# Incremental Machine Learning Techniques for Document Layout Understanding

S. Ferilli, M. Biba, T.M.A. Basile, F. Esposito
*Dipartimento di Informatica - Università di Bari*
{*ferilli, biba, basile, esposito*}*@di.uniba.it*

## Abstract

*In real-world Digital Libraries, Artificial Intelligence techniques are essential for tackling the automatic document processing task with sufficient flexibility. The great variability in document kind, content and shape requires powerful representation formalisms to catch all the domain complexity. The continuous flow of new documents requires adaptable techniques that can progressively adjust the acquired knowledge on documents as long as new evidence becomes available, even extending if needed the set of recognized document types. Both these issues have not yet been thoroughly studied. This paper presents an incremental first-order logic learning framework for automatically dealing with various kinds of evolution in digital repositories content: evolution in the definition of class definitions, evolution in the set of known classes and evolution by addition of new unknown classes. Experiments show that the approach can be applied to real-world.*

## 1. Introduction

In a world and society strongly based on documents for all key activities and branches, the introduction of computer technology has boosted the production and exchange of documents in electronic format, and raised the need for digital systems and repositories to manage, store, organize and suitably retrieve the documents at need. Such systems, called Digital Libraries, must thus deal with a huge and ever-increasing amount of documents, which motivated much work by the Artificial Intelligence community in the last decades [4, 8]. Typically, different kinds of documents must be processed in different ways, and are characterized by different layout appearance. Thus, distinguishing documents according to their layout appearance can be useful to focus only on their most important components

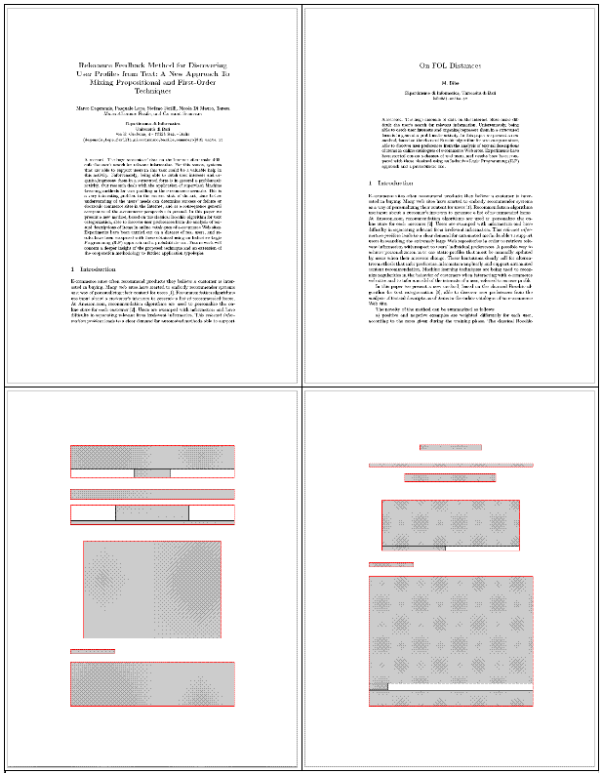

**Figure 1. Documents in the same class**

when extracting information aimed at indexing, rendering and collaboration, this way improving effectiveness and efficiency. While for some kinds of documents having a strict layout standard (such as forms) static models based on templates and on size/position/kind of layout components can be sufficient for recognizing them and properly extracting the needed information, most document classes show a high variability in the components' size and position (see Figure 1), so that very flexible techniques are needed to understand them. In such cases, the mutual relationships among components are more important than the components'

