Combining Optimal and Atomic Decomposition of Terminology Association graphs

Marie-Jean Meurs MARIE-JEAN.MEURS@UNIV-AVIGNON.FR
LIA - University of Avignon,France

Eric SanJuan ERIC.SANJUAN@UNIV-AVIGNON.FR
LIA - University of Avignon,France

Keywords: clustering, optimization, graph separators, text mining, triangulated graphs

Abstract

We introduce novel approaches of graph decomposition based on optimal separators and atoms generated by minimal clique separators. The decomposition process is applied to co-word graphs extracted from Web Of Science database. Two types of graphs are considered: co-keyword graphs based on the human indexation of abstracts and terminology graphs based on semi-automatic term extraction from abstracts.

1. Introduction

Topic mapping consists in representing and visualizing the main topics of a knowledge domain represented by a corpus of documents (Schiffrin & Börner, 2004), often a large set of peer reviewed abstracts from a bibliographic database. Most of the methods are based on term $\times$ document matrix where terms can be words, name entities, noun phrases or any type of bibliographic information. Terms are related if they appear together in several documents. Self Organized Maps are among the most popular approaches and have been implemented in powerful enterprise information retrieval systems (see for example www.infocodex.com). Formal Concept Analysis has generalized pair-wise associations to frequent item mapped into a Galois lattice. In this context, graph theory has been intensively used not only to build efficient data structures but also to define indicators following Social Network theory. The emergence of powerful graph visualization applications (for example: www.aisee.com) has brought back the attention on structure analysis of association graphs where vertices are terms and edges represent relations. On these graphs, the fact that maximal cliques correspond to closed frequent item sets has impulse the development of specialized clustering algorithms that converge towards dense subgraphs. In this paper we combine a divide and conquer method (revealing high connected components) based on an optimization algorithm with the atom decomposition based on Lex-M algorithm. Our experiments show whenever the optimization approach is tractable.

Preliminary work. Under review by the International Workshop on Mining and Learning with Graphs (MLG). Do not distribute.

2. Association graphs

First we formally define what we mean exactly by association graph. Let us consider a binary relation R defined on a pair of sets U, V, (i.e. $R \subseteq U \times V$). For example, U is a set of documents and V is a set of bibliographic items like keywords, terms, authors, etc. Then, $(d, t) \in R$ if t is a keyword indexing d, or an author of d, or a term appearing in the abstract of d, etc.

Let us now consider the non directed graph $G_R = (V, E)$ where $E = \{\{v_1, v_2\} : v_1 \neq v_2, (\exists u \in U)\{(u, v_1), (u, v_2) \subseteq R\}$ and a real weight function f on E. In practice f measures the strength of an association. It can simply be co-occurrence $f(v_1, v_2) = |\{u \in U : \{(u, v_1), (u, v_2)\} \subseteq R\}|$ or some more sophisticated measures like mutual information, likelihood, etc. In practice, f is often a combination of several measures. Even if these measures permit to remove weak edges, they are often contradictory when they have to state which is the most important relation. Therefore, in many applications, edge weakness appears to be a local property just depending on an edge weighting function but not edge or vertex relevance. This fact has lead Social Network Theory to introduce much more

elaborated indicators like betweenness centrality that relies on whole set of graph geodesics.
We thus define association graphs as being subgraphs $A_s = (V, E_s)$ of G_R where $E_s = \{e \in E : f(e) > s\}$ and s is a fixed positive real. Hence, an association graph is a set of valid associations, where valid means that their strength is over some threshold.

2.1. Extraction from Bibliographic databases

One large category of association graphs contains those on keywords and terms built from a set of references resulting from a query on large bibliographic databases. Since queries are often short Multi Word Terms, these graphs have a dense central kernel formed around the query terms.
In this paper we have considered the set U of $6,692$ abstracts extracted from Web of Science (scientific.thomson.com) with the query "magnetic interaction" covering the last decade. The set V contains $53,735$ terms which were semi automatically extracted from abstracts using TermWatch system (https://daniel.iut.univ-metz.fr/TermWatch/index.pl). We also considered the set ID of $17,486$ keywords indexing the references. Only pairs of terms (key-words) co-occurring at least twice have been considered. The weight function $f(v_1, v_2)$ we chose to select relevant edges, is the product $P(v_1/v_2) \times P(v_2/v_1)$ of conditional probabilities of finding a term of the edge $\{v_1, v_2\}$ knowing the presence of the other. This function allows to remove noisy edges generated by very frequent terms (key-words). We then selected three thresholds $s_{1000}, s_{1500}, s_{2000}$ in the following way: s_x is the minimal value such that the maximal cardinality of connected components in A_{s_x} is less than x. We found out that all the graphs A_{s_x} have a central component with more than 95% of vertices. In the sequel, we shall identify these huge components with the graph itself. It is worth mentioning that these graphs have the small world property: a small diameter and a high clustering coefficient.

3. Decomposition approaches

The problem that we handle here is the separation of association graphs into equilibrate dense subgraphs with a small overlap between them whenever they exist. This problem is a central problem to all topic mapping approaches based on association graphs. We combined the search of minimal separators with a structural graph decomposition into atoms.
Let us state our objective in a formal way. We want to find a balanced minimum-weight separator in a n-vertex graph that partitions the graph into two components of similar sizes. The instance consists of a connected undirected graph $G = (V, E)$, with $|V| = n$, an integer $\beta(n)$, upper bound of the size of each component such that $1 \leq \beta(n) \leq n$ and a weight w_i associated with each vertex $i \in V$. The vertex separator problem (VSP) is NP-hard (Bui et al., 1994).
In order to favour separators that are central in the graph we compute for each vertex the number of geodesics that cross it. We take this score as betweenness centrality evaluation. When there exists several possible minimal separators, the one with maximal betweenness is chosen.

