Applied Artificial Intelligence, 21:317–342

ISSN: 0883-9514 print/1087-6545 online
DOI: 10.1080/08839510701252551

Taylor & Francis
Taylor & Francis Group

RELATIONAL DATA MINING AND ILP FOR DOCUMENT IMAGE UNDERSTANDING

Michelangelo Ceci, Margherita Berardi, and Donato Malerba □ *Dipartimento di Informatica, Universitàdegli Studi di Bari, Bari, Italy*

□ *Document image understanding denotes the recognition of semantically relevant components in the layout extracted from a document image. This recognition process is based on domain-specific knowledge that can be acquired automatically by applying data mining techniques. The spatial dimension of page layout makes classification methods developed in inductive logic programming (ILP) and multi-relational data mining (MRDM) the most suitable candidates for this specific task. In this paper, both approaches are considered and empirically compared on three different data sets consisting of multi-page articles published in an international journal and historical documents. The ILP method is able to learn recursive logical theories that express dependencies between logical components, while the MRDM method extends the naïve Bayesian classifier to data stored in multiple tables of a relational database. Experimental results confirm the importance of the spatial dimension for this application and show that the ILP method tends to be conservative with a high (low) percentage of omission (commission) errors, while the probabilistic nature of the MRDM method allows us to tradeoff between the two types of error.*

Document image analysis is the subfield of digital image processing that aims to convert document images to symbolic form for modification, storage, retrieval, reuse, and transmission (Nagy 2000). This conversion is a complex process articulated into several stages. Initial processing steps include binarization, skew detection, noise filtering, and segmentation. Then document image is decomposed into several constituent items which represent coherent components of the documents (e.g., text lines or halftone images), without any knowledge of the specific format. This layout analysis step precedes the interpretation or understanding of document images, whose aim is that of recognizing semantically relevant layout components (e.g., title and abstract) as well as extracting abstract relationships between layout components (e.g., reading order).

Address correspondence to Donato Malerba, Dipartimento di Informatica, Via Orabona 4, 70126 Bari, Italy. E-mail: malerba@di.uniba.it

Domain-specific knowledge appears essential for document image understanding: In the literature, there are no examples of attempts to develop a system that can interpret arbitrary documents (Nagy 2000). The importance of knowledge representation and acquisition methods in the interpretation of document images has led some distinguished researchers to claim that document image understanding should be considered a branch of artificial intelligence (Tang et al. 1994). In many applications presented in the literature, a great effort is made to hand-code the necessary knowledge according to some formalism, such as block grammars (Nagy et al. 1992), geometric trees (Dengel et al. 1992), and frames (Wenzel and Maus 2001). However, hand-coding domain knowledge is time-consuming and limits the application of document analysis systems to predefined classes of documents.

To alleviate the burden of developing and customizing document analysis systems, data mining methods can be profitably applied to extract the required domain-specific knowledge. For instance, the discovery of association rules on document layout structures can help to extract spatial relationships between logical components (Berardi et al. 2003a), while hierarchical clustering can be used to recognize and interpret tables in documents (Hu et al. 2001). In this paper we investigate the induction of classifiers that can be used to automatically recognize semantically relevant layout components. Classifiers are constructed from a set of training documents whose layout structures have already been interpreted by the users and described according to some representation formalism. Therefore, the customization of a document analysis system for a specific class of documents can be performed by extracting the layout structures of training documents, by manually annotating them in order to specify the semantically relevant layout components (logical structures), and then by automatically inducing a set of classifiers to be operationally used on a set of new documents. In this way, human intervention is limited to annotating layout structures.

In the literature, several methods have been proposed for the construction of classifiers to be used in document image understanding. A brief review is reported in the next section. Most of them assume that training data are represented in a single table of a relational database, such that each row (or tuple) represents an independent example (a layout component) and columns correspond to properties of the example (e.g., height of the layout component). This single-table assumption, however, is too strong for at least three reasons. First, layout components cannot be realistically considered independent observations, because their spatial arrangement is mutually constrained by formating rules typically used in document editing. Second, spatial relationships between a layout component and a variable number of other components in its neighborhood cannot be properly represented by a fixed number of attributes in a table. Even more

so, the representation of properties of the other components in the neighborhood, because different layout components may have different properties (e.g., the property "brightness" is appropriate for half-tone images, but not for textual components). Third, logical components, that is, the components of the logical structures, may be related to each other as well. For instance, the logical components "title" and "author" of a printed paper are often interrelated sequentially (the author follows the title). Since the single-table assumption limits the representation of relationships (spatial or non) between examples, it also prevents the discovery of this kind of pattern, which can be very useful in document image understanding.

All these issues are ultimately due to the fact that document layout structures are a kind of spatial data and, as such, they are subject to spatial autocorrelation.[1] As pointed out by Malerba and Lisi (2001), methods investigated both in inductive logic programming (ILP) (Muggleton 1992; De Raedt 1992; Nienhuys-Cheng and de Wolf 1997) and in multi-relational data mining (MRDM) (Džeroski and Lavrač 2001) are the most suitable for spatial data, since they allow spatial relations between layout components to be effectively and naturally represented. Indeed, ILP uses computational logic as the representation formalism of both training observations and induced hypotheses. Therefore, n-ary predicates ($n > 1$) can be used to represent spatial relationships; training data are stored as Prolog programs and the induced classifier corresponds to a logical theory. On the contrary, MRDM approaches operate on data distributed in a set of tables (not a single one) and look for *relational patterns* that involve multiple tables from a relational database. Spatial relationships can be easily represented by means of a table and some particular integrity constraints named foreign keys. As pointed out by Knobbe et al. (1999), MRDM differs from ILP in three aspects: 1) it is restricted to the discovery of non-recursive patterns; 2) the semantic information in the database is exploited explicitly; and 3) the emphasis on database primitives ensures efficiency.

The limits of some methods for document image understanding and the recent developments in the field of MRDM motivate this work, whose main scope is that of evaluating and systematically comparing two distinct approaches to classifier construction for document image understanding, namely, a logical approach based on theoretical advances in the field of ILP and a statistical approach based on concepts and principles typical of MRDM. The systems that implement the methods developed according to these two distinct approaches are ATRE (Malerba 2003) and Mr-SBC (Ceci et al. 2003). The former is an ILP system that can autonomously discover concept dependencies and recursive theories and for this reason it is able to deal with autocorrelation on spatially lagged response and explanatory variables. The latter can only deal with spatially lagged explanatory variables, but presents some complementary advantages, such as greater

efficiency and the computation of a degree of confidence (a posterior probability) in the predicted class, which convey information on the potential uncertainty in classification.

In order to compare the two approaches, both ATRE and Mr-SBC have been integrated in the document analysis system WISDOM++ (http://www.di.uniba.it/~malerba/wisdom++) (Altamura et al. 2001), whose applicability has been investigated in the context of the IST project COLLATE (http://www.collate.de/). WISDOM++ permits the transformation of document images into XML format by means of several complex steps: 1) preprocessing of the raster image of a scanned paper document; 2) segmentation of the preprocessed raster image into basic layout components; 3) classification of basic layout components according to the type of content (e.g., text, graphics, etc.); 4) identification of a more abstract representation of the document layout (layout analysis); 5) classification of the document on the basis of its layout and content; 6) identification of semantically relevant layout components (document image understanding); 7) application of OCR only to those textual components of interest; and 8) storage in a relational database and generation of a document in XML format that conveys all information extracted in previous steps. In the WISDOM++ context, document image understanding is limited to mapping the layout structure of a document into the corresponding logical structure, that is, abstract relationships between layout components are not extracted. The mapping can be performed by means of a set of classification rules, hence the need of automatically inducing some classifiers from data representing the layout of a set of training documents.