attributes for defining their role, which makes useless classical attribute-value representation formalisms and processing techniques. We propose the use of first-order logic (FOL for short) as a powerful formalism able to express relations between objects. In particular, *Horn clauses* are FOL formulæ of the form $l_0$ :- $l_1, \ldots, l_n$ where the $l_i$'s are *atoms* (predicates applied to terms), to be interpreted as "$l_0$ holds if $l_1$ and ... and $l_n$ hold". In our case, $l_0$ states a document's class or a component's role, while the premises describe its layout according to components size, type, absolute and relative position. A model in this setting consists of a set of clauses such as "ecai(A) :- first_page(A, B), frame(B, C), text(C), height_very_very_small(C), on_top(D, C), text(D), pos_upper(D), frame(B, D)" ($A$ is an ECAI paper if its first page $B$ contains two text frames, $C$ and $D$, the latter in upper position and ontop of the former, which is very very small). Moreover, in real-world (digital) libraries known documents classes can adopt new layout standards, and completely new document types can be considered during time, raising the complex problem of evolution in document shapes and types. Unfortunately, incrementality in FOL learning is not limited to parameter adjusting, but often involves complete restructuring of the model components.

The novel contribution of this paper lies in the presentation and validation of a full incremental framework to learn and adapt on-the-fly FOL theories for document management whenever needed, not only as concerns class definitions but even for completely new (and possibly unknown) classes. The proposed process of document assimilation was implemented in DOMINUS, a prototypical document processing and management system characterized by the intensive exploitation of intelligent techniques [2] whose architecture includes a Learning Server module devoted to solve different Machine Learning tasks involved in document processing. In particular, two modules are specifically related to the subject of this paper: an incremental first-order logic learning system that performs theory revision, and a conceptual clustering system in charge of discovering new document layout classes in rejected documents.

## 2. An Incremental Framework for Tackling Evolution in digital repositories

Document image understanding is essential to support the organization of the ever-increasing documents having different nature and, more importantly, different standards. Statistical techniques have been proved successful in efficiently and effectively carrying out such tasks on documents with a fixed layout. Furthermore, the classical approaches require that the classes and the layout components to be learned are known at the beginning of the learning process, thus whenever a new class/component is available the learning process must be restarted from scratch. The continuous flow of new documents in Digital Libraries calls for incremental abilities of the system in order to revise a faulty knowledge previously acquired for recognizing the class and the logical structure of a document. Indeed, the great variety of document layout standards requires the system to be flexible in dealing with both completely new classes of documents or different layout templates of documents belonging to the same class. However, strict time constraints do not allow to restart learning from scratch every time a new problematic item comes up. For this reason, we propose the exploitation of incremental strategies in document management systems that have to deal with continuous assimilation of documents.

If the class and layout components type can be correctly recognized by the system for a new document using the current model, the document is automatically filed and the significant components undergo text extraction and indexing. In case of failure (which can be recognized by inability of assigning a document/component to a class, or by multiple inconsistent assignments, or by a low assignment confidence, or by an explicit intervention of an expert), the system rejects the document and/or its layout components and puts them in stand-by. If the expert can provide the system with the correct document class (resp. components type), and they are already known to the system, the incremental supervised learning algorithm revises (without completely rejecting it or restarting the learning process from scratch) the previously learned theory. Otherwise, in case the class and/or component types provided by the expert are new to the document management system, the incremental supervised learning algorithm extends the theory with definitions for the new classes/components in order to correctly recognize future items of the same kind. In case the expert is unable to provide the system with the correct document class and/or layout components type, after a sufficient amount of unlabeled rejected documents/components is collected, unsupervised learning is started to discover new classes among them, according to regularities in their layout structure.

### 2.1 Theory revision

Supervised learning in our framework works according to the following procedure. An initial theory for recognizing classes of documents and their significant layout components is learned, based on labeled observations given by the expert. Starting from a given (even

empty) theory, examples are considered one by one as long as they become available, continuously revising the learned theory until it passes the desired accuracy threshold. From that moment on, the theory becomes operational and serves as a classifier for new unseen documents, thus making the system completely automatic. When a new document is wrongly classified, or cannot be classified at all by the current theory, a reject occurs. If the correct class is provided by the expert, a revision is started and the current theory is automatically tuned so that it can account also for the problematic example, and can improve its performance on future documents as well. The important point is that, if the document belongs to a previously unknown class, a brand new class can be learned (even from just one instance) and added to the theory, this way avoiding the need for a knowledge engineer that sets up again the learning task from scratch. Thus, not only the set of learning examples can be partially known, but the set of target classes as well.

