Discovering Association Rules in Semi-structured Data Sets

M. Montes-y-Gómez[1], A. Gelbukh[1], A. López-López[2]

[1] Centro de Investigación en Computación (CIC-IPN), México.
mmontesg@susu.inaoep.mx, gelbukh@cic.ipn.mx

[2] Instituto Nacional de Astrofísica, Optica y Electrónica (INAOE), México.
allopez@inaoep.mx

Abstract. The discovery of association rules is one of the classic problems of data mining. Typically, it is done over well-structured data, such as databases. In this paper, we present a method of discovery of association rules in semi-structured data, namely, in a set of conceptual graphs. The method is based on conceptual clustering of the data and constructing of a conceptual hierarchy. A feature of the method is the possibility of using different levels of generalization.

1. Introduction

Nowadays, institutions have great capabilities of generating and collecting data. This situation has generated the need for new tools that assist transforming the vast amounts of data in useful information and knowledge.

Examples of such tools are the *data mining* systems (Han and Kamber, 2001). Typically, these systems allow extracting implicit patterns from large databases, but they cannot adequately manage a set of non-structured or semi-structured objects, such as a text collection.

This paper is related to this problem. It is focused on the automated analysis of a set of complex objects –possibly texts– represented as *conceptual graphs* (Sowa, 1984; Sowa, 1999).

There are some previous methods for the analysis of a set of conceptual graphs. Some of these methods consider their comparison (Myaeng and López-López, 1992; Mugnier and Chein, 1992; Mugnier, 1995; Montes-y-Gómez *et al*., 2000; Montes-y-Gómez *et al*., 2001a), other their use in information retrieval (Myaeng, 1992; Ellis and Lehmann, 1994; Huibers *et al*., 1996; Genest and Chein, 1997), and others their clustering (Mineau and Godin, 1995; Godin *et al*., 1995; Bournaud and Ganascia, 1996; Bournaud and Ganascia, 1997; Montes-y-Gómez *et al*., 2001b).

In this paper, we introduce the problem of discovering *association rules* in a set of conceptual graphs. Basically, we propose to use the conceptual clustering of the graphs as a kind of index of the collection, and to take advantage of this structure when searching for the associations.

This approach not only facilitates the identification and construction of the final association rules, but also allows constructing association rules at the different levels of generalization.

The rest of the paper is organized as follows. Section 2 describes the clustering of the conceptual graphs. This section focuses on the description of the main characteristics of the resulting conceptual hierarchy. Section 3 presents the method for discovering association rules. In the first part, the problem is formally defined; in the second part, the procedure for the discovery is detailed and illustrated with a simple example. Finally, section 4 drawns some preliminary conclusions.

2. Clustering of Conceptual Graphs

In some previous work, we presented a method for the *conceptual clustering* of conceptual graphs (Montes-y-Gómez *et al*., 2001b). There, we argued that the resulting conceptual hierarchy expresses the hidden organization of the collection of graphs, but also constitutes an abstract or index of the collection that facilitate the discovery of other kind of hidden patterns, e.g., the association rules. Following, we briefly explain the main characteristics about this conceptual hierarchy.

Conceptual clustering –unlike the traditional cluster analysis techniques– allows not only to divide the set of graphs into several groups, but also to associate a description to each group and to organize them into a hierarchy.

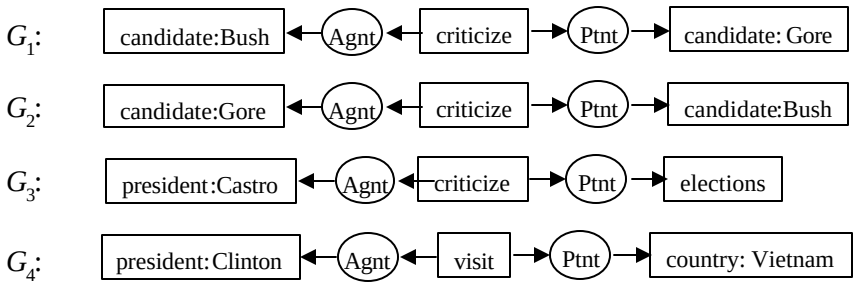

Figure 1. A small set of conceptual graphs

The resulting hierarchy H is not necessarily a tree or lattice, but a set of trees (a forest). This hierarchy is a kind of inheritance network, where those nodes close to the bottom indicate specialized regularities and those close to the top suggest generalized regularities[1]. For instance, given the small set of graphs of the figure 1, the hierarchy of the figure 2 expresses one possible conceptual clustering.

