Graph decomposition approaches for terminology graphs

Mohamed Didi Biha[1], Bangaly Kaba[2], Marie-Jean Meurs[3], and Eric SanJuan[3] **

[1] LANLG, 33 rue Louis Pasteur, 84000 Avignon, France
mohamed.didi-biha@univ-avignon.fr
[2] LIMOS, Université Blaise Pascal Clermont 2, 63177 AUBIERE cedex, France
kaba@isima.fr
[3] LIA, Université d'Avignon, BP 1228 84911 Avignon, Cedex 9, France
{marie-jean.meurs,eric.sanjuan}@univ-avignon.fr

Abstract. We propose a graph-based decomposition methodology of a network of document features represented by a terminology graph. The graph is automatically extracted from raw data based on Natural Language Processing techniques implemented in the TermWatch system. These graphs are Small Worlds. Based on clique minimal separators and the associated graph of atoms: a subgraph without clique separator, we show that the terminology graph can be divided into a central kernel which is a single atom and a periphery made of small atoms. Moreover, the central kernel can be separated based on small optimal minimal separators.

Key words: graph algorithms, graph decomposition, polyhedral approach, text mining, topic vizualisation

1 Introduction

Terminology graphs that include explicitly defined properties and relationships developed for human-curated semantic networks, such as controlled ontologies, are used for organizing and communicating information. At the core of these terminologies are discrete elements of knowledge, or entities, which carry meaning. The way in which these entities are arranged and encoded in electronic format is a key concern in informatics [1].

The TermWatch system [2] aims to automatically extract a terminology graph from texts based on Natural Language Processing (NLP) approaches originally introduced in [3].

In this paper we show how these graphs can be structured in coherent subnetworks in order to allow its visualisation and to approximate a real concept network. For that we use two recent graph decomposition approaches. The first

** Corresponding author.

one is based on the concept of graph of atoms (an atom is a subgraph without clique separator). The important point is that this decomposition is unique. However, there is no upper limit to atom size and we have observed that terminology graphs are made of small atoms gravitating at the peripheral of a huge central one. It is necessary to break this central atom into equal parts without loosing its internal structure. We show that this can be efficiently accomplish using optimal separators.

The rest of the paper is organised as follows. In section 2 we recall the features of terminology graphs extracted using TermWatch system. In section 3, we formally define the process of graph decomposition into atoms. In section 4 we show how optimal separators can be found. In section 5 we experiment the whole process on a real corpus. Finally, we conclude on related work and perspectives.

2 TermWatch system

This system comprises three modules: a term extractor, a relation identifier which yields the terminological network and a visualisation module.

2.1 Term extraction

This module performs term extraction based on shallow NLP, using the LTPOS tagger and LTChunker[4]. LTChunker identifies simplex noun phrases (NPs), i.e., NPs without prepositional attachments. In order to extract more complex terms, we wrote contextual rules to identify complex terminological NPs, i.e. those with a prepositional attachment. The number of words in a term is not limited. This choice is based on the observation that most concepts in the technical domain are long multi-word terms.

2.2 Identifying semantic nearest neighbours (S-NN) of terms

This module identifies the different semantic variants of the same term based on surface and internal linguistic operations between MWTs and the use of an external resource, here WordNet.

Morphological variants are identified using the LTPOS tagger. Lexical variants are identified based on word changes in terms. However, the definition of lexical variants is restricted in order to allow the change of only one word in the same position so as to avoid generating spurious relations. The change can take place either in a modifier position (*T-cell line / fibroplast line*) or in the head position (*T-cell line / T-cell lymphoma*). The head in a noun phrase is the term focus (subject) while the modifier plays the role of a qualifier. Syntactic variants involve structural changes in terms, for instance a permutation: "*retrieval of information*" and "*information retrieval*". Other syntactic operations called expansions, involve the addition of modifier or head words in a term. Modifier

[4] (C) Andrei Mikheev 1996–2000 (C) LTG, University of Edinburgh 1996–2000.

