (1) nous pouvons lever l'équation matricielle $\gamma \cdot \lambda = C'_I \cdot \gamma$. L'indice maximal d'intrication sera égal à la valeur propre maximale de la matrice C'_I. Nous nous intéressons ainsi aux valeurs relatives γ_t. La section suivante présente à partir du vecteur γ_t et de la valeur λ *l'intensité d'intrication* et *l'homogénéité d'intrication* comme mesures globales de l'intrication d'un graphe.

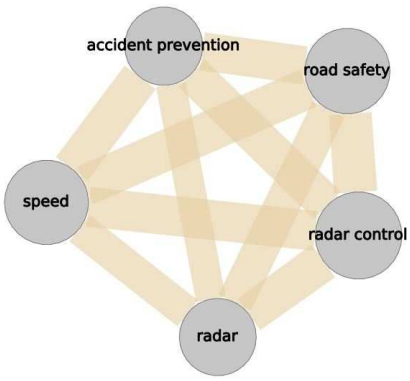


FIG. 2 – *Un groupe de documents avec une intrication optimale a pour réseau d'interaction des termes une clique où chaque terme interagit avec les autres identiquement.*

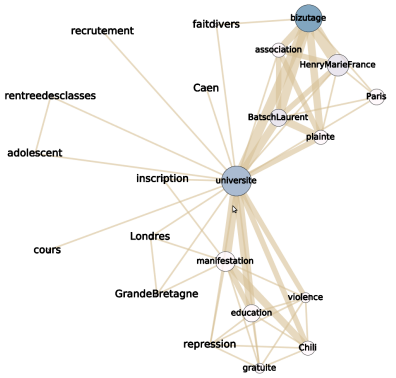


FIG. 3 – *Un réseau d'interaction sous forme d'étoile regroupe (a) un(des) terme(s) central(aux), avec un(des) indice(s) d'intrication plus élevé. Les termes périphériques peuvent former des sous-groupes denses avec des indices d'intrication* localement *plus élevés.*

2.3 Profils d'intrication

La topologie du réseau d'interaction des termes $I = (T, F)$ fournit une information complémentaire sur la façon dont les termes contribuent à l'intrication sémantique globale d'un groupe ou sous-groupe de documents. L'attention se porte ici sur les interactions entre les termes au sein d'une collection

L'archétype d'un *groupe de documents optimal cohésif* est une configuration dans laquelle tous les documents sont indexés exactement par les mêmes termes (et par conséquent tous les termes sont intriqués entre eux de manière maximale). Le graphe $I = (T, F)$ correspond alors à une *clique*. Dans ce cas, toutes les entrées $n_{t,t'}$ de la matrice N_I coïncident, et toutes les entrées de la matrice C_I valent 1. La valeur propre maximale C'_I est dans ce cas égale à $\lambda = |F|$, et tous les indices γ_t coïncident. La théorie de Perron-Frobenius sur les matrices non-négatives (Ding et Zhou, 2009, Chap. 2) montre en plus que $\lambda = |F|$ est la valeur propre maximale possible pour une telle matrice avec ses entrées comprises dans l'intervalle $[0, 1]$.

La situation opposée est celle où le réseau d'interaction des termes n'est pas connexe. Les termes se séparent en plusieurs sous réseaux qui n'interagissent jamais entre eux. C'est une force de la représentation du réseau de pouvoir identifier directement les composantes connexes (qui doivent être analysées chacune indépendamment). En effet, lorsque $I = (T, F)$ est non connexe, la matrice C_I est considérée comme *réductible* ; à l'inverse, lorsque $I = (T, F)$ est connexe, la matrice C_I est *irreductible*. Dans ce cas, la théorie des matrices non négatives nous apprend que la matrice C_I a une valeur propre maximale $\lambda \in \mathbb{R}$ avec un seul vecteur propre associé γ_t. Ce vecteur propre a des entrées réelles et non négatives [(Ding et Zhou, 2009, Theorem 2.6)]. Nous considérons par la suite que C_I est irréductible.

Un autre cas intéressant apparaît typiquement lorsque quelques termes sont centraux et le

reste des termes périphériques : d'une part, tous les documents partagent quelques termes en commun, se regroupant donc autour de quelques termes centraux ; d'autre part, ces documents forment des sous-groupes autour de termes ou sous-sujets secondaires. Le réseau d'interaction des termes présente alors une structure en forme d'étoile (figure 3). L'indice d'intrication est plus élevé pour les termes centraux alors que les autres termes ont des indices bien plus faibles (les noeuds de la figure 3 sont colorés en fonction de leur indice d'intrication). Les termes périphériques peuvent aussi former des sous-réseaux plus denses entre eux. Lorsque l'on calcule les indices d'intrication restreints à ce sous-réseau de termes, les valeurs se rapprochent de celles du scénario de la clique. Les études de cas présentées ci-dessous développent cette situation.

Nous pouvons déterminer l'intrication au niveau du groupe de document par comparaison avec les valeurs de l'archétype de la configuration optimale. Nous avons vu auparavant que la valeur propre maximale est délimitée par $|F|$, ainsi le ratio $\frac{\lambda}{|F|} \in [0,1]$ mesure combien intense est l'interaction des termes dans un groupe de documents. Ce ratio nous fournit une mesure de *l'intensité d'intrication* parmi tous les documents, *sous considération du jeu de termes* T.

De la même manière, lorsque les poids $c_{t,t'}$ sont identiques nous obtenons un vecteur propre avec des entrées identiques. Ce vecteur propre détermine alors l'espace diagonal généré par le vecteur diagonal $1_T = (1, 1, \ldots, 1)$. Ceci motive la définition d'une seconde mesure liée à l'homogénéité de la distribution de l'intrication parmi les termes. Nous déterminons ainsi une similarité cosinus $\frac{\langle 1_T, \gamma \rangle}{\|1_T\| \|\gamma\|} \in [0,1]$ qui exprime la proximité du groupe de document avec la situation d'intrication optimale. Nous l'appellerons *l'homogénéité d'intrication*.

3 Etude de cas

Cette section présente deux cas d'usage illustrant l'utilisation des mesures d'intrication et du réseau d'interaction des termes pour l'exploration d'un groupe de documents.

Chacun de ces cas d'usage a été construit à partir des extraits de journaux télévisés (JTs) couvrant plusieurs sujets sur une période de 100 jours. Les documents ont été manuellement indexés par l'INA. Les groupes de documents ont été identifiés en utilisant des approches de clustering classiques (ces approches n'étant pas le sujet de ce papier).

3.1 La sécurité routière et les radars

Nous considérons tout d'abord un jeu de 20 documents, tous liés à la sécurité routière. Bien que petit, cet ensemble de documents présente des caractéristiques intéressantes. La sécurité routière est devenue un sujet d'intéret suite à la promotion par le gouvernement des radars automatiques, impliquant une augmentation du nombre d'amendes. Ce sujet a bien sûr été abordé par chacun des JTs. Les documents ont été par la suite annotés avec des termes tels que *arrestation, automobiliste, conduite, danger, prévention des accidents, radar, sécurité routière, société, vitesse, etc.* La figure 4 montre le réseau d'interaction des termes en résultant.

Les noeuds les plus sombres ont un indice d'intrication plus élevé. La taille des noeuds correspond au nombre de liens associés au terme *dans le graph* G_D. Le dessin du graphe nous montre que les noeuds centraux *prévention des accidents* et *vitesse* ont des indices d'intrication plus élevés (respectivement 0.38 et 0.44). L'intensité d'intrication pour tout le réseau $I = (T, F)$ est de $\lambda/|F| = 0.33$, et l'homogénéité (similarité cosinus 2.3) est de 0.81.

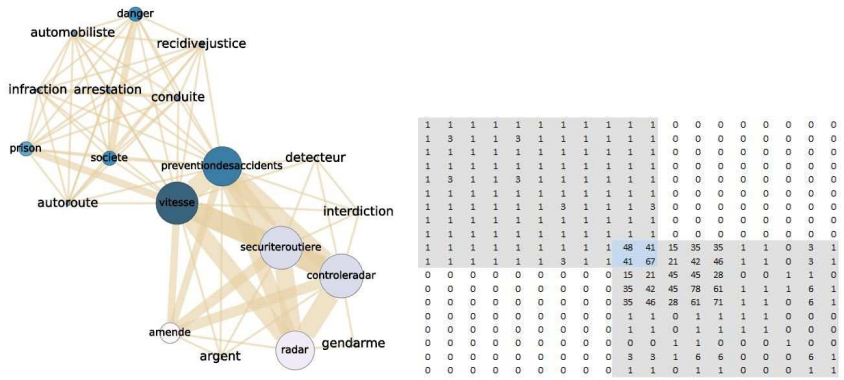


FIG. 4 – *Réseau d'interaction des termes déterminé à partir d'un jeu de documents liés à la sécurité routière et aux radars. Le réseau se divise en deux composantes organisées autour des termes centraux* prévention des accidents *et* vitesse*, ainsi menant à une évidente structure en blocs de la matrice d'interaction (une fois ordonnée).*

La figure 4 montre que les termes se divisent en deux groupes indiquant pourquoi l'intrication sémantique maximale ne peut pas être observée. La matrice d'interaction N_I présente de même une structure en blocs (en grisé), avec des valeurs hors diagonale nulles. Les termes centraux sont dans la partie bleue de la représentation matricielle, interagissant avec tous les autres termes. La partie supérieure de la matrice correspond à la partie supérieure du réseau d'interaction et montre que tous ces termes interagissent ensemble de manière peu fréquente (c.a.d. les termes indexent un petit sous-ensemble de tous les documents). La partie inférieure de la matrice présente un comportement complètement différent, où cinq termes interagissent de manière plus intensive ensemble, mais pas avec tous les autres termes de la composante.