Formally, each node $h_i \in H$ is represented by a triplet[2] ($cov(h_i)$, $desc(h_i)$, $coh(h_i)$). Here $cov(h_i)$, the coverage of h_i, is the set of graphs covered by the regularity h_i; $desc(h_i)$, the description of h_i, consists of the common elements of the graphs of $cov(h_i)$, i.e., $desc(h_i)$ is the overlap of the graphs covered by h_i; $coh(h_i)$, the cohesion of h_i, indicates the less similarity among any two graphs of $cov(h_i)$, i.e., $\forall G_i, G_j \in cov(h_i), similarity(G_i, G_j) \geq coh(h_i)$.

In this hierarchy, the node h_j is a descendent of the node h_i, expressed as $h_j < h_i$, if and only if:

1. The node h_i covers more graphs than the node h_i: $cov(h_j) \subset cov(h_i)$.
2. The description of the node h_i is a generalization of the description of the node h_j: $desc(h_j) < desc(h_i)$.
3. The cohesion of the graphs of the cluster h_i is less or equal than the cohesion of the graphs of the cluster h_j: $coh(h_i) \leq coh(h_j)$.

[1] The construction of the conceptual hierarchy is a knowledge-based procedure (Montes-y-Gómez *et al*., 2001b). Basically, a concept hierarchy (defined by the user in accordance with his interests) handles the generalization/specialization of the graphs when the conceptual hierarchy is constructed.

[2] This notation was adapted from (Bournaud and Ganascia, 1996); where each node h_i was represented by a pair ($cov(h_i)$, $desc(h_i)$).

3. Discovery of asociation rules

The general problem of discovering association rules was introduced in (Agrawal *et al*., 1993). Given a set of transactions, where each transaction is a set of items, an *association rule* is an expression of the form $X \Rightarrow Y$, where X and Y are subsets of items. These rules indicate that transactions that contain X tend to also contain Y. For instance, an association rule is: "30% of the transactions that contain beer also contain diapers; 2% of all transactions contain both items". In this case, 30% is the *confidence* (c) of the rule and 2% it *sopport* (s).

Thus, the discovery of association rules is definded as the problem of finding all the association rules with a confidence and support greater than the user-specified values *minconf* and *minsup* respectivally.

Tipically, this problem is divided in the following two subproblems:

1. Find all the combinations of items with a support greater than *minsup*. These combinations are called the *frequent item sets*.
2. Use the frequent item sets to generate the desire association rules. The general idea is that if, say, {a,b} and {a,b,c,d} are frequent item sets, then the association rule {a,b} ⇒ {c,d} can be determined by computing the radio c = *support*({a,b,c,d})/*support*({a,b}). In this case, the rule holds only if $c \geq$ *minconf*.

3.1 Association rules among conceptual graphs

Given a set of conceptual graphs $C = \{G_i\}$, we define an association rule as an expression of the form $g_i \Rightarrow g_j (a, b)$, where g_i is a generalization of g_j ($g_j < g_i$); c is the confidence of the rule and s it support.