expansions are subdivided into "Insertions" and "Left-expansions". Head expansions are either expansions solely on the head position of a term or double expansions (both in the modifier and head position). Semantic variants are used to identify more semantically bound terms amongst lexical substitutions as the latter can be noisy, especially on binary terms. For instance, a chain of lexical head substitutions can link all of these terms: *T-cell line, T-cell lineage, T-cell lymphoma, T-cell lysate, T-cell malignancy, T-cell maturation, T-cell mitogen, T-cell mitogenesis*. This can capture semantically close terms like "*T cell line*" and "*T cell lineage*" but this would be purely accidental. To identify semantic substitutions amongst the lexical ones, WordNet is used to filter those variants where the substituted words are in a WordNet relation.[5] We distinguish WordNet substitutions according to the two grammatical functions of the substituted word: head or modifier.

The variations described above can be further refined according to the number and position of inserted words for expansion variants. Thus we distinguish further between strong and weak expansions. Strong expansions are those variants where only one word is added (*B cell lymphoma line / human B cell lymphoma line*) while those involving the addition of more than one word are considered as weak expansion variants (*TSH receptor / TSH receptor (TSHR)-specific T cell line*).

2.3 Graph visualisation module

For visualization purposes, graphs are clustered. For this task, we use a variant of the single link clustering (SLC), called CPCL (Classification by Preferential Clustered Link) originally introduced in [3] to form clusters of keywords related by geodesic paths made of strength associations. The advantages of SLC clustering are that it produces a unique output and that it runs in linear time on the number of edges. The CPCL variant also has these properties. It merges iteratively clusters of keywords related by an association strongest than any other in the external neighborhood. In other words, CPCL works on local maximal edges instead of absolute maximal values like in standard SLC. CPCL output is unique such as in SLC while reducing the chain effect. We refer the reader to [4] for a detailed description in the graph formalism. The CPCL algorithm has been optimised to run in $O(|E|)$ time, where $|E|$ is the number of edges of the graph.

Finally, using the interactive interface AiSee (http://www.aisee.com) and its optimized bi-scale force directed layout, we obtain a two level access to the network of terms and clusters.

AiSee needs as input a file in Graph Description Language (GDL). Our GDL generator uses edge width to visualize the strength of the link. Clusters are then

[5] Despite the fact that general resources cannot capture the explicit conceptual relation between specialized domain terms, we still highly improved the precision of the substitutions variants using WordNet, in the sense that 97% of the WordNet substitutions linked semantically related terms.

represented by ovals whose size depends on the number of clustered vertices. Finally, special clusters can be unfolded in a wrapped form that allows to visualize the transitions to other clusters.

3 Atom Graph decomposition

3.1 Introduction to graph decomposition

We will recall some preliminary graph notions which will be helpful to follow our approach.

A graph is denoted $G = (V, E)$ where V is a finite set of vertices and E is a finite set of edges. The graphs on which we work are undirected. $G(A)$ is the subgraph induced by a vertex set A(included in V). A clique in a graph is a set of pairwise adjacent vertices. A connected component graph is a maximal vertex set wich induces a maximal connected subgraph. A tree is a connected graph without cycle. A graph is said to be chordal graph iff there is no chordless cycle of length more than 3. In the simple graphs that are the trees, the articulations are the vertices which are not leaves (vertices which have at maximum one incident edge). The removal of an articulation vertex defines several subgraphs. To decompose a tree, we copy an articulation in each subgraph its removal defines. Graphs subjacent to corpus are not trees. To decompose them, we use instead of articulation vertex in the case of tree, the groups of vertices called 'minimal separators'. A subset S of vertices is a ***minimal separator*** of a connected graph $G = (V, E)$ if $G(V - S)$ has at least two connected components. In general, a graph has an exponential number of minimal separators. However, it has been proved that the number of clique minimal separators (separators that are completely connected) is weak and less than the number of vertices. In fact the decomposition of the graph and the enumeration of all clique minimal separators can be done in linear time $O(|V||E|)$. This has been dealt by Tarjan in [?]. This process is based on minimal triangulation algorithms that embed a graph into a chordal graph by the addition of an inclusion-minimal set of edges.

