Concept lattice classifier : a first step towards an iterative process of recognition of noised graphic objects

Stéphanie Guillas, Karell Bertet, and Jean-Marc Ogier

L3I laboratory, Université de La Rochelle
av M. Crépeau, 17042 La Rochelle Cedex 1, France
{sguillas, kbertet, jmogier}@univ-lr.fr

Abstract. In this paper, we propose a generic description of the concept lattice as classifier in an iterative recognition process. The experimentation is realized on the noised symbols of GREC database [6]. Our experimentation presents a comparison with the two classical numerical classifiers that are the bayesian classifier and the nearest neighbors classifier and some comparison elements for an iterative process.

1 Introduction

The work presented in this paper takes place in the field of automatic retro-conversion of technical documents [16] and proposes to use concept lattice to recognize graphic objects, and more precisely to classify noised symbols images of GREC database [6]. This graph issued from Formal Concept Analysis (a theory of data analysis) [18], has often been used in data mining [11]. A recent study [10] gives a comparison of several supervised classification methods based on concept lattice, and clearly shows the interest of its use in classification.

In study [7], we showed that concept lattice has a structure which looks like the decision tree, and that its bigger size gives more robustness to the noise than the decision tree. We also highlighted the recognition parameters and use the concept lattice as a classifier in a one-step process.

Here, we present a description of an iterative process (Fig. 1), where we repeat the recognition process with selection of new attributes (or characteristics) in the signatures at each iteration. In the field of symbols recognition, an iterative process is attractive because various techniques (structural, statistical) enable to extract new data from images. In our process, discretization and in particular selection of attributes are necessary to reduce the context size. Otherwise, we have chosen to build the concept lattice because we need the graph to navigate and to progressively validate attributes to classify the noised data.

Recognition process (Fig. 1) is usually composed of the *learning* stage and the *classification* stage (section 2). In part 2.1, we describe the data learning where data are discretized and the concept lattice is built. Classification and especially navigation in the concept lattice is described in part 2.2. Part 3 proposes a comparison in cross-validation with the bayesian classifier and the nearest neighbors classifier and at last, conclusion and extensions are presented in part 4.

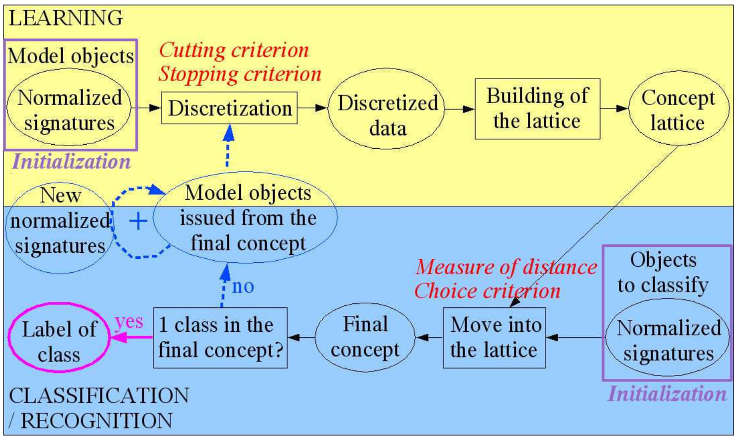

Fig. 1. The iterative recognition stages

2 Process

The iterative recognition process follows a coarse-to-fine strategy with selection of new attributes at each iteration. The recognition process (Fig. 1) is composed of: learning and classification. We first have a set of *model* objects (classes are known) and a set of objects to classify. Classification aims to attribute a class label to each object. After each iteration, we propose a *final concept* (defined in part 2.2) which contains one or several classes. When it contains only one class, the process is finished, otherwise, the signatures don't discriminate enough the classes, and another selection of attributes is needed to determine the class label.

2.1 Learning

In the general case, the learning stage consists in organizing a concept lattice data issued from a set of objects. In our case, objects are graphic images described by equal size normalized numerical signatures: [15, 14]. Learning stage (Fig. 1) is composed of: *discretization of data* and *building of the lattice*.

Discretization Discretization [4, 3, 13] consists in organizing the signatures $p = (p_i)_{i \leq n}$ issued from the objects set O, in intervals, that characterize each class of objects. At each step of discretization, an interval is selected to be cut. This selection depends on a *cutting criterion*, and the cutting process is repeated until a *stopping criterion* is validated. In study [7], we selected the maximal distance as non supervised criterion and the Hotelling's coefficient as supervised criterion.

