# Field Extraction by Hybrid Incremental and a-priori Structural Templates

Vincent Poulain d'Andecy
*Laboratory L3i, Univ. of La Rochelle*
*Avenue Michel Crepeau, 17042*
*La Rochelle, France*

Emmanuel Hartmann
*Yooz*
*1 rue Fleming, 17000*
*La Rochelle, France*

Marçal Rusiñol
*Computer Vision Center*
*Edifici O, Univ. Autònoma de Barcelona*
*08193 Bellaterra (Barcelona), Spain.*

***Abstract*—In this paper, we present an incremental framework for extracting information fields from administrative documents. First, we demonstrate some limits of the existing state-of-the-art methods such as the delay of the system efficiency. This is a concern in industrial context when we have only few samples of each document class. Based on this analysis, we propose a hybrid system combining incremental learning by means of itf-df statistics and a-priori generic models. We report in the experimental section our results obtained with a dataset of real invoices.**

***Keywords*-Layout Analysis, information extraction, incremental learning.**

## I. Introduction

For more than 30 years, digitization of administrative documents has driven scientists and industrial actors to face both the issue of classification and information extraction from administrative documents. An issue is to automatically feed Document Management Systems while trying to minimize human intervention. Early applications have proposed solutions for invoices [1], govemmental registers [2], orders [3], forms [4], etc. Industrial applications such as EMC-CAPTIVA, ITESOFT, ABBYY, etc. have commercialized automatic form processing systems. Today, such applications should deal with heterogenous workflows of administrative documents: *invoices*, *orders*, *tax forms*, *financial reports*, *mail*, *questionaires*, etc. This process is known as Digital Mailroom [5], [6]. However, even if commercial solutions exist, it is far from being a solved problem. The bottleneck comes from the document variability both in terms of document classes and in terms of intra-class layout variability. For instance, each organization produces its own invoice layout. In addition of the heterogeneity, the document distribution is unbalanced. A user can capture a dozen of orders per day, but it captures only few invoices per month and only one or two tax notices per year.

Thus, a challenge is to define performant algorithms to cope with the two classic issues and one original issue:

- support any variability of layouts,
- minimize the end-user effort,
- train and adapt quickly the application [5] based on a one-sample or very few samples.

In this study, we focus on the field extraction issue.

The rest of the paper is organized as follows. We study the state of art in Section II. We demonstrate our choice for selecting an approach fitting with our objectives. In Section III we propose an end-to-end system based on the incremental structural templates algorithm introduced in [6]. A contribution is the experimentation of its robustness during the learning of first samples. We point out some limitations. Another contribution, we propose in Section IV to enhance this incremental approach by combining with an a-priori template. We conclude in Section V.

## II. Motivations

The information extraction process has been a fundamental research topic for decades in the document analysis domain. All technics are always driven by a model. We have found in the state of art several kinds of strategies for both modeling a document and training such models [7].

To cope with the layout structure variability, most of approaches describe the layout in a logical way, i.e. global and local relative positions between the data to extract, and either its page position, or the presence of graphical or textual information in the neighborhood [1], [4]. A very formal way to describe this logical structure is to use a grammar like the DMOS method presented in [2]. The flexibility of using a grammar avoids the training stage and does not require many samples. But the configuration time (the grammar description) is very time consuming and prevent quick adaptations.

Systems like INFORMYS [1], DAVOS [4] and more recently INTELLIX [5] define strategies that can be trained on labeled documents. In [1], the authors define rules for *invoices*. The rules are based on parameters and keywords which are learned from samples. DAVOS and INTELLIX enable the learning from samples of various logical layouts. However, INTELLIX offers an approach where the training data are given by the end-user along the document process. Training data are not given during an a priori training stage. This approach matches our objectives of minimizing the human efforts. Moreover, the end-user only labels the target data. He does not care anymore about technical or structural description. This system performs spatially based approaches by combining both global position information (for fixed position field) and a local template for floating position fields (eg. *amounts*). For floating position fields, few local templates are learned based on top and bottom position of neighbouring words. Authors note that usually 5 to 10 templates must be learned for floating position data. But they do not discuss on robustness. Successfully $85\%$ of fields are extracted.