La topologie du réseau suggère d'examiner de plus près les documents liés aux termes de la partie inférieure, positionnés sous les termes centraux *prévention des accidents* et *vitesse*. Nous considérons un sous-graphe I' formé à partir des termes *amende*, *gendarme*, *radar*, *sécurité routière*, etc. Les termes de I' indexent tous les documents à l'exception d'un seul. Pour ce sous-réseau I', nous mesurons une intensité de 0.31 et une homogénéité de 0.72, ce qui est légèrement inférieur aux mesures du réseau complet I. Ces valeurs sont plus faibles car beaucoup de termes comme *amende*, *argent* et *detecteur* n'indexent que peu de documents (comme leur taille le suggère) et les termes de I' se distribuent de manière inégale sur les arêtes entre documents en comparaison de la distribution de tous les termes de I (comme suggéré par les entrées nulles de la partie en bas à droite de la matrice). Notons que la fonction $\cos(-)$ est non linéaire pour bien interpréter toute variation dans la mesure d'homogénéité.

Nous pouvons maintenant nous concentrer sur la clique de 5 noeuds de la figure 2 (section 2.3) associée à 18 des 20 documents. Comme prévu elle atteint des valeurs d'intrication plus élevées avec 0.6 en intensité et 0.98 en homogénéité, ce qui confirme que ces 18 documents forment bien un tout cohésif autour de ces 5 termes.

Nous terminons ce premier exemple en regardant la partie supérieure du réseau, formée par les termes positionnés au dessus des termes centraux *prévention des accidents* et *vitesse*. Nous

considérons un sous-graphe composé des termes *danger*, *automobiliste*, *autoroute*, *prison*, etc. Nous obtenons le sous-réseau I'' en haut de la figure 4, où les termes indexent 19 des 20 documents originaux. En restreignant l'analyse à ce sous-réseau I'', l'intensité et l'homogénéité d'intrication sont à 0.57 et 0.98. Après suppression des termes centraux *prévention des accidents* et *vitesse*, les mesure d'intensité et d'homogénéité montent à 0.71 et 0.99. Nous trouvons ainsi que ces documents forment une unité cohésive sémantiquement formant incidentellement une clique avec des poids d'interaction inégaux $n_{t,t'}$. Cette conclusion doit malgré tout être modérée parce que les termes de ce second sous-réseau ne concernent que 4 documents. On constate ici que de fortes valeurs en intensité et en homogénéité sont bien plus faciles à atteindre sur des ensembles réduits de documents et de termes.

3.2 À propos des profils d'intrication

Nous retournons maintenant sur la notion de profils d'intrication en considérant l'exemple discuté précédement. Nous avons utilisé l'intensité $\frac{\lambda}{|F|}$ et l'homogénéité $\frac{\langle 1_T,\gamma\rangle}{\|1_T\|\|\gamma\|}$ comme deux mesures distinctes afin de fournir une information complémentaire au réseau d'interaction des termes. Cet exemple montrent des situations où ces mesures varient beaucoup. Bien que l'on suspecte que ces quantités ne varient pas indépendament, nous pouvons néanmoins placer ces variables dans un plan où l'intensité serait notée sur l'axe x et l'homogénéité sur l'axe y (figure 5). N'importe quel réseau d'interaction peut alors être placé comme un point sur ce plan $(x, y) = (\frac{\lambda}{|F|}, \frac{\langle 1_T,\gamma\rangle}{\|1_T\|\|\gamma\|})$, ainsi nous pouvons diviser ce plan en zones correspondant à des profils différents. Une supposition naïve est de diviser ce plan en 4 zones plus ou moins rectangulaires (lignes en pointillés). C'est assez loin de la réalité, on suspecte que les véritables zones suivent des formes plus complexes et le problème reste à étudier plus amplement, mais nous observons malgré tout 4 catégories.

La clique qui était présentée comme l'archétype du réseau d'interaction optimal se situe dans la partie la plus en haut à droite du plan $(x, y) = (1, 1)$. La zone supérieure droite correspond aux réseaux relativement denses et homogènes. La clique de 5 noeuds du "Radar" de la figure 2 appartient à cette catégorie de profil tout comme le sous réseau supérieur de la "Sécurité routière" (figure 4, section 3.1).

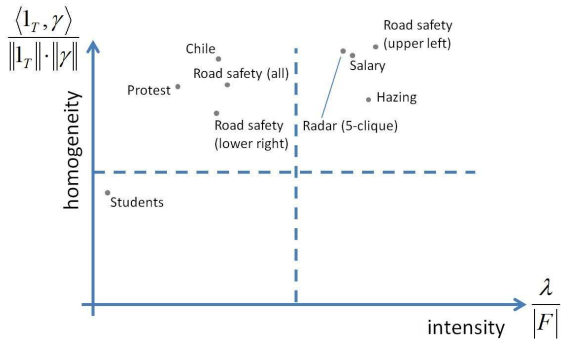


FIG. 5 – *Les profils d'intrication peuvent être grossièrement catégorisés en combinant intensité d'intrication et homogénéité d'intrication pour identifier 4 zones critiques.*

La partie supérieure gauche correspond à une homogénéité relativement élevée avec une intensité relativement faible : les termes interagissent pratiquement tous entre eux, mais pas autant que le graphe des documents G_D ne l'autorise en théorie. Les matrices N_I ne sont pas creuses, et ont une diagonale plutôt large avec des entrées hors diagonales plus faibles. La partie inférieure du réseau sur la "Sécurité Routière" (Figure 4), en est un bon exemple.

La situation de la partie inférieure droite apparaît lorsque les termes sont imbriqués un peu comme s'ils exprimaient des concepts similaires à différents niveaux de généralité. Cette situation se traduit par des inclusions consécutives des termes dans les arêtes des documents (c.a.d. les liens de G_D associés avec un terme t incluent tous les liens associés au terme t' avec en plus quelques autres liens).

La partie inférieure gauche rassemble des réseaux avec de faibles intensités et homogénéités d'intrication. C'est un cas commun qui bien souvent rassemble des documents et des termes avec peu d'interaction. Dans cette situation, beaucoup de termes sont satellites de termes centraux, avec parfois plusieurs centres. Un jeu de termes couvrant un paysage sémantique large induit inévitablement ce genre de situation. Un réseau typique de cette situation présente une faible densité (peu d'arêtes), menant à une matrice N_I creuse avec des entrées d'ordre ϵ. Le réseau d'interaction de départ de notre exemple tombe dans cette catégorie.

Bien que ces zones nous fournissent une grille intéressante pour l'évaluation des profils de réseaux, nous avons besoin de plus d'expérimentations pour confirmer ces catégories et déterminer les limites de ces zones et pour estimer de quelle manière elles sont peuplées.

4 Étude utilisateur

Notre objectif est de fournir à l'utilisateur des outils pour évaluer plus rapidement et avec plus d'assurance le contenu d'une collection de documents. Nous avons testé nos outils auprès de 4 documentalistes experts (2 séniors et 2 juniors) de l'INA. Bien que restreinte, cette expérience a été conçue autour d'un protocole strict : 4 jeux de notices documentaires de reportages de JT concernant 4 évènements médiatiques ont été générés avec des techniques de clustering classique, avec des tailles et des valeurs d'intensités et d'homogénéités d'intrication variées.

Les utilisateurs disposaient de deux interfaces : la première propose une liste de documents permettant d'inspecter librement les titres, leur contenu et leurs termes d'indexation ; la seconde interface proposait une représentation interactive et synchronisée du graphe de documents et du réseau d'interaction des termes précédemment introduit. Les indices d'intrications sont calculés sur les sous-graphes sélectionnés par l'utilisateur.

Les utilisateurs avaient 10 minutes pour se familiariser avec la tâche et les interfaces. Les jeux de données et les interfaces ont été distribués aléatoirement de manière à éviter les biais. Un entretien et un questionnaire clôturaient cette évaluation, d'une durée moyenne de 2h30.

Les tâches demandées aux utilisateurs étaient les suivantes :

- évaluer la cohésion sémantique globale de chacun des groupes de documents ;
- éliminer les documents "bruitant" une collection afin de renforcer la cohésion sémantique globale de la collection et indiquer les documents qui leur apparaissaient mal indexés (mauvais termes) ;
- dans une collection donnée, trouver des documents correspondant à une requête donnée ;
- raconter l'histoire expliquant le contenu d'une collection de documents (qui devrait alors plus ou moins correspondre avec l'évènement couvert par les JT) ;

– exprimer leur confiance dans leur analyse (par exemple, avoir écarté les *bons* documents, et raconté la *bonne* histoire).

La compilation des interviews réalisées rapportent les observations suivantes : les utilisateurs ont apprécié le réseau d'interaction pour son utilité à identifier/discriminer les différents termes, pour la concision de sa représentation. Ils ont confirmé son utilité quant à la compréhension d'une collection de documents dans son ensemble. Les commentaires recueillis lors des entretiens révèlent que les utilisateurs pouvaient développer une bonne intuition du type de collection de documents à partir de la forme du réseau, et utiliser cette intuition pour identifier ses caractéristiques importantes (comme les termes centraux et périphériques, le découpage d'un thème en plusieurs petites histoires). Les utilisateurs ont aussi confirmé le lien entre la cohésion sémantique perçues et les mesures d'intrication retournées. Les documents annotés par des termes de faible indice d'intrication correspondent bien à des documents " à la limite " des sujets principaux des jeux et cet indicateur est donc très pertinent. Finalement, les utilisateurs ont trouvé que l'utilité apportée par le réseau d'interaction des termes grandissait avec l'augmentation de la taille du jeu, ce que semblent confirmer nos premières mesures de temps (bien qu'on ne puisse le dissocier pour le moment d'un effet d'apprentissage de l'interface).

5 Conclusion

Ce papier présente des outils facilitant l'exploration et l'évaluation approfondies du contenu sémantique d'une collection de documents. Deux mesures d'intrication et un graphe de termes associés à la collection sont introduits pour permettre une exploration interactive récursive des données. L'étude de cas (section 3) montre l'apport du graphe d'interaction, dont la topologie exprime certaines caractéristiques sémantiques de la collection analysée. Les mesures d'intensité et d'homogénéité d'intrication permettent des analyses plus complètes qu'une simple proximité sémantique et offrent à l'utilisateur une nouvelle appréciation sur la sémantique de la collection ou d'une sous collection sélectionnée. Ces mesures nous ont permis de distinguer 4 profils génériques d'intrication sémantiques. L'étude utilisateur confirme l'intérêt de ces approches pour une étude de contenu détaillée d'une collection de documents.