An association rule of this kind indicates that the conceptual graphs of the collection that contains the graph g_i, c% of the times also contains the more specialized graph g_j; also indicates that s% of the graphs of the collection contains the graph g_j. For instance, the following

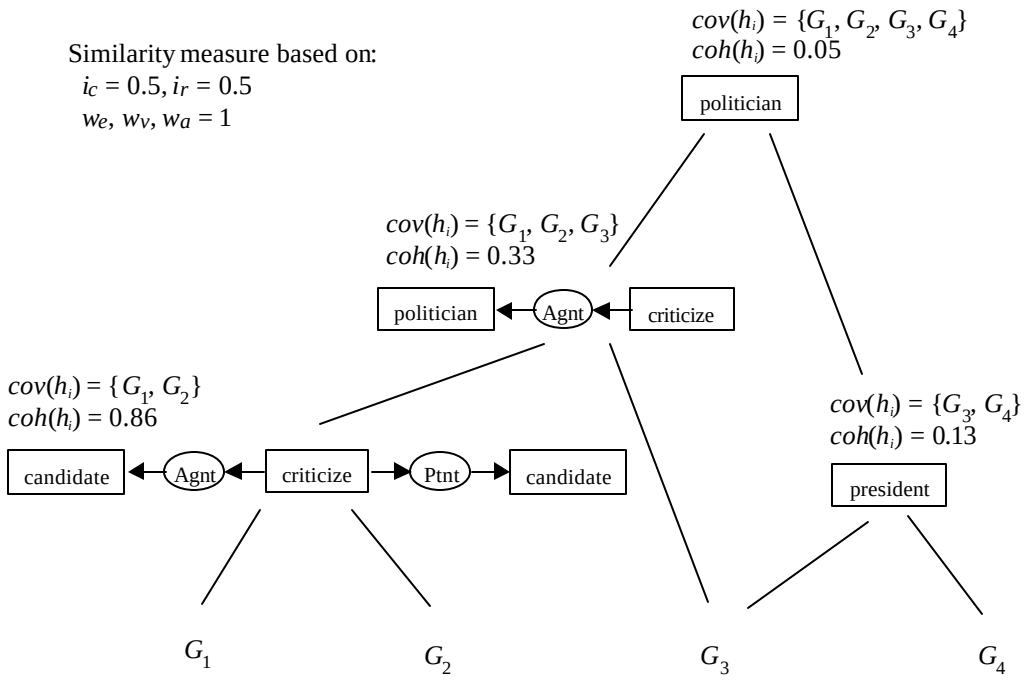

Figure 2. The conceptual clustering of the set of graphs

association rule corresponds to the collection of graphs of figure 1:

[criticize] ⇒ [politician]←(agt)←[criticize] (1/.75)

This rule indicates that: "*all* the graphs mentioning a criticism, describe a criticism does by a politician, and that 75% of the graphs of the collection mention a criticism does by a politician".

In consequence, the discovering of associations in a set of conceptual graphs is defined as the problem of finding all association rules $g_i \Rightarrow g_j\,(a, b)$, such that $c \geq minconf$ and $s \geq min\ sup$.

3.1.1. Procedure of Discovery

Basically, the discovery of association rules in a set of conceptual graphs is based on their conceptual hierarchy *H*. Each node $h_i \in H$ expresses a regularity, where it description $desc(h_i)$ is a common generalization of two or more graphs of *C*. Additionally, any conceptual graph *g* implicit in h_i, i.e. any graph *g* such that: $desc(h_i) < g$ and $\nexists h_k \in H : desc(h_i) < desc(h_k) < g$, is also an *implicit* common generalization of the same subset of graphs of *C*.

Figure 3 shows some common generalizations implicit in the conceptual hierarchy of figure 2. In this figure, the highlighted nodes are part of the original hierarchy, and the rest of the nodes are the implicit common generalizations.

In accordance with this description, we determine the following two kinds of association rules from a conceptual hierarchy.