In DOMINUS such a task is carried out by INTHELEX [1], an incremental learning system able to induce and revise FOL theories for multiple classes from examples. It incorporates two inductive refinement operators to fix the wrong behavior of the current theory. In case a positive example is rejected, the system generalizes the definition of the corresponding concept by dropping some conditions (ensuring consistency with all the past negative examples), or adding to the current theory (if consistent) a new alternative definition of the concept. When a negative example is explained, a specialization of one of the theory definitions that concur in explaining the example is attempted by adding to it some (positive or negative) conditions which characterize all the past positive examples and discriminate them from the current negative one. In case no complete/consistent refinement can be found, exceptions can be added to the theory.

### 2.2 Document clustering

On-line digital libraries are bound to face, sooner or later, the show-up of new documents that do not belong to any known class, and for which no label is provided by the expert. Indeed, new kinds of documents can enter the areas of interest of the library, or new shapes of existing classes can come up in time. These documents cannot be incorporated in the library structure as-is. When a sufficient amount of such documents is collected, the system should autonomously discover new classes among them, according to regularities in their layout structure.

The *Clustering* task aims at organizing a collection of unlabeled patterns into groups (clusters) of homogeneous elements based on their similarity [5]. Conceptual Clustering also generates a concept description for each cluster, driven by both the inherent structure of the data and the description language used. The similarity measure exploited to evaluate the distance between elements determines the effectiveness of the clustering algorithms. Differently from the case of attribute-value representations [6], completely new comparison criteria and measures must be defined for FOL descriptions since they do not induce a Euclidean space. Our proposal is based on a similarity assessment technique for Horn clauses introduced in [3], and exploits the resulting similarity value as a distance measure in a classical $k$-means clustering algorithm, based on *medoids* instead of *centroids* as cluster prototypes guiding the partitioning process (due to the non-Euclidean space issue).

The evaluation of similarity between two items $i'$ and $i''$ is based on the number of common and different features among them [7], through a formula that yields a value ranging between 0 and 1. Since FOL formulæ are complex objects, it is not easy to straightforwardly count the number of common and different features for two of them, and hence after assigning a similarity to basic components, the similarity of higher-level ones is assessed on the grounds of the similarity of their subcomponents plus the new features introduced by their combination. Terms (representing objects, related to each other by means of predicates) are compared according to the properties they own, usually expressed by unary predicates (e.g., `text(X)`), and the roles they play in the observations, represented by their position in the $n$-ary predicate arguments (e.g., in `on_top(X,Y)` $X$ plays the role of the upper object and $Y$ the role of the lower one). At the relational level, atoms are compared based on their neighborhood, i.e. the set of atoms having terms in common with them, also considering the similarity of those terms. Finally, the similarity between clauses is obtained according to their (least general) generalization (i.e., a maximal overlapping between their atoms), smoothed by the the similarity among the couples of terms and atoms resulting from the generalization associations.

## 3. Experiments

The proposed framework was tested on a real-world dataset made up of 353 scientific papers belonging to 4 classes: Elsevier journals, SVLN series, Journal of Machine Learning Research and Machine Learning Journal. Title, Abstract, Author, Logo and Keywords were selected as layout components of interest, for a total of 1567 instances. Overall, this yields 1312 posi-

tive/negative examples for classification and 6268 for understanding. Note that, in FOL learning, the number of examples is not important, since the descriptions are human-understandable and hence the expert can select fewer but significant cases. FOL layout descriptions of papers were automatically generated by DOMINUS according to the ODA/ODIF standard. The complexity of such a dataset is considerable: the journals layout styles are quite similar (and hence mismatch is easy), and the overall description of the 353 documents consists of 67920 atoms, for an average of 192+ atoms per description (some descriptions are made up of more than 400 atoms). All experiments were run on a PC endowed with a 2.13 GHz processor and 2GB RAM.