Thus we propose an algorithmic process which decomposes a graph of terms subjacent to a given corpus of textual data into connected groups of terms which are called 'atoms' and that do not have clique separators. One of the interesting advantages of this decomposition is that atoms we define are not disjoint, but can have an overlap. The process of decomposition consists in copying a 'clique minimal separator' into different parts of the graph, so that each overlap between two 'atoms' is a 'clique minimal separator'. Previous works have proved that the intersection graphs of subtrees in trees are exactly the chordal graphs [?]. In our application, we have found that our graphs of atoms are chordal. To keep the structure of the graph, they have to be copied in each different part defined by their removal. One of the important features of this method is that the decomposition is unique.

3.2 Algorithmic decomposition

We have introduced our method in [4]. To implement our program, first we computed a minimal triangulation in linear time O(n.m) followed by the process of graph decomposition dealt by Tarjan. Thus, we produced all clique minimal separators of the graph. Then we decompose the graph in a set of atoms. If there are cycles with more than three vertices in the graph, they will be retrieved into atoms. The algoritm decomposition is described as below:

Decomposition algorithm

Input : A graph $G = (V, E)$ and moplex ordering κ.
Output : Set of graph components

Initialization: i⟵1,j⟵1.
begin
  For i=1 to $|\kappa|$-1 do ;
  $S_i \leftarrow N_G(\kappa(i))$; $N_G(\kappa(i))$ is the neighborhood of i
  if S_i is a clique **then**
    $C \leftarrow i$ and its neighborhood;
    $C \leftarrow G$-C;
    $Comp(j) \leftarrow C \bigcup S_i$;
    $j \leftarrow j + 1$;
  $Comp(j) \leftarrow G$ (Last component);
end

3.3 Atom decomposition of Small World Graphs

A graph is said to be SWG when it simultaneously shows both low diameter and high clustering measure, (i.e., high density of edges in the neighborhood of each vertex). According to [5], the path length $L(p)$ and the clustering coefficient $C(p)$ are the two structural measurements that characterize the SWGs.

The usual approach to visualize a SWG consists in computing a decomposition into highly connected components and to offer to the user an abstract view of the network to start with [6].

We adopt a similar approach except that we compute overlapping atoms [4] instead of disjoint connected components. The atoms of a graph can be defined based on the concept of (a, b)-**clique separators**.

By definition an atom A of a graph G contains at least one complete separator S of G, however S is not a separator of A. Atoms overlap if they contain the same separator of G.

In our experiments, we have observed that graphs have a central atom with long cycles that involves almost 50% of the vertices and numerous peripheral atoms of small size that are almost chordal (cycles have less than three elements).

To visualize small atoms and their interactions on a map, we shall define a valued graph exclusively based on the structure of $G = (V, E)$. Each atom A

is labeled by the vertex w_1 having the highest degree defined as the number of edges linking w_1 to another vertex w_2 in A. Atoms having the same label are merged together. The valued graph of atoms that we shall denote by $G(At) = (V_{At}, E_{At}, a_{At})$ is defined as follows.

The vertex of $G(At)$ are pairs of the form (k, l) where k is a vertex of G_k and l is the label of an atom containing k. An edge $e = (w_1, w_2)$ is defined between two vertices $w_1 = (k_1, l_1)$ and $w_2 = (k_2, l_2)$ if one of the following happens:

1. $l_1 = l_2$ and (k_1, k_2) is an edge of G. In this case the value s_{At} of the edge e is set to 1.
2. $k_1 = k_2$ and there exists a clique separator S in G that separates the atom l_1 from the atom l_2. In this case $a_{At}(w1, w2)$ is set to the ratio between the number of elements in S and the total number of elements in atoms l_1 and l_2.