Les tests ont été effectués sur des échantillons de petite taille, limités à quelques centaines de documents. Néanmoins, une analyse humaine approfondie ne peut s'exercer sur des collections de milliers de documents à la fois. Cet outil trouve sa place en complément de procédure d'agrégation à grande échelle, dans des contextes de qualité d'accès aux collections spécifiques, comme les bibliothèques ou les centres documentaires.

Nos travaux futurs porterons sur l'intégration de notre prototype dans un contexte d'usage réel afin de mesurer les retours utilisateurs dans une situation d'exploitation réaliste. Les mesures d'intrication étant génériques, elles peuvent s'appliquer à tous types de réseau mais avec une interprétation spécifique. Nous effectuons actuellement des tests sur des données issues des échanges collaboratifs.

Références

Arun, R. et al (2010). On finding the natural number of topics with latent dirichlet allocation : Some observations advances in knowledge discovery and data mining. Volume 6118 of

Lecture Notes in Computer Science, pp. 391–402. Springer.

Beaza-Yates, R. et B. Ribeiro-Neto (1999). *Modern Information Retrieval*. ACM Press.

Blei, D., A. Ng, et M. Jordan (2003). Latent dirichlet allocation. *Journal of Machine Learning Research 3*(5), 993–1022.

Brin, S. et L. Page (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems 30*(1-7), 107–117.

Burt, R. et T. Scott (1985). Relation content in multiple networks. *Social Science Research 14*, 287–308.

Croft, B., D. Metzler, et T. Strohman (2009). *Search Engines : IR in Practice*. Addison Wesley.

Deerwester, S., S. Dumais, G. W. Furnas, T. K. Landauer, et R. Harshman (1990). Indexing by latent semantic analysis. *Journal of the Society for Information Science 41*(6), 391–407.

Dhillon, I. S. (2001). Co-clustering documents and words using bipartite spectral graph partitioning. In *7th ACM SIGKDD*, San Francisco, California, pp. 269–274. ACM.

Ding, J. et A. Zhou (2009). *Nonnegative Matrices, Positive Operators and Applications*. Singapore : World Scientific.

Dumais, S. T. et al (1988). Using latent semantic analysis to improve access to textual information. In *SIGCHI*, pp. 281–285. ACM.

Kohonen, T. et al. (2000). Self organization of a massive document collection. *Neural Networks, IEEE Transactions on 11*(3), 574 –585.

Salton, G. (1983). *Automatic Text Processing : The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley.

Slonim, N. et N. Tishby (2000). Document clustering using word clusters via the information bottleneck method. In *23rd ACM SIGIR*, SIGIR '00, pp. 208–215. ACM.

Thomas, H. (1999). Probabilistic latent semantic indexing. In *22nd ACM SIGIR*, Berkeley, California, United States, pp. 50–57. ACM.

Xu, W., X. Liu, et Y. Gong (2003). Document clustering based on non-negative matrix factorization. In *26th Annual International ACM SIGIR*, SIGIR '03, pp. 267–273. ACM.

Zhao, Y., G. Karypis, et U. Fayyad (2005). Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery 10*, 141–168.

Summary

With the booming growth of online resources, efficiently exploring document collections has become a daily task for most users, and yet still remains a challenge for research. In this article, we present three tools to help users perceive the semantic content of a set of documents : a weightened interaction network of terms associated with documents, as well as intensity and homogeneity measures of semantic intrication reflecting the main features of term distribution within a set of documents. These measures can also help to identify different types of set of documents.

Carte auto-organisatrice pour graphes étiquetés

Nathalie Villa-Vialaneix*, Madalina Olteanu**
Christine Cierco-Ayrolles*

*Unité BIA, INRA de Toulouse, Auzeville, France
{nathalie.villa,christine.cierco}@toulouse.inra.fr,
http://www.nathalievilla.org
http://carlit.toulouse.inra.fr/wikiz/index.php/Christine_CIERCO-AYROLLES
**SAMM, Université Paris 1, Paris, France
madalina.olteanu@univ-paris1.fr
http://samm.univ-paris1.fr/OLTEANU-Madalina

Résumé. Dans de nombreux cas d'études concrets, l'analyse de données sur les graphes n'est pas limitée à la seule connaissance du graphe. Il est courant que des informations supplémentaires soient disponibles sur les sommets et que l'utilisateur souhaite combiner ces informations à la structure du graphe lui-même pour comprendre l'intégralité des données en sa possession. C'est ce problème que nous souhaitons aborder dans cet article, en nous focalisant sur une méthode de fouille de données qui combine classification (non supervisée) et visualisation : les cartes auto-organisatrices. Nous expliquons comment l'utilisation de méthodes à noyaux permet de combiner de manière efficace des informations de natures diverses (graphe, variables numériques, facteurs, variables textuelles...) pour décortiquer la structure des données et en offrir une représentation simplifiée. Notre approche est illustrée sur divers exemples : un premier exemple, sur des données simulées, permet de comprendre comment se comporte l'algorithme. Un second exemple illustre la méthode sur un graphe réel de plusieurs centaines de sommets, qui modélise un corpus de documents médiévaux.

1 Introduction

Dans de nombreuses applications dans lesquelles les données sont modélisées par un graphe (ou réseau), il est courant que des informations additionnelles (qui n'ont pas de relation directe avec la structure relationnelle du réseau) soient disponibles. Ces informations peuvent qualifier les sommets ou les arêtes du réseau ; dans le premier cas, on parle de « graphes étiquetés ». Les étiquettes peuvent être multiples et de natures diverses (numériques, catégorielles, textuelles). Les analyses statistiques qui visent à aider l'utilisateur à comprendre ses données doivent alors permettre de tenir compte de l'intégralité de l'information disponible et pas seulement de la structure du graphe. La question peut être abordée sous plusieurs angles : en étudiant la corrélation entre distance dans le réseau et similarité entre étiquettes (comme dans le cas des études sur l'homophilie dans les réseaux sociaux : voir Adamic et al. (2003); Crandall et al.

(2008); Aiello et al. (2010); Cointet et Roth (2010)), en faisant appel à des modèles de diffusion pour modéliser une dynamique dans les étiquettes du réseau (voir Valente et al. (1997); Newman (2002); Christakis et Fowler (2007)) ou bien en utilisant des outils issus de la statistique spatiale pour trouver et quantifier des phénomènes d'auto-corrélation dans le réseau (voir Laurent et Villa-Vialaneix (2011)).

Une méthodologie standard pour la fouille de graphe est la recherche de communautés : celle-ci consiste à partitionner les sommets du graphe en groupes de sommets denses qui partagent (comparativement) peu de liens entre eux. Deux articles de revue Fortunato (2010); Schaeffer (2007) présentent les principales approches développées dans ce domaine. Ici, nous présentons une approche pour détecter des communautés telles que non seulement les sommets d'une même communauté soient fortement inter-connectés mais aussi aient des étiquettes similaires. Cruz et al. (2011) aborde ce problème en utilisant une approche en deux temps, recherchant d'abord des communautés de manière classique par optimisation de la modularité, puis rafinant ces communautés en optimisant une enthropie basée sur la description des sommets. Dans cet article, nous proposons une approche en un temps, basée l'algorithme de carte auto-organisatrice Kohonen (2001) : celui-ci permet, en effet, de combiner classification non supervisée et visualisation en projetant les individus étudiés (ici les sommets du graphe) dans des neurones organisés topologiquement sur une grille de faible dimension (généralement égale à 2). La version initiale de l'algorithme est destinée à analyser des individus décrits par des variables numériques mais diverses variantes ont été proposées pour étendre son utilisation à des données décrites par des variables catégorielles (Cottrell et Letrémy, 2005) ou plus généralement à des données décrites par une dissimilarité (Kohohen et Somervuo, 1998; El Golli et al., 2006; Rossi et al., 2007; Hammer et al., 2011; Olteanu et al., 2012) ou par un noyau (Mac Donald et Fyfe, 2000; Villa et Rossi, 2007; Boulet et al., 2008).

Dans cet article, nous étendons l'algorithme SOM à noyau stochastique (décrit dans Mac Donald et Fyfe (2000); Villa et Rossi (2007)) en introduisant une approche multi-noyaux permettant l'intégration d'informations diverses dans la carte produite. La méthodologie proposée est décrite dans la section 2. Elle est ensuite illustrée sur deux exemples, un exemple simulé, utilisé pour montrer comment la méthode se comporte en présence d'informations contradictoires sur les données et un exemple réel issu d'un graphe décrivant des relations entre individus à partir d'un corpus d'actes notariés médiévaux.

2 Une carte auto-organisatrice pour graphe étiqueté

Dans la suite, nous supposerons donné un graphe $\mathcal{G}$ avec n sommets $\{1,\dots,n\}$, simple et pondéré par des poids $(W_{ij})_{i,j=1,\dots,n}$ ($W_{ij} \geq 0$ et $W_{ij} = W_{ji}$). En outre, chaque sommet i est décrit par D « étiquettes » (variables) $(c_i^d)_{d=1,\dots,D}$.

2.1 Noyaux

Les relations entre sommets et les similarités entre étiquettes sont décrites au moyen d'autant de noyaux. Un noyau K sur l'espace abstrait $\mathcal{X}$ est une application de $\mathcal{X} \times \mathcal{X}$ dans $\mathbb{R}$, symétrique ($K(x,x') = K(x',x)$) et positive ($\forall\, i = 1,\dots,N \sum_{kk'} \alpha_k \alpha_{k'} K(x_k,x_{k'}) \geq 0$). Aronszajn (1950) montre qu'une telle application est un produit scalaire d'une projection ϕ des données de $\mathcal{X}$ dans un espace de Hilbert $(\mathcal{H},\langle .,. \rangle)$: $K(x,x') = \langle \phi(x),\phi(x') \rangle$. L'intérêt

croissant autour de ce type de similarités vient du fait qu'une fois le noyau choisi, ni ϕ, ni $\mathcal{H}$ n'ont besoin d'être explicites pour pouvoir calculer des distances entre individus.