Here are some stopping criteria: $crit_{class\ separated}$ is *"to stop when classes are separated"*; $crit_{nb\ steps}$ is *"to stop when the discretization steps number equals a constant nb"*; $crit_{nb\ classes\ max}$ means that the final concept contains at most nb classes; and $crit_{cutting\ min}$ limits the cutting criterion above a minimal value.

When discretization is performed, objects $p \in O$ are characterized by intervals $I = I_1 \times I_2 \times \ldots \times I_n$ with I_i the intervals set of each attribute $i = 1 \ldots n$, and the membership relation $\mathcal{R}$ between objects and intervals can be deduced.

Building of the concept lattice Building of the lattice immediately follows the discretization stage and is totally determined by the membership relation $\mathcal{R}$ between objects and intervals without criterion or parameter.

A concept lattice is composed of a set of *concepts* ordered by inclusion, which forms a graph (that has the lattice properties [1]). We associate to a set of objects $A \subseteq O$, the set $f(A)$ of intervals in relation $\mathcal{R}$ with A: $f(A) = \{x \in I \mid p\mathcal{R}x \; \forall \, p \in A\}$. Dually, for a set of intervals $B \subseteq I$, we define the set $g(B)$ of objects in relation $\mathcal{R}$ with B: $g(B) = \{p \in O \mid p\mathcal{R}x \; \forall \, x \in B\}$.

A *formal concept* is a pair objects-intervals (A, B) with $A \subseteq O$, $B \subseteq I$, $f(A) = B$ and $g(B) = A$. Two concepts (A_1, B_1) and (A_2, B_2) are in relation in the concept lattice if they verify the inclusion property: $(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow A_1 \supseteq A_2$ equivalent to $B_1 \subseteq B_2$. Let $\prec$ be the transitive reduction associated to $\leq$. The minimal concept $(O, f(O))$ according to the relation $\leq$ contains the whole objects O and the set $f(O)$. Note that $f(O) = \emptyset$ when intervals shared by all the objects are removed. Dually, the maximal concept is $(g(I), I)$. For more information about Galois connection and concept lattice, see [1].

Several algorithms can generate the concept lattice: Bordat [2], Ganter [5], Valtchev et al. [17] and Nourine and Raynaud [12] which has the best theoretic complexity (quadratic complexity by element of the produced lattice). The main limit of concept lattice is its cost in time and space. Indeed, its size is bounded by $2^{|S|}$ in the worst case, and by $|S|$ in the best case. The main advantage of this graph is its good readability because it is easy to interpret.

2.2 Classification

Concept lattice can be seen as a search space in which we move by validation of the intervals issued from the discretization stage. During the classification, the signature $s = (s_1, \ldots, s_n)$ of the object to recognize is introduced in the concept lattice starting from the *minimal concept*: $(O, f(O))$ meaning that the whole classes of objects are *candidates* to recognition and no interval is validated. We progress step by step in the Hasse diagram of the concept lattice by validation of new intervals and consequently by reduction of the objects set and their corresponding classes, until we reach a *final concept*.

A concept is a *final concept* when it is the last concept in the classification progress containing objects of some class. A final concept (A, B) corresponds to the sup-irreducibles of the lattice. (see [1]) and is characterized by:

$$|GetClasses((A,B))|! = \sum_{(A',B') \succ (A,B)} |GetClasses((A',B'))|$$

From a current concept, an elementary step of classification consists in selecting an interval from a set of intervals S, to progress toward a new concept. More precisely, S is a family of intervals obtained from the n immediate successors $(A_1, B_1), \ldots, (A_n, B_n)$ of the current concept (A, B) and defined by: $S = \bigcup_{i=1}^{n} B_i \backslash B = \{X_1, \ldots, X_n\}$. Thus, the *choice criterion* parameter consists in *choosing a subset* X_i *of intervals among* S using a distance measure d.

In our experiments, symbols are noised and thus values of their signature can be modified. To make supple the boundaries of intervals we can introduce the fuzzy theory. Then, the distance measure would be $d(s_i, x) = \mu_A(x)$, with μ the *likelihood degree* of the assertion $x \in A$, and A a fuzzy set.

3 Experimental results

Our previous work [7] showed that concept lattice is more appropriated to the classification of noised graphic objects than the decision tree. Otherwise, experimental results showed that the Radon signature [14], the Hotelling's cutting criterion seem to be the most appropriate and are used in these new tests.

3.1 Tests with separation of classes

In this experiment, concept lattice is compared to bayesian classifier and nearest neighbors classifier (k-NN). For the concept lattice, we use $crit_{class\ separated}$ as stopping criterion, so one iteration is required to obtain a label of class. Our data consist of 2 sets of 10 classes of symbols of GREC2003 [6] (namely cl1-10 and cl11-20), where each class contains 1 "model" symbol and 90 symbols (Fig. 2 (left)) noised by the Kanungo method [8]. We use another data set composed of 25 classes (namely cl1-25) of GREC2005 database (Fig. 2 (right)). This symbols set is more noised than those of GREC2003, and is composed of 175 symbols.