RELATED WORK

In the literature there are already several works on document image understanding. Akindele and Belaïd (1995) proposed to apply the R-XY-Cuts method on training data in order to extract the layout structures to match against an initial model defined by an expert. The aim of the matching is to discard training documents whose layout structure is very different from the expected one. Then, a generic model of the logical structure is built by means of a tree-grammar inference method applied to validated layout structures with associated labels. Therefore, this approach is based on a demanding human intervention, which is not limited to layout labeling but also involves the specification of an initial model.

Walischewski (1997) proposes to represent each document layout by a complete attributed directed graph with one vertex for each layout object. The vertex attributes are pairs (l, c), where l denotes the type of layout component while c denotes the logic label of the layout object. Edges have thirteen attributes corresponding to Allen's qualitative relations on intervals

(Allen 1983). An attribute of the edge (v_i, v_j) is a pair (h, v) describing qualitatively the relative horizontal/vertical location between the two vertices v_i and v_j. The learning algorithm returns triples $[(c_i, c_j), (h, v), (w_h, w_v)]$ stating that Allen's relation h (v) holds between c_i and c_j along the horizontal (vertical) axis with strength w_h (w_v). All together, the triples define an attributed directed graph representing the model. Recognition is based on an error tolerant subgraph isomorphism between the graphs representing the document and the model. This approach, although relational, only handles qualitative information and has been tested on simple layout structures.

In the work by Palmero and Dimitriadis (1999) a document is viewed as a sequence of objects, whose labels depend both on the geometrical properties of the block (size, position, etc.) and on the decisions taken for previous sequence items. As in the work by Walischewski, there is an implicit recognition of the importance of considering autocorrelation on logical labels, although the original bidimensional spatial autocorrelation boils down to one-dimensional temporal autocorrelation, which is handled by a recursive neuro-fuzzy learning algorithm. The effect of sequence ordering on blocks is not examined.

Probabilistic relaxation (Rosenfeld et al. 1976) deals with autocorrelation by first classifying objects on the basis of their own properties and then by iteratively adjusting assigned labels by referring to their neighbors. Le Bourgeois et al. (2001) tested this approach on blocks delimiting words and compared it to naïve Bayesian classification by taking into account both word features and features of neighboring words. Experimental results are in favor of the naïve Bayesian classifier, thus justifying its extension to more complex situations in which blocks can be spatially related to many other blocks.

Aiello et al. (2002) applied decision tree learning to textual logical components. Seven attributes are considered: two for geometrical properties of the block (aspect ratio, area ratio), four for the textual content (font size ratio, font style, number of characters, number of lines), and one for spatial closeness to a figure. Experiments show that these features are enough to learn accurate trees for some logical components, but results refer to the ideal situation of ground truth data for layout structures and textual content. Moreover, the recognition of logical components is based on their textual content, which means that document image understanding is based on OCR results and not viceversa as in WISDOM++.

Interestingly, none of the studies reported here relies on logic-based approaches to layout representation and learning, although the representation of spatial properties is very natural in first-order logic. The first attempt to apply logic-based learning methods to document image understanding is reported in Esposito et al. (1994). The document analysis system PLRS processes document images in order to extract their layout components that are later classified by means of a set of rules. These rules are automatically

induced from a set of training documents whose layout structures are described by means of a first-order logic formalism. Although experiments reported in the paper are limited to few documents, the problem of learning rules that express dependencies between logical components is clearly identified and three experimental settings are compared: 1) learning under the independence assumption; 2) learning with a user-defined dependency graph; and 3) learning with a dependency order derived through statistical techniques.

Subsequent works have led to the development of a new document analysis system, named WISDOM++, with several new features, such as full integration of all modules, graphical user interface for different classes of users, knowledge-based layout analysis, facilities for layout-analysis correction, management of multi-page documents, processing of document images of various size and resolution, color-based image segmentation, integration with commercial OCR and text reading facilities, document storage on database (Oracle 9i™ and MS Access™), generation of documents in XML format, and graphical rendering on Web browsers (Altamura et al. 2001; Malerba et al. 2003; Altamura et al. 2007). From the viewpoint of data mining, several new problems have been identified, namely, blocks classification (Altamura et al. 1999), layout analysis correction (Berardi et al. 2003b), discovery of association rules (Berardi et al. 2003a), and content-based classification of semantic components (Berardi et al. 2005). In this paper, the problem of interpreting document images is reconsidered in the light of recent developments of ILP and MRDM, which allows us to handle numeric attributes and relations in first-order rule learning, to automatically discover dependencies between logical components, to deal with efficiency issues through caching strategies, to exploit the semantic information in the database, as well as to design a probabilistic classifier for multi-relational data according to the naïve Bayesian framework. In addition, we consider the new challenging applications of WISDOM++ to collections of historical document images, which are characterized by a low-quality layout due to the presence of frames, stamps, signatures, ink specks, and manual annotations that overlap those layout components involved in the understanding or annotation processes.

ILP APPROACH TO DOCUMENT IMAGE UNDERSTANDING

Supervised concept learning has long been a principal area of machine learning research. A supervised concept learning system is supplied with information about several entities whose class (or concept) membership is known and produce from this a characterization of each class. In ILP, concepts to be learned are represented by means of predicate symbols and the result of the learning process is a logical theory. In particular, the ILP system ATRE embedded in WISDOM++ solves the following problem.

Given:

- a set of *target* concepts $C_1, C_2, \ldots, C_r$ to be learned
- a set of *observations* O described in a language L_O
- a *background* (or *domain*) *knowledge* BK expressed in a language L_{BK}
- a *language of hypotheses* L_H
- a user's *preference criterion* PC

Find

a (possibly recursive) *logical theory* T for the concepts $C_1, C_2, \ldots, C_r$, such that T is *complete* and *consistent* with respect to O and satisfies the preference criterion PC.

In the document understanding domain, observations correspond to descriptions of the layouts extracted from document images. Properties or attributes of a layout component are represented by unary descriptors (e.g., *height(X)*), while spatial relations between two layout components are represented by binary descriptors (e.g., *on_top(X,Y)*). The complete list of descriptors used in this work is reported in Table 1. Some additional

TABLE 1 Attributes and Relations Used to Describe the Layout Extracted from a Document Image

Type	Name/arguments	Description	Values
Containment	part_of/2	True if a layout component is part of a page	Boolean
Page position	page/1	Position of the page in the document	{first, intermediate, last_but_one, last}
Locational	x_pos_centre/1	Position of the centroid of the layout component w.r.t. the x axis	Natural
	y_pos_centre/1	Position of the centroid of the layout component w.r.t. the y axis	Natural
Geometrical	height	The height in pixels of a logical component	Natural
	width	The width in pixels of a logical component	Natural
Topological	on_top/2	True if a block is on top/above another block	Boolean
	to_right/2	True if a block is to the right of another	Boolean
	alignment/2	Defines the type of vertical (col) or horizontal (row) alignment between two layout components	{right_col, left_col, middle_col, both_columns, middle_row, lower_row, upper_row, both_rows}
Content	type_of	The content type of a logical component	{image, text, horizontal line, vertical line, graphic, mixed}

predicates (e.g., *author*) are used for the logical labels and target concepts correspond to true values (e.g., *author(X) = true*). This means that no clause in T is generated for false predicates.