Dans le cas d'un graphe, plusieurs noyaux décrivant les similarités entre sommets peuvent être utilisés. Les plus populaires sont des versions régularisées du Laplacien L du graphe (voir Smola et Kondor (2003)), comme le noyau de la chaleur $e^{-\beta L}$ (*heat kernel*, Kondor et Lafferty (2002)) ou bien le noyau de temps de parcours (*commute time kernel*, Fouss et al. (2007)), qui n'est autre que l'inverse généralisée du Laplacien du graphe et s'interprète comme la mesure du temps moyen nécessaire pour relier deux sommets du graphe par une marche aléatoire sur les arêtes.

Pour décrire les similarités entre étiquettes des sommets, plusieurs choix sont possibles, dépendant de la nature des données :

- si les étiquettes $(c_i^d)_i$ sont numériques ($\in \mathbb{R}^M$), le noyau le plus simple est le noyau linéaire : $K_d(c_i^d,c_{i'}^d) = (c_{i'}^d)^T c_i^d$. D'autres noyaux permettent d'appliquer une transformation non linéaire sur les données, permettant de capter des corrélations plus complexes que la corrélation linéaire comme, par exemple, $K_d(c_i^d,c_{i'}^d) = e^{-\beta\|c_i^d - c_{i'}^d\|^2}$ (noyau Gaussien) ou bien $K_d(c_i^d,c_{i'}^d) = (1 + (c_{i'}^d)^T c_i^d)^P$ (noyau polynomial de degré P) ;
- si les étiquettes sont des variables catégorielles, une approche courante est de recourir au codage disjonctif de ces variables (c'est-à-dire à leur recodage par modalité en 0/1) et d'utiliser un noyau pour variable numérique. Notons que, dans le cas où le noyau linéaire est utilisé, cette opération conduit à utiliser comme noyau entre deux individus, le nombre d'attributs communs aux deux individus ;
- si les étiquettes sont des mots ou plus généralement du texte, plusieurs noyaux ont été proposés, tous basés sur le nombre d'occurrences de sous-parties communes aux deux textes comparés (voir Watkins (2000)). Une implémentation de ces noyaux est proposée dans le package **kernlab** du logiciel libre R (voir R Development Core Team (2012); Karatzoglou et Feinerer (2010)).

Le noyau final retenu pour mesurer la similarité globale entre les sommets i et i' est alors

$$K_T(i,i') = \alpha_0 K_0(i,i') + \sum_d \alpha_d K_d(c_i^d,c_{i'}^d)$$

où K_0 est le noyau choisi pour mesurer la similarité induite par la structure du graphe et les $(\alpha_d)_{d=0,\dots,D}$ sont des réels positifs tels que $\sum_d \alpha_d = 1$. Il est facile de montrer qu'un tel noyau satisfait aux conditions de l'article Aronszajn (1950).

2.2 Carte auto-organisatrice multi-noyaux

Une fois que les diverses informations sur les sommets du graphe ont été combinées pour calculer une mesure de proximité globale entre ces sommets, nous utilisons le noyau résultat K_T comme produit scalaire dans l'algorithme de carte auto-organisatrice pour plonger les sommets du graphe dans une carte de faible dimension. De manière plus précise, la version stochastique de l'algorithme de carte auto-organisatrice à noyau, comme décrite dans Mac Donald et Fyfe (2000); Villa et Rossi (2007), est utilisée pour positionner sur la carte les sommets du graphe. L'organisation des sommets sur la carte tient alors compte, à la fois, de la structure du graphe mais aussi des proximités entre les diverses étiquettes.

De manière plus précise, les sommets du graphe $\mathcal{G}$, $\{1,\dots,n\}$ sont projetés sur une carte de faible dimension composée de M neurones, $\{U_1,\dots,U_M\}$. Une relation de voisinage, h, est

définie entre les neurones et évolue au cours de l'algorithme de manière à converger progressivement vers un voisinage restreint au neurone lui-même. Chaque neurone U_j est représenté par un prototype p_j qui prends ses valeurs dans l'espace image, $\mathcal{H}$, induit implicitement par le noyau K_T. Si ϕ est la projection implicite induite par K_T dans $\mathcal{H}$, les prototypes sont définis comme $p_j = \sum_{i=1}^{n} \gamma_{ji}\phi(i)$, tels que $\gamma_{ji} \geq 0$ et $\sum_{i=1}^{n} \gamma_{ji} = 1$. Les $(\gamma_{ji})_{i,j}$ peuvent être initialisés de manière aléatoire.

L'algorithme alterne alors, de manière itérative,

– sélection aléatoire d'un sommet i et affectation dans le neurone pour lequel le prototype est le plus proche au sens de la distance dans $\mathcal{H}$:

$$f(i) \leftarrow \arg\min_j \|\phi(i) - p_j\|_{\mathcal{H}}, \tag{1}$$

– mise à jour des prototypes :

$$\gamma_{ji'} \leftarrow \gamma_{ji'} + \mu^t h^t(f(i),j)\,(\delta_{i'i} - \gamma_{ji'})$$

où $\delta_{i'i} = 1$ si $i = i'$ et 0 sinon. Le paramètre μ^t est adaptatif, généralement décroissant en $1/t$.

L'équation (1) est résolue en remarquant que la norme dans $\mathcal{H}$ est déduite du produit scalaire K. La minimisation s'effectue donc sur les quantités $\sum_{kk'} \gamma_{jk}\gamma_{jk'}K(k,k') - 2\sum_i \gamma_{jk}K(i,k)$. L'algorithme est stoppé à stabilisation de la classification, qui intervient après que la relation de voisinage ait été réduite au seul neurone : dans cette phase finale, l'algorithme est similaire à un algorithme k-means et sa convergence est donc assurée. Une dernière étape affecte tous les sommets du graphe à un neurone selon le critère de l'équation (1).

3 Applications

3.1 Données simulées

Dans cette partie, nous présentons un exemple simple sur des données simulées pour comprendre comment l'algorithme se comporte en présence de données de natures diverses. Pour cela, nous avons généré aléatoirement un jeu de données de 150 observations réparties en 6 groupes de 25 observations chacun, de la manière suivante :

– des relations entre les 150 observations ont été simulées par un graphe simple non pondéré qui ressemble au modèle aléatoire « planted 3-partition » décrit dans Condon et Karp (2001). Les sommets des groupes 1 et 2, des groupes 3 et 4 et des groupes 5 et 6 sont indicernables du point de vue de ce graphe : les arêtes entre les sommets de ces ensembles sont générées de manière aléatoire avec une probabilité égale à $0{,}3$ (modèle de Erdös Reyni, Erdõs et Rényi (1959)). Les arêtes entre les sommets d'ensembles distincts sont générées selon le même modèle aléatoire mais avec une probabilité moindre : $0{,}01$ (entre les sommets des groupes 1 ou 2 et les sommets des groupes 3 ou 4 et entre les sommets des groupes 3 ou 4 et les sommets des groupes 5 ou 6) ou $0{,}005$ (entre les sommets des groupes 1 ou 2 et les sommets des groupes 5 ou 6).
Les graphes simulés sont représentés dans la figure 1 (à gauche).

– des données numériques entre les 150 observations ont été simulées selon deux gaussiennes à deux dimensions : les groupes 1, 3 et 5 et les groupes 2, 4 et 6 sont indistinguables du point de vue des valeurs numériques prises. De manière plus précise, pour

chacun des deux ensembles, des variables aléatoires de distribution Gaussienne, centrée respectivement en $(0,0)$ et en $(1,1)$ et de matrice de covariance $\sigma^2\mathbb{I}_2$ (ou $\mathbb{I}_2$ est la matrice identité) avec $\sigma = 0{,}3$.
Les données numériques sont représentées dans la figure 1 (à droite).

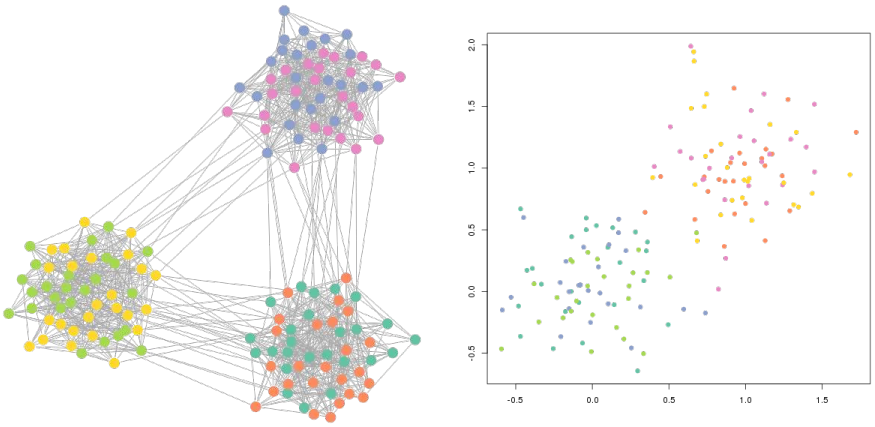


FIG. 1 – *À gauche : graphes aléatoires générés par des modèles de graphes en classes et représentés par un algorithme de force comme décrit dans Fruchterman et Reingold (1991) et implémenté dans le package* R **igraph** *; Csardi et Nepusz (2006)). À droite : distribution des variables numériques. Dans les deux graphiques, les 6 groupes de 25 sommets sont représentés par des couleurs différentes (groupe 1 : vert, groupe 2 : jaune, groupe 3 : bleu/vert, groupe 4 : rouge, groupe 5 : bleu et groupe 6 : rose)*

L'algorithme de cartes auto-organisatrices à noyau a été appliqué pour produire une topologie des données en utilisant

- pour le graphe, le noyau de temps de parcours ;
- pour les données numériques, un noyau Gaussien, dont le paramètre a été calibré de manière automatique selon la méthode décrite dans Caputo et al. (2002).