The first case corresponds to edges in atoms. To ensure that the related vertices will not be separated by any clustering procedure, we set the value of such edges to 1, the maximum. The second case deals with edges relating copies of G vertices in different atoms. This valued graph can be displayed as described here below.

Now to vizualize the central atom we shall look for optimal minimal separators that allow to split the atom into parts of equal size.

4 Graph decomposition by optimal separators

Combinatorial optimization is a lively field of applied mathematics, combining techniques from combinatorics, linear programming, and the theory of algorithms, to solve optimization problems over discrete structure. Combinatorial optimization searches for an optimum object in a finite collection of objects. Typically, the collection has a concise representation, like a graph, while the number of objects is huge. Combinatorial optimization problems are usually relatively easy to formulate mathematically, but most of them are computationally hard. The basic idea behind polyhedral techniques is to derive a good linear formulation of the set solutions by identifing linear inequalities that can proved to be necessary in the description of the convex hull of feasible solutions.

4.1 Polyhedral approach for ab-separator problem

Finding a balanced minimum-weight separator in a n-vertex graph that partitions the graph into two components of similar sizes, smaller than $2n/3$, is relevant in many problems. Formally, the vertex separator problem (VSP) can be stated as follows. The instance consists of a connected undirected graph $G = (V, E)$, with $|V| = n$, an integer $\beta(n)$ such that $1 \leq \beta(n) \leq n$ and a cost c_i associated with each vertex $i \in V$. The problem is to find a partition

$\{A, B, C\}$ of V such that :

$$E \text{ contains no edge } (i,j) \text{ with } i \in A \ , j \in \ B, \tag{1}$$

$$\max\{|\mathrm{A}|, |\mathrm{B}|\} \leq \ \beta(n) \ , \tag{2}$$

$$\sum_{j \in C} c_j \text{ is minimized} \tag{3}$$

The vertex separator problem (VSP) is NP-hard [7]. In 2005, Egon Balas and Cid De Souza provide the first polyhedral study of the vertex separator problem (VSP) [8]. Recently, Didi Biha and Meurs [in preparation], starting from the Balas and De Souza'work, studied the vertex separator polyhedron and gived several new valid inequalities for this polyehdron.

4.2 The polyhedron of separators

For a given graph $G = (V, E)$, we consider the particular case of (VSP) in which two non-adjacent vertices a and b are given and we look for a partition $\{A, B, C\}$ which satisfies (1) and (2) with $a \in A$, $b \in B$ and $|C|$ is minimum. This particular case is called *ab-separator* problem in this paper.We can solve (VSP) by solving at most $\frac{n(n-2)}{2}$ ab-separator problems.
Given the non-adjacent vertices a and b, the incidence vector of a partition $\{A, B, C\}$ of V which satisfies (1) and (2) with $a \in A$ and $b \in B$ is
$X = (x_{1a}, ..., x_{(n-2)a}, x_{1b}, ..., x_{(n-2)b}) \in \{0,1\}^{2(n-2)}$, with $x_{ia} = 1 \Leftrightarrow i \in A$, $x_{ib} = 1 \Leftrightarrow i \in B$, $\forall i \in V \setminus \{a, b\}$.
Let P_{ab} be the polyhedron associated to the ab-separator,
i.e. $P_{ab} = Conv\left\{X \in R^{2(n-2)}\right\} : X$ is an incidence vector for some ab-separator partition $\{A, B, C\}$.
Let Γ_{ab} be a simple chain between a and b. Let $I(\Gamma_{ab})$ be the set of intern vertices of Γ_{ab}. The inequality $\sum_{i \in I(\Gamma_{ab})} (x_{ia} + x_{ib}) \leq |I(\Gamma_{ab})| - 1$ is valid for P_{ab} (4.2).
In fact, if $\{A, B, C\}$ is an ab-separator partition, then every chain from a to b contains at least one vertex of C. For all couple of non-adjacent vertices $(i, j) \in V$, let α_{ij} be the maximum number of disjoint chains between i and j.

