

A Frame-Based System for Automatic Classification of Semi-Structured Data

Bernardo Pereira Nunes¹

Marco Antonio Casanova¹

Abstract: The problem of data classification goes back to the definition of taxonomies covering knowledge areas. With the advent of the Web, the amount of data available increased several orders of magnitude, making manual data classification impossible. This work presents a tool to automatically classify semi-structured data, represented by frames, without any previous knowledge about structured classes. The tool uses a variation of the K-Medoid algorithm and organizes a set of frames into classes, structured as a strict hierarchy.

1 Introduction

In this paper, we present a process based on an unsupervised learning technique to automatically classify semi-structured data. The process consists of the following key steps: determine the number of clusters in a data set; find the basic-level category; and refine the class hierarchy. For the last step, we propose three techniques, that we call hybrid, abstract and medoid. Finally, to demonstrate the process, we present an example of the abstract technique.

¹ Departamento de Informática, PUC-Rio, Caixa Postal 38097
{bnunes, casanova}@inf.puc-rio.br}

2 Basic Definitions

Categorization is the process of grouping ideas or objects using some purpose or relationship between them. According to Rosch et al [1], people categorize “things” in term of prototypes. The prototype theory was initially known as the concept of basic-level categorization. However, finding the basic-level categorization depends on the concept in question. For example, the basic-level categorization of concepts such as “furniture” or “animal” could be “chair” and “robin”, respectively. Thus, for each concept approached, the basic-level categorization can be more specific or more general.

Lakoff and Johnson [2] approached this question not just by the objectivist view, i.e., not just by taking into account the inherent properties of objects. Instead, they included interactional properties, such as perceptual properties, motor-activity properties, purposive properties, functional properties, etc. So, non-prototypical objects should be categorized by their relationships with the prototypes or by their similarity with the prototypes.

The *prototype* is defined as the most central object in its category. Each category is structured as a “radial structure”, i.e., some objects are more representative (closer to the prototype) in a category than others (far from the prototype).

In this paper, our strategy adopts the prototype theory to automatically classify semi-structured data using frames. Intuitively, a frame [7] is a data structure defined to represent a concept or a stereotyped situation, such as “being in a certain kind of living room” or “going to a child’s party”. More precisely, a *frame* [3] is a set of slots with distinct names. A *slot* is an expression of the form “ $P:V$ ” or of the form “ $P:$ ”, where P and V , called the *slot name* and the *slot value*, satisfy one of the following conditions:

1. P is an attribute of the entity being described, and V , if defined, is a single value (the attribute is single-valued, by assumption), or
2. P is of the form $R/1$, where R is a binary relationship in which the entity is the first participant, and V , if defined, is a single value or a set of values (the relationship is non-total and multi-valued, by assumption), or
3. P is of the form $R/2$, where R is a binary relationship in which the entity is the second participant, and V , if defined, is a single value or a set of values (the relationship is non-total and multi-valued, by assumption)

The *top frame* is the empty set. An *instance frame* is a frame whose slots are all of the form “ $P:V$ ”, and a *class frame* is a frame with at least one slot of the form “ $P:$ ”.

3 Classification process

The classification process we propose is based on the notions of radial structure and prototype, represented by the most central object in its cluster. The process is based on a variation of the k-Means algorithm [4], called k-Medoid [5], which maintains the radial

structure. In the k-Medoid algorithm, the *medoid* is defined as the most central element in its cluster, and the average dissimilarity to all objects in a cluster is minimal. Thus, we may consider the medoid of a cluster as the most representative object of the cluster or the *prototype*. The closeness criterion used takes into account just the slot names, or slot names and slot values.

To illustrate our process, we describe an example of classification using the abstract technique. Suppose that we start with a set of frames of three different classes (unknown to the algorithm), represented by the following class frames [3]: Person [name:, age:], Employee [name:, age:, works:, area:, salary:], and Student [name:, age:, level:, area:, fee:].

The first step is to determine the number of clusters. We then run the k-Medoid algorithm, varying k , and validating each cluster through the global silhouette width [6]. The silhouette method is based on the index $s(i)$, defined as

$$s(i) = \frac{b(i) - a(i)}{\max(\{a(i), b(i)\})} \quad (1)$$

where $a(i)$ is the average distance between object i and the objects in its cluster A , and $b(i)$ is the average distance between object i and the objects in its “second closest” cluster B . Intuitively, $s(i)$ represents how well matched is the object i in the cluster A . The average $s(i)$ of a cluster is a measure of how tightly grouped all data in the cluster are. Thus, the average $s(i)$, called the *global silhouette width* of the entire data set, is a measure of how appropriately the data has been clustered. It helps determining the number of clusters, denoted by k . We stress that this step will always be executed each time a cluster needs to be split.