The measure is done on 10 types of field in an automatic archiving solution.

With a similar paradigm for the training, Rusiñol et al. [6] and Santosh et al. [8] have proposed a graph-model trained by end-user information for processing *invoice* fields and *table* contents respectively. In both cases, end-user information are samples of the target data. The end-user just need to point out and label the data that he wants to output. No document structure information is required from users. Compared to INTELLIX, these approaches enable the learning of a model with only one sample for either fixed position fields or floating position fields. Obviously, these approaches simplify the human effort and cope with our fast adaption constraint.

About the table extraction, Santosh et al. [8] models a table row structure with a graph. The graph is built by analyzing the redundancy of row structures. Nodes of the graph are sequences of words sharing the same format and the same horizontal organization for several text lines. The end-user just needs to label a node in the first detected row. It specifies the information that the system shall output. For instance, he can just label a node as *quantity* to extract *quantities*. Only one page is required to model a table when there is enough row redundancy within the page. The approach is based on a-priori generic knowledge of usual table structure. A table is a sequence of rows. Many rows repeat the same pattern. The limit is in the expectation that the row structure is a discriminant pattern. However, a relevant idea is to base the graph conception on a generic a-priori knowledge of document organization.

About field extraction Rusiñol et al. [6], propose an incremental structural model based on a star graph [9]. The field to extract is defined as the center node of the star graph. All the words around in the neighborhood are satellite nodes. Vertices represent the geometric relation between the center node and satellites. The matching of the satellite nodes within the document infers the field position. Nodes are weighted by an *inverse term frequency-document frequency* (itf-df) statistic. This weight enables the selection of stable satellite nodes among all satellite nodes. Weights are computed along the workflow processing. Even if a single sample is enough to initialize a model, the graph is optimized by pruning non stable nodes after several samples. In any case, the end-user only labels one word: the target data. The rest of the model is automatically estimated and incrementally tuned along the processing of documents. The system seems applicable for non-frequent document classes because only one sample is required to create a model. However, we can wonder about the robustness of this approach while the *itf-df* has no "pruned" the satellite nodes. This method reachs more than $92\%$ of invoice field extraction.

In conclusion, we have chosen to study more deeply the approach proposed in [6]. It fits our main objectives of single-sample training. By the way, the incremental learning of additional samples seems to assume the high performance in extraction for recurrent document types.

## III. Incremental Structural Template Experiments

In this section, we propose the global incremental application based on Rusiñol et al. algorithm. Then, we focus on Rusiñol et al. algorithm to briefly recall the specifications. We study the incrementality in our content. Finally, we discuss our experiments to conclude on identified limitations.

### A. Global test application

We integrated the Incremental Structural Template in a global platform which is summarized by Fig. 1.

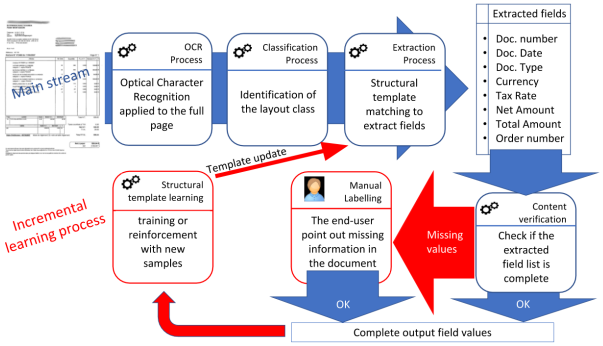


Figure 1. Overview of global system

In the Main stream, documents are recognized by an Optical Character Recognition software (OCR). Next, a classification process is used to identify the layout. Then, fields are extracted by the structural templates learned for the layout. After these steps, we have a list of extracted fields. A verification procedure is performed to check the list completion. Either if the field list is empty (new document type) or some information is missing, the document is pushed to an end-user. The end-user fills the field list by labelling the text which refers to the field in the document. In any case, we output a complete list of fields. This final output list is pushed to the incremental structural learning process. This process initializes new layouts or reinforces existing layouts.