4.3 Model

Our model is formally described as follows:
Data :

- A connected undirected graph $G = (V, E)$, with $|V| = n$,
- An integer $\beta(n)$
- $a \in A$, $b \in B$ virtual vertices,
- $\alpha_{min} = \mathrm{Min}\{\alpha_{ij}, \quad i \in V, \ j \in V, \ (i,j) \notin E\}$

The (VSP) can be formulated as the following mixed integer programming :

$$\text{Maximize } \sum_{i=1}^{n}(x_{ia}+x_{ib})$$

$$s.c.:$$

$$x_{ia} \in \{0,1\}, \qquad \forall i \in V \tag{4}$$

$$x_{ia}+x_{ib} \leq 1, \qquad \forall i \in V \tag{5}$$

$$x_{ia}+x_{jb} \leq 1, \qquad \forall (i,j) \in E \tag{6}$$

$$x_{ja}+x_{ib} \leq 1, \qquad \forall (i,j) \in E \tag{7}$$

$$\sum_{i=1}^{n}(x_{ia}+x_{ib}) \leq n-\alpha_{min} \tag{8}$$

$$1 \leq \sum_{i=1}^{n} x_{ia} \leq \lfloor \frac{n-\alpha_{min}}{2} \rfloor \tag{9}$$

$$1 \leq \sum_{i=1}^{n} x_{ib} \leq \beta(n)\ ,\ 1 \leq \sum_{i=1}^{n} x_{ia} \leq \beta(n), \tag{10}$$

The constraint (5) is valid since A and B are disjoint sets. The constraints (6) and (7) are valid since there is no edge between the two sets A et B. The constraint (8) comes from (4.2). Without loss of generality, we may assume that $|A| \leq |B|$. Furthermore, $|A|+|B| \leq n-\alpha_{min}$, thus $|A| \leq \lfloor \frac{n-\alpha_{min}}{2} \rfloor$, that is corresponding to the constraint (9) in our model. The constraints (10) are valid since the ab-separators satisfy (2).

5 Application

Based on graph decomposition technique into atoms described here above, the methodology used here for viewing results merges two graphs into the same visual output: the graph of term-term associations and of variation links and, a second graph of author-term association (ATCA here below), enabling us to link authors to the clusters of terms used (research topics) in their publications.

5.1 Methodology summary

The method works in seven phases described hereafter:

1. **Term extraction and selection:** noun phrases (NPs) are extracted from ISI abstracts. NPs are merged together based on COMP relations described in section 2: spelling variants, left expansions, insertions, modifier substitution and WordNet synonyms. The resulting clusters are the connected components of the graph where vertices are terms and there is one edge between each term and its variants. We shall refer to these clusters as Term

components. Only Term components with at least two vertices are considered. Each of these Term components are labelled by the NP having the most number of variants. Supplementary NPs are considered based on CLASS relations. These relations are: left expansions and head substitutions on NPs with at least three words. NPs involved in such variations are added as supplementary Term components but not clustered. Each of them is isolated in a separated component. This a way to select NPs based on surface linguistic variation.

2. **Terminology graph extraction:** associations are computed between Term components or authors, this results in the extraction of a new graph where vertices can be both previous term components or authors. An edge is drawn between two of these vertices whenever a valid association is found. Associations are computed in the following way. For each component or author x, we denote by $D(x)$ the set of documents where abstracts have at least one NP in x if x is a component or where x is one of the authors otherwise. Then two vertices c_1 and c_2 are associated if there are at least two documents in $D(c_1).D(c_2)$. In this case, an equivalence $E(c_1, c_2)$ coefficient is computed between c_1 and c_2 in the following way:

