Visualisation de graphes avec Tulip : exploration interactive de grandes masses de données en appui à la fouille de données et à l'extraction de connaissances.

David Auber*, Yves Chiricota **
Maylis Delest *
Jean-Philippe Domenger *
Patrick Mary *
Guy Melançon***

*LaBRI UMR 5800
Université Bordeaux I
351 Cours de la Libération
33405 Talence Cedex – France
{auber,maylis,domenger,mary}@labri.fr
www.labri.fr
**Département de mathématiques et d'informatique
Université du Québec à Chicoutimi
555, boulevard de l'Université
Chicoutimi, G7H 2B1 – Canada
Yves.Chiricota@uqac.ca
wwwdim.uqac.ca
***INRIA Futurs & LIRMM UMR 5506
161 rue Ada
34392 Montpellier Cedex 5 – France
Guy.Melancon@lirmm.fr
www.inria.fr – www.lirmm.fr

Résumé. Cet article décrit une étude de cas exhibant les qualités de la plateforme de visualisation de graphes Tulip, démontrant l'apport de la visualisation à la fouille de données interactive et à l'extraction de connaissances. Le calcul dŠun graphe à partir d'indices de similarité est un exemple typique où l'exploration visuelle et interactive de graphes vient en appui au travail de fouille de données. Nous penchons sur le cas où l'on souhaite étudier une collection de documents afin d'avoir une idée des thématiques abordées dans la collection.

1 Introduction

La plate-forme de visualisation de graphes Tulip[1] (Auber, 2003) développée au LaBRI est dédiée à l'exploration de grands graphes. Elle autorise un utilisateur expert à visualiser un graphe à partir d'algorithmes de dessin parmi les plus récents. Elle facilite le calcul et le rendu visuel de statistiques sur les graphes afin d'en rechercher les propriétés structurelles. Conçue pour la manipulation et le calcul du clustering de grands graphes, l'interface utilisateur permet de fouiller l'information ainsi découpée en inspectant les clusters et leurs éléments, tout particulièrement lorsque le clustering produit une hiérarchie de sous-graphes.

De fait, Tulip se prête à la fouille interactive de données munies de relations binaires et enrichies d'attributs, numériques ou textuels. Le calcul d'un graphe à partir d'indices de similarité est un exemple typique où l'exploration interactive vient en appui au travail de fouille de données et d'extraction de connaissances. On peut penser au cas où des documents sont liés deux à deux par un indice de similarité (de leur contenu). En seuillant cet indice, on peut induire sur l'ensemble des documents une structure de graphes qu'il est alors utile d'examiner.

Nous proposons dans cet article de décrire une étude de cas exhibant les qualités de la plate-forme Tulip et démontrant l'apport de la visualisation à la fouille de données et à l'extraction de connaissances.

2 Exploration visuelle et fouille de données

Nous nous pencherons sur le cas où l'on souhaite étudier une collection de documents afin d'avoir une idée des thématiques abordées dans la collection de manière globale, pour savoir si un sujet rassemble une majorité de documents, ou si au contraire il ne concerne qu'une toute petite part d'entre eux, par exemple. Typiquement, on extraira des documents un ensemble de mots-clés qui capturent leur contenu à divers degré. Le plus souvent, la présence de mots-clés donnent lieu au calcul d'indices de similarité entre documents : deux documents seront d'autant plus similaires qu'ils ont des mots-clés en commun et que ceux-ci y apparaissent souvent. (Nombre de variantes existent ; voir (Hammouda et Kamel, 2004) par exemple, ou (Dubois et Bothorel, 2006) pour une approche récente intégrant les usages.) A l'inverse, on peut associer les mots-clés deux à deux afin de *voir* émerger certains mots au rang de concept (Figure 1). Les réseaux de co-citation, mêlant auteurs et publications scientifiques, sont un autre exemple de graphes qui se prêtent à l'examen visuel afin de définir une stratégie d'analyse des données sous-jacentes.