In this study, we use ABBYY as OCR and we simulated manually the layout classification because it was not the focus of our study. However, we assume that technics like the Blur Shape Model [10] can be used to implement a layout classifier.

### B. Incremental Structural Template system from [6]

The overview of the system is summarized by the Fig. 2

We assume that the end-user in our global system can label a target term within a sample document. This term is the target field $F_j$. A star graph $S_j$ is computed automatically by encoding pairwise term relationships between $F_j$ and each other terms $W_i$ of the page (Fig 3). $W_i$ (red boxes) and $F_j$ (blue box) are linked by a spatial relationship quantified by polar coordinates $(\theta_{ij}; r_{ij})$.

But not all $W_i$ have the same semantic importance within the document [11]. Some $W_i$ have not the same impact in the relationship when they are not structurally

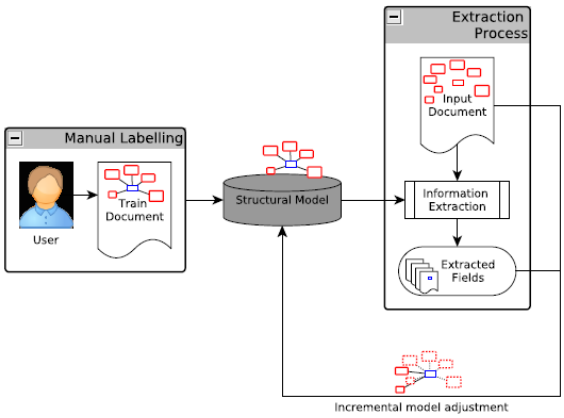


Figure 2. Overview of the incremental system (Fig from [6])

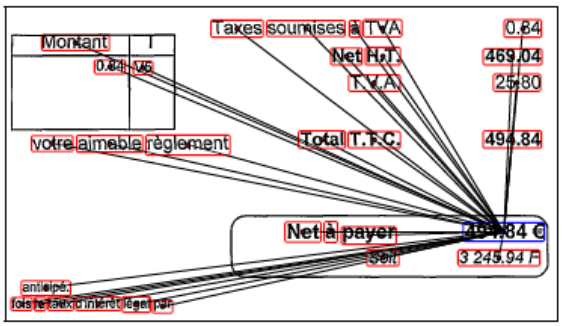


Figure 3. Star graph (Fig from [6])

correlated. Authors have proposed to weight each $W_i$ with an *Inverse Term Frequency-Document Frequency (itf-df)*, which is a reformulation of the classic *Term Frequency-Inverse Document Frequency (tf-idf)*. Given a term transcription $w$ and a document $d$ in a corpus $D$, the *itf-df* weighting scheme is formulated as:

$$\text{itf-df}_{w,d} = \text{itf}_{w,d} \times \text{df}_w, \tag{1}$$

with

$$\text{itf}_{w,d} = \frac{1}{\text{f}(w,d)}, \tag{2}$$

where $f(w,d)$ is the number of times that the term $w$ occurs in document $d$. $\text{itf}_{w,d}$ approaches to 0 the more times $w$ occurs in $d$.

$$\text{df}_w = \log_{10}\left(1 + \frac{9 \times \text{g}(w)}{N}\right), \tag{3}$$

defines the document Frequency where $g(w)$ in the number of documents where the term $w$ appears, and $N$ is the total number of document in $D$. $\text{df}_w$ tends to 0 the less the term $w$ is recurrent in $D$. It models the stability of the term.