$$E(c1, c2) = \frac{|D(c1).D(c2)|^2}{|D(c1)|.|D(c2)|}$$

   Only associations for which $E(c_1, c_2) > 0.05$ are considered. Let us call this graph ATCA (Associations between Term Components and Authors).
3. **Graph components:** connected components of ATCA are computed. It is usual that in this kind of graph, there are lots of small components and only one really big component that contains more than two thirds of the vertices. We shall refer to it as the main ATCA component.
4. **Atom decomposition:** the main ATCA component is decomposed into atoms. An atom is a sub-graph where there is no clique separator. Again in this kind of graph, it often occurs that there is a central huge atom and several small ones. We shall refer to the biggest atom as the central ATCA atom.
5. **peripheral atom layout:** peripheral atoms and their interactions are visualized by generating the atom graph as described in section 3.3. Since atoms overlap, each atom is labelled by its central vertex (the vertex with the highest degree). A vertex in several atoms is duplicated. Each copy is labelled by the vertex label and the atom label. An edge is drawn between two copies of ATCA vertices if they share the same atom label or if they are copies in separated atoms that involve a common ATCA separator.
6. **central atom separation:** we also implemented the splitting of the central atom based on optimal separators as described in §4.
7. **Central atom visualisation:** the central atom componenrts are visualized based on Single Link Clustering that groups together vertices whose association equivalence coefficient between is higher than any other one in the neighbourhood (local maximums). This allows us to reduce the size while preserving the graph structure.

5.2 Results

We experimented this method on the same corpus as [9] on terrorism extracted from ISI bibliographic database. In this corpus, 57,855 NPs were extracted from 3,366 ISI abstracts. These NPs were clustered into 3,293 Term components with at least two NPs. The maximal size of Term components is 30. 8,357 supplementary terms having at least one CLASS variant are added to the set of Term components. The ATCA graph has 16,258 edges. Its main component has 9,324 edges over 1,070 vertices. This component involve 489 atoms. The central atom has 2,070 edges over 307 vertices and can be splitted in three parts using two separators of four vertices each. All the other atoms have less than 29 vertices.

Upon closer inspection, these three sub-networks corroborate the findings in Chen's study (2006) on the same corpus but on a shorter period (1990-2003). In [9], three major groups of clusters were identified by author co-citation and term networks using CiteSpace II: a cluster on 'body injuries in terrorist bombing', a second bigger cluster on 'health care response to the threat of biological and chemical weapons', a third biggest and more recent cluster on pyschological and pyschiatric impacts of the september 11, 2001 terrorist attack with terms like 'United States' and 'posttraumatic stress disorder' (PTSD) being very prominent. We found a similar demarcation in the internal structure of the central atom that has the following separators:

1. 'health care provider', 'specific clinical', 'AU: Tonat_K' and 'physical injury'.
2. 'public health', 'AU: Tracy_M', 'terrorist attack victim' and 'AU: Pfefferbaum_B'.

Figure 1 shows the atom graph of the terminology graph computed on this corpus and its central atom before applying the decomposition based on optimal minimal separators.

6 Conclusion

To the best of our knowledge, this paper is the first attempt to apply graph atom decomposition to knowledge domain mapping [11, ?,12]. The advantage of atom graph decomposition is that it is unique since it is based on the intrinsic structure of the graph. Its main drawback is that small atoms do not always exists in a graph. The results obtained on the corpus used in this experiment tend to show that: atom decomposition is tractable on a large corpus of documents and that central atoms can be separated using optimal separators.

Previous experimentations on other bibliographic corpura dealing with *Information Retrieval*, *genomics* or *Organic Chemistry* have confirmed these results.