Explicit associations: For each pair of nodes h_i, $h_j \in H$, such that the node h_j is a descendent of the node h_i, i.e. $h_j < h_i$, the following association rule can be constructed:

$$desc(h_i) \Rightarrow desc(h_j) \quad \left(c = \frac{|\text{cov}(h_j)|}{|\text{cov}(h_i)|},\ s = \frac{|\text{cov}(h_j)|}{|C|} \right)$$

Implicit associations: For each conceptual graph *g* implicit in the node h_i, the following association rules are valid:

$$g \Rightarrow desc(h_i) \quad \left(c = 1, s = \frac{|\text{cov}(h_i)|}{|C|} \right)$$

- $\forall h_j \in H$: $desc(h_j) < desc(h_i)$

$$g \Rightarrow desc(h_j) \quad \left(c = \frac{|\text{cov}(h_j)|}{|\text{cov}(h_i)|}, s = \frac{|\text{cov}(h_j)|}{|C|} \right)$$

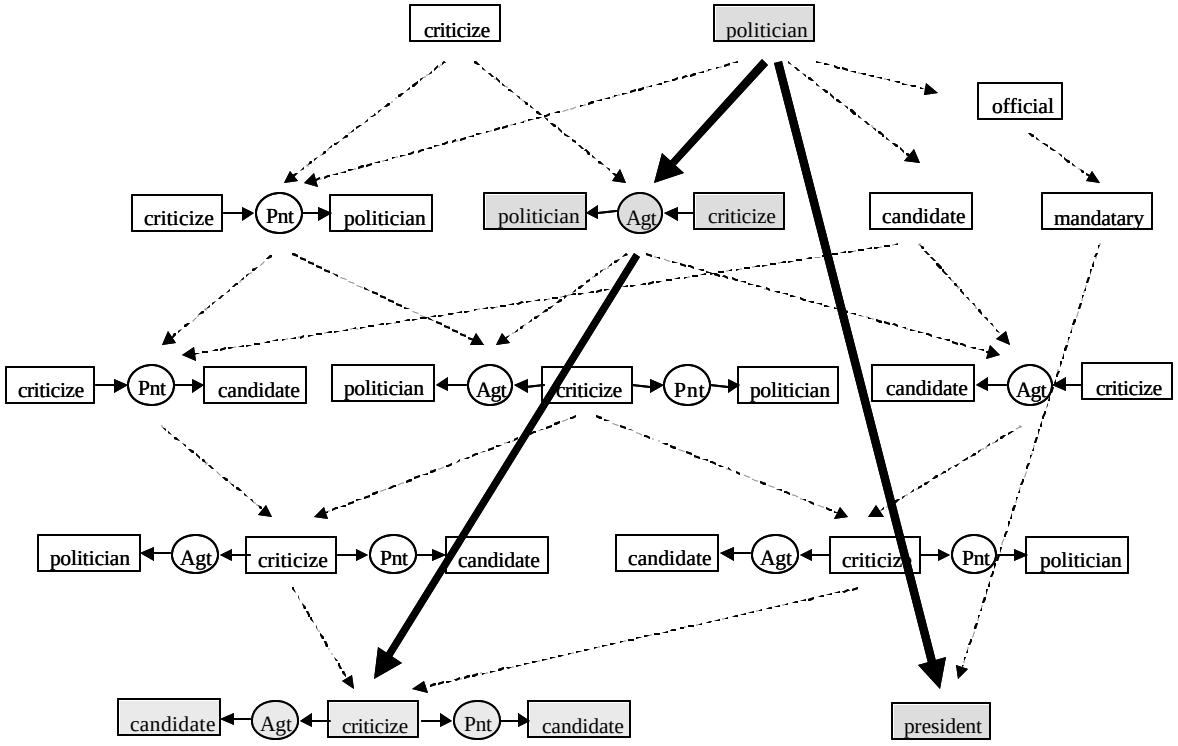

Figure 3. Common generalizations implicit in *H*

- $\forall h_j \in H$: $desc(h_i) < desc(h_j)$ y $g < desc(h_j)$

$$desc(h_j) \Rightarrow g \quad \left(c = \frac{|\mathrm{cov}(h_i)|}{|\mathrm{cov}(h_j)|},\ s = \frac{|\mathrm{cov}(h_i)|}{|C|} \right)$$

On the basis of these definitions, it is possible to discover *all* the association rules in a set of conceptual graphs. Usually, the set of all these associations is too big and has too much *redundant* information. Therefore, the redundant associations must be deleted or, in the best of the cases, never constructed.