The logical theory T is a set of definite clauses[2] (Lloyd 1987) such as:

$$author(X1) \leftarrow alignment(X1, X2) = middle.col, abstract(X2), height(X1) \in [7..13],$$

which can be easily interpreted as follows: If a layout component (*X1*) whose height is between 7 and 13 is above and centrally aligned with another layout component (*X2*) labeled as "abstract," then it can be classified as "author." This clause exemplifies the concept of dependency between two logical components (author and abstract).

An observation is represented by means of a ground multiple-head clause, called *object*, whose body describes the layout of a page while its head describes the logical labels associated to layout components. All literals in the head of the clause are *examples* (either positive or negative) of the concepts $C_1, C_2, \ldots, C_r$, and multiple labeling of a layout component is allowed. An instance of object is reported in the following:

```
class(1) = tpami,  affiliation(2) = false,...,  paragraph(2) =
false,  title(3) = true,...,  table(3) = false,...,  affiliation(15) =
false,..., references(15) = false, paragraph(15) = true
  ← page(1) = first,
  part_of(1, 2),..., part_of(1, 13),
  width(2) = 391,..., width(13) = 263,
  height(2) = 9,..., height(13) = 58,
  type_of(2) = text,..., type_of(13) = image,
  x_pos_centre(2) = 354,..., x_pos_centre(13) = 411,
  y_pos_centre(2) = 29,..., y_pos_centre(13) = 753,
  on_top(2, 4),..., on_top(12, 13),
  to_right(11, 12), ..., to_right(3, 6),
  alignment(3, 8) = only_left_col,...,
  alignment(8,10) = only_upper_row.
```

where the constant 1 denotes the whole page, while the constants 2, 3,...,15 denote the layout components. The descriptor *class(X)* in the head is reported for the sake of completeness and denotes the class of the document, namely "paper published in *IEEE Trans. on Pattern Analysis and Machine Intelligence*." It is considered only in the document classification step.

In this application domain, the following background knowledge has been defined:

$$at\_page(X) = first \leftarrow part\_of(Y, X),\ page(Y) = first$$
$$at\_page(X) = intermediate \leftarrow part\_of(Y, X),\ page(Y) = intermediate$$

$$at\_page(X) = last\_but\_one \leftarrow part\_of(Y, X),\ page(Y) = last\_but\_one$$
$$at\_page(X) = last \leftarrow part\_of(Y, X), page(Y) = last$$
$$alignment(X, Y) = both\_rows \leftarrow alignment(X, Y) = only\_lower\_row,$$
$$\quad alignment(X, Y) = only\_upper\_row$$
$$alignment(X, Y) = both\_columns \leftarrow alignment(X, Y) = only\_left\_row,$$
$$\quad alignment(X, Y) = only\_right\_col$$

The first four clauses allow information on the page order to be automatically associated to layout components, since their logical labeling may depend on the page order. The last two clauses define the alignment by both rows/columns of two layout components.

The *completeness* property of the output theory T holds when T explains all observations in O of the r concepts C_i, while the *consistency* property holds when T explains no counter-example in O of any concept C_i. The satisfaction of these properties guarantees the correctness of the induced theory with respect to O. Whether the theory T is actually correct, that is, whether it classifies correctly all other examples not in O, is an extra-logical matter, since no information on the generalization accuracy can be drawn from the training data themselves. In fact, the selection of the "best" theory is always made on the ground of an *inductive bias* (Mitchell 1997) expressed in the form of preference criterion (*PC*). In this work, short rules, which explain a high number of positive examples and a low number of negative examples, are preferred.

At the high level, the learning strategy implemented in ATRE is *sequential covering* (or *separate and conquer*) (Mitchell 1997), that is, one clause is learned (*conquer stage*), covered examples are removed (*separate stage*), and the process is iterated on the remaining examples. The most relevant novelties of the learning strategy implemented in ATRE are embedded in the design of the conquer stage.

First, the conquer stage of our algorithm aims at generating a clause that covers a specific positive example, called *seed*. Second, the separate-and-conquer strategy is traditionally adopted by single concept learning systems that generate clauses with the same literal in the head at each step. In ATRE, clauses generated at each step may have different literals in their heads. In addition, the body of the clause generated at the i-th step may include all literals corresponding to those target concepts $C_1, C_2, \ldots, C_r$ for which at least a clause has been added to the partially learned theory in previous steps. In this way, dependencies between target concepts can be automatically discovered.

Obviously, the order in which clauses of distinct target concepts have to be generated is not known in advance. This means that it is necessary to generate clauses with different literals in the head and then to pick one

of them at the end of each step of the separate-and-conquer strategy. Since the generation of a clause depends on the chosen seed, several seeds have to be chosen such that at least one seed per incomplete concept definition is kept. Therefore, the search space is actually a forest of as many search-trees (called *specialization hierarchies*) as the number of chosen seeds. A directed arc from a node (clause) C to a node C′ exists if C′ is obtained from C by adding a literal (single refinement step).

The forest can be processed in parallel by as many concurrent tasks as the number of search-trees (hence the name of *separate-and-parallel-conquer* for this search strategy). Each task traverses the specialization hierarchy top-down (or general-to-specific), but synchronizes traversal with the other tasks at each level. Initially, some clauses at depth one in the forest are examined concurrently. Each task is actually free to adopt its own search strategy, and to decide which clauses are worth to be tested. If none of the tested clauses is consistent, clauses at depth two are considered. Search proceeds towards deeper and deeper levels of the specialization hierarchies until at least a user-defined number of consistent clauses is found. Task synchronization is performed after that all "relevant" clauses at the same depth have been examined. A supervisor task decides whether the search should carry on or not on the basis of the results returned by the concurrent tasks. When the search is stopped, the supervisor selects the "best" consistent clause according to the user's preference criterion. This *separate-and-parallel-conquer* search strategy provides us with a solution to the problem of *interleaving* the induction process for distinct concept definitions. It has the advantage that simpler consistent clauses are found first, independently of the predicates to be learned. Moreover, the synchronization allows tasks to save computational effort when the distribution of consistent clauses in the levels of the different search trees is uneven. Details on the search strategy and its optimization through caching techniques are reported in Malerba (2003) and Berardi et al. (2004).

MRDM APPROACH TO DOCUMENT IMAGE UNDERSTANDING

Mr-SBC (Multi-Relational Structural Bayesian Classifier) extends to multi-relational data the naïve Bayesian classifier (Domingos and Pazzani 1997), which was originally defined for training data represented in a single table. The problem solved by the system can be formalized as follows.

Given:

- a training set represented by means of h relational tables $S = \{T_0, T_1, \ldots, T_{h-1}\}$ of a relational database D
- a set *PK* of primary key constraints on tables in S
- a set *FK* of foreign key constraints on tables in S

- a target relation $T \in S$
- a target discrete attribute y in T, different from the primary key of T, whose domain is $\{C_1, C_2, \ldots, C_r\}$

Find

a multi-relational naïve Bayesian classifier which predicts the value of y for some individual represented as a tuple in T (with possibly UNKNOWN value for y) and related tuples in S according to *FK*.

The solution implemented by Mr-SBC is based on the idea that for an individual I to be classified it is possible to find a set R of first-order definite clauses that classifies I into one of the classes $\{C_1, C_2, \ldots, C_r\}$. The class $f(I)$ associated to I maximizes the posterior probability $P(C_i|R)$:

$$f(I) = \arg\ \max_i\ P(C_i|R).$$

By applying Bayes theorem, we have:

$$f(I) = \arg\ \max_i\ P(C_i|R) = \arg\ \max_i \frac{P(C_i)P(R|C_i)}{P(R)}.$$

Since $P(R)$ is independent of the class C_i, it does not affect $f(I)$, that is,

$$f(I) = \arg\ \max_i\ P(C_i)P(R|C_i).$$

The construction of the set R is based on the notion of foreign key path.