The creation of a $S_j$ happens in our system when there is no pair $(L_k, S_j)$ in the Structural Model Database for a given document $k$ attached to a class $L_k$. We initialize the structural template $S_j$ with the current document $k$ for each field $j$. $N$ and $g(w)$ are set to 1. We link $S_j$ to the given class $L_k$. The pair $(L_k, S_j)$, $N$ and $g(w)$ are stored in the Structural Model Database. A pair $(L_k, S_j)$ is created for each field $j$ in the output field list. Hence, the creation of a new template is automatically done when appears the first document sample of a document class.

The incremental reinforcement is performed iteratively along the document processing. We update $S_j$ for each processed document $k$ e.i. for each field j in the output field list. If necessary, we enlarge the graph with new nodes when new terms are observed in the document $k$ for $S_j$. New term $g(w)$ is set to 1. We update existing nodes according to terms observed in the sample. $g(w)$ may be increased by 1. We prune existing nodes when the *itf-df* is low. The pair $(L_k, S_j)$, $N$ and $g(w)$ are updated in the Structural Model Database. In our implementation, terms are raw words given by the OCR with no linguistic pre-procession such as stemming, stopword filtering or lemmmatisation.

Table I
FIELD EXTRACTION RATE (%)

| Run | Doc nbr | Doc date | Doc type | Currency | Net Amount | Total Amount | Tax Rate |
|---|---|---|---|---|---|---|---|
| #1 | 84 | 93 | 95 | 100 | 95 | 95 | 94 |
| #2 | 87 | 95 | 93 | 100 | 95 | 96 | 100 |

### C. Experimentation and Limitations

The incremental reinforcement on volume has been demonstrated in [6] on invoices. The iterative learning enhances the field extraction accuracy between 0.74 and 5.24 percentage points. A contribution of our proposal is to study the robustness of this approach on a one-sample training when the *itf-df* weighting is not yet relevant. We chose to experiment in the same domain, i.e. invoice documents, to make the comparison easiest.

Our study dataset is composed of 270 documents. In detail, we have 3 or 4 samples for 80 classes of invoices chosen randomly among an end-user corpus. We assume this dataset is representative of real layout variabilities. We groundtruthed (GT) 7 fields for each document: DocNbr, DocDate, DocType, Currency, NetAmount, TotalAmount, TaxRate. These 7 fields define our output field list. Our test protocol is the following: the end-user picks up one document from each class to initialize the learning. Therefore, the manual labelling is triggered on each sample. The end-user labels each field of the output field list. The 2 or 3 remaining documents of each class are feed into the system to measure the field extraction rate. The field extraction rate is the ratio between the number of extracted field matching the GT by the total number of field. The experience is performed twice with two different documents for the learning. Table I shows the result.

We observe:

- the extraction rate is very high despite using a single document instance for training. But the learning

sample has an impact (runs #1 and #2 do not produce the same result for 5 fields among 7),
- Doc Nbr is 10% less performant than all others,
- Tax Rate behaves very differently depending on which is the training document.

A deep analysis of these cases reveals that two main limitations can explain almost all these facts: small vertical shifting context (VSC) and the human labeling error (HLE).

VSC is illustrated by fig 4 and 5. There, the insertion of few text lines invalidates the spatial relationship between $F_j$ and most $W_i$ terms (above or below the insertion depending $F_j$ position). In fig 4, $S_{netamount}$ matches "by luck". The insertion leads most $W_i$ to a non-solution. This dead-end gives a chance for a second solution which matches with only few terms. In Figs 5 and 6, $S_{DocNbr}$ fails because a line in the neighborhood has been inserted. This insertion introduces a shift between the above and below terms. Unfortunately, Most bottom terms infer a wrong term $F_j$ against top terms inference.