En d'autres mots, partant d'une collection de N documents, on se ramène à l'étude d'un graphe dont les sommets représentent les documents. La mesure de similarité est interprétée comme la donnée des arêtes qui sont alors valuées (voire orientées). Lorsque le graphe ainsi obtenu est complet ou lorsque son nombre d'arêtes avoisine $\binom{N}{2}$ (ou $N(N-1)$ dans le cas d'un graphe orienté, ou N^2 si on admet les boucles ou *auto-référence*), on peut filtrer certaines des arêtes pour alléger le graphe, en espérant ne retenir que les relations les plus marquées, c'est-à-dire les plus à même de rendre la structure inhérente à la collection de documents. Cette idée souvent empruntée remonte à (Tenenbaum et al., 2000) et s'avère utile, malgré quelques limitations attribuables à la difficulté du problème de la réduction de la dimension d'un jeu de

[1] Le chapitre de l'ouvrage cité ici renvoie aux premières versions de Tulip, qui en est maintenant à la version 3.0.0. Voir le site `www.tulip-software.org` pour plus d'informations et pour télécharger la version la plus récente.

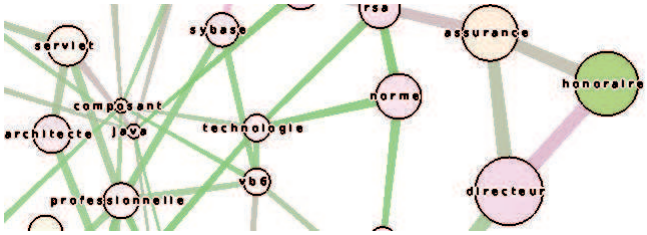

FIG. 1 – *Vue partielle d'un réseau sémantique (mots-clés extraits de documents) induits d'indices de similarités.*

données (dimensionality reduction) – voir (Balasubramanian et Schwartz, 2002). La Figure 1 illustre ce procédé ; dans cet exemple, les mots-clés sont associés selon leurs indices de co-occurences dans la collection de documents.

Toutefois, l'utilisateur peut rapidement être dépassé par le nombre d'éléments du graphe (sommets et arêtes) à étudier, et par la complexité du réseau qu'ils forment. C'est ici que la visualisation interactive entre en jeu : l'utilisateur pourra interagir sur la carte et acquérir une compréhension du réseau à travers sa navigation, plutôt que d'en rester à une carte statique. L'analyste peut s'appuyer sur une visualisation de l'ensemble des documents et l'explorer interactivement afin de percevoir des motifs structuraux dans l'organisation des liens. Soulignons que la visualisation n'est, la plupart du temps, pas une fin en soi mais permet de définir une stratégie d'analyse du corpus. A l'inverse, la visualisation évoluera en proportion de la compréhension que gagne l'analyste sur cet ensemble de données complexes. Ce scénario a été testé en taille réelle sur un jeu de données diffusé sur le web (Delest et al., 2004)[2] ; il a aussi été repris plus récemment dans le contexte bio-informatique (Iragne et al., 2005)[3] et pour l'exploration de grands réseaux spatiaux en géographie (Amiel et al., 2005)[4] – voir Figure 2. Il s'agit là, d'une certaine manière, d'un scénario très souvent mis en oeuvre avec Tulip pour venir en appui à la fouille et à l'exploration interactive de données complexes[5].

3 Visualisation de graphes

La plate-forme Tulip prend le parti d'offrir en priorité à l'utilisateur un grand choix d'algorithmes de dessin de graphes (Graph Drawing[6]), c'est-à-dire de représentations des graphes dans le plan ou dans l'espace à l'aide de sommets (des points) et d'arêtes (traits rectilignes).

[2]Voir l'URL `www.cs.umd.edu/hcil/iv04contest/`. Le jeu de données comprend environ 600 articles de la conférence InfoVis rassemblant 1600 auteurs et faisant référence à près de 5000 autres articles référencés dans la ACM Digital Library. Les données comprenaient, outre les références, titre des articles et noms des auteurs, les résumés et/ou les mots-clés donnant lieu au calcul de différents indices de similarités.

[3]Le nombre de protéines de la levure, souvent étudiée, est estimé à 6 000 et leur nombre d'interactions à 30 000. Outre leur taille, les graphes d'interactions de la protéomique ont une structure qui les rend difficile à analyser.