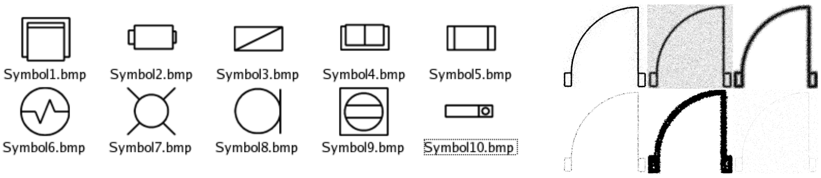

Fig. 2. 10 examples of "model" symbols of GREC2003 database (left) and 6 examples of noised symbols of GREC2005 database (right)

We test the 3 classifiers by the cross-validation technique [9]. Symbols are partitioned in n blocks of equal size. Each block is used as a learning set, and the other blocks are tested. The test result is the average of the n recognition rates. On GREC2003 symbols, we try: 5 blocks of 182 symbols (test 1), 10 blocks of 91 symbols (test 2) and 26 blocks of 35 symbols (test 3). On GREC2005 symbols, we try 5 blocks of 35 symbols (test 4). Recognition rates are shown in Figure 3.

For test 4, results are really low due to the high level of noise. From these results, we deduced that k-NN classifier gives the best rates, and bayesian classifier gives better rates than the concept lattice only when the size of the learning set is important (tests 1 and 2). Notice that concept lattice only needs between 6 and 15 attributes of the Radon signature among the 50 values, on the contrary to the bayesian and the k-NN classifiers. The relatively good results of these tests indicate that an iterative process is an interesting way to explore.

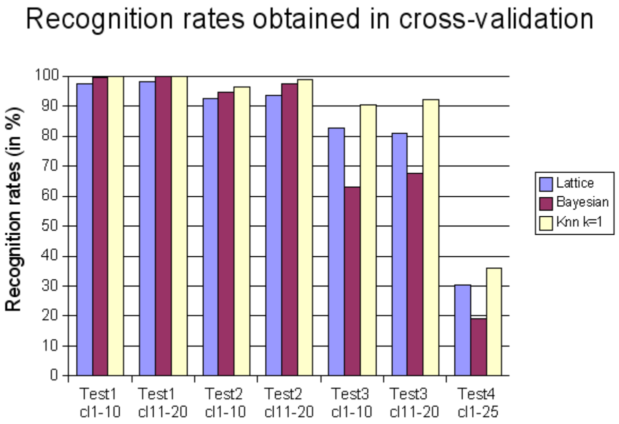

Fig. 3. Results of cross-validation for the 3 classifiers

3.2 Tests without separation of classes

In order to set up an iterative process, we need to define a stopping criterion of discretization. We evaluate the potential of recognition in a one iteration process for the following stopping criteria: $crit_{class\ separated}$, $crit_{nb\ steps}$ with nb $=5$ or 10 and $crit_{cutting\ min}$ with Hotelling's cutting criterion $> 0, 5$. These tests are performed on symbols of GREC2005 (presented in test 4).

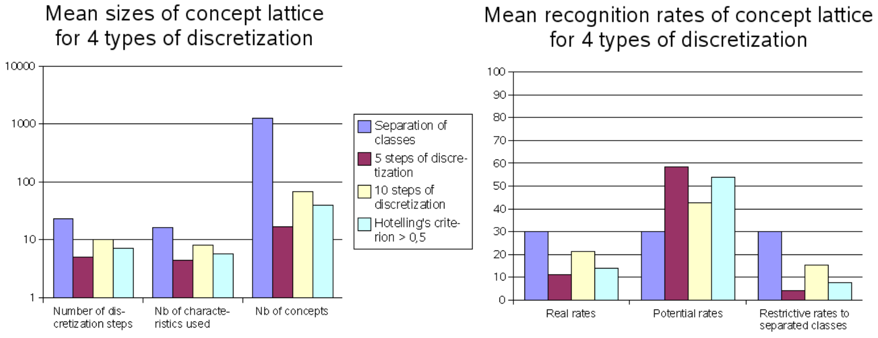

Fig. 4. Mean results at first iteration of recognition process on symbols of GREC2005