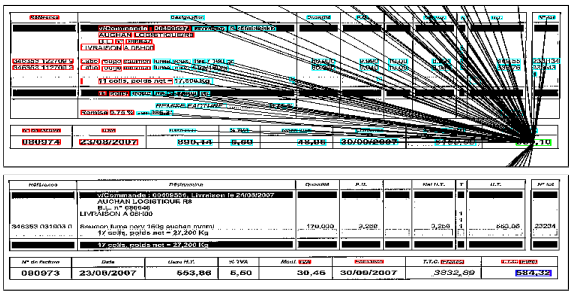

Figure 4. Learned $S_{netamount}$ and matching success even if the line the bottom of table has disappeared. It marches the second table header.

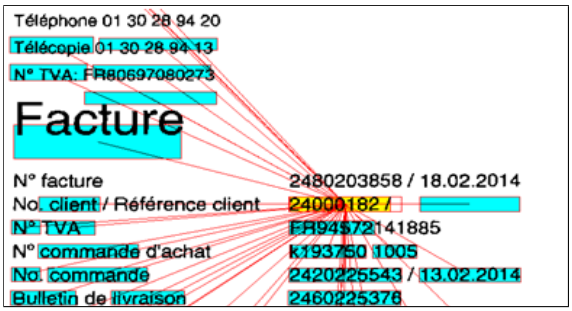


Figure 5. Learned $S_{DocNbr}$ matching failure because the line "No. client / Reference client 24000182/" has been inserted. Unfortunately bottom nodes match better that top nodes.

The iterative learning of *itf-df* weights should enable the selection of stable pairs $(F_j, W_i)$ to cope with this issue. To verify, we extend our test set with 100 additional documents. These documents are from only one class where we observed a VSC issue on a $S_{DocNbr}$ (Fig 5). However, this test set contains 2 different categories of pages: pages with the shift and pages without the shift.

Starting with a virgin Structural Model Database, we feed sequentially our system with the 100 samples. $S_{DocNbr}$ is initialized on the first sample. $S_{DocNbr}$ is iteratively updated according to our system.

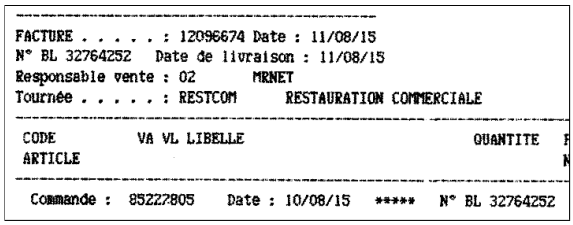


Figure 6. Ambiguous $F_j$, the target "No BL 32764252" appears twice (1) below "Facture" and (2) within the table (bottom). Which is the good one?

Along this iterative runtime, we measure sample per sample how many $F_{DocNbr}$ have been correctly extracted since the beginning of the test. We run this protocol with 3 different pruning thresholds: 0.5, 0.7 and, 0.8. Fig 7 shows the measure. We report also in Fig 7, the page category of the processed sample (green line). Page category is set to 0.1 if the page contains a shift. We set 0.2 when the page does not contain the shift.

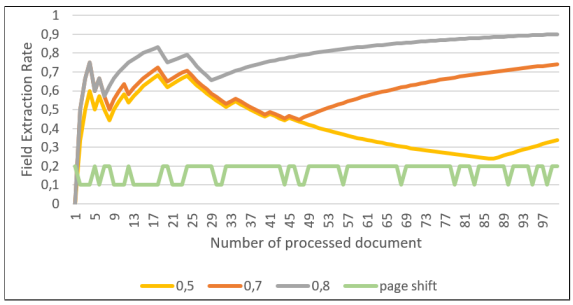


Figure 7. Extraction rate of a $S_{DocNbr}$ along the iterative processing of 100 smaples from one document class containng shifts witch impact the extraction.