Trois résultats ont été produits : dans le premier, les deux noyaux ont été combinés (avec $\alpha_1 = \alpha_2 = \frac{1}{2}$) et dans les deux autres, chacun des deux noyaux a été utilisé seul. La carte obtenue est donnée dans la figure 2. Chaque diagramme circulaire représente un neurone de la carte. La taille du diagramme est proportionnelle au nombre d'observations classées dans ce neurone et les couleurs représentent la répartition des six classes initiales dans le neurone. Les arêtes reliant les diagrammes sont de largeur proportionnelle au nombre total d'arêtes reliant les sommets classés dans les deux classes respectivement. Les cartes basées sur une combinaison des noyaux sont les seules à retrouver les 6 classes et à proposer une organisation de celles-ci qui corresponde à l'organisation des données initiales, selon le graphe et les données numériques sous-jacentes.

La même expérience a été répétée 100 fois sur des générations aléatoires de données correspondant à un modèle un peu plus complexe (dans lesquels les groupes de sommets sont moins facilement identifiables) : dans celui-ci, les arêtes d'ensembles distincts sont générées avec une probabilité plus forte, respectivement égale à $0{,}1$ et $0{,}05$ et les écarts types des distri-

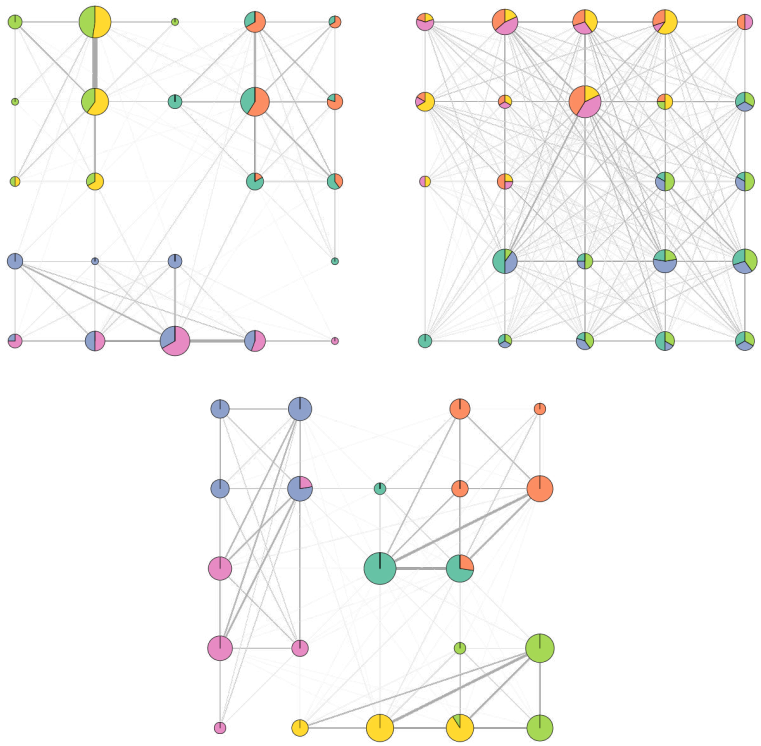


FIG. 2 – *Cartes obtenues à partir du premier jeu de données : en haut à gauche avec le graphe seulement, en haut à droite avec les données numériques seulement et en bas avec la combinaison des deux données. Voir le texte pour une description plus détaillée des graphiques.*

butions Gaussiennes ont été augmentées à $\sigma = 0{,}6$. Enfin, l'information mutuelle normalisée, (voir Danon et al. (2005)) entre les classes sous-jacentes et les classes retenues par l'algorithme a été calculée : cette mesure permettant de quantifier l'adéquation entre deux partitions, est comprise entre 0 et 1 et vaut 1 lorsque les deux partitions sont identiques. Les résultats correspondant à l'utilisation du graphe seul, des données numériques et des deux types de données sont fournies dans la boîte à moustaches de la figure 3. L'information mutuelle normalisée obtenue par le modèle combinant les deux noyaux est largement améliorée par rapport à l'information mutuelle utilisant une seule des deux informations : la combinaison des deux noyaux permet donc bien d'incorporer les informations des deux provenances de manière cohérente.

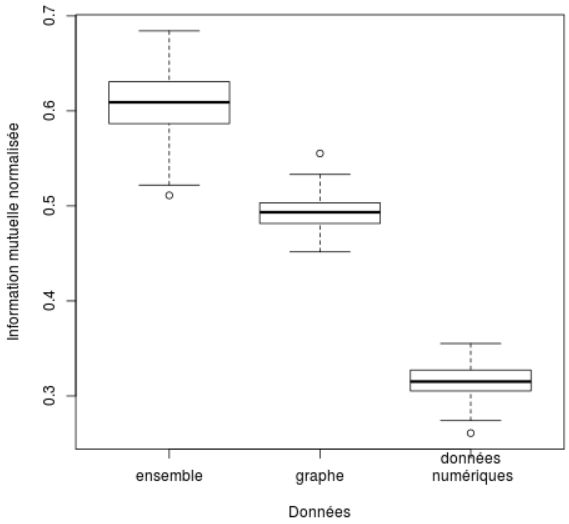


FIG. 3 – *Boîte à moustaches des informations mutuelles normalisées par rapport à la classification de référence sur 100 réalisations aléatoires des données pour diverses cartes auto-organisatrices utilisant l'un ou l'autre des noyaux (sur le graphe ou les données numériques) ou bien une combinaison des deux noyaux (« ensemble »).*

3.2 Données issues d'un corpus d'archives médiévales

Le graphe étudié dans cet exemple est issu d'un corpus d'actes notariés médiévaux décrit dans Boulet et al. (2008); Rossi et Villa-Vialaneix (2011); Hautefeuille et Jouve (2012). Ce corpus est le travail original d'un feudiste qui a été employé pour collecter et retranscrire tous les actes notariés mentionnant des rentes qui auraient été rédigés sur les seigneurie de "Castelnau Montratier" entre 1250 et 1700 (approximativement). Ce travail était destiné au nouveau propriétaire de la seigneurie, pour lui permettre de lever les loyers qui devaient lui revenir. Les documents sont donc tous de nature assez similaire et limité à une aire géographique étroite (d'environ 300 km^2).

Chacun des actes du corpus contient une ou plusieurs trasactions qui ont été digitalisés dans une base de données consultable librement à `http://graphcomp.univ-tlse2.fr`. Les transactions elles-mêmes contiennent des informations variées, avec des degrés de précision divers selon les transactions. En général, sont au moins mentionnés les noms des participants et la date de la transaction (au moins l'année). Un graphe a été tiré de ces informations dans lequel

- les sommets modélisent les individus directement impliqués dans les transactions (les notaires et les confronts parfois mentionnés ne sont donc pas inclus). Le graphe contient 1 446 individus et 3 192 arêtes.
- deux sommets sont reliés par une arête si les deux individus ont été impliqués dans une transaction commune ;
- les sommets sont étiquetés par la date moyenne d'activité de l'individu (variable numérique) et par le nom de famille de l'individu (variable textuelle).

L'utilisation de ces trois informations permet donc de regrouper des individus de la même famille, ayant des relations sociales similaires et une activité à des dates proches. Les noyaux suivants ont été combinés avec un poids identique ($\alpha_d = 1/3$ pour $d = 1,..,3$) :

- noyau de temps de parcours pour le graphe ;
- noyau linéaire pour les dates ;
- noyau spectral pour les distances entre noms de famille (voir Karatzoglou et Feinerer (2010), ici nous avons fixé le paramètre de taille des séquences communes comptées à 4).

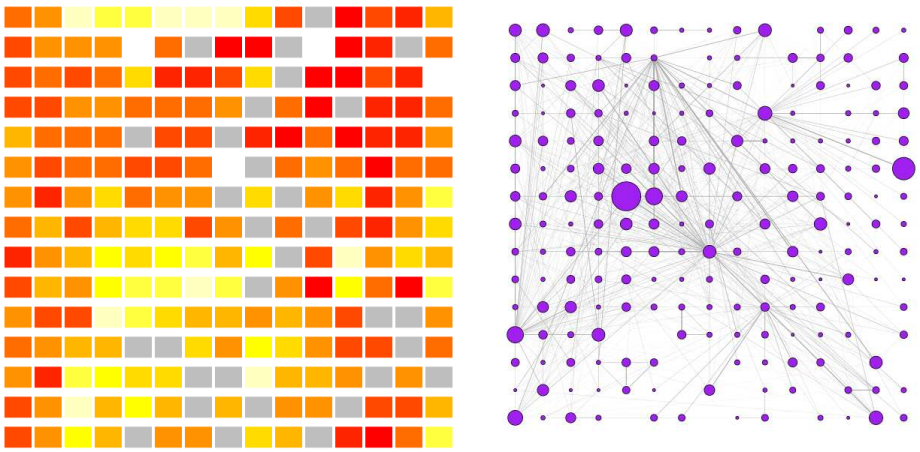


FIG. 4 – *Carte des dates moyennes par neurones (à gauche : les dates les plus récentes sont en jaunes, les plus anciennes en rouge) et carte du réseau (à droite : les aires des sommets sont proportionnelles au nombre d'individus classés dans le neurone considéré et les épaisseurs des arêtes sont proportionnelles au nombre total de liens entre les individus des deux neurones divisé par la racine carré du produit des deux effectifs des neurones pour éviter de sur-représenter des arêtes entre sommets de fort effectif).*

Les résultats sont présentés dans les figures 4 et 5. On y observe une bonne organisation des noms de famille, avec un regroupement spatialisé des noms de famille comme par exemple, la

FIG. 5 – *Carte des principaux noms de famille (au maximum deux par neurone) et effectif d'apparition du nom dans le neurone.*

famille Combelcau (en bas à droite), proche des familles Combe et Lacombe qui sont similaires au niveau des noms (dont l'orthographe dérive assez facilement à cette époque) ou bien la famille Bosseran en haut au centre ou la famille Estairac, aussi en haut au centre. De manière répétée dans la carte, le neurone contient des individus de même nom, avec des occurences fréquemment supérieures à 5. La carte présente également une bonne organisation selon les dates. Certains neurones proches correspondent à des dates assez différentes : en bas à droite, par exemple, les dates des neurones sont différentes mais les noms de famille similaires. La topologie reflète donc une similarité entre individus (même famille) et les différents clusters permettent de séparer les individus selon leur période d'activité. Du point de vue du réseau,

la carte présente aussi une bonne organisation, avec des neurones plus fortement connectés au centre de la carte et des neurones plus isolés sur les bords. Les communautés extraites semblent cohérentes du point de vue des trois données d'entrée, fournissant à l'utilisateur historien, un moyen de se focaliser sur des communautés locales, homogènes du point de vue des dates, des relations et des familles impliquées.

