A Vectorial Representation for the Indexation of Structural Informations

Nicolas Sidère, Pierre Héroux, Jean-Yves Ramel

A vectorial representation for the indexation of structural informations

Nicolas Sidere[1,2], Pierre Heroux[1], and Jean-Yves Ramel[2]

[1] Université de Rouen
LITIS EA 4108
BP 12 - 76801 Saint-Etienne du Rouvray, FRANCE
[2] Université François Rabelais Tours
Laboratoire Informatique de Tours
64 Avenue Jean Portalis
37200 Tours, FRANCE
{nicolas.sidere,jean-yves.ramel}@univ-tours.fr
pierre.heroux@univ-rouen.fr

Abstract. This article presents a vectorial representation of structured data to reduce the complexity of dissimilarity computations in an information retrieval context. This representation enables, via a computation of an adapted measure, to approximate the distance between structural representations in both context of distance between graphs and searching occurrences of subgraphs. Preliminary results show that the proposed representation offers comparable performance with those of the literature.

Key words: graph signature, structured information indexation and retrieval

1 Introduction

The evolution of digitization techniques, the easiness of broadcasting by the way of the Internet and the wish to keep and access digitized collections of documents put the themes of indexation and retrieval in the heart of many research axes. With the diversity of thematics (preserving old books, archiving administrative data, automatic reading, ...) the amount of data generated grows up and up. This contributes to the multiplication of works in this domain. If some solutions come out, these are often restricted to a specific application domain or corpus. Many uses a keyword-based indexing extracted by optical recognition characters systems which are inefficient on particular documents (old books archives, graphical documents, ...) or with a manual annotation limited by the size of the corpus or by the subjectivity of the user. So, one can see a certain interest in a new querying modalities. Consequently, the nature of works are more and more aimed to a new characterization of documents with indices such as structure. This is the point we intend to study in this article. According to the situation, the structure analyses can be used in many ways :

- the layout description. For example, the layout of a phone book, which is quite significant.
- the logical organization (title, section, paragraph, ...) can be used to differenciate some documents, a newspaper from a novel for example.
- the frequency of the same element, in a technical drawing with several occurences of a symbol.

The document retrieval consists in measuring how a structural description is relevant with respect to the user's need, which can be expressed with a structural representation of the request too. The purpose is to return k documents ranked according to their relevance. In fact, as the notion of structure of a document can be expressed by different ways according to the user, it is quite important that the user has the final choice.

Most of the times, this information is represented by graphs. Many methods can be found to label a graph to obtain this representation. Nevertheless, the computation of a graph-to-graph distance reveals to be a NP-Complex problem. This complexity grows in an exponential way with the number of nodes.

Our work aims at reducing this complexity. An interesting approach consist in extracting a numerical characteristic vector which embeds a part of the graph topological information. The comparison of two graphs is reduced to a more simple computation of a distance between two vectors in an euclidean space. More, an information retrieval context where the documents are described by structural informations, indexing can be done offline. Some works have already been done :

- A first method is presented in [1]. This representation is based on vertex degrees. The simplicity of the description combined with a comparison of two graphs reduced to a linear time allow to find topologically similar graphs in the most of cases.
- In the second approach presented in [3] where the representation of a document is focused on occurent elements (subgraphs). The document is described with a bag of symbol representation. A rich knowledge is necessary to enable the final identification.

The characterization of a graph by a vector gives, in the case presented above, advantages, but also drawbacks. For example, the description proposed by Lopresti and Wilfong ([1]) is not bijective. Thus, two non-isomorphic graphs can have the same vectorial description for which the distance is null. This ambiguity is due to the vector construction method. It is only based on the vertex degree. Several configurations from the same set of nodes with different graphs but the same signature may occur. We think that this description does integrate the informations on the graph topology in a superficial way.

Barbu's approach need a rich knowledge of the domain, that does not allow to treat a document set with a huge heterogenieity. So, we can notice that the relevance of the results are dependant of the graph vector characteritics.

