Text segmentation of digitized clinical texts

Cyril Grouin
LIMSI, CNRS, Université Paris-Saclay, F-91405 Orsay
cyril.grouin@limsi.fr

Abstract

In this paper, we present the experiments we made to recover the original page layout structure into two columns from layout damaged digitized files. We designed several CRF-based approaches, either to identify column separator or to classify each token from each line into left or right columns. We achieved our best results with a model trained on homogeneous corpora (only files composed of 2 columns) when classifying each token into left or right columns (overall F-measure of 0.968). Our experiments show it is possible to recover the original layout in columns of digitized documents with results of quality.

Keywords: Digitized texts; Document layout; Natural Language Processing

1. Introduction

1.1. Digitization and page decomposition

The digitization process generally involves several steps (Randriamasy and Vincent, 1994): (i) page decomposition in order to recognize structural and logical units (e.g., blocks, columns, lines) within a page, (ii) characters recognition using either Optical Character Recognition (OCR) or Handwritten Text Recognition (HTR), and (iii) format and layout analysis so as to produce the final layout of a digitized document. The performances of the page decomposition step is crucial for the final output. The delimitation of text blocks, columns, and lines is achieved using the spatial properties of the elements on a page (Ha et al., 1995). Sylwester and Seth (1995) designed a specific algorithm for the bloc segmentation task, based on the XY Cut[1] algorithms series, which relies on costs defined for each possible block decomposition from the image.

More recently, new techniques have been designed to deal with complex document layout (multi-columns, noisy areas, ornamental characters, etc.) or for a particular domain. Specific pre- and post-processing techniques tailored for a given domain tend to improve text extraction from digitized text, as done by Xu et al. (2008) for the biomedical domain. Nikolaou et al. (2010) proposed an algorithm for ancient machine-printed documents (e.g., old books from the 18th century) based on the run length smoothing algorithm (RLSA). They improved the results achieved by the RLSA algorithm for both line, word and character recognitions. Kaur et al. (2013) demonstrated that a pre-processing stage can improve the page decomposition task. Alternatively, post-processing techniques are also used to correct OCR errors which constitute obstacles to many further NLP processes. Those post-processing techniques aim at identifying tables within the text (Kieninger, 1998; Ng et al., 1999) or correcting tokenization errors (Furrer, 2013).

Since performances of page decomposition also depends on the metrics used, Shafait et al. (2007) compared six metrics to evaluate the page decomposition functions from OCR systems in order to select the metric that provides the best decomposition outputs.

[1]XY Cut or top-down algorithms represent the whole document as the root of a tree. Each final segmentation is represented as a leaf in this tree.

1.2. Recovering the original document layout

Nevertheless, the global layout of digitized documents can be lost, especially frontiers of text columns, either because the digitization process does not provide any information of structure, or in case of process done on the OCR outputs using a system which accidentally deletes multi spaces. This can be a real issue for NLP tasks if one can not access the files with original layout structure. This point mainly concerns corpora for which additional process must be done before distribution to partners (namely, a de-identification process in order to preserve the privacy of people mentioned in documents, e.g., patients in clinical records).

In this paper, we investigate how to recover the original page layout structure into two columns from digitized files, using machine-learning approaches through two kinds of experiment: (i) identifying the column each token belongs to and (ii) identifying the separator between both columns. To the best of our knowledge, there is no previous study on this issue.

2. Material and methods

2.1. Corpus

2.1.1. Presentation

Corpus layout and pre-processing Our corpus is composed of digitized clinical texts from the fetopathology unit of the Armand-Trousseau University Hospital (APHP), written in French between 1991 and 1999. A few documents from this corpus is formatted into two columns. This layout corresponds to a specific kind of document (laboratory results), where contact information is given in the left column (i.e., all medical doctor names and phone numbers) while the clinical core content is present in the right column (i.e., chromosomal formulæ). Nonetheless, the text resulting from the OCR function does not include any mark of blocks or columns of text. The content of the two columns appears on the same physical line, where the physical space between both columns is represented by multi-spaces.

In order to preserve the privacy of patients mentioned in those clinical records, an automatic de-identification process has been done to remove personal health identifiers (PHI) through the MEDINA toolkit (Grouin, 2013; Grouin and Zweigenbaum, 2013). De-identification replaced PHI with tags composed of the PHI category and a unique ID

```
Exam no:REF-3206
PATrrOroGiCaL Requested by: Dr FIRST-3207 LAST-3208
EMBRYOTOGY
AND CYTOGENETICS
TABORATORY
Dear Colleague,
Head of Department:
Pr FIRST-1572 LAST-1349
Cytogenetics Unit The amniotic fluid cultures chromosomal exam for your patient:
Phone: TEL-1350
Pr FIRST-1544 LAST-2126
Pr FIRST-1360 LAST-3209 Mrs. FIRST-909 LAST-907
M.D..Ph.D.:
Dr FIRST-1576 LAST-1370 obtained following results
Dr FIRST-1403 LAST-1363
Chromosomal formula: 47, XX, + 13
Conclusion: Regular and Free Trisomy 13.

Sincerely yours.
Doctor LAST-2181 Professor FIRST-1737 LAST-1758
```

Figure 1 – Sample digitized and de-identified clinical text, original layout document (two columns) is lost

```
                          Exam no:REF-3206
PATrrOroGiCaL             Requested by: Dr FIRST-3207 LAST-3208
EMBRYOTOGY
AND CYTOGENETICS
TABORATORY
                          Dear Colleague,
Head of Department:
Pr FIRST-1572 LAST-1349
Cytogenetics Unit         The amniotic fluid cultures chromosomal exam for your patient:
Phone: TEL-1350
Pr FIRST-1544 LAST-2126
Pr FIRST-1360 LAST-3209   Mrs. FIRST-909 LAST-907
M.D..Ph.D.:
Dr FIRST-1576 LAST-1370   obtained following results
Dr FIRST-1403 LAST-1363
                          Chromosomal formula: 47, XX, + 13
                          Conclusion: Regular and Free Trisomy 13.

                          Sincerely yours.
                          Doctor LAST-2181 Professor FIRST-1737 LAST-1758
```