Definition 1. A foreign key path is an ordered sequence of tables $\vartheta = (\mathrm{T}_{i_1}, \mathrm{T}_{i_2}, \ldots, \mathrm{T}_{i_S})$, where

- $\forall j = 1, \ldots, s, \quad \mathrm{T}_{i_j} \in S$
- $\forall j = 1..s-1, \quad \mathrm{T}_{i_{j+1}}$ has a foreign key to the table T_{i_j}.

All predicates in definite clauses in R are binary and can be of two different types.

Definition 2. A binary predicate p is a *structural* predicate associated to a table $T_i \in S$ if a foreign key in T_i exists that references a table $T_{i_1} \in S$. The first argument of p represents the primary key of T_{i_1} and the second argument represents the primary key of T_i.

Definition 3. A binary predicate p is a *property* predicate associated to a table $T_i \in S$ if the first argument of p represents the primary key of T_i and the second argument represents another attribute in T_i which is neither the primary key of T_i nor a foreign key in T_i.

Definition 4. A first-order definite clause associated to the *foreign key path* ϑ is a clause in the form:

$$p_0(A_1, y) \leftarrow p_1(A_1, A_2), p_2(A_2, A_3), \ \ldots, p_{s-1}(A_{s-1}, A_s), p_s(A_s, c).$$
$$\text{or} \quad p_0(A_1, y) \leftarrow p_1(A_1, A_2), p_2(A_2, A_3), \ \ldots, p_{s-1}(A_{s-1}, A_s),$$

where

1. p_0 is a property predicate associated to the target table T and to the target attribute y.
2. $\vartheta = (T_{i_1}, T_{i_2}, \ldots, T_{i_s})$ is a *foreign key path* such that for each $k = 1, \ldots, s-1$: p_k is a structural predicate associated to the table T_{i_k}.
3. p_s is an optional property predicate associated to the table T_{i_s}.

Mr-SBC searches for all possible definite clauses R_j associated to foreign key paths of a user-defined maximum length and covering the individual I. Then, the probability $P(R|C_i) = P(\bigcap_{R_j \in R} R_j | C_i)$ is computed by applying the naïve Bayes independence assumption on the minimal factor of the formula $\bigcap_{R_j \in R} R_j$. More details are reported in Ceci et al. (2006).

The relational nature of the probabilistic classification performed by Mr-SBC makes the system suitable for the document image understanding domain, where classes C_i are logical labels that can be associated to layout components (individuals to be classified). In addition, tightly coupling with a relational DBMS allows Mr-SBC to work, by means of views on the database used by WISDOM++ to store data on documents. In this way, Mr-SBC takes advantage of the database schema that provides useful knowledge of the data model without asking the user to specify some background knowledge. The logical view that Mr-SBC has on the layout and logical structures of document images is reported in Figure 1.

The application of Mr-SBC to the document image understanding domain is not straightforward and requires some adjustments. First, it is necessary to modify the search strategy in order to allow cyclic paths. As observed by Taskar et al. (2002), the acyclicity constraint hinders the representation of many important relational dependencies. This is particularly true in the task at hand, where a relation between two logical components is modeled by means of a table. For example, suppose that we need to model the relation *on_top* between two layout components. From a database point of view, this is realized by means of the table "block" and a table "on_top" that contains two foreign keys to the table "block." In the original formulation of the problem solved by Mr-SBC, first-order classification rules do not consider the same table twice (Ceci et al. 2003), therefore it is not possible to explore the search space by considering first the table "block," then the table "on_top," and again the table "block." To avoid this problem, we modified Mr-SBC, allowing cyclic paths. For this purpose, we considered a new definition of foreign key paths.

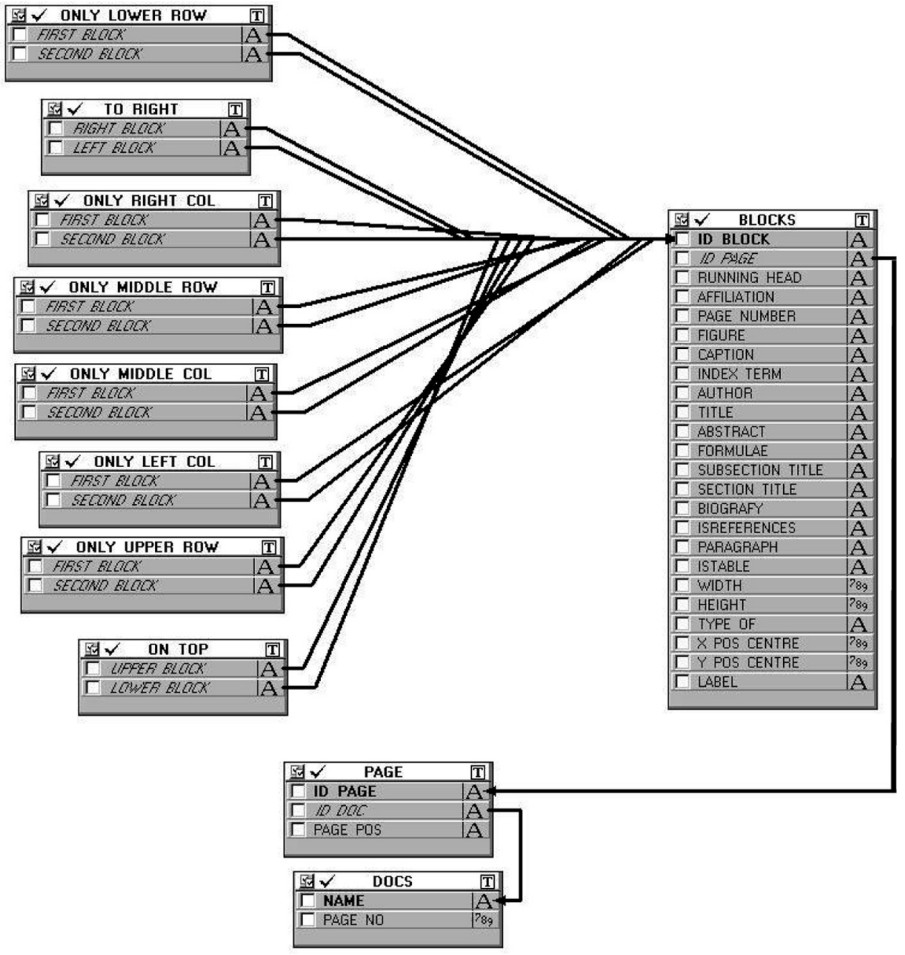

FIGURE 1 Logical view of the database input to Mr-SBC.

Definition 5. A foreign key path is an ordered sequence of tables $\vartheta = (T_{i_1}, T_{i_2}, \ldots, T_{i_S})$, where

- $\forall j = 1, \ldots, s,\ T_{i_j} \in S$
- $\forall j = 1..s-1,\ T_{i_{j+1}}$ has a foreign key to the table T_{i_j} or T_{i_j} has a foreign key to $T_{i_{j+1}}$.

The second adjustment concerns the classification of layout components. In document image understanding, it is possible that the same layout component is associated with two different logical labels because the layout analysis process has not been able to generate a distinct layout component for each logical component. To handle these situations, a binary classifier is

built for each class, such that it discriminates examples assigned to that class from all the others. However, this leads to the problem of "unbalanced data sets," since data can be characterized by a predominant number of negative examples with respect to the number of positive examples.