4 Conclusion

Dans cet article, nous avons présenté une approche permettant d'obtenir des communautés dans un graphe en tenant compte d'informations additionnelles connues sur les sommets. L'utilisation d'une combinaison de noyaux semble être une méthodologie pertinente qui arrive à tirer profit de l'ensemble des données disponibles. Dans les applications présentées, la combinaison linéaire a été réalisée au moyen de poids identiques, chaque noyau étant donc considéré avec la même importance. Une extension de ce travail, soumis pour publication (Olteanu et al., 2013), permet l'apprentissage adaptatif des poids de la combinaison des noyaux pour optimiser la classification en donnant plus ou moins d'importance à certains types d'informations.

Références

Adamic, L., O. Buyukkokten, et E. Adar (2003). A social network caught in the web. *First Monday 8*.

Aiello, L., A. Barrat, C. Cattuto, G. Ruffo, et R. Schifanella (2010). Link creation and profile alignment in the aNobii social network. In *Proceedings of the Second IEEE International Conference on Social Computing (SocialCom)*, Minneapolis, USA.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society 68*(3), 337–404.

Boulet, R., B. Jouve, F. Rossi, et N. Villa (2008). Batch kernel SOM and related laplacian methods for social network analysis. *Neurocomputing 71*(7-9), 1257–1273.

Caputo, B., K. Sim, F. Furesjo, et A. Smola (2002). Appearance-based object recognition using svms : which kernel should i use ? In *Proceedings of NIPS workshop on Statistical methods for computational experiments in visual processing and computer vision*, Whistler.

Christakis, N. et J. Fowler (2007). The spread of obesity in a large social network over 32 years. *New England Journal of Medicine 357*, 370–379.

Cointet, J. et C. Roth (2010). Local networks, local topics : Structural and semantic proximity in blogspace. In *Proceedings of the International Conference on Weblogs and Socail Media (ICWSM)*, Washington, DC, USA.

Condon, A. et R. Karp (2001). Algorithms for graph partitioning on the planted partition model. *Random Structures Algorithms 18*(2), 116–140.

Cottrell, M. et P. Letrémy (2005). How to use the Kohonen algorithm to simultaneously analyse individuals in a survey. *Neurocomputing 63*, 193–207.

Crandall, D., D. Cosley, D. Huttenlocher, J. Kleinberg, et S. Suri (2008). Feedback effects between similarity and social influence in online communities. In *Proceedings of the 14th SIGKDD*, pp. 160–168.

Cruz, J., C. Bothorel, et F. Poulet (2011). Entropy based community detection in augmented social networks," , 2011 international conference, pp.163-168 doi : 10.1109/cason.2011.6085937. In *Proceedings of Computational Aspects of Social Networks (CASoN)*, pp. 163–168.

Csardi, G. et T. Nepusz (2006). The igraph software package for complex network research. *InterJournal Complex Systems*.

Danon, L., A. Diaz-Guilera, J. Duch, et A. Arenas (2005). Comparing community structure identification. *Journal of Statistical Mechanics*, P09008.

El Golli, A., F. Rossi, B. Conan-Guez, et Y. Lechevallier (2006). Une adaptation des cartes auto-organisatrices pour des données décrites par un tableau de dissimilarités. *Revue de Statistique Appliquée LIV*(3), 33–64.

Erdõs, P. et A. Rényi (1959). On random graphs. i. *Publicationes Mathematicae 6*, 290–297.

Fortunato, S. (2010). Community detection in graphs. *Physics Reports 486*, 75–174.

Fouss, F., A. Pirotte, J. Renders, et M. Saerens (2007). Random-walk computation of similarities between nodes of a graph, with application to collaborative recommendation. *IEEE Trans Knowl Data En 19*(3), 355–369.

Fruchterman, T. et B. Reingold (1991). Graph drawing by force-directed placement. *Software Pract Exper 21*, 1129–1164.

Hammer, B., A. Gisbrecht, A. Hasenfuss, B. Mokbel, F. Schleif, et X. Zhu (2011). Topographic mapping of dissimilarity data. In *Proceedings of WSOM 2011*, pp. 1–15.

Hautefeuille, F. et B. Jouve (2012). Defining rural elites in the 13th to 15th centuries at the crossroads of historical, archeological and mathematical approaches. Submitted to the Journal of Interdisciplinary History.

Karatzoglou, A. et I. Feinerer (2010). Kernel-based machine learning for fast text mining in R. *Computational Statistics and Data Analysis 54*, 290–297.

Kohohen, T. et P. Somervuo (1998). Self-organizing maps of symbol strings. *Neurocomputing 21*, 19–30.

Kohonen, T. (2001). *Self-Organizing Maps, 3rd Edition*, Volume 30. Berlin, Heidelberg, New York : Springer.

Kondor, R. et J. Lafferty (2002). Diffusion kernels on graphs and other discrete structures. In *Proceedings of the 19th International Conference on Machine Learning*, pp. 315–322.

Laurent, T. et N. Villa-Vialaneix (2011). Using spatial indexes for labeled network analysis. *Information, Interaction, Intelligence (i3) 11*(1).

Mac Donald, D. et C. Fyfe (2000). The kernel self organising map. In *Proceedings of 4th International Conference on knowledge-based intelligence engineering systems and applied technologies*, pp. 317–320.

Newman, M. (2002). Spread of epidemic disease on networks. *Phys Rev E 66*(016128).

Olteanu, M., N. Villa-Vialaneix, et C. Cierco-Ayrolles (2013). Multiple kernel self-organizing maps. Submitted.

Olteanu, M., N. Villa-Vialaneix, et M. Cottrell (2012). On-line relational som for dissimilarity data. In P. Estevez, J. Principe, P. Zegers, et G. Barreto (Eds.), *Advances in Self-Organizing Maps (Proceedings of WSOM 2012)*, Volume 198 of *AISC (Advances in Intelligent Systems and Computing)*, Santiago, Chile, pp. 13–22. Springer Verlag, Berlin, Heidelberg.

R Development Core Team (2012). *R : A Language and Environment for Statistical Computing*. Vienna, Austria. ISBN 3-900051-07-0.

Rossi, F., A. Hasenfuss, et B. Hammer (2007). Accelerating relational clustering algorithms with sparse prototype representation. In *6th International Workshop on Self-Organizing Maps (WSOM)*, Bielefield, Germany. Neuroinformatics Group, Bielefield University.

Rossi, F. et N. Villa-Vialaneix (2011). Représentation d'un grand réseau à partir d'une classification hiérarchique de ses sommets. *Journal de la Société Française de Statistique 152*(3), 34–65.

Schaeffer, S. (2007). Graph clustering. *Computer Science Review 1*(1), 27–64.

Smola, A. et R. Kondor (2003). Kernels and regularization on graphs. In M. Warmuth et B. Schölkopf (Eds.), *Proceedings of the Conference on Learning Theory (COLT) and Kernel Workshop*, Lecture Notes in Computer Science, pp. 144–158.

Valente, T., S. Watkins, M. Jato, A. van der Straten, et L. Tsitsol (1997). Social network associations with contraceptive use among comeroonian women in voluntary associations. *Social Science & Medecine 45*, 677–687.

Villa, N. et F. Rossi (2007). A comparison between dissimilarity SOM and kernel SOM for clustering the vertices of a graph. In *6th International Workshop on Self-Organizing Maps (WSOM)*, Bielefield, Germany. Neuroinformatics Group, Bielefield University.

Watkins, C. (2000). Dynamic alignment kernels. In A. Smola, P. Bartlett, B. Schölkopf, et D. Schuurmans (Eds.), *Advances in Large Margin Classifiers*, Cambridge, MA, USA, pp. 39–50. MIT P.

Summary

In a number of real-life applications, the user is interested in analyzing the graph together with additional information known on its nodes. The combination of all the sources of information can help him to better understand the dataset in its whole. The present article focus on such an issue, by using self-organizing maps, that combine clustering and visualization. Using a kernel version of the algorithm makes it possible to combine various types of information (graph, numerical values, factors, strings...). Several simplified representations of the data can then be derived from the obtained map. The approach is illustrated on two examples: the first one is simulated data that seeks at proving the usefulness of the method. The second example is a real-life application where the graph, that contains several hundred nodes, has been extracted from a corpus of medieval documents.

Primates: Un Système de gestion de la confidentialité dans les réseaux sociaux

Imen BEN DHIA*, Talel ABDESSALEM*, Mauro SOZIO*

*46, Rue Barrault, 75013 Paris,
name.lastname@telecom-patistech.fr,
http://www.telecom-paristech.fr

Résumé. While online social networks (OSN) present unprecedented opportunities for sharing information and multimedia content among users, they raise major privacy issues as users could often access personal or confidential data of other users. Most social networks provide some basic access control policies, which however seem to be very limited given the diversity of user relationships in the current social networks (e.g. friend, acquaintance, son) as well as the needs of social network users who might want to express sophisticated access control policies (e.g. "*invite all children of my colleagues to my child's birthday party*"). In this paper, we present `Primates` a privacy management system for social networks. `Primates` allows users to specify access control rules for their resources and enforces access control over all shared resources. The set of users who are allowed to access a given resource is defined by a set of constraints on the paths connecting the owner of a resource to its requester in the social graph. We demonstrate the accuracy of our access control model and the scalability of our system.