Our approach relies on these works. The idea is to build a lexicon with model graphs differenciated by their topology and their size. The vector will be built

according to the occurences of these figures present in the target graph. All the problematic is on the choice of the lexicon, which must answer to the genericity problem but also needs to be performant and cute in results

In the next section we present the lexicon that we chose and how it can be built. The section 3 shows the construction process of a vectorial representation of a graph. Section 4 presents two distances operator using this vectorial description, both distances correspond to two different cases of use. Section 5 presents the first experiments showing that the representation offers performance equivalent to Lopresti and Wilfong in a context of classification. Finally, Section 6 evaluates and sets out a number of prospects for the continuation of this work.

2 The lexicon construction

As said before, the lexicon is the basement of the construction of our graph vectorial signature. So, the lexicon content is quite determining in the relevance of the vectorial representation. We have seen ([3]) that it is possible to build it from the n most frequent subgraphs of the database, for example. Nevetheless, this is only possible when the base presents a high homogeneity. The frequency is revealing a certain semantic. Thus, in the work of Barbu, subgraphs are frequently associated with graphic symbols, entities which carry meaning in a technical documentation.

Unfortunately, there are many cases where these assumptions are not verified. For example, old books are different depending on the author, publisher, date ... Therefore, to keep a generic nature, it is preferred to use a lexicon totally independent from the database content. However, this lexicon must be sufficiently comprehensive to ensure that these terms can afford to discriminate a graph from another.

We have therefore decided to take as a baseline the non-isomorphic graphs network presented in [2]. The network presents all graphs composed of n edges up to N (N is the maximum number of edges). This network is built iteratively from a graph made up of a single vertex. At each iteration, it is possible to construct a graph of rank n adding an edge to a graph of rank $n-1$ with the ability to add a vertex if needed. All solutions are being considered which makes the network complete. A graph with rank n built from a graph with rank $n-1$ is called successor. Conversely, the graph of $n-1$ is called predecessor. A graph of this network may have several successors. Similarly, several graphs with rank $n-1$ can rise to a single successor. Ways of construction of this non-isomorphic graph network can be stored to build all predecessors and successors of a graph.

Thereafter, the term *pattern* will refer to a subgraph element of the non-isomorphic graph network. So, the lexicon is composed of all patterns until the defined rank.

For example, the figure 1 shows the non-isomorphic graph network until the fourth rank giving a lexicon of 11 patterns. The dotted arrows indicate the path of construction of the network, the arrows are directed from the predecessors towards the successors.

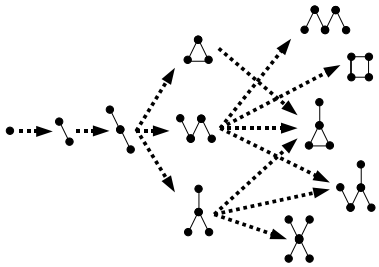


Fig. 1. The non-isomorphic graph network

Table 1 gives the number of elements in the lexicon depending on the maximum rank of the non-isomorphic graph network.

Rank	Size
0	1
1	2
2	3
3	6
4	11
5	23
6	51
7	117
8	276

Table 1. Size of the lexicon depending on the rank of the non-isomorphic graph network

We can notice that the number of patterns increases exponentially with the rank. The size of the lexicon is a parameter to determine according to several criteria. Indeed, the complexity of the transformation to a vectorial representation is directly dependent of the number of patterns. However, the more the size of the lexicon increases, the bigger the patterns it integrates are. The vectorial representation then integrates more information on topology. Therefore, it is necessary to find a trade-off between expressiveness and complexity.