Several solutions to the problem of the unbalanced data sets have been proposed. Some are based on a sampling of examples in order to have a balanced data set (Mladenic and Grobelnik 1999). Others are based on a different idea: a) for each class C_i, examples in the test set are ranked from the most probable member to the least probable member; b) for each test example, a correctly calibrated estimate of the true probability that it belongs to class C_i is computed (Zadrozny and Elkan 2001); and c) a probability threshold that delimitates the membership and the non-membership of a given test example to the class C_i is computed. This approach fits our case well, since the naive Bayesian classifier for two-class problems tends to rank examples well (even if the classifier does not return a correct probability estimate) (Zadrozny and Elkan 2001). In our solution, the threshold is determined by maximizing the AUC (Area Under the ROC Curve) (Provost and Fawcett 2001) according to a cost function:

$$cost = P(C_i) \cdot (1 - TP) \cdot c(\neg C_i; C_i) + P(\neg C_i) \cdot FP \cdot c(C_i; \neg C_i),$$

where $P(C_i)$ ($P(\neg C_i)$) is the prior probability that an example does (not) belong to the class C_i, $c(\neg C_i; C_i)(c(C_i; \neg C_i))$ is the cost of classifying a positive (negative) example as negative (positive) for the class C_i, TP is the true positive rate and FP is the false positive rate. In the experiments reported in the next section, different values of $CostRatio = c(\neg C_i; C_i)/c(C_i; \neg C_i)$ have been considered.

EXPERIMENTAL RESULTS

For a fair comparison of the two learning methods, both Mr-SBC and ATRE are trained on three different data sets.[3] The first data set consists of multi-page articles published in an international journal. In particular, we considered 21 papers, published as either regular or short, in *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, in the January and February issues of 1996. Each paper is a multi-page document; therefore, we processed 197 document images in all and the user manually labeled 2,436 layout components, that is, on average, 116 components per document, 12.37 per page. The components that have not been labeled are "irrelevant" for the task in hand or are associated to "noise" blocks; they are automatically considered *undefined*.

The second and the third data sets have been provided by two distinct European film archives, namely, Deutsches Filminstitut (DIF) and Filmarchiv Austria (FAA) in the context of the EU funded project COLLATE. In both

TABLE 2 Considered Logical Labels for Each Data Set

Source	Labels
TPAMI	*abstract, affiliation, author, biography, caption, figure, formulae, index_term, reference, table, page_no, paragraph, running_head, section_title, subsection_title, title.*
DIF	*cens_signature, cert_signature, object_title, cens_authority, chairmen, assessors, session_data, representative*
FAA	*dep_signature, adhesive_stamp, stamp, registration_au, date_place, department, applicant, reg_number, film_length, film_producer, film_genre, film_title*

data sets, documents represent rare historic film censorships from the 20's and 30's. In the DIF data set, documents are generally composed of two pages and the user manually labeled 149 layout components out of 950 components in all. The FAA data set is composed of one-page documents and the user manually labeled 140 layout components over 503 components in all. In DIF and FAA data sets, the percentage of *undefined* components is relatively high with respect to the TPAMI data set. This is mainly due to the presence of ink specks, holes, and manual annotations on historical documents. Table 2 reports logical labels considered in this study, while Figure 2 shows two examples of labeled document images.

The performance of the learning tasks is evaluated by means of a 5-fold cross-validation on all data sets, that is, for each data set, the set of documents is first divided into five folds, and then, for every fold, ATRE and Mr-SBC are trained on the remaining folds and tested on the

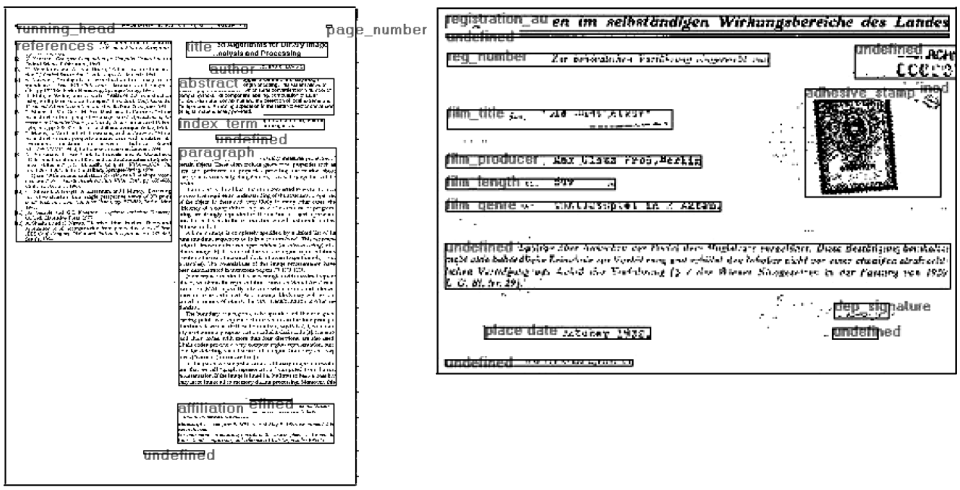

FIGURE 2 Two processed document images. On the left: an example of a TPAMI short paper. On the right: an example of a FAA censorship card.

TABLE 3 TPAMI Data Set Description: Distribution of Pages and Examples per Document Grouped by Five Folds

Fold no.	No. of Documents	No. of pages	No. of labeled components	Total no. of components
1	4	40	476	597
2	4	36	519	684
3	4	41	481	697
4	4	42	541	774
5	5	38	419	549
Total	21	197	2436	3301

hold-out fold. In Tables 3, 4, and 5 a brief description of the data sets is reported.

For each learning problem, the number of omission/commission errors is recorded. An omission error occurs when the logical labeling of a layout component is missed, while a commission error occurs when a wrong logical labeling is "recommended" by the classifier. In our study we do not base the evaluation of the classifiers on the standard classification accuracy, because for each learning task, the number of positive and negative examples is strongly unbalanced and, in most cases, the trivial classifier that returns always "undefined" would be the classifier with the best accuracy.

Henceforth, experimental results are commented for each data set. They are reported in the order of quality of the extracted layout, namely, TPAMI, DIF, and FAA. Indeed, TPAMI documents are well structured, and the extracted layout generally separates the logical components, while layout extraction is more difficult for historical documents due to the presence of stamps, ink specks, and so on. Nonetheless, there is a difference between DIF and FAA documents; the former is better structured and less noisy.

TABLE 4 DIF Data Set Description: Distribution of Pages and Examples per Document Grouped by Five Folds

Fold no.	No. of Documents	No. of pages	No. of labeled components	Total no. of components
1	5	8	28	200
2	5	9	30	196
3	5	9	33	201
4	5	8	25	152
5	5	9	33	201
Total	25	43	149	950

TABLE 5 FAA Data Set Description: Distribution of Pages and Examples per Document Grouped by Five Folds

Fold no.	Document name	No. of pages	No. of labeled components	Total no. of components
1	5	5	34	125
2	5	5	29	115
3	5	5	36	93
4	5	5	26	113
5	5	5	15	57
Total	25	25	140	503

TPAMI Data Set

Figure 3 shows results of Mr-SBC for different values of *CostRatio* when $n = 2$ and $n = 3$, where n is the number of predicates in the body of a first-order definite clause associated to a *foreign key path* (see Definition 4). By increasing *CostRatio*, more importance is given to the cost of false negative $c(\neg C_i; C_i)$ and the ability of the classifier to correctly classify positive examples increases, while the precision of the classification decreases because negative examples are erroneously classified as positive. This behavior is somehow expected and occurs also in the case of DIF and FAA documents. What is really surprising is the fact that when moving from $n = 2$ to $n = 3$ the number of commission errors increases. This is also clarified in Table 6, where the absolute number of omission/commission errors is reported for each fold. This means that while probability values returned by Mr-SBC for positive and negative examples of each logical components are well separated when $n = 2$, they become closer when $n = 3$, thus causing some problems to the automated threshold determination procedure. Indeed, if we rank each positive/negative example according to the probability value returned by Mr-SBC, we observe that most of positive (negative) examples have high (low) ranking when $n = 2$ (see Figure 4), while the ranking distribution is less skewed when $n = 3$.