Keywords: Access control, Design, Privacy, Online social networks

1 Introduction

In the last few years, we have been witnessing an explosion of online social networks (OSNs) such as Facebook, LinkedIn, and Twitter which nowadays account hundred million users across the globe using them on a daily basis. At the time this paper is written, Facebook reports over 901 million active users [1], while Twitter has 140 million active users [2].

In an OSN, each user can easily share information and multimedia content (e.g., personal data, photos, videos, contacts, etc.) with other users in the network, as well as organize different kind of events (e.g., business, entertainment, religion, dating etc.). While this presents unprecedented opportunities, it also gives raise to major privacy issues, as users could often access personal or confidential data of other users. Most social networks provide some basic access control policies, e.g., a user can specify whether a piece of information shall be publicly

1. `http://www.facebook.com/press/info.php?statistics`
2. `https://business.twitter.com/basics/what-is-twitter/`

available, private (no one can see it) or accessible to friends only. However, the set of relationships represented in an OSN nowadays is quite rich and diverse, with relative relationships as well as the possibility of distinguishing between acquaintances and close friends becoming increasingly common ; as a result there is an increasing need to providing more sophisticated access control policies. For instance, one would like to say "*invite all children of my colleagues to our child's birthday party*" or "*show this picture of myself wearing a funny costume to my friends and the friends of my friends while not to my colleagues*".

In this paper, we present `Primates` a privacy management system for social networks implementing the access control model we proposed in Abdessalem et Dhia (2011). Our model can be described as follows. We represent a social network as a labeled directed graph where nodes represent users and edges represent social relationships between them. Labels on the edges specify the type of relationships between users, such as 'friends', 'acquaintances', 'son', etc. Each edge can also be associated with a weight measuring the degree of trust between two given users.

Privacy preferences are expressed in our system by a set of access rules, each one being associated with a given resource (i.e, a shared resource) and specifies, through a reachability constraint, the set of users who can access such a resource. Each reachability constraint is represented by a path expression over the social graph, that combines constraints on labels, edge directions, label order, and/or distance between nodes. For instance, one would like to express the constraint that *Alice* can access the content published by *Bill* only if *Bill* is a friend of a friend of *Alice*, while he is not a colleague of *Alice*. In the social network graph, this corresponds to verify whether there is a path of length at most two between *Alice* and *Bill* whose edges are labeled 'friend' and there is no edge labeled 'colleagues' between the two users. We also allow resource owners to grant access to other users who are trusted "enough". Trust between two users is measured by the weight associated to the edge connecting them, if any, or it can be computed using trust propagation models Liu et al. (2008); Guha et al. (2004) if the there is no such an edge. We do not discuss trust propagation models any further, as this is out of the scope of this paper.

Determining whether a requester should be granted access is done at query time, that is when a requester tries to access a shared resource. Our objective in this paper, is to demonstrate the accuracy of `Primates` as well as its scalability.

The remainder of the paper is organized as follows. In the rest of this section, we discuss some related work. In Section 2, we present the access control model used in `Primates`. In Section 3, we describe the way we enforced this access control model. We, then, describe the high-level architecture of our system in Section 4, and, detail our demonstration scenario.

Related Work Previous work on access control in social networks can be classified into two main categories : (i) *machine learning-based* approaches as in Fang et LeFevre (2010); Shehab et al. (2010), which try to automatically configure user privacy settings, based on available explicit access authorizations and the underlying graph structure (i.e., communities), and, (ii) *rule-based* approaches as in Carminati et al. (2009), which introduced trust and distance in the social graph as the key criteria for access rules. Our work can be classified as a rule-based approach. It generalizes access constraints by taking into account the properties of the users, the paths connecting them, and allows expressing complex relationships (i.e, sequence of direct relationships of different types).

2 Access control model

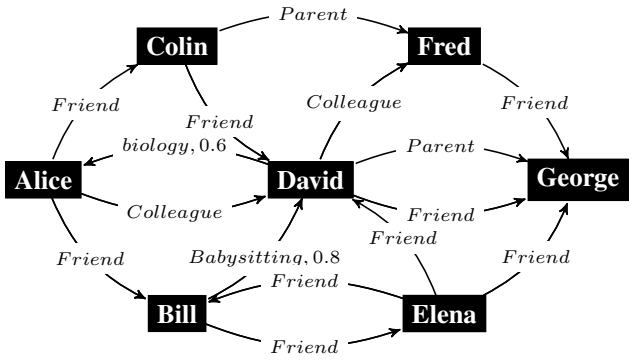


FIG. 1 – *An Online Social Network Subgraph*

In our model, an OSN is represented as a labeled directed graph, where nodes represent users and edges denote social relationships between users. User properties (age, gender, etc.) are expressed as attributes of the graph nodes.

Privacy preferences are expressed by a set of access rules, each one being associated with a given resource (i.e, a shared resource) and specifies, through a reachability constraint, the set of users who can access such a resource. Each reachability constraint is represented as a path expression over the OSN graph.

Online Social Network (OSN) We formally represent an OSN as a directed graph $G = (V, E)$, where V is a set of nodes denoting social network users and E is a set of directed edges representing social relationships between them. A labeling function $l : E \rightarrow 2^{\Sigma}$ specifies the set of labels (where Σ is a set of labels such as friend, colleague, etc.) associated to each edge, while a function $t : E \rightarrow [0, 1]$ measures trust between users. In case a trust value is not specified, a default value (e.g, 0.5) can be used. Each node is associated with a set of pairs (attr,value) specifying a set of attributes and their values for the corresponding user.

Access Rules An access rule expresses a set of constraints that should be met in order to access a given resource. Formally, an access rule is defined as a tuple (u, r, P, C) where u denotes the owner of a resource r, C is a set of constraints on the attributes of the requester (such as location='Paris' or trust ≥ 0.8) and $P = p_1, \ldots, p_k$ expresses a set of constraints on the paths connecting the requester to the owner of the resource ; each p_j is defined as a triple $p_j = (l, dir, I)$ where l is a label in Σ, $I = (min, max)$ is a pair of integers specifying the minimum and maximum length of p_j and $dir \in \{\leftarrow, \rightarrow, \Leftrightarrow\}$ indicates the direction of p_j. Given a requester v for resource r, an access rule (u, r, P, C) is satisfied if all constraints C are satisfied and there is a path between u and v satisfying P ; v is granted access to r if there is an access rule at least that is satisfied.

For instance $P = ('friend', \rightarrow, (1,2)), ('colleague', \rightarrow, (1,1))$ expresses the constraint that there must be a path between the owner and a requester that first traverses a path of length in $(1,2)$ whose edges are labeled 'friend' and then an edge labeled 'colleague'.

Suppose that *Elena* wants to make her baby-sitting advertisement (identified as *ad*) accessible to her direct friends. She can specify an access rule as follows :

$$AR_1 = (Elena, ad, ('friend', \rightarrow, (1,1)), -)$$

If *Elena* wants to extend access to her indirect friends (i.e, the friends of her friends) the path that should be specified would be $('friend', \rightarrow, (1,2))$. Again, if she wants to extend it to the users that consider her as a friend, she should specify an other access rule which is the following :

$$AR_2 = (Elena, ad, ('friend', \leftarrow, (1,2)), -)$$

If *Elena* wants to change the access rules associated to her baby-sitting advertisement and make it accessible to the trustworthy (trust threshold = 0.8) baby-sitters of her friends within 2 hops then she should specify the following access rule :

$$AR_3 = (Elena, ad, \{('friend', \rightarrow, (1,2)), \\ ('babysitter', \rightarrow, (1,1))\}, [trust = 0.8])$$

Finally, the authorization can be limited to the baby-sitters living in Paris by adding an additional condition to the access rule as follows :

$$AR_4 = (Elena, ad, \{('friend', \rightarrow, (1,2)), \\ ('babysitter', \rightarrow, (1,1))\}, [location = Paris])$$

3 Access Control Enforcement

The access control enforcement mechanism is performed by the *reference monitor*, which is a trusted software module that intercepts each access request submitted by a requester to access a resource, and, based on the specified access policy, determines whether access should be granted or denied to the requester. Suppose that a user *v* is requesting for a resource *r*. When *v* submits his/her access request to access the resource *r*, the system retrieves the set of access rules associated to that resource. If there are no access rules related to the requested resource, then, the system will apply the *default access rule* which is defined by the user and applied whenever there are no access rules associated to the requested resource. This prevents the access control strategy from being too loose (by setting resources having no associated rules to public) or too restrictive (by setting resources having no associated rules to private). Then, the system evaluates the set of retrieved access rules and stops, either when the requester satisfies one of these rules, or when all the rules were evaluated and the requester satisfies no rule. In the former case, the requester is authorized to access the resource that he asked for. In the latter case, the requester is denied access because his profile is not consistent with the target audience.

Reachability paths evaluation The enforcement of an access rule consists in evaluating the path P that is associated to it. The problem of evaluating these paths boils into a reachability problem in graph databases, which is well-known in the database community. Evaluating a reachability query, in our case, consists in determining whether two nodes u and v (for instance, the owner and the requester) in the graph are connected through a path with constraints on labels and distance. We devised an algorithm which is an adapted version of the breadth-first-search (BFS) algorithm applied to the graph together with the specified constraints to reduce the search space. The trust computation process between u and v is done at the same time, when the graph is explored. However, since the focus in `Primates` is on reachability, we implemented a simple trust propagation function and consider transitivity as the only way of propagation. The inferred trust value between two nodes u and v, connected through a path p, is computed by multiplying the explicit trust values associated with each edge in p.