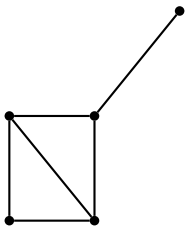
3 The construction of the vectorial representation

The construction of the vector consists on determining the frequency for each patterns of the lexicon in the graph to describe. The dimension of the vector is the size of the lexicon. Its construction can become very costly in time. However, the indexing phase, which is to build the vectorial representation of all graphs of the base, can be done offline. The complexity of construction of the vector is only critical when processing the query graph. Indeed, this graph can be

obtained from a sample document presented by a user. The extraction method of representation will be the same that is used for the base and may be costly.

The lexicon is sorted in the order of the subgraphs network, the first value of the vector describing a graph is then the number of vertices, the second the number of edges, the third the number of subgraphs with two edges, ...

The search for patterns may be different depending on constraints. The recording (or not) of patterns that share one or several components (vertices or edges) directly influences the vectorial representation of a same graph. The use of this type of constraints can be justified by the need to find a bijection between the graph and its vectorial representation. Indeed, for the vectorial description to be closest to the graph, a pattern discovered in a graph must be removed from it. Elements (edges or vertices) can not belong to another occurrence of the same pattern. However, the complexity of the extraction of the vectorial representation with these constraints increases. Following the case, it is not necessary to apply such constraints. We give details on this point in the next section



(a) Graph G

Motif						
Frq.	5	6	10	2	10	3

(b) Number of occurences of each pattern in G

Motif						
Frq.	5	6	3	1	1	1

(c) Number of edge-disjoint occurences of each pattern in G

Motif						
Frq.	5	2	1	1	1	1

(d) Number of vertex-disjoint occurences of each pattern in G

Fig. 2. A simple graph and its vectorial descriptions

As a didactic example, the figure 2 represents the vectorial description associated with a undirected and unlabelled simple graph (Fig. 2(a)) for a lexicon of size 6 (order maximum of patterns : 3) without constraints (Fig. 2(b)) with edge-disjoint constraint (Fig. 2(c)) and vertex-disjoint constraint (Fig. 2(d)).

In the next section, we present two ways to use the vectorial description. We will show the advantage of being able to decline the representation according to the selected use case.

4 Examples of measure depending on the case of use

A computation of distances in a grah space can quantify the difference between two graphs. However, the complexity related to the computation of the distance between graphs prohibits its use in a case of seeking information where the purpose is to order all documents of the basis (or the k nearest) depending on their proximity to a query. The vectorial representation that we propose will enable to approximate this distance by a dissimilarity measure between structural representations of documents on the one hand, and a request expressed by the same structural mean, on the other. Even if the extraction of this vectorial representation requires a important cost, this task of indexing can be done offline and can be tolerated.

The various dimensions of the vectorial representation that we offer presents a certain redundancy. Indeed, if a pattern of order n is counted, all its predecessors are also included in the statement. This redundancy is the integration of a degree of robustness we wished to provide in our presentation. It seemed appropriate to take into consideration the disturbances that could infer on structural representations often extracted automatically : appearance or disappearance of vertices or edges. Thus, if two identical graphs have strictly identical vectorial descriptions, two graphs which one is a noisy version of the other have at least some patterns below in common.

Finally, we justify our proposal of vectorial description by the fact that it can be used in two cases of use. In the first practice, it is to find in the indexed graph database the nearest to the query graph. A second application is to find graphs of the database containing the largest number of occurrences of the query graph. Both following subsections introduced measures on our vectorial description and applying to these two cases of use.

4.1 Measuring dissimilarity graph to graph

In this case of use, the purpose is to find, among a graph database, those closest to a query graph. The graphs in the database and the query graph are represented by a vectorial features. The distances between graphs of the database on the one hand and the query graph on the other hand is approximated by a dissimilarity measure. This measure of dissimilarity corresponds to the distance between the vectorial representations of graphs.

This problem has already been raised in the literature. We have chosen to use a Euclidean distance. In a n dimensions space, the distance is expressed in this form:

$$D(G_1, G_2) = \sum_{i=1}^{N} \sqrt{(k_{1i} - k_{2i})^2}$$

with G_1 and G_2 two graphs to compare, k_{i1} and k_{i2} the occurrences of the pattern i in G_1 and G_2.