Figure 4 shows the mean number of: discretization steps, attributes used, and concepts (left); and 3 mean rates: real rates, potential rates and restrictive rates to separated classes (right) described below. Notice that the scale is a logarithmic one (left graphic). Let r_s be a boolean where $r_s = 1$ when the class of symbol s is included in the final concept (A, B) chosen by the process; and $r_s = 0$ otherwise. Notice that these 3 rates are equal when the stopping criterion is $crit_{class\ separated}$. These 3 rates can be formally defined by:

- Real rate $= \frac{1}{symbols\ nb} \sum_s (\frac{r_s}{|GetClasses((A,B))|})$
- Potential rate $= \frac{1}{symbols\ nb} \sum_s r_s$
- Restrictive rate $= \frac{1}{symbols\ nb} \sum_s \{r_s \text{ such that } |GetClasses((A, B))| = 1\}$

These results show that with only one iteration of recognition, we obtain real rates near to those with $crit_{class\ separated}$. Moreover, even if the iterative process requires the construction of several concept lattices, their size (i.e. number of concepts) is really lower than the concept lattice built in case of $crit_{class\ separated}$ (see Fig. 4). In conclusion, when correctly adjusted, a stopping criterion can give a nice compromise between recognition rates and size of the lattice.

4 Conclusion

The experimentations show that concept lattice gives relatively close recognition rates than the famous k-NN classifier. Otherwise, the iterative recognition approach described here is interesting to handle big sets of classes, what was relatively costly, and the first results are promising. Moreover, this iterative system could be useful when classes are few separable. Indeed, we could inject a more discriminating signature to characterize these classes. We would like to manage a new stopping criterion of discretization combination of the proposed criteria together with the maximal number of classes in the final concepts. Indeed, this criterion could be useful to have a better control of the discretization.

References

1. M. Barbut and B. Monjardet. *Ordre et classification, Algèbre et combinatoire.*
2. J. Bordat. Calcul pratique du treillis de Galois d'une correspondance. *Math. Sci. Hum.*, 96:31–47, 1986.
3. J. Dougherty, R. Kohavi, and M. Sahami. *Supervised and unsupervised discretization of continuous features.* Morgan Kaufman, 1995.
4. U. M. Fayyad and K. B. Irani. *Multi-interval discretization of continuous-valued attributes for classification learning.* Morgan Kaufman, 1993.
5. B. Ganter. Two basic algorithms in concept analysis. *Technische Hochschule Darmstadt (Preprint 831)*, 1984.
6. GREC. www.cvc.uab.es/grec2003/symreccontest/index.htm.
7. S. Guillas, K. Bertet, and J.-M. Ogier. A generic description of the concept lattices' classifier : application to symbol recognition, revised and extended version of paper first presented at sixth iapr international workshop on graphics recognition (grec'05), hong kong, china, august 2005. In *GREC*, Lecture Notes in Computer Science. Springer Verlag, 2006.

8. T. Kanungo and al. Document degradation models: parameter estimation and model validation. In *IAPR Workshop on machine vision applications, Kawasaki (Japan)*, pages 552–557, 1994.
9. D. Krus and E. Fuller. Computer assisted multicrossvalidation in regression analysis. *Educational and Psychological Measurement*, 42:187–193, 1982.
10. E. Mephu NGuifo and P. Njiwoua. Treillis des concepts et classification supervisée. In *Technique et Science Informatiques (à paraître), RSTI).* Hermès - Lavoisier, Paris, France, 2005.
11. I. Nafkha, S. Elloumi, and A. Jaoua. Conceptual information retrieval based on cooperative conceptual data reduction. *Information and communication technologies: from theory to applications*, pages 547–553, 2004.
12. L. Nourine and O. Raynaud. A fast algorithm for building lattices. In *Third International Conference on Orders, Algorithms and Applications*, Montpellier, France, august 1999.
13. R. Rakotomalala. *Graphes d'induction.* PhD thesis, Université Claude Bernard, Lyon I, Décembre 1997.
14. S. Tabbone and L. Wendling. Recherche d'images par le contenu l'aide de la transforme de radon. *Technique et Science Informatiques*, 2003.
15. M. Teague. Image analysis via the general theory of moments. *Journal of Optical Society of America (JOSA)*, 70:920–930, 2003.
16. K. Tombre and B. Lamiroy. Graphics recognition - from re-engineering to retrieval. *Proceedings of 7th ICDAR, Edinburgh (Scotland, UK)*, pages 148–155, August 2003.
17. P. Valtchev, R. Missaoui, and P. Lebrun. A partition-based approach towards constructing galois (concept) lattices. *Discrete Mathematics*, 3(256):801–829, 2002.
18. R. Wille. Restructuring lattice theory: an approach based on hierarchy on contexts. *Ordered sets*, pages 445–470, 1982.