The first run of the k-Medoid is considered equivalent to the basic-level categorization. In the example shown in Fig. 1 (step 1), the result of the basic-level categorization is represented by the following three ($k=3$) class frames: Person_C, Employee_F and Student_F. Once the basic-level categorization is found, the process continues and the specialization step, Fig. 1 (step 2), is executed. This step works as follows: for each cluster found on the basic-level categorization, the k-Medoid runs recursively (each cluster is split) until a stop criterion is reached. The stop criterion is a measure of the quality of the generated clusters. In our example, for each of the clusters Person_C, Employee_F and Student_F, the algorithm determined the ideal number of clusters. Hence, we discovered that Employee_F could be split into two clusters and Student_F, into three. The specialization process continues for each new cluster (Employee_E, Employee_A, Student_A, Student_F, Student_M), until no more specializations become necessary, and the process stops. In the example, each cluster was split based on attribute values, i.e., cluster Employee_E has an attribute “area” whose value is “engineering”, whereas the value for the Employee_A is “law”.

The next step is concept generalization, as in Fig.1 (step 3). The algorithm works in two steps. The first step is to generalize the concepts above the basic-level categorization, as

shown in Fig.1 (step 2), the algorithm *merges* the two closest frames and creates a new frame, called “Abstract”. This is repeated until no frames are close enough to be merged, or there is just one frame on the top.

Given two frames, F and G , we define their *merge* [3], denoted “ $F \Delta G$ ”, as the frame M such that a slot $s \in M$ iff $s \in F$ and there is $g \in G$ such that g subsumes s , or $s \in G$ and there is $f \in F$ such that f subsumes s .

Briefly, the algorithm first merges the closest frames (Employee_F and Student_F) into Abstract_2. Then, Person_C is merged with Abstract_2, creating Abstract_1. The process continues and generalizes all concepts found under the basic-level categorization. The abstract strategy uses the frames of the medoids to create the new representative frame. The current medoid is reallocated with the others frames on the bottom of the hierarchy. For example, Abstract_3 was created by merging Employee_E with Employee_A, the medoid Employee_F was replaced by Abstract_3, and reallocated under the clusters Employee_E or Employee_A. Likewise, Abstract_4 resulted by merging Student_A with Student_F, and then with Student_M.

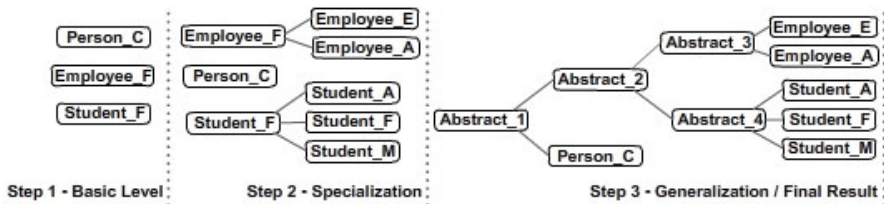


Fig. 1 Classification steps of the Abstract technique.

The final classification is shown in Fig.1 (step 3), and their categories are defined as Abstract(1) [name:, age:], Abstract(2) [name:, age:, area:], Abstract(3) [name:, age:, works:, area:, salary:] and Abstract(4) [name:, age:, level:, area:, fee:]. In this case, we do not have a class frame with a value determining a type of a class. Indeed, observing Fig. 1, the leaves are represented by frames from the data set, and intermediate nodes are represented by abstract frames. Each leaf represents a subset of the original set of objects, in the sense that its frame is the medoid of the subset.

4 Conclusions

We presented a process based on an unsupervised learning technique to automatically classify semi-structured data. To refine the class hierarchy, we proposed three techniques but, due to space limitations, we illustrated here just one of them.

The example of Section 3 used synthetic data set to demonstrate the efficacy of the process. However, we also tested the process with real data and compared the result with the

original (manual) classification, with success. We refer the reader to <http://www.neovisual.com.br/automaticClassification/> for the details.

References

1. Rosch, E.; Mervis, C.B.; Gray, W.; Johnson, D.; Boyes-Braem, P. (1976) "Basic Objects in Natural Categories", *Cognitive Psychology*, Vol.8, No.3 (July 1976).
2. Lakoff, G.; Mark, J. (1980) *Metaphors We Live By*. Univ. Chicago Press.
3. Barbosa, S. D. J.; Breitman, K. K.; Furtado, A. L.; Casanova, M. A. (2007) "Similarity and Analogy over Application Domains". In: *Anais do XXII Simposio Brasileiro de Banco de Dados*. pp.238-254.
4. MacQueen, J. B. (1967): "Some Methods for classification and Analysis of Multivariate Observations", *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, University of California Press, 1:281-297.
5. Han, J.; Kamber, M. (2001) "Cluster Analysis". In: Morgan Kaufmann Publishers (eds.), *Data Mining: Concepts and Techniques*, 1 ed., chapter 8, New York, USA, Academic Press.
6. Rousseeuw, P. (1987) "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis". *Computational and Applied Mathematics*, 20.
7. Kaufman, L.; Rousseeuw, P. (1990) *Finding Groups in Data*. Wiley, New York, NY.
8. Minsky, M. (1975) "A Framework for Representing Knowledge". In: *The Psychology of Computer Vision*, P. Winston (Ed.), McGraw-Hill.