[4]Les données ITA pour l'année civile 2000 font état d'environ 16500 liaisons aériennes entre un peu plus d'un milliers de villes.

[5]L'article (Melançon, 2006) fait état de nombre de cas réels qui entrent tous dans le cadre de la fouille interactive et visuelle de données.

[6]Voir l'URL `www.graphdrawing.org`

C'est en quelque sorte une spécialisation de Tulip à un type précis d'abstractions visuelles ou de transformations visuelles, pour emprunter les termes de la taxonomie de Chi (Chi, 2000). Ainsi, on peut facilement faire appel aux algorithmes de dessin d'arborescences [(Reingold et Tilford, 1981) (Eades, 1992) (Grivet et al., 2004)], de graphes orientés sans circuit (DAG), de graphes planaires et de graphes plus généraux à l'aide des algorithmes basés sur les analogies physiques (souvent appelés *masses-ressort*) [(Fruchterman et Reingold, 1991) (Fricke et al., 1994)] ou s'appuyant sur l'analyse spectrale [(Koren et Harel, 2002) (Harel et Koren, 2002)].

Cela dit, l'interface, le format de description et les fonctionnalités internes à Tulip permettent à l'utilisateur de représenter les sommets par différentes objets graphiques simples. En particulier, Tulip autorise l'utilisateur à calculer, outre les positions des sommets dans le plan, divers attributs numériques pour les sommets et/ou les arêtes. La visualisation de ces attributs est alors naturellement traduits sous forme d'indices visuels comme la couleur ou la taille des sommets. Cette idée très simple a fait ses preuves pour aider l'analyste dans l'exploration de grands graphes (voir (Herman et al., 2000) par exemple).

Comme on peut s'y attendre, l'utilisateur est à même d'affecter à chaque sommet une forme différente et paramétrable, une couleur qui varie en fonction d'un indice structurel ou d'un attribut contextuel. S'il est possible d'effectuer ce choix au moment de la visualisation, on peut tout aussi bien l'insérer au niveau du fichier de description, ou encore concevoir un *plug-in* spécifique qui s'intégrera facilement à la plate-forme (voir Section 5). La plate-forme offre déjà nombre d'algorithmes calculant des indices structuraux sur les graphes – indices de centralité (Freeman, 2000), indice de clustering (Watts et Strogatz, 1998), ... – et les indices plus habituels comme le degré entrant ou sortant, etc. D'autres algorithmes permettent de tester certaines propriétés comme l'acyclicité ou la connexité, et effectueront la sélection de composantes connexes, d'arbres couvrants ou de graphes acycliques couvrants, par exemple.

Ces dernières manipulations s'avèrent des plus pratiques pour sélectionner un sous-ensemble d'un jeu de données et le faire passer au rang de *sous-graphe*. Cet objet affiché dans une fenêtre qui lui est propre permettra de travailler à plus petite échelle, tout en héritant des propriétés calculées sur le graphe dont il est issu (voir Figure 5).

3.1 Cas d'étude

La Figure 1 montre une vue partielle d'un graphe associant des mots-clés extraits d'une collection de documents. Les co-occurrences de ces mots-clés donnent lieu au calcul d'un indice de similarité dont l'histogramme apparaît à la Figure 2. Le graphe formé de ces mots-clés et de toutes les associations comporte 352 sommets et 11443 liens, soit près de 33 fois plus d'arêtes que de sommets. C'est sans surprise que l'on constate qu'une majorité de mots-clés sont associés au travers d'un indice plus faible. La première opération consistera à filtrer les liens pour ne garder que le quart d'indices le plus élevés (au moins égal à $\sim 0,33$), réduisant le nombre de liens à 2725, soit un peu plus de 8 fois son nombre d'arêtes – le graphe demeure donc relativement dense malgré qu'on ait filtré une majorité d'arêtes[7]. L'objectif de cette opération est double : gagner en lisibilité et extraire du réseau un squelette formé des liens les plus forts.