3.1. Optimal separators

We first try to solve the problem directly. In 2005, Egon Balas and Cid De Souza provide the first polyhedral study of the vertex separator problem (VSP) (Balas & de Souza, July 2005). In (DidiBiha et al., 2007), the polyhedral approach for graph decomposition is presented using a combinatorial model searching optimal separators with the smallest size. The weight constraint consisted of minimizing the size of the separator, i.e. all the vertices have the same weight equal to 1.
In the work we present, the weight w_k given to a vertex k is equal to one plus the inverse of its betweenness. The size constraints we chose bound the size of one component of the partition by $2n/3$ and the other by $n/2$. The choice of $2n/3$ provides a well-balanced partition and the choice of $n/2$ improves the resolution time without loss of generality. Hence, the optimization problem we solve consists of finding a partition $\{A, B, C\}$ of V such that E contains no edge (i, j) with $i \in A$, $j \in B$, $|A| \leq 2n/3$, $|B| \leq n/2$ and $\sum_{k \in C} w_k$ is minimized.
To solve the mixed integer programming optimization problem associated with this formal model, we use the object oriented APIs of Concert Technology for C++ from the Ilog CPLEX 9.0 solver [1]. The resolution provides exact solutions and optimal separators.
The resolution is based on a branch and bound algorithm exploring a search tree. At each node, CPLEX solves the relaxation of the initial problem. We use the default CPLEX search settings which branching rules consist in selecting the more fractional variable at each node of the search tree. All the nodes are explored according to the best first strategy, i.e. the best expected objective function value.
We ran the optimizer with and without using the betweenness centrality. On small graphs we found out that the decomposition was a bit slower (less than

[1] http://www.ilog.com/products/cplex/

10%) using betweenness centrality. However on large instances, the resolution is faster using betweenness centrality: 2.1 times faster for $x = 1000$, 0.27 times faster for $x = 1500$ and higher. The acceleration in the case $x = 1000$ is due to the fact that the set of minimal separators for $A_{s_{1000}}$ is much bigger. The betweenness criteria allows to consistently reduce this set.

3.2. Clique minimal separators

Despite the use of betweenness to reduce the solution space and the use of a 3.06GHz CPU with 4GB of RAM, this direct approach only got a solution for $x = 1000$. For higher x the optimizer was still running after five days.

We then got the idea of reducing the input based on clique separators. A clique separator is a completely connected subgraph whose removal disconnects the graph. Clique separators allow to define the concept of atom graph as maximal connected subgraph without clique separators. The atom enumeration can be done in $O(|V|.|E|)$ time. As pointed out in previous works, terminology association graphs issued from a query on a bibliographic database have a central atom that often covers 2/3 of the vertices (DidiBiha et al., 2007). The key point is that atom decomposition is much more tractable by the optimizer. The computation time decrease to less than 1 hour against several days for graphs of similar size whithout atom decomposition. This is because the optimizer does not get lost by linear sub-graphs which can be decomposed in many different ways and thus increase exponentially the number of solutions to explore. They have almost as many separators as disconnected subgraphs.

Thus, instead of trying of decomposing the initial graph, we look for its main atoms and decompose them. In terminology association graphs, there is often a unique central atom connected to the root clique separator of several disjoint trees of atoms. In this case, since an optimal separator avoids cliques, any optimal separator atom is an optimal graph separator.

3.3. Algorithm decomposition

Previous observation lead us to consider the following algorithm 1 that alternates atom search and optimal separation untill the graph is splitted into highly connected components. Indeed, the components returned by the algorithm do not have nor complete separators, nor optimal separators.

Algorithm 1 OptiSep

Input: graph G
$TODO = (G)$; $RESULT = ()$
repeat
  G'=shift($TODO$)
  for v **vertex in** G' **do**
    $f(v)$ = number of geodesics crossing v
    for P_i **connected component of** G' **do**
      if exists atom H in P_i such that $|H|/|P_i| > 2/3$ **then**
        (A, B, C) =optimal_separation(G,P_i,f)
        if $|C| < (|A| + |B|)/6$ **AND** $||A| - |B|| > 1/3(|A| + |B|)$ **then**
          $TODO = TODO.(G - B, G - A)$
          ELSE $RESULT = RESULT.(G')$
        end if
      end if
    end for
  end for
until $TODO$ is empty
return $RESULT$

4. Discussion

Using atom decomposition, we have shown that a large family of terminology graphs considered in topic mapping can be decomposed, meanwhile the problem is NP-complete. Next we plan to study co-citation graphs. Most of them contain a threshold subgraph that only has clique separators because they are triangulated.

References

Balas, E., & de Souza, C. C. (July 2005). The vertex separator problem : a polyhedral investigation. *Mathematical Programming*, *103*, 583–608.

Bui, T., Fukuyama, J., & Jones, C. (1994). The planar vertex separator problem : Complexity and algorithms. *Manuscript*.

DidiBiha, M., Kaba, B., Meurs, M.-J., & SanJuan, E. (2007). Graph decomposition approaches for terminology graphs. *Proceedings of MICAI 2007, Aguascalientes, Mexique*.

Schiffrin, R., & Börner, K. (2004). Mapping knowledge domains. *Publication of the National Academy of Science (PNAS)*, *101*, 5183 – 5185.