Depending on the application framework, this measure can evolve. In technical document retrieval, for example, experts can provide *a priori* knowledge on semantics of symbols present in the document. This can "punish"or "prime" the presence or absence of a symbol. Of course, our approach can be expanded to all types of documents where sufficient knowledge allows to consider the relevance of patterns. That is rendered by weighting α_i each pattern i.

$$D(G_1, G_2) = \sum_{i=1}^{N} \sqrt{\alpha_i (k_{1i} - k_{2i})^2}$$

The weights α_i can then be determined by optimization algorithms or artificial learning depending on case of use. Similarly, if that vectorial representation is used on a uniforme database, it is possible to apply feature selection methods aiming at the reduction of dimensionality. The purpose of this reduction of the vector is to improve performance by removing aberrant or unnecessary patterns. It is also possible to combine patterns. There are several methods in literature, such as principal components analysis or feature selection.

In this case, the use of representations without constraints give a redundancy of the information in order to increase the precision of the vectorial representation.

4.2 Finding occurences of a query graph

Other applications do not need to measure a distance between two vectors to quantify the similarities between them, but investigate the presence and number of occurrences of the query graph in a graph, even in a pre-indexed database.

For example, finding a specific electrical component in a plan is a possible application. Here, the aim is to find the number u of occurrences of the query graph S in the graph G.

V_G is the vectorial representation of G and V_S for S. There is v_{Gi} (respectively v_{Si}) the occurences number of the pattern i in G (respectively in S).

$$V_G = \begin{bmatrix} v_{G1} \\ \vdots \\ v_{GN} \end{bmatrix}$$

$$V_S = \begin{bmatrix} v_{S1} \\ \vdots \\ v_{SN} \end{bmatrix}$$

So,

$$\forall i \in \mathbb{N}, 1 \leq i \leq N, v_{Gi} \geq u.v_{Si}$$

In other words, if a graph G includes u occurrences of a query graph S, then each subgraph S is present in G u times at least.

This scheme may be applied to all subgraphs of S and particularly on the patterns of the lexicon. Thus, for all i dimensions of the vectorial representation.

$$\forall i \in \mathbb{N}, 1 \leq i \leq N, u \leq u_i = \frac{v_{Gi}}{v_{Si}} \text{ avec } v_{Si} \neq 0$$

Consequently,

$$u \leq \min_{i/v_{Si} \neq 0} \left(\frac{v_{Gi}}{v_{Si}} \right)$$

Then, the quantity $\min_{i/v_{Si} \neq 0} \left(\frac{v_{Gi}}{v_{Si}} \right)$ can be used to approximate u, the occurences number of S in G.

It is noteworthy that in this case of use, a vectorial representation extracted with tight constraints, as we have seen in the third paragraph, must be used. Indeed, research occurrences of a subgraph often needs that all occurrences must be fully present in the graph. They can not share edges or node. The choice of constraint depends on the context and then requires an *a priori* knowledge of an expert.

5 Experiments

We describe in this section the first experiments on the vectorial description of graphs we conducted. During these tests, our vectorial description of graph is compared to the vectorial description proposed by Wilfong and Lopresti.

For both descriptions, we have limited the size of the vector to dimension 6. Thus, the vectorial description Lopresti and Wilfong graphs are in the vertices of degree 1 to 6 and our description are the patterns with 3 edges or less.

Both descriptions are compared on a classification task by nearest neighbours using the technique known as "leave one out", ie that each element is ranked considering all others belong to the learning database.

The first comparison was made on synthetic graphs. In this database, each class is associated to a random generating model with two parameters that are n the number of vertices of the graph and d the average degree of vertices. 20 classes have been generated for n taking values among 5, 10, 20, 50 and 100 and d taking values 0.2, 0.5, 1 and 2. Thus, 20 graphs were generated for each class, making a database of 400 graphs.