L'interface permet d'associer différentes mesures à des attributs graphiques. L'association de variables quantitatives à différentes variables visuelles est une idée qui prend sa source en

[7] Voir (Melançon, 2006) pour une discussion sur la densité attendue des graphes dans divers domaines d'application.

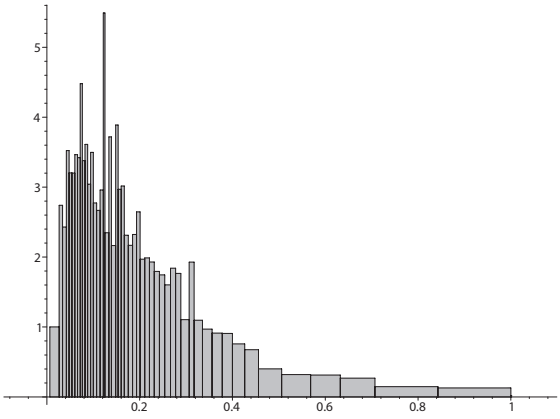

FIG. 2 – *Histogramme des indices de similarités (normalisés).*

cartographie (voir (Bertin, 1998), (Denègre, 2005)). Dans la Figure 1, la taille d'un sommet correspond à son degré dans le graphe. Le degré donne en effet une première indication du rôle que joue un sommet dans le réseau, au moins dans son voisinage immédiat. La couleur du sommet est, elle, associée à la connexité de son voisinage (selon l'indice de clustering de Watts (Watts et Strogatz, 1998)). Le choix des couleurs peut lui aussi être varié. L'exploration interactive permet à l'utilisateur, après avoir repéré un phénomène saillant (voisinage dense, sommet de degré plus élevé, etc.) de circonscrire son exploration à une région particulière du graphe.

La Figure 3 illustre une manipulation typique, aisément exécutée avec Tulip. L'utilisateur aura sans aucun doute perçu la complexité du voisinage du sommet (concept) *administration*, et pourra alors sélectionner le sommet. Par déplacement, il peut ensuite isoler ce sommet en l'écartant de son voisinage. L'ensemble des liens suit et on est à même de percevoir comment est tissé cette région. Les arêtes du sommet sont mises en exergue par un effet de coloration pour signaler leur sélection, rendant à la fois l'effet de la manipulation et cette partie de la carte plus lisible et permettant à l'utilisateur de constater les liens avec les concepts *processus*, *surveillance*, ou *programme*, par exemple.

4 Clustering de graphes

Dans nombre de cas cependant, le dessin de graphes et le calcul d'attributs numériques et/ou d'indices visuels s'avèrent impuissants face au volume des données à visualiser et à explorer. L'utilisateur peut alors mettre en marche des algorithmes de clustering afin d'identifier dans les données des sous-groupes homogènes. Les structures de données internes à Tulip ont été spécifiquement conçues à cet effet. Un algorithme de clustering classique produit une partition ensembliste des sommets du graphe. Si on itère la procédure sur chacun des clusters, on obtient alors une arborescence encodant l'imbrication de sous-graphes. Tulip donne explicitement accès à cette hiérarchie de sous-graphes permettant à l'utilisateur, d'une part, d'accéder à