Redundant implicit association: The implicit association rule $g_i \Rightarrow g_j\ (a, b)$ is redundant, if and only if, one of the following conditions is satisfied:

- There is another implicit association rule $g_k \Rightarrow g_l\ (a/b)$, such that g_k is a generalization of g_i ($g_i \leq g_k$), and/or g_l is a specialization of g_j ($g_l \leq g_j$).
- There is another implicit association rule $g_i \Rightarrow g_k\ (1/\gamma)$ in combination with the explicit association rule $g_k \Rightarrow g_j\ (a/b)$.

For instance, given the conceptual hierarchy of figure 2, the following list represents all the *not*-redundant association rules. In this list, the rules are ordered by their confidence and support values.[3]

[criticize] Þ [politician]←(agt)←[criticize]	(1/.75)
[criticize]→(pnt)→[politician] Þ [candidate]←(agt)←[criticize]→(pnt)→[candidate]	(1/.5)
[candidate] Þ [candidate]←(agt)←[criticize]→(pnt)→[candidate]	(1/.5)
[official] Þ [president]	(1/.5)
[politician] Þ [politician]←(agt)←[criticize]	(.75/.75)
[politician]←(agt)←[criticize] Þ [candidate]←(agt)←[criticize]→(pnt)→[candidate]	(.66/.5)
[politician] Þ [candidate]←(agt)←[criticize]→(pnt)→[candidate]	(.5/.5)
[politician] Þ [president]	(.5/.5)

The basic algorithm for the discovery of association rules in a conceptual hierarchy of graphs is described below. This algorithm traverses all the hierarchy (using a *bottom-up* approach), and for each node h_i identifies those not-redundant associations with confidence and support

[3] In this list, all associations with a confidence value $c = 1$ are implicit association rules.

greater that the user-specified values *minconf* and *minsup* respectively.

The explicit and implicit associations are constructed apart. The explicit associations are constructed relating the node h_i with each one of it ancestor nodes, while the implicit associations are constructed relating the node h_i with each implicit conceptual graphs g that satisfy the following condition[4]: There is not other graph g' implicit in h_i such that $g < g'$.

```
1  procedure FindAllAssociations (H)
2     for each node h_i ∈ H: |cov(h_i)| == 1
3          // search for all association related with the node h_i
4          AsociationsWithNode (h_i)
5     endfor
4  endprocedure

1  procedure AsociationsWithNode (h_base)
2     h_base ← visited
3      // if it support is sufficiently great then search it associations
4     if |cov(h_base)| / |C| ≥ minsup
5          ImplicitAssociations(h_base)
6          // constructs the explicit associations with the parent nodes
7          for each node h_p ∈ P(h_base)
8               ExplicitAssociation (h_p, h_base)
9     endif
10    // begins recursion, the parent nodes of h_base are analyzed
11    for each node h_p ∈ P(h_base)
12       if node h_p was not visited
13            AsociationsWithNode (h_p)
14    endfor
15 endprocedure

// Defines an explicit association between h_left and h_right. In the hierarchy
// the node h_left is a descendent of the node h_right
1  procedure ExplicitAssociation (h_left, h_right)
2     support     = |cov(h_right)| / |C|
3     confidence = |cov(h_right)| / |cov(h_left)|
4     if confidence ≥ minconf
5          OUT ← "h_left → h_right, confidence, support"
6          // Construct the rules with h_right and the parent nodes of h_left
7          for each node h_p ∈ P(h_left)
8               ExplicitAssociation (h_p, h_right)
9     endif
10 endprocedure

1  procedure ImplicitAssociation (h_base)
2     // computes the confidence and support on the basis of the node h_base
3     confidence = 1
4     support     = |cov(h_base)| / |C|
5     // associations with implicit generalizations without relations
6     for each concept c ∈ h_base
7          g ← c' : c ≤ c' y ∄c_p ∈ P(h_base): c_p ≤ c'
8          OUT ← "g → h_base, confidence, support"
9     endfor
10    // associations with implicit generalizations with one relation
11    for each relation r ∈ h_base
12       if r is not covered by P(h_base)
13            // generalization g is defined as a star graph
14            g ← r
15            for each concept c in the neighborhood of r
16                 g ← maximal generalization of concept c
17            OUT ← "g → h_base, confidence, support"
18       endif
19    endfor
20 endprocedure
```