Table 6 also shows omission and commission errors performed by ATRE. First, we observe that the number of positive examples for each fold is lower

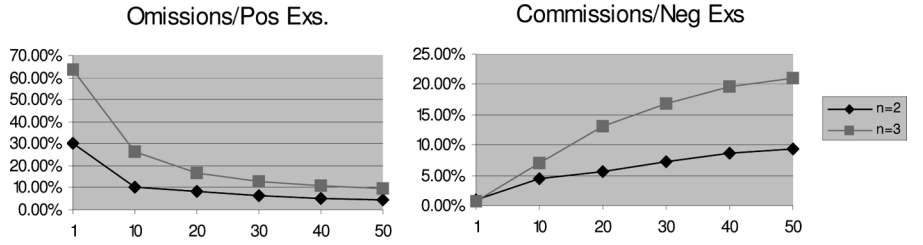

FIGURE 3 Percentage of omission/commission errors computed for different values of *CostRatio* and number of predicates in the body (n).

TABLE 6. Mr-SBC vs. ATRE

		ATRE		Mr-SBC ($n = 2$)				Mr-SBC	Mr-SBC ($n = 3$)			
	Folds	No. Ex	Errors	1	10	20	30	No. Ex	1	10	20	30
Omission errors	1	295	120	174	69	58	56	476	304	157	97	72
	2	312	99	155	60	48	34	519	366	135	88	76
	3	322	107	140	50	42	34	481	285	114	67	54
	4	334	115	166	36	19	13	541	362	134	77	60
	5	261	105	105	40	28	21	419	241	102	82	51
			36.06%	30.25%	10.53%	8.07%	6.56%		63.61%	26.36%	17.01%	12.85%
Commission	1	8860	66	127	500	564	592	9076	79	617	1337	1685
	2	9948	84	106	506	598	948	10425	89	687	1282	1668
	3	10133	95	103	448	600	716	10671	62	759	1541	1762
	4	11276	76	76	485	677	879	11843	93	810	1533	1976
	5	7974	52	90	300	439	561	8365	50	628	969	1347
			0.77%	1.02%	4.45%	5.71%	7.29%		0.74%	6.97%	13.20%	16.77%

Average number of omission errors over positive examples and commission errors over negative examples on TPAMI data set. Mr-SBC results are obtained varying *CostRatio* in {1,10,20,30} and n in {2,3}.

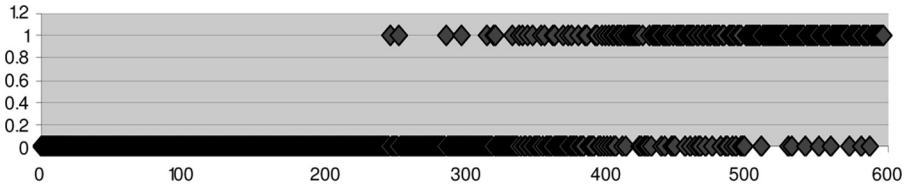

FIGURE 4 Ranking of 597 testing examples of the logical component *Paragraph* for the TPAMI data set. Each point represents an example *E*. Examples are ranked on the basis of $P(Paragraph = \text{“}true\text{”}|E)$. Positive examples are reported above (Y = 1) while negative examples are reported below (Y = 0). Results are obtained by training Mr-SBC on folds 2,3,4,5 and testing on fold 1 with $CostRatio = 10$ and $n = 2$.

for ATRE than for Mr-SBC (compare columns “No. Ex” of the two systems). This is due to the fact that ATRE does not converge towards a solution when examples of the logical component “*Paragraph*” are considered. The training set contains many examples of Paragraph, but finding characterizing properties proved to be a difficult task for both systems. By comparing the results of the two systems, we observe that ATRE is quite precise (0.77% of commission errors), but also presents a high number of omission errors (36.06%). Mr-SBC with $n = 2$ seems to offer a good tradeoff between recall and precision for this data set, especially considering the fact that it learns a classifier for all logical components; *Paragraph* included. For the specific choice of $n = 2$ and $CostRatio = 10$, Table 7 reports results concept by concept, so that it is possible to understand what are the logical components that both systems found difficult to recognize. ATRE is very conservative and has an omission rate below 30% only for *running_head*, *page_number*, and *figure*, while Mr-SBC has an omission rate below that threshold for nine logical components. The main problem with Mr-SBC is the high number of commission errors for some concepts. This is due to the maximization of the AUC according to the cost function defined previously that leads Mr-SBC to set very low thresholds, which do not properly filter negative examples.

For the sake of completeness, some examples of rules learned by ATRE are reported in the following.

1. $running\_head(X1) \leftarrow width(X1) \in [390..414], y\_pos\_centre(X1) \in [19..100]$
2. $figure(X1) \leftarrow type\_of(X1) = graphic, height(X1) \in [35..218]$
3. $figure(X1) \leftarrow type\_of(X1) = image, alignment(X1, X2) = only\_middle\_col, figure(X2)$
4. $table(X1) \leftarrow height(X1) \in [56..75], width(X1) \in [370..445], on\_top(X1, X2), figure(X2)$
5. $title(X1) \leftarrow at\_page(X1) = first, y\_pos\_centre(X1) \in [67..88]$
6. $subsection\_title(X1) \leftarrow x\_pos\_centre(X1) \in [74..84], width(X1) \in [105..115]$.

They can be interpreted easily. For instance, the first rule states that the running head is a large block located at the top of the document. The first

TABLE 7. Mr-SBC vs. ATRE on TPAMI

	Omissions/Pos. Ex.		Commissions/Neg. Ex.	
	ATRE	Mr-SBC	ATRE	Mr-SBC
Abstract	42.86%	**19.05%**	**0.15%**	0.91%
Affiliation	59.09%	**13.64%**	**0.40%**	0.37%
Author	44.00%	**40.00%**	**0.09%**	0.40%
Biography	33.33%	**28.57%**	**0.03%**	0.24%
Caption	74.86%	**30.60%**	**1.83%**	14.95%
Figure	14.33%	**4.78%**	**1.31%**	3.47%
Formulae	40.67%	**12.23%**	**5.45%**	12.81%
Index_term	90.91%	**45.45%**	**0.03%**	0.46%
Page_number	7.22%	**0.56%**	0.42%	**0.35%**
Paragraph	–	3.62%	–	22.39%
References	72.50%	**30.00%**	**0.52%**	3.19%
Running_head	12.32%	**5.42%**	**0.16%**	4.58%
Section_title	66.13%	**33.87%**	**0.74%**	4.72%
Subsection_title	100.00%	**60.00%**	**0.06%**	3.14%
Table	76.09%	**34.78%**	**0.68%**	4.06%
Title	43.48%	**26.09%**	**0.27%**	0.95%

Average number of omission errors over positive examples, commission errors over negative examples. Mr-SBC results are obtained with $n = 2$ and $CostRatio = 10$. The best results are in bold.

three rules refer to logical components for which both learning systems show a low omission error rate, while the other three rules refer to logical components for which ATRE has a quite high percentage of omission errors. It is noteworthy that the syntactic complexity of rules cannot be easily associated with the performance of the learning systems; while the first six rules appear quite meaningful, the last is simple but does not seem to capture any actual regularity. Finally, we observe that some rules (3 and 4) express meaningful concept dependencies, even through recursive definitions.