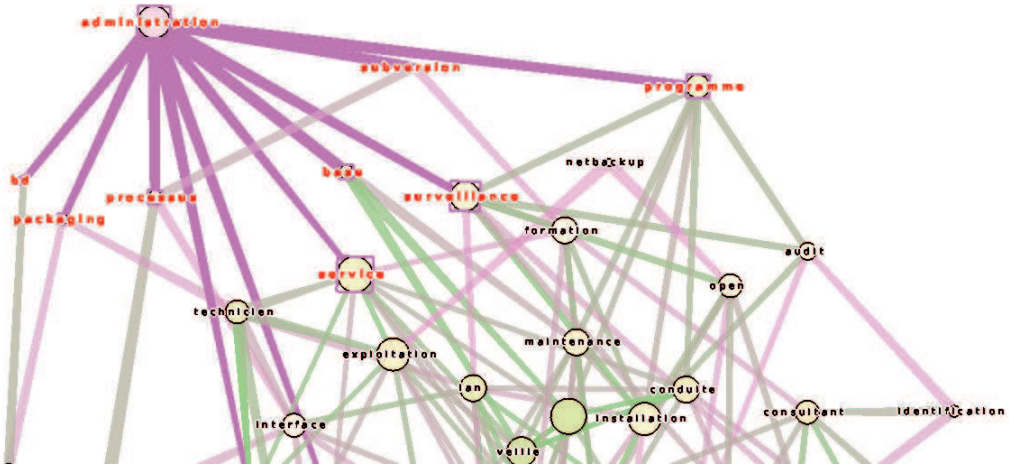

FIG. 3 – *Sélection et mise en exergue du voisinage d'un sommet ("administration").*

l'intérieur d'un sous-graphe pour examiner sa structure, d'autre part de visualiser la structure du graphe quotient : le graphe rendant compte des relations entre les sous-graphes.

La Figure 4 donne un exemple de graphe *quotient* illustrant les liens qu'entretiennent les clusters d'un graphe. L'algorithme utilisé (Auber et al., 2003a) permet d'isoler certaines composantes plus fortement connectées, tout en les organisant autour d'un noyau du graphe d'origine. Remarquez que le noyau (au centre de la Figure) est organisé de manière similaire et suit une topologie en étoile typique de la méthode utilisée ici. Cette image est elle-même une vue partielle de tout le réseau : nous avons en effet accédé au cluster illustré à la Figure 4 et à son organisation multi-niveaux en zoomant à l'intérieur de l'une des composantes du graphe de départ. Observez le cluster contenant le mot *administration* qui avait retenu notre attention à la Figure 3, qui se trouve sur la droite. Avec le mot *subversion* auquel il est lié de manière significative, il forme une paire indissociable dans le graphe quotient. Les autres voisins du mot *administration* sont passés au second ordre, et reste encapsulé dans le noyau.

On peut d'une certaine manière penser aux composantes placées en périphérie du noyau comme aux mots-clés formant avec certains de leurs voisins des composantes indissociables, mais qui se détachent suffisamment du noyau central. Typiquement, tout algorithme de clustering est susceptible de repérer un voisinage formant une clique – un sous-graphe complet, comme celui dans la partie gauche de la Figure 5. Au niveau le plus bas, on trouvera des composantes *irréductibles*, dans le sens où l'algorithme de clustering ne saura trouver de critère pour en extraire un sous-graphe. C'est évidemment le cas pour les cliques, mais aussi pour les graphes comme celui situé à droite dans la Figure 5, qui sans être une arborescence est principalement constitué de longues chaînes de mots sans voisinage touffu (sauf peut-être pour le voisinage du mot *java*). Incidemment, on peut chercher à suivre dans ce sous-graphe les chemins allant d'un mot à l'autre pour y trouver une proximité de sens et ainsi interpréter le contenu de la collection de documents.