For supervised learning, 10-fold cross-validation was exploited. In each fold, the distribution of examples from the 4 classes was uniform. Results were evaluated according to Predictive Accuracy and F-measure (equally weighting Precision and Recall). Classification outcomes show 98% average accuracy and 96% average F-measure. For the understanding task, overall averages are 95% for accuracy and 89% for F-measure. All are statistically significant according to a Wilcoxon test. Each revision takes on average a few seconds for accomplishment, ensuring applicability of the approach to real-world environments. Moreover, even for more strict time requirements, the old version of the theory is still available for use during refinement, so there is no need for hanging up the system.

As to unsupervised learning, for performance evaluation purposes, it was simulated on the same dataset, by hiding the documents class and asking the learner to group them into 4 clusters according to their layout similarity. Then, the results were compared to the correct classes (*supervised clustering*) to evaluate precision, recall and purity. Analysis of the results revealed on average a 91.60% precision, 93.28% recall and 92.35% purity, indicating that the proposed method is highly effective in autonomously recognizing the original classes in spite of the representation-related difficulties. Time performance in this case, being quadratic in the number of documents, is in the order of hours, but the implemented prototype was not optimized for runtime savings and, in any case, the invention of new classes from unknown documents can be done off-line, and new classes can be made available after computation, this way not affecting the normal system behaviour and performance.

## 4. Conclusion

Automatic document processing in Digital Libraries requires intelligent techniques for tackling the domain complexity in a sufficiently flexible way. This paper presented an incremental first-order logic learning framework for dealing with evolution in digital repositories. Indeed, relations can tackle the great variability in kind, content and shape of available documents. Incremental supervised and unsupervised learning techniques can deal with the continuous flow of new documents in digital repositories, progressively adjusting the acquired knowledge on documents as long as new evidence becomes available. Experiments in real-world cases proved the framework to be very effective both in dealing with changes in known classes definition and in integrating new (known or unknown) classes in the library organization, with accuracy always well above 90%, also in the unsupervised case. Future work will concern experimenting with more document types, and the introduction of statistical techniques to further improve performance.

**Acknowledgments**

This work was partially funded by the DDTA "Distretti Digitali per il Tessile-Abbigliamento" Project, Regione Puglia, Italy.

## References

[1] F. Esposito, S. Ferilli, N. Fanizzi, T. M. Basile, and N. Di Mauro. Incremental multistrategy learning for document processing. *Applied Artificial Intelligence: An Internationa Journal*, 17(8/9):859–883, 2003.

[2] F. Esposito, S. Ferilli, N. D. Mauro, and T. Basile. Incremental learning of first order logic theories for the automatic annotations of web documents. In *Proceedings of ICDAR-2007*, pages 1093–1097. IEEE Computer Society, Los Alamitos, CA, September 2007.

[3] S. Ferilli, T. Basile, N. D. Mauro, M. Biba, and F. Esposito. Generalization-based similarity for conceptual clustering. In *MCD-2007*, volume 4944 of *Lecture Notes in Artificial Intelligence*, pages 13–26. Springer, 2008.

[4] E. Fox. How to make intelligent digital libraries. In *Proceedings of ISMIS94*, volume 869 of *Lecture Notes in Artificial Intelligence*, pages 27–38. Springer, 1994.

[5] A. Jain, M. Murty, and P. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[6] M. Li, X. Chen, X. Li, B. Ma, and P. Vitanyi. The similarity metric, 2003.

[7] D. Lin. An information-theoretic definition of similarity. In *Proc. 15th International Conf. on Machine Learning*, pages 296–304. Morgan Kaufmann, 1998.

[8] G. Nagy. Twenty years of document image analysis in PAMI. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):38–62, 2000.