DIF Data Set

Table 8 shows experimental results on the historical documents provided by DIF. This time the number of training examples is the same for both systems, since ATRE has no problem learning logical theories for all logical components. As *CostRatio* increases, the percentage of omission errors performed by Mr-SBC decreases at the cost of a slight increase of commission errors. For this data set, the choice of $n = 2$ and $CostRatio = 10$ seems to offer a good tradeoff between omission and commission errors.

FAA Data Set

Table 9 shows experimental results for the FAA data set. As in the previous case, the number of training examples is the same for both

TABLE 8 Mr-SBC vs. ATRE

		ATRE		Mr-SBC ($n = 2$)				Mr-SBC	Mr-SBC ($n = 3$)			
	Folds	No. Ex	Errors	1	10	20	30	No. Ex	1	10	20	30
Omission errors	1	28	9	11	4	2	2	28	20	15	8	6
	2	30	10	9	6	6	4	30	23	14	13	12
	3	33	5	13	4	4	4	33	26	16	10	8
	4	25	8	6	5	4	4	25	18	11	8	7
	5	33	13	14	5	5	5	33	23	21	15	11
			30.40%	35.02%	16.31%	14.08%	12.75%		73.72%	51.27%	35.93%	29.40%
Commission	1	1372	1	3	15	23	23	1372	2	31	82	115
	2	1342	15	13	28	28	52	1342	13	179	209	226
	3	1347	6	4	13	13	13	1347	0	65	102	119
	4	1039	3	6	18	28	42	1039	14	65	99	121
	5	1374	5	6	18	18	18	1374	9	65	118	145
			0.46%	0.50%	1.43%	1.74%	2.37%		0.62%	6.26%	9.42%	11.22%

Average number of omission errors over positive examples and commission errors over negative examples on DIF dataset. Mr-SBC results are obtained varying *CostRatio* in {1,10,20,30} and n in {2,3}.

TABLE 9 Mr-SBC vs. ATRE

		ATRE		Mr-SBC ($n = 2$)				Mr-SBC	Mr-SBC ($n = 3$)			
	Folds	No. Ex	Errors	1	10	20	30	No. Ex	1	10	20	30
Omission errors	1	34	21	27	17	16	13	34	32	27	24	19
	2	29	15	20	15	13	12	29	26	20	16	15
	3	36	27	29	20	18	16	36	30	32	15	10
	4	26	14	16	10	9	9	26	22	19	14	12
	5	15	8	7	5	4	4	15	9	7	3	3
			59.50%	67.43%	45.81%	40.63%	37.07%		82.34%	71.40%	48.25%	40.31%
Commission	1	1466	9	12	42	59	98	1466	0	27	120	245
	2	1351	16	10	66	99	135	1351	0	36	121	154
	3	1080	3	3	32	72	94	1080	7	36	204	296
	4	1330	6	5	44	76	116	1330	0	27	140	213
	5	669	7	7	56	68	94	669	39	162	285	341
			0.71%	0.65%	4.48%	6.78%	9.63%		1.30%	6.82%	17.83%	24.50%

Average number of omission errors over positive examples and commission errors over negative examples on FAA dataset. Mr-SBC results are obtained varying *CostRatio* in {1,10,20,30} and n in {2,3}.

TABLE 10 Mr-SBC vs. ATRE

	ATRE	Mr-SBC ($n = 2$)	Mr-SBC ($n = 3$)
TPAMI	46,770.5	756.4	10,186.9
DIF	532.20	94.7	774.4
FAA	514.2	108.3	736.1

Average learning times. Results are expressed in seconds. Mr-SBC results are obtained with *CostRatio* = 10.

systems. Results confirm initial observation on the complexity of this learning task because of the poor layout structure of many censorship cards. ATRE misclassifies few examples, but it suffers from a high rate (almost 60%) of omission errors. Mr-SBC also has a high percentage of omission errors but it is possible to keep it under control with an appropriate choice of *CostRatio* (once more, a good tradeoff is $n = 2$ and *CostRatio* = 10).

We conclude with two general considerations.

1. Mr-SBC is more efficient than ATRE (see Table 10), despite of the fact that most of computations are performed by database queries (ATRE works only in main memory).
2. The number of rules learned by ATRE varies considerably for each data set (see Table 11) and seems more related to the number of training examples rather than the number of classes. On the contrary, the number of classification queries (i.e., probabilities) performed (estimated) by Mr-SBC seems to depend strongly on the number of predicates in the body of a first-order definite clause associated to a foreign key path. The high number of probabilities to be estimated when $n = 3$ can be a cause of the low performance of Mr-SBC. Indeed, as observed by Bellman (1961), the number of examples should increase exponentially with the number of features to maintain a given level of accuracy ("curse of dimensionality").

TABLE 11 Mr-SBC vs. ATRE

	TPAMI	DIF	FAA
ATRE – avg no. of learned rules	278.8	28.4	41.6
Mr-SBC – Avg no. of classification queries ($n = 2$)	3,494	1,265	1,287
Mr-SBC – Avg no. of classification queries ($n = 3$)	40,273	16,093	8,025
No. classes	16	8	12

Complexity of induced models. Mr-SBC results are obtained with *CostRatio* = 10.

CONCLUSIONS

In this work the induction of classifiers for the automated recognition of semantically relevant layout components has been investigated. In particular, the resort to a relational approach has been motivated by observing that document layout structures are a kind of spatial data which can be properly modeled by means of either first-order logic or multiple tables of a relational database. Two distinct relational approaches to classifier construction have been described and empirically compared, namely, a logical approach based on theoretical advances in the field of inductive logic programming and a statistical approach based on concepts and principles typical of multi-relational data mining. The former, implemented in ATRE, supports the autonomous discovery of concept dependencies and is able to deal with autocorrelation of spatially lagged response and explanatory variables. The latter, implemented in Mr-SBC, can only deal with spatially lagged explanatory variables but is well founded in the Bayesian theory and returns a degree of confidence (a posterior probability) for each class.

The empirical comparison of the two systems has been performed on three data sets with various degree of accuracy in extracted layout. Experimental results show that: 1) spatial relations occur in many clauses of logical theories learned by ATRE, which motivates the resort to a relational approach; 2) brittleness of logical theories makes ATRE quite conservative (low rate of commission errors and high rate of omission errors); 3) ATRE is not always applicable (see logical component *Paragraph* in TPAMI data); 4) the performance of Mr-SBC strongly depends on the number n of predicates in the body of definite clauses associated to foreign key paths, in particular, commission errors noticeably increase with n, probably because of the greater difficulty to find a proper threshold for probability values associated with positive/negative examples; 5) the probabilistic classification performed by Mr-SBC allows us to tradeoff between omission and commission errors; and 6) Mr-SBC is more scalable than ATRE on this application domain.

For future work, we intend to improve both Mr-SBC by including contextual discretization of numerical attributes and ATRE by weakening the subsumption test in order to recover omission errors.

REFERENCES

Aiello, M., C. Monz, T. Todoran, and M. Worring. 2002. Document understanding for a broad class of documents. *International Journal of Document Analysis and Recognition* 5(1):1–16.

Akindele, O. T. and A. Belaïd. 1995. Construction of generic models of document structures using inference of tree grammars. In *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, pages 206–209, Montreal, Canada.

Allen, J. F. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM* 26(11):832–843.