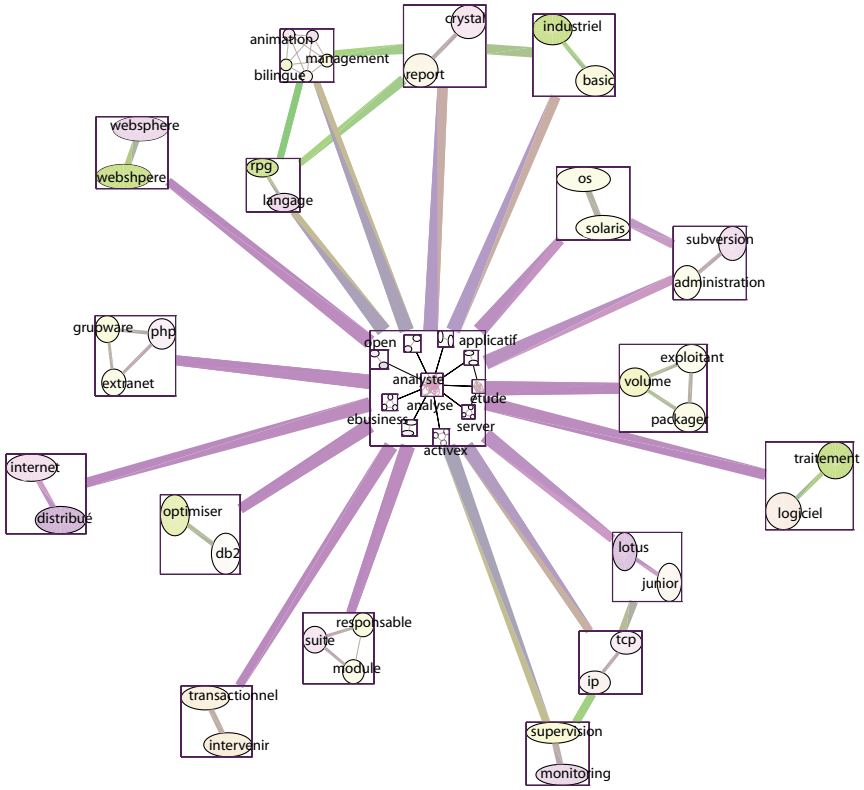

FIG. 4 – *Graphe quotient résultant du clustering des mots/concepts en sous-groupes.*

5 Conclusion

Plus qu'une suite logicielle, Tulip est une plate-forme de développement de laquelle peuvent être déclinées des applications spécifiques [EVAT (Auber et al., 2003b) pour la visualisation et la comparaison de grandes arborescence – arborescences de fichiers, classification des espèces, etc. – ProViz (Iragne et al., 2005) pour la visualisation de réseaux d'interaction de protéines, ARNA (Gainant et Auber, 2004) pour la visualisation et la comparaison de structures secondaires d'ARN ou encore SWViz (Auber et al., 2003a) pour la visualisation de grands réseaux sociaux].

L'architecture de la plate-forme facilite l'ajout de composants dédiés sous-forme de *plug-in* qui sont chargés dynamiquement. Cette caractéristique essentielle favorise grandement la diffusion et l'adoption de la plate-forme. Elle permet aussi à l'utilisateur d'ajouter ses propres algorithmes de calculs d'indices structuraux, ou encore d'indices propres aux données visualisées s'appuyant sur des attributs spécifiques (sémantiques ou structuraux). Le format de description de Tulip s'inspire volontairement d'un format texte très dénué afin d'assurer les meilleures performances au chargement des données ; cela dit, le format permet d'embarquer tous types d'attributs – textuels, numériques, textures pour enrichir les sommets (imagettes), etc. L'ajout