We observe that (1) whatever the pruning threshold, the extraction fails at the beginning (0% of extraction, it is conform with the VSC failure). (2) After several iterations, the performance improves. But the result is chaotic depending the type of page (with or without the shift). (3) after more training, the model becomes more stable. The extraction becomes robust. We see a delay depending on the pruning threshold. For 0,5, we need around 85 samples to stabilize the weighting. For 0.7, we need around 50. For 0.8, only 30 are enough. The pruning deletes inrelevant pairs $(F_j, W_i)$. Obviously, the selection of the best pairs i.e. highest *itf-df* improves the performance. With the threshold at 0.8, we have around 80% of success after 50 samples only.

In conclusion, it proves the benefit of the incremental learning approach. But the drawback is a delay to get good performances.

HLE issue comes from an ambiguous context in the training document. The value of the target field appears twice in the document. The end-user picks up the wrong one. The issue is because the semantic of the two terms is close to. It traps the user. In consequence, a wrong $S_n$ is learned (Fig 6). If the user does not repeat the error, the incremental learning will fix the issue itself [6].

In conclusion, we propose an incremental learning sys-

tem based on end-user samples. This system relies on an existing approach. This is not a novelty but a contribution is a deep analysis of algorithm limitations. Despite some advantages like the one-sample learning, this system is not robust in this case. We propose in next section, an evolution of the algorithm to speed-up the learning in order to enable a performant one-sample training.

## IV. Combination of Incremental and a-priori

Inspired by [1], [5], a field can be localized by modelling a semantic spatial horizontal and vertical relationship between the field and somes terms in the document. In Figs. 4, 5 and 6, we can observe this fact. For instance, the invoice number is on the same line, on the left, than the term "n facture". The pair $(F_{docnumber}, W_{nfacture})$ seems the most relevant.

We propose to use a such a-priori knowledge about the "semantic spatial relationship" as a structural model to boost the incremental learning. We propose to enhance the term weight $TW$ of a term $w$ involved in the graph $S_j$ by a combination of the *itf-df* and an a-priori model $M$ given for the field $F_j$.

$$TW_{w,f,d} = M_{w,f} \times \text{itf-df}_{w,d} \tag{4}$$

where itf-df$_{w,d}$ is given by eq. 1, $M_{w,f}$ is the given a-priori structural model of a known spatial relationship between a term $w$ and a field $f$. $M_{w,f}$ filters the itf-df$_{w,d}$ by muting terms which do not fit the a-priori template.

This formula is very generic to cope with many known context. For instance in a form, for a regular field which is designated by a keyword $KY$ on its left, $M_{w,f}$ can be defined as "the closest term $KY$ on the left of f". Formally, a possible formula could be

$$M(w,f) = \{1/(1+r_{wf})\ if\ \theta_{wf} = 270\ and\ w = KY\ else\ 0\}$$

where $(\theta_{wf}, r_{wf})$ is the polar coordinate computed for encoding the spatial relationship between $w$ and $f$ in the star graph $S_f$. In this case, $TW_{w,f,d}$ is valorized only if the end-user train a field having $KY$ on its left . The star graph $S_f$ is limited to only one pair $(F, WKY)$.

For a cell in a table, $M_{w,f}$ can be defined as "terms $w$ close to the document left margin and same line of $f$ or terms above $f$". Formally, a possible formula could be

$$\begin{aligned} M(w,f) = &\{(1/(1+L)\ if\ \theta_{wf} = 180\ else\ 1)\ or \\ &(1\ if\ \theta_{wf} = 0\ else\ 0)\} \end{aligned} \tag{5}$$

Where $L$ is the left coordinate of the term $w$.

In our study, we consider the following contest to define our a-priori structural model.

- Documents respect a Manhattan organization. Horizontal and vertical alignment have more sense than diagonal.
- Fields are usually introduced by a keyword on the same line and, on their left. But sometimes we can figure out a field by an information on its right. For instance, the currency symbol (\$, €, ...) can designate an amount (\$200, 60€) on both direction.
- Even if fields are introduced by keywords, we do not want to define any a-priori dictionary. Indeed the building of the dictionaries could be time consuming.
- Keywords which designate a field are usually the closest words.