The rate of correct classification using the description of Lopresti and Wilfong reached 99.45 %. The few confusions are between classes of graphs C (n = 5, d = 0.2) and C (n = 5, d = 0.5). The classification using the proposed description given strictly identical results, the same confusion occuring on the same elements.

In a second experiment, graphs correspond to structural representations of symbols from GREC competition (recognition of symbols). We used 10 classes of symbols which have been alterated by rotation and vectorial bending. The structural representations are adjacency regions graphs corresponding to the black and white connected components. The classes contain between 5 and 19 elements per class, the total number of elements of the database rise to 88. Figure 3 shows examples of symbols from the database.

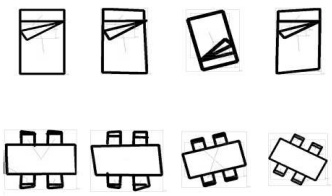


Fig. 3. Examples of symbols from the GREC database

The rate of correct classification obtained through the representation of Lopresti and Wilfong amounts to 51.14 % while the results obtained through our representation reached 53.41 %.

Good recognition rate achieved in the first experimentation should be put into perspective considering that generative models of random graphs as they are defined clearly distinguish the classes. Indeed, classes could be classified without any confusion by considering only the number of vertices and the number of edges.

The smaller rate classification observed on the application of recognition of symbols must be relativised because the structural representations which are used are not labelled and they only reflect the topology of the adjacency regions graphs without qualifying the nature of the region.

	Lopresti & Wilfong method	Our method
Synthetic graphs	99.45%	99.45%
GREC	51.14%	53.41%

Table 2. Summary of the first results

In one case as in others, it is interesting to note that the rate classification obtained with the two vectorial descriptions are comparable. This indicates that the description we propose can be used in a search for nearest neighbour.

6 Conclusions

In this article, we have presented a new vectorial description of graphs in order to reduce the complexity of computing distances needed for information retrieval applications with structured data. This representation is proposed to be applied to two cases of different use:

1. Finding similar graphs;
2. Finding graphs containing multiple occurrences of a query graph.

The results of preliminary tests highlight that our vectorial representation is equivalent in terms of performance to other approaches proposed in the literature to approximate distance between graphs in a task of classification.

However, we have seen throughout this article that some points remain outstanding. It now seems important to measure the influence of the size of the lexicon on the construction of the description and its complexity. Indeed, table 1 shows the need to limit the description to low rank to reduce the complexity of its construction. However, initial tests have proved that the relevance of the description was better with a important number of patterns. future work will naturally focus on the compromise between expressiveness and dimensionality of our representation.

To evaluate the relevance of our vectorial representation in the context of information retrieval to order graphs based on the number of occurrences of a query graph we have used structural representations of old document images formerly extracted from the platform AGORA (cf. [4]). We intend to build a set of queries with relevant documents to assess details and reminders for the k first documents according to the value of k.

The first experiments have shown that in the case of structural representations with many edges, the extraction of representation required a consequent computation time. The algorithmic optimizations would be realised.

7 Thanks

The works described in this article were done with the support of the ANR within the project NAVIDOMASS ANR-06-MDCA-12.

References

1. Lopresti, D.,Wilfong, G.: A fast technique for comparing graph representations with applications to performance evaluation, Int. J. Doc. Anal. Recognit., vol. 6, nb. 4, 219–229, 2003
2. Jaromczyk, J., Toussaint, G. : Relative neighborhood graphs and their relatives Proceedings of the IEEE, 1992
3. Barbu, E., Heroux, P., Adam, S., Trupin, E.: Clustering document images using a bag of symbols representation, ICDAR, 1216–1220, 2005
4. Ramel, J.Y., Busson S., Demonet M.L.: AGORA:the Interactive Document Image Analysis Tool of the BVH Project, DIAL, 145–155, 2006