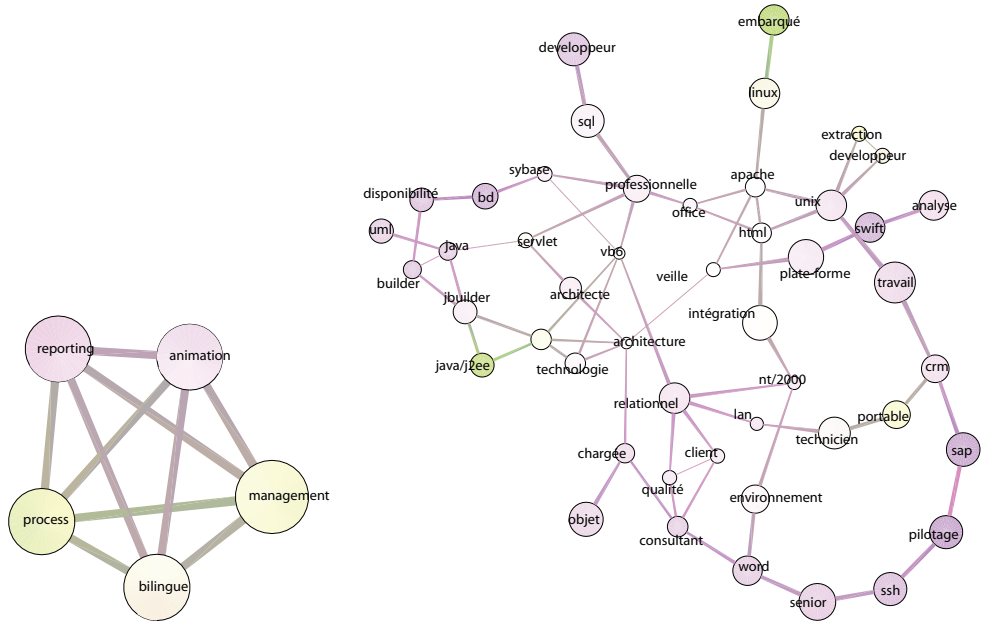

FIG. 5 – *(Gauche) Sous-graphe formant une clique et identifié comme tel lors du clustering. Ce sous-graphe a donc été extrait du graphe d'origine et placé en marge du noyau,* mais en relation avec ses clusters voisins *dans le graphe quotient (voir au haut de la Figure 4). (Droite) Sous-graphe se situant au niveau le plus bas, en ce sens que l'algorithme (Auber et al., 2003a) ne reconnaît aucune composante à extraire.*

d'algorithmes de clustering ou de dessin de graphes est tout aussi facile.

A l'inverse d'autres outils de visualisation comme IVTK (Fekete, 2004) ou *prefuse* (Heer et al., 2005), Tulip est développé en C++, permettant ainsi de manipuler des données massives. La déclinaison du cœur de Tulip en une application dédiée nécessite l'utilisation de la librairie QT[8], pour ce qui concerne l'environnement de fenêtrage et les modalités de l'interface graphique, auquel vient se greffer le noyau Tulip pour la gestion de la structure de graphes (hiérarchiques) et l'utilisation d'algorithmes spécifiques. Notons au passage que Tulip est inclus dans plusieurs distributions Linux (Suse, Debian, ...). Tulip est distribué sous licence GPL et est hébergé sur le site SourceForge[9].

Les fonctionnalités de la plate-forme Tulip constituent sans aucun doute une boîte à outils des plus utiles pour découvrir, concevoir et tester les hypothèses des chercheurs œuvrant dans le domaine de l'extraction et la gestion des connaissances. Idéalement, la visualisation sera couplée aux méthodes et techniques propre à ce domaine afin d'offrir aux utilisateurs une cartographie sémantique interactive pour fouiller les grands espaces d'information, structurés ou non, ou encore comme interface d'accès à l'information.

[8] Voir le site `www.trolltech.com`

[9] Voir aussi le site de Tulip, `www.tulip-software.org`

Références

Amiel, M., G. Melançon, et C. Rozenblat (2005). Réseaux multi-niveaux : l'exemple des échanges aériens mondiaux. *M@ppemonde 79*(3-2005).

Auber, D. (2003). Tulip - a huge graph visualization framework. In P. Mutzel et M. Jünger (Eds.), *Graph Drawing Software*, Mathematics and Visualization Series. Springer Verlag.

Auber, D., Y. Chiricota, F. Jourdan, et G. Melançon (2003a). Multiscale Navigation of Small World Networks. In *IEEE Symposium on Information Visualisation*, Seattle, GA, USA, pp. 75–81. IEEE Computer Science Press.

Auber, D., M. Delest, J.-P. Domenger, P. Ferraro, et R. Strandh (2003b). EVAT : an Environment for the Visualization and Analysis of Trees (2nd place - InfoVis Contest). In *IEEE Symposium on Information Visaulization*, pp. 124–126. IEEE Computer Society.

Balasubramanian, M. et E. L. Schwartz (2002). The Isomap Algorithm and Topological Stability. *Science 295*, p. 7.

Bertin, J. (1998). *Sémiologie Graphique : Les Diagrammes, Les Réseaux, Les Cartes*. Les ré-impressions. Ecole des Hautes Etudes en Sciences Sociales.

Chi, E. H. (2000). A Taxonomy of Visualization Techniques Using the Data State Reference Model. In *IEEE Symposium on Information Vizualization*, pp. 69. IEEE Computer Society.

Delest, M., T. Munzner, D. Auber, et J.-P. Domenger (2004). Exploring InfoVis Publication History with Tulip (2nd place - InfoVis Contest). In *IEEE Symposium on Information Visualization*, pp. 110. IEEE Computer Society.