Based on these specifications, we propose a very simple and generic model for $M_{w,f}$ as "the closest term $w$ on the same line of $f$". Formally, we propose this implementation $M_{w,f} = MH(w,f) \times MD(w,f)$ where $MH$ weights the horizontal alignment of $w$ and $f$ and $MD$ weights the distance between $w$ and $f$.

$$MH(w,f) = \{1\ if\ |\theta_{wf}| < \alpha\ or\ |\theta_{wf} - 180| < \alpha)\ else\ Mute\}$$

$$MD(w,f) = 1/(1+r_{wf})$$

where $\alpha$ is a horizontality decision parameter (threshold on the angle). $Mute$ is a parameter to filter words which are not aligned with $F$.

MH keeps in the graph any horizontal terms aligned with the center node $F_j$. To accept left or right relative position, both angle 0 and 180 are evaluated. The threshold parameter on the angle introduces a tolerance on small skews. Non-horizontal relationship are penalized by the $Mute$ factor. This factor can be interpreted as "how many non-horizontal terms can compensate one missing horizontal term".

We set our parameter $\alpha$ to 1.08 by experiments. We set $Mute$ to a positive value to enable the learning without any horizontal terms. 0.01 was fixed by experiments to not impact the graph matching when horizontal terms exist. We have replicated the same protocol as described in section III. We reused our first corpus of 270 documents. We performed 2 runs with the new method to measure the stability. We have extended this corpus with 106 new invoices, 7 layouts, 10 fields each. Compared to the first corpus, 3 new fields are introduced: delivery Nbr, Order Nbr, Due Date. The table III summarizes our measure, compared with the best run with original incremental approach (itf-df).

The performance and the stabitlity demonstrate that we can learn quickly and efficiently with one-sample. The $DocNbr$ $DeliveryNbr$, $OrderNbr$ extraction rates are coherent with other fields. Most of the remaining issues are OCR issues.

The main reason of the improvement comes from the early graph pruning. The a-priori model selects best nodes without the need of the itf-df. A test on the 100 samples document class (section III) shows that we reach now the best performance at the first sample while we need 30 samples with the itf-df.

However, the itf-df remains interesting in the formula.

It refines horizontal terms. The a-priori model boosts the selection of horizontal terms. At the end, the itf-df prunes the most stable terms among horizontal terms. It

Table II
FIELD EXTRACTION RATE (%)

| Field | Corpus 1 (270) | | | Corpus 2 (106) | | |
|---|---|---|---|---|---|---|
| | itf-df | itf-df* $M_{w,f}$ 1 | itf-df* $M_{w,f}$ 2 | itf-df | itf-df* $M_{w,f}$ 1 | itf-df* $M_{w,f}$ 2 |
| Doc Nbr | 87% | 94% | 94% | 88% | 93% | 93% |
| Doc Date | 95% | 97% | 97% | 99% | 99% | 99% |
| Doc Type | 93% | 96% | 96% | 100% | 100% | 100% |
| Currency | 100% | 100% | 100% | 99% | 99% | 99% |
| NetAmount | 95% | 95% | 95% | 90% | 95% | 95% |
| TotalAmount | 96% | 96% | 96% | 92% | 97% | 97% |
| Tax Rate | 100% | 100% | 100% | 97% | 97% | 97% |
| Delivery Nbr | - | - | - | 82% | 97% | 97% |
| Order Nbr | - | - | - | 87% | 93% | 93% |
| Due Date | - | - | - | 95% | 95% | 95% |

gives a more accurate template. Unfortunately, our small test corpus does not show any benefits.