4 System Description

4.1 Global architecture

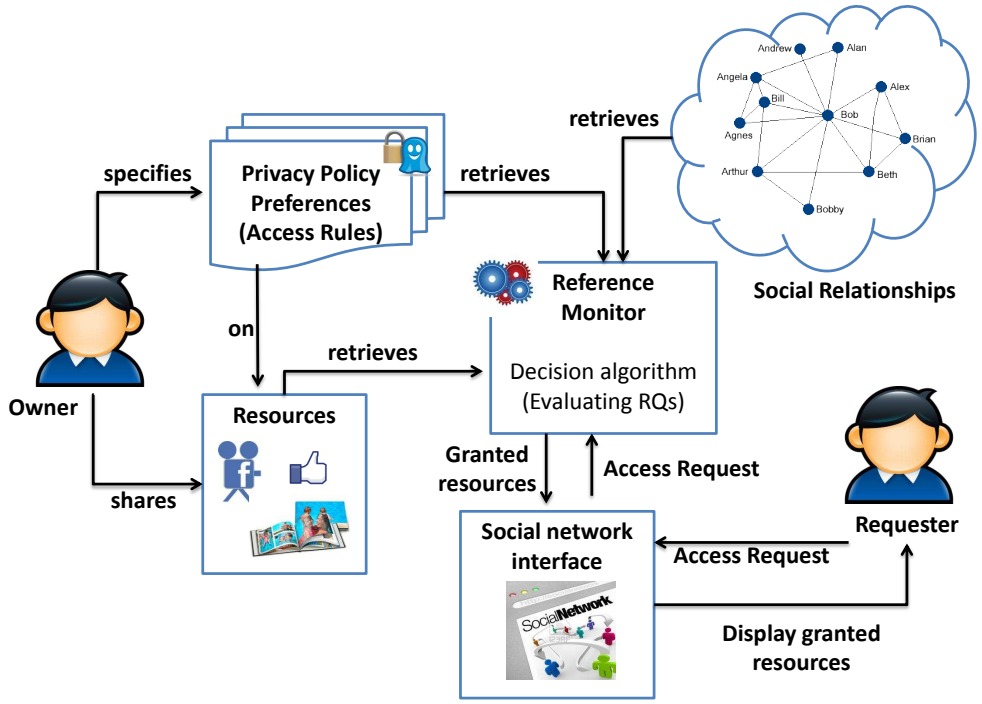


FIG. 2 – *System Architecture*

The global architecture of our system is depicted in Figure 2. Each user can share multiple resources. Resources are information (photos, videos, comments, etc.) to which access may need to be controlled. A user (the owner) can express his privacy policy preferences for his

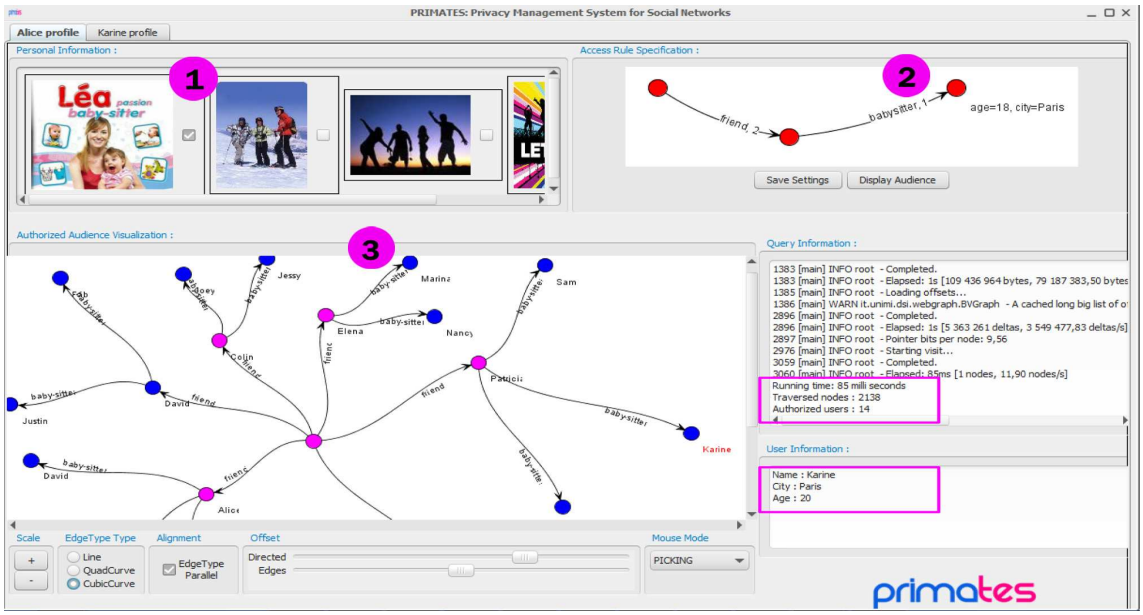


FIG. 3 – *User interface for specifying access rules*

resources by specifying one or several access rules to them. The requester, also called subject, is a user who is trying to access some resources of another user (the owner). The subject can send a request to access resources via the social network interface. This request is sent to the reference monitor, which is the component that implements user privacy preferences. It takes as input the access request and based on the access rules that are associated to the requested resource and social connections in the social netwok graph, it will authorize rendering only the resources for which the requester is part of its authorized profiles. We run our system on a compressed version of a snapshot of the *liveJournal* graph consisting of 5M nodes and 80M edges. The graph was compressed using the *WebGraph* framework Boldi et al. (2011) which provides algorithms for accessing compressed graphs without any decompressing. The compressed version of the *liveJournal* graph fits into main memory and can be accessed efficiently. User relationships have only a single type, which is *Friend*. For this reason, we artificially introduced other relationship types and associated them to edges on the *liveJournal* graph. The introduction of these types was done with respect to natural characteristics of human relations (e.g., a person can have on average 3 children and at most 2 close friends, etc.). We associated access rules to randomly selected user information. The response time is less than 1 sec (on average).

4.2 Demonstration Scenario

The demonstration shows two contributions of our work. The first one is the design of an access control model that allows users to specify more sophisticated privacy policies which fit their privacy needs. The second one is the enforcement of this model in such a way that allows

users to intuitively specify their privacy settings and efficiently visualize the set of users that are allowed to get access to their information.

As shown in Figure 3, using `Primates`, visitors can select some information and express the desired privacy settings for it either from a set of access rules predefined by the system or by expressing new access rules in a user-friendly graphical way. They can express constraints on edge label and distance, and, node properties. Provinding users with a graphical interface to specify access rules allows them to express their privacy settings in a simple and intuitive way rather than expressing it in terms of paths as they are defined in the model. A parser converts then user actions into path patterns that would be evaluated as reachability queries.

Once access rules are specified, visitors will have the possibility to browse and visualize the graph of authorized users according to the specified privacy preferences. They can then navigate this graph and see explicitly who are the authorized users and can eventually refine access rules based on that. Displayed users on the authorized audience graph can be clicked on to to display their profile information such as the name, age, city, etc.

Visitors can also see how each user in the social network sees only information to which he is authorized to get access. They can see the profile (all shared personal information) of a user as an information owner on one side and try to get access to that profile from the perspective of another user on the other side. They will clearly see that not all the information in the owner's profile is displayed, but only a subset of that information to which he/she is allowed to access. An example scenario run on `Primates` can be found in the following url : `http://perso.telecom-paristech.fr/~ibendhia/demo.html`.

5 Conclusion

In this paper, we developed an access control model for OSNs that enables a fine-grained description of privacy policies. These policies are specified in terms of constraints on the type, direction, depth of relationships and trust levels between users, as well as on the users properties. This model relies on a reachability-based approach, where a subject requesting to access an object must satisfy policies determined by the object owner.

Références

Abdessalem, T. et I. B. Dhia (2011). A reachability-based access control model for online social networks. In *Proc. of the 1st ACM SIGMOD Workshop on Databases and Social Networks (DBSocial)*, pp. 31–36.

Boldi, P., M. Rosa, M. Santini, et S. Vigna (2011). Layered label propagation : A multiresolution coordinate-free ordering for compressing social networks. In *Proceedings of the 20th international conference onWorld Wide Web*.

Carminati, B., E. Ferrari, et A. Perego (2009). Enforcing access control in web-based social networks. *ACM Trans. Inf. Syst. Secur.*, 6 :1–6 :38.

Fang, L. et K. LeFevre (2010). Privacy wizards for social networking sites. WWW, pp. 351–360.

Guha, R., R. Kumar, P. Raghavan, et A. Tomkins (2004). Propagation of trust and distrust. WWW, pp. 403–412.

Liu, H., E.-P. Lim, H. W. Lauw, M.-T. Le, A. Sun, J. Srivastava, et Y. A. Kim (2008). Predicting trusts among users of online communities : an epinions case study. In *EC*, pp. 310–319.

Shehab, M., G. Cheek, H. Touati, A. C. Squicciarini, et P.-C. Cheng (2010). Learning based access control in online social networks. WWW, pp. 1179–1180.

Summary

De nos jours, les réseaux sociaux jouent un rôle important dans le partage d'informations et de contenus multimédia. Cependant, ils ne sont pas dotés de mécanismes appropriés pour permettre aux utilisateurs de contrôler de façon flexible et efficace l'accès à leurs données. La plupart des réseaux sociaux n'offrent que des solutions basiques, qui ne tiennent pas compte de la nature des liens entre les utilisateurs et la diversité des besoins de confidentialité des utilisateurs. Dans cet article, nous présentons le système `Primates` pour la gestion de la confidentialité dans les réseaux sociaux. `Primates` permet aux utilisateurs de définir des règles de contrôle d'accès pour leurs ressources et applique ces règles d'accès aux ressources partagées. Les utilisateurs qui sont autorisés à accéder à une ressource donnée sont spécifiés à l'aide d'un ensemble de contraintes décrivant le chemin (path) qui les relie dans le graphe social au propriétaire de la ressource. Nous montrons dans cet article la pertinence de notre modèle de contrôle d'accès et les performances de notre système.

Partenaires :