References

1. Bales ME., Johnson SB.: Graph theoretic modeling of large-scale semantic networks. J. Biomed Inform. **39**(4) (2006) 451–464

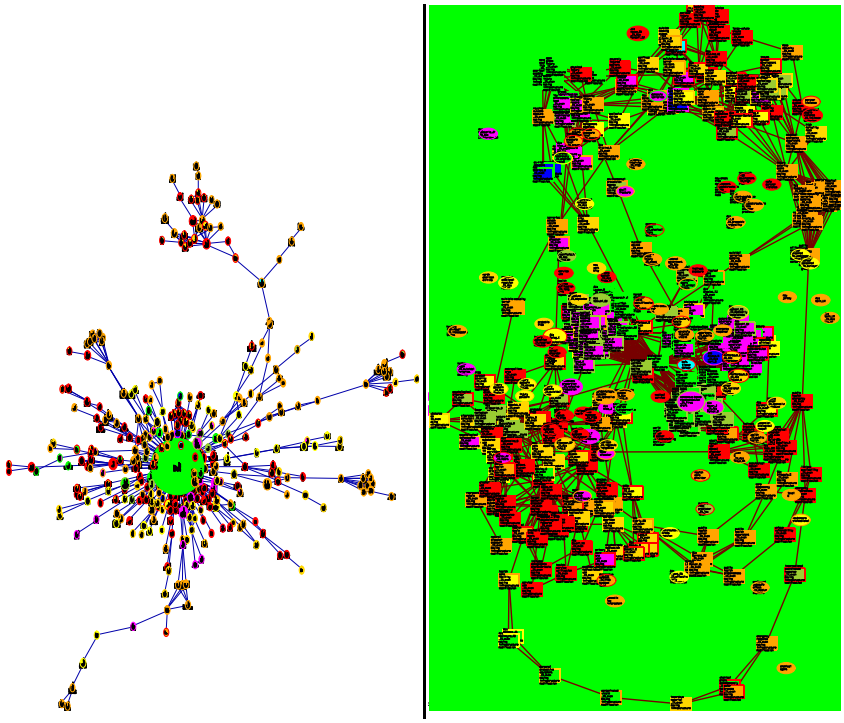

Fig. 1. Atom graph (left) and its central atom (rigth)

2. SanJuan, E., Ibekwe-SanJuan, F.: Text mining without document context. Information Processing and Management **42** (2006) 1532–1552
3. Ibekwe-SanJuan, F.: A linguistic and mathematical method for mapping thematic trends from texts. In: Proc. of the 13th European Conference on Artificial Intelligence (ECAI), Brighton, UK (1998) 170–174
4. Berry, A., Kaba, B., Nadif, M., SanJuan, E., Sigayret, A.: Classification et désarticulation de graphes de termes. In: Proc. of the 7th International conference on Textual Data Statistical Analysis (JADT 2004), Louvain-la-Neuve, Belgium (2004) 160–170
5. Ferrer-i-Cancho, R., Sole, R.V.: The small world of human language. Proceedings of The Royal Society of London. Series B, Biological Sciences **268**(1482) (2001) 2261–2265
6. Auber, D., Chiricota, Y., Jourdan, F., Melancon, G.: Multiscale visualization of small world networks. In: IEEE Symposition on Information Visualisation, IEEE Computer Society (2003) 75–81
7. Bui, T., Fukuyama, J., Jones, C.: The planar vertex separator problem : Complexity and algorithms. Manuscript (1994)
8. Balas, E., de Souza, C.C.: The vertex separator problem: a polyhedral investigation. Math. Program. **103**(3) (2005) 583–608
9. Chen, C.: Citespace ii: Detecting and visualizing emerging trends and transient patterns in scientific literature. JASIST **57**(3) (2006) 359–377

10. Neumann, A., Gräber, W., Tergan, S.O.: Paris - visualizing ideas and information in a resource-based learning scenario. In: Knowledge and Information Visualization. (2005) 256–281
11. Braam, R., Moed, H., A., A.V.R.: Mapping science by combined co-citation and word analysis. 2. dynamical aspects. Journal of the American Society for Information Science **42**(2) (1991) 252–266
12. Schiffrin, R., Börner, K.: Mapping knowledge domains. Publication of the National Academy of Science (PNAS) **101**(suppl 1) (2004) 5183 – 5185