Conclusions

The discovery of association rules is one of the classic problems of data mining. Typically, it is done over well-structured data, such as databases. In this paper, we presented our first ideas about the discovery of association rules over semi-structured data. Basically, we focused on the *discovery of association rules in a set of conceptual graphs.*

The method we presented bases the discovery of the associations in the existence of a *conceptual clustering* of the set of graphs. It uses this clustering as an index of the collection, and takes advantage of it structure at the moment of searching for an association. Basically, the association rules are identified and constructed by traversing the conceptual hierarchy.

The use of a conceptual clustering is not only important because it makes easy the discovery and construction of the association rules, but also because it allows discovering association rules at *different levels of generalization*.

Currently, we are implementing a system for the discovery of the associations among conceptual graphs. As future work, we plan to test the performance of this system in different conditions and domains.

[4] The other conceptual graphs implicit h_i are not considered because they produce redundant implicit associations.

Acknowledgements

Work done under partial support of CONACyT, CGEPI-IPN, and SNI, Mexico.

References

Agrawal, Imielinski, and Swarni (1993), Mining Association Rules between Set of Items in Large Databases, Proc. Of the ACM SIGMOD Conference of Management of Data, 1993.

Bournaud and Ganascia (1996), Conceptual Clustering of Complex Objects: A Generalization Space based Approach, Lecture Notes in Artificial Intelligence 954, Springer, 1996.

Bournaud and Ganascia (1997), Accounting for Domain Knowledge in the Construction of a Generalization Space, Lectures Notes in AI (1257), Springer-Verlag, 1997.

Ellis and Lehmann (1994), Exploiting the Induced Order on Type-Labeled Graphs for fast Knowledge Retrieval, Lecture Notes in Artificial Intelligence 835, Springer-Verlag 1994.

Genest, and Chein (1997). An Experiment in Document Retrieval Using Conceptual Graphs, Conceptual structures: Fulfilling Peirce´s Dream. Lecture Notes in artificial Intelligence 1257, August 1997.

Godin, Mineau and Missaoui (1995), Incremental Structuring of Knowledge Bases, International KRUSE Symposium, August 11-13, Santa Cruz, California, 1995.

Han and Kamber (2001), Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 2001.

Huibers, Ounis and Chevallet (1996), "Conceptual Graph Aboutness", Lecture Notes in Artificial Intelligence, Springer, 1996.

Mineau and Godin (1995), Automatic Structuring of Knowledge Bases by Conceptual Clustering, IEEE Transactions on Knowledge and Data Engineering, 7(5), 1995.

Montes-y-Gómez, Gelbukh and López-López (2000), Comparison of Conceptual Graphs, Lecture Notes in Artificial Intelligence 1793, Springer 2000.

Montes-y-Gómez, Gelbukh, López-López and Baeza-Yates (2001a), Flexible Comparison of Conceptual Graphs, submitted to DEXA-2001.

Montes-y-Gómez, Gelbukh, López-López and Baeza-Yates (2001b), Text mining with Conceptual Graphs, submitted to SEPNL-2001.

Myaeng and López-López (1992), Conceptual Graph Matching: a Flexible Algorithm and Experiments, Journal of Experimental and Theoretical Artificial Intelligence, Vol. 4, 1992.

Myaeng and Khoo (1994), Linguistic Processing of Text for a Large-Scale Conceptual Information Retrieval System, Lecture Notes in AI 835, Springer-Verlag 1994.

Sowa (1984), Conceptual Structures: Information Processing in Mind and Machine, Addison-Wesley, Reading, M.A., 1984.

Sowa (1999). Knowledge Representation: Logical, Philosophical and Computational Foundations, 1st edition, Thomson Learning, 1999.