It enables to fix the HLE issue. In case of human error, the learned graph Sj merges 2 kinds of horizontal nodes. The correct terms and the similar, but wrong, terms cannot be distinguished by the a-priori model. The itf-df allows to distinguish them. For instance after 4 samples where 1 term is an error occurring 1 time and, 1 term is correct recurring 3 times: the weight for the outlier term is $0,51 = log10(1+9/4)$ while the weight of the good term is 0,88.

## V. CONCLUSIONS AND PERSPECTIVES

The configuration and adaptation efforts for Automatic Document Processing system is an important concern. To face this issue for the fields extraction, we propose an incremental learning framework which can be initialized with a one-sample training. A contribution is to deeply study an existing algorithm and demonstrate limitations for the one-sample training. Then, another contribution is to enhance the incremental model with a generic concept of a-priori structural model. We propose an instance for forms. It is based on a simple and efficient horizontal and distance model. Experiences on real documents show a general improvement and moreover, a gain from 87% to 94% on difficult fields. The global performance is excellent.

In perspective, the approach can be extended and experimented with more complex a-priori model. In particular, it can be studied for table extraction.Another perspective is to introduce more semantic in both the apriori model and itd-df based model. Today, terms are raw words from the OCR. Semantic grouping of terms could make sense. In a reverse point of view, the learning of different models for a same type of field should help the building of onlologies, for instance by merging different graphs.

Finally, an interesting perspective is to replace terms by Word2Vec vectors. A question would be to turn a dedicated model from a layout into a general model for several semantically equivalent layouts.

## ACKNOWLEDGMENTS

This work was supported by the Spanish projects TIN2014-52072-P and TIN2017-89779-P and by the CERCA Programme / Generalitat de Catalunya.

## REFERENCES

[1] F. Cesarini, M. Gori, S. Marinai, and G. Soda, "INFORMys : A flexible invoice-like form-reader system," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 7, pp. 730–745, July 1998.

[2] B. Coüasnon, "DMOS, a generic document recognition method: application to table structure analysis in a general and in a specific way," *International Journal of Document Analysis and Recognition*, vol. 8, no. 2, pp. 111–122, June 2006.

[3] A. Belaïd, Y. Belaïd, L. Valverde, and S. Kebairi, "Adaptive technology for mail-order form segmentation," in *Proceedings of the 6th International Conference on Document Analysis and Recognition*, 2001, pp. 689–693.

[4] A. Dengel and F. Dubiel, "Computer understanding of document structure," *International Journal of Imaging Systems and Technology*, vol. 7, no. 4, pp. 271–278, December 1996.

[5] D. Schuster, K. Muthmann, D. Esser, A. Schill, M. Berger, C. Weidling, K. Aliyev, and A. Hofmeier, "Intellix – end-user trained information extraction for document archiving," in *Proceedings of the 12th International Conference on Document Analysis and Recognition*, 2013.

[6] M. Rusiñol, T. Benkhelfallah, and V. dAndecy, "Field extraction from administrative documents by incremental structural templates," in *Proceedings of the 12th International Conference on Document Analysis and Recognition*, 2013.

[7] A. Belaïd, V. d'Andecy, H. Hamza, and Y. Belaïd, "Administrative document analysis and structure," in *Learning Structure and Schemas from Documents*, 2011, pp. 51–72.

[8] K. Santosh and A. Belaïd, "Pattern-based approach to table extraction," in *Proceedings of the Iberian Conference on Pattern Recognition and Image Analysis*, 2013, pp. 766–773.

[9] F. Haray, *Graph Theory*. Addison-Wesley, 1994.

[10] A. Fornés, S. Escalera, J. Lladós, G. Sánchez, and J. Mas, "Hand drawn symbol recognition by blurred shape model descriptor and a multiclass classifier," in *Proceedings of the International Workshop on Graphics Recognition*, 2007, pp. 29–39.

[11] C. Peanho, H. Stagni, F. Soares, and C. da Silva, "Semantic information extraction from images of complex documents," *Applied Intelligence*, vol. 37, no. 4, pp. 543–557, September 2012.