Denègre, J. (2005). *Sémiologie et conception cartographique*. Hermes Science.

Dubois, V. et C. Bothorel (2006). From semantic to social : an integrated approach for content and usage analysis. In *Workshop on Semantic Network Analysis (co-located in ESWC 2006)*, Budva, Montenegro.

Eades, P. (1992). Drawing Free Trees. *Bulletin of the Institute for Combinatorics and its Applications 5*, 10–36.

Fekete, J.-D. (2004). The infovis toolkit. In *10th IEEE Symposium on Information Visualization*, Austin, TX, pp. 167–174. IEEE Press.

Freeman, L. C. (2000). Visualizing Social Networks. *Journal of Social Structures 1*(1).

Fricke, A., A. Ludwig, et H. Mehldau (1994). A Fast Adaptive Layout Algorithm for Undirected Graphs. In *Symposium on Graph Drawing GD '93*, Volume 894 of *Lecture Notes in Computer Science*, Berlin, pp. 389–403. Springer Verlag.

Fruchterman, T. et E. Reingold (1991). Graph Drawing by Force-Directed Placement. *Software – Practice & Experience 21*, 1129–1164.

Gainant, G. et D. Auber (2004). Arna : Interactive comparison and alignment of rna secondary structure. In *IEEE Symposition on Information Visualisation*, pp. 8. IEEE Computer Society.

Grivet, S., D. Auber, J.-P. Domenger, et G. Melançon (2004). Bubble Tree Drawing Algorithm. In R. S. Kozera, L. Noakes, H. Palus, W. Skarbek, B. Smolka, et K. Wojciechowski (Eds.), *ICCVG International Conference on Computer Vision and Graphics*, Volume 32 of *Computer Vision and Imaging*, Warsaw, Poland, pp. 633–641. Springer.

Hammouda, K. et M. Kamel (2004). Document Similarity Using a Phrase Indexing Graph Model. *Knowledge and Information Systems 6*(6), 710–727.

Harel, D. et Y. Koren (2002). Graph Drawing by High-Dimensional Embedding. In *International Symposium on Graph Drawing*, Volume 2528 of *Lecture Notes in Computer Science*, pp. 207–219. Springer-Verlag.

Heer, J., S. K. Card, et J. A. Landay (2005). prefuse : a toolkit for interactive information visualization. In *SIGCHI conference on Human factors in computing systems*, Portland, Oregon, USA, pp. 421–430. ACM Press.

Herman, I., M. S. Marshall, et G. Melançon (2000). Graph Visualisation and Navigation in Information Visualisation : A Survey. *IEEE Transactions on Visualization and Computer Graphics 6*(1), 24–43.

Iragne, F., M. Nikolski, B. Mathieu, D. Auber, et D. Sherman (2005). Proviz : protein interaction visualization and exploration. *Bioinformatics 21*(2), 272–274.

Koren, Y. et D. Harel (2002). ACE A Fast Multiscale Graph Algorithm. In *IEEE Symposium on Information Visualization*, Boston, USA. IEEE CS.

Melançon, G. (2006). Just how dense are dense graphs in the real world ? A methodological note. In E. Bertini, C. Plaisant, et G. Santucci (Eds.), *BELIV Workshop (AVI Conference)*, Venice, Italy, pp. 75–81. ACM Press.

Reingold, E. et J. Tilford (1981). Tidier Drawing of Trees. *IEEE Transactions on Software Engineering 7*(2), 223–228.

Tenenbaum, J. B., V. d. Silva, et J. C. Langford (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science 290*, 2319–2323.

Watts, D. et S. H. Strogatz (1998). Collective dynamics of "small-world" networks. *Nature 393*, 440–442.

Summary

The paper presents a case study describing features of the Graph Visualization Framework Tulip, assessing of its contribution to interactive data mining and knowledge extraction. The computation of a graph out of data equipped with similarity indices is a typical example where visual and interactive graph exploration supports data mining. We look at how topics emerging from a collection of documents can be interactively explored through graph visualization and navigation.