Altamura, O., F. Esposito, and D. Malerba. 1999. WISDOM++: An interactive and adaptive document analysis system. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 366–369. IEEE Computer Society.

Altamura, O., F. Esposito, and D. Malerba. 2001. Transforming paper documents into XML format with WISDOM++. *International Journal on Document Analysis and Recognition* 4(1):2–17.

Altamura, O., M. Berardi, M. Ceci, D. Malerba, and A. Varlaro. 2007. Using color information to understand censorship cards of film archives. *International Journal on Document Analysis and Recognition* IJDAR, Springer, DOI:10.1007/s10032-006-0021-1.

Bellman, R. E. 1961. *Adaptive Control Processes*. Princeton: Princeton University Press.

Berardi, M., M. Ceci, and D. Malerba. 2003a. Mining spatial association rules from document layout structures. In *Proc. of the 3rd Workshop on Document Layout Interpretation and its Application (DLIA 2003)*, pages 9–13. Deutsches Forschungszentrum fur Kunstliche Intelligenz, GmbH, Germany.

Berardi, M., M. Ceci, F. Esposito, and D. Malerba. 2003b. Learning logic programs for layout analysis correction. In *proceedings of the International Conference on Machine Learning (ICML2003)*, pages 27–34.

Berardi, M., A. Varlaro, and D. Malerba. 2004. On the effect of caching in recursive theory learning. In: *Inductive Logic Programming: ILP 2004, Lecture Notes in Artificial Intelligence*, 3194, eds. R. Camacho, R. D. King, and A. Srinivasan, 44–62. Berlin: Springer.

Berardi M., M. Ceci, and D. Malerba. 2005. A hybrid strategy for knowledge extraction from biomedical documents. Proceedings of NNLDAR, pages 18–22, Seoul, Korea.

Ceci, M., and A. Appice. 2006. Spatial associative classification: Propositional vs structural approach. *Journal of Intelligent Information Systems* 27(3):191–213.

Dengel, A., R. Bleisinger, R. Hoch, F. Fein, and F. Hones. 1992. From paper to office document standard representation. *Computer* 25(7):63–67.

De Raedt, L. 1992. *Interactive Theory Revision*. London: Academic Press.

Domingos, P. and M. Pazzani. 1997. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning* 29(2-3):103–130.

Dzeroski, S. and N. Lavrac. 2001. *Relational Data Mining*. Berlin: Springer-Verlag.

Esposito, F., D. Malerba, and G. Semeraro. 1994. Multistrategy learning for document recognition. *Applied Artificial Intelligence: An International Journal* 8(1):33–84.

Hu, J., R. Kashi, D. Lopresti, and G. Wilfong. 2001. Experiments in table recognition. In *Proceedings of the Workshop on Document Layout Interpretation and Applications*, Seattle, Washington, USA. http://www.science.uva.nl/events/dlia2001.

Knobbe, A. J., A. Siebes, and D. M. G. Van der Wallen. 1999. Multi-relational decision tree induction. In *Proceedings of the 3rd European Conference on Principles of Data Mining and Knowledge Discovery*, pages 378–383. Springer-Verlag.

Le Bourgeois, F., S. Souafi-Bensafi, J. Duong, M. Parizeau, M. Coté, and H. Emptoz. 2001. Using statistical models in document images understanding. Workshop on Document Layout Interpretation and its Applications, DLIA Seattle, Washington, USA. http://www.science.uva.nl/events/dlia2001.

Lloyd, J. W. 1987. *Foundations of Logic Programming, 2nd ed.* Berlin: Springer-Verlag.

Malerba, D. and F. A. Lisi. 2001. Discovering associations between spatial objects: An ILP application. In: *Inductive Logic Programming, Lecture Notes in Artificial Intelligence, 2157*, eds. C. Rouveirol and M. Sebag, 156–163. Berlin: Springer.

Malerba, D. 2003. Learning recursive theories in the normal ILP setting. *Fundamenta Informaticae* 57(1):39–77.

Malerba, D., M. Ceci, and M. Berardi. 2003. XML and knowledge technologies for semantic-based indexing of paper documents. In: *Database and Expert Systems Applications, 14th International Conference, DEXA 2003, Lecture Notes in Computer Science, 2736*, eds. V. Marík, W. Retschitzegger, and O. Stepánková, 256–265. Berlin: Springer.

Mitchell, T. M. 1997. *Machine Learning*. McGraw-Hill, New York, USA.

Mladenic, D. and M. Grobelnik. 1999. Feature selection for unbalanced class distribution and naive bayes. In *Proc. of the 16th International Conference on Machine Learning (ICML)*, pages 258–267.

Muggleton, S. 1992. *Inductive Logic Programming*. London: Academic Press.

Nagy, G. 2000. Twenty years of document image analysis in PAMI. *IEEE Trans. PAMI* 22(1):38–62.

Nagy, G., S. C. Seth, and S. D. Stoddard. 1992. A prototype document image analysis system for technical journals. *IEEE Computer* 25(7):10–22.

Nienhuys-Cheng, S.-W. and R. de Wolf. 1997. *Foundations of Inductive Logic Programming*. Heidelberg: Springer.

Palmero, G. I. S. and Y. A. Dimitriadis. 1999. Structured document labeling and rule extraction using a new recurrent fuzzy-neural system. *International Journal of Document Analysis and Recognition* 181–184. Proc. of the 5th/conference on/KDAR/Bangalore, India.

Provost, F. and T. Fawcett. 2001. Robust classification for imprecise environments. *Machine Learning* 42(3):203–231.

Rosenfeld, A., R. A. Hummel, and S. W. Zucker. 1976. Scene labeling by relaxation operations. *IEEE Transactions SMC* 6(6), 420–433.

Tang, Y. Y., C. D. Yan, and C. Y. Suen. 1994. Document processing for automatic knowledge acquisition. *IEEE Trans. on Knowledge and Data Engineering* 6(1):3–21.

Taskar, B., P. Abbeel, and D. Koller. 2002. Discriminative probabilistic models for relational data. In *Proc. of Int. Conf. on Uncertainty in Artificial Intelligence*, pages 485–492, Alberta, Canada.

Walischewski, H. 1997. Automatic knowledge acquisition for spatial document interpretation. In *Proc. of the 4th International Conference on Document Analysis and Recognition (ICDAR)*, pages 243–247, Bangalore, India.

Wenzel, C. and H. Maus. 2001. Leveraging corporate context within knowledge-based document analysis and understanding. *International Journal on Document Analysis and Recognition* 3(4):248–260.

Zadrozny, B. and C. Elkan. 2001. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Proc. of the 18th International Conference on Machine Learning (ICML)*, pages 609–616, Williamstown, Massachusetts.

ENDNOTES

1. In statistics, spatial autocorrelation indicates the fact that the effect of an explanatory (independent) or response (dependent) variable at any location may not be limited to the specific location.
2. For the sake of completeness, we point out that the actual representation of the clause in ATRE is:

$$\begin{aligned} \mathit{author}(X1) = \mathit{true} \leftarrow\ & \mathit{alignment}(X1, X2) \\ = \mathit{middle\_col},\ & \mathit{abstract}(X2) = \mathit{true},\ \mathit{height}(X1) \in [7..13], \end{aligned}$$

   where the truth value of a predicate is made explicit. However, it is easy to transform ATRE's clauses into definite clauses, extended with built-in predicates. Details are reported in Malerba (2003).
3. Data in the first order logic format are available on-line at the following url: http://www.di.uniba.it/~ceci/micFiles/5fold%20cross%20validation%20Tpami.rar