Figure 2 – Expected output (left and right columns correctly identified and separated)

(e.g., "FIRST-1572" for a first name, "LAST-1349" for a last name, "TEL-1350" for a phone number, etc.).[2] The de-identification process is based on a CRF system which takes as input a tabular file (i.e., one token per line), in order to predict the category each token belongs to (either a PHI category—*first name, last name, date,* etc.—or a null value). Since the tokenization process reduced each multi-spaces into one single space, the original number of spaces between two tokens—even more between tokens from the left and right columns—is lost. Moreover, the token position in the text is not expressed in terms of character offsets. The original document layout can not be easily reproduced. As a consequence, the content from the left and right columns is separated by only one space (see Figure 1). The succession of tokens from left and right columns produces unexpected sequences of tokens which can have a negative impact on further NLP processes.

Pre-processing issue Figure 1 presents a de-identified digitized document from our corpus. The digitization process produced incorrect tokens (e.g., "PATrrOroGiCaL" in-

[2]In this corpus, all tags "FIRST-1572" refer to the same first name (e.g., John). Conversely, "FIRST-1572" and "FIRST-1573" refer to distinct first names or distinct forms from the same first name due to OCR error (e.g., John vs. Jobn). This solution allows to keep the original distribution of data in the whole corpus while preserving the privacy of patients.

stead of "PATHOLOGICAL") and the column segmentation is lost (all tokens from the two columns appear on the same line).
Four kinds of physical lines can be found in this corpus (the right arrow represents the expected separation between left and right columns):

- lines composed of tokens from both left and right columns (e.g., line #2: *PATrrOroGiCAL → Requested by: Dr FIRST-3207 LAST-3208*);
- lines only composed of tokens from the left column (e.g., line #3: *EMBRYOTOGY →*);
- lines only composed of tokens from the right column (e.g., line #1: *→ Exam no:REF-3206*);
- blank lines (e.g., line #18).

As shown on this sample, the sentences are not correctly chained (the dots between square brackets replace the elements from the left column): *The amniotic fluid cultures exam for your patient: [...] Mrs. FIRST-909 LAST-907 [...] obtained following results [...] Chromosomal formula: 47, XX, + 13*. The combination of the two columns produces either sequences of tokens which are not linguistically correct (e.g., *Cytogenetics Unit The amniotic* on line #9) or correct sequences which do not correspond to the original text (e.g., *Dr FIRST-1576 LAST-1370 obtained following results* on line #14). As a consequence, it is necessary to recover the original segmentation into columns, either to remove the left column or to only extract the content of the right column, in order to improve results from further NLP processes on the clinical core content of the document. Figure 2 presents the expected output after automatic columns identification.

2.1.2. Constitution and annotation

Corpus constitution We selected a total number of 265 files from our digitized and de-identified initial corpus, based on the following repartition: all available files formatted into two columns (i.e., 134 single files), and a close number of files formatted into one column (here, 133 files), preserving the wholeness of a clinical record (a clinical record being composed of several files).
We then split this corpus into training and test sets, according to a 60%/40% ratio, considering two kinds of corpora:

- **CORPUS-ALL:** a training set composed of 162 files (50% of files formatted into one column and 50% of files formatted into two columns) and a test set composed of 105 files (same balanced repartition between one or two columns formatted files);
- **CORPUS-2COL:** a sub-corpus only composed of files formatted into two columns: training set of 81 files and test set of 53 files.

Figure 3 presents the corpus production process we followed.

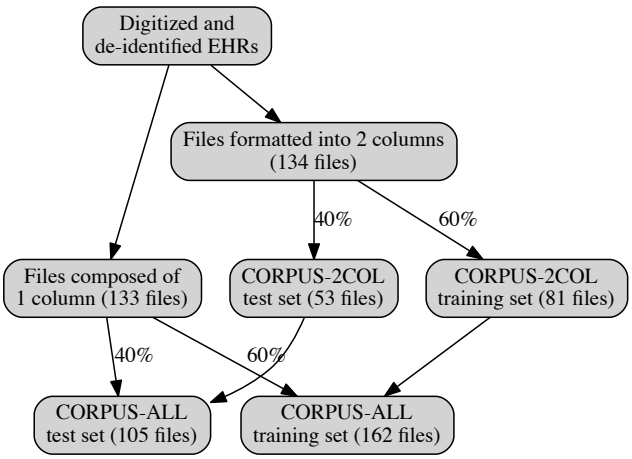

Figure 3 – Corpus constitution

Corpus annotation Since our aim is to identify columns of text from digitized files, we designed a very simple annotation schema based on the following annotation rules:

- for each physical line, to annotate the first token belonging to the right column;
- for lines only composed of tokens from the left column, do not annotate any token;
- for lines only composed of tokens from the right column, to annotate the first token of this line.

Following those principles, only files annotated into two columns (CORPUS-2COL) need to be annotated. One human annotated those 134 files in 70 minutes, using the BRAT annotation tool developed by Stenetorp et al. (2012).

2.1.3. Statistics

Table 1 shows the number and percentage of tokens found in left and right columns, in both training and test corpora from the CORPUS-2COL.

Corpus	Training (81 files)		Test (53 files)	
Left column	8,388	57.6%	6,104	58.8%
Right column	6,172	42.4%	4,281	41.2%
Overall	14,560	100.0%	10,385	100.0%

Table 1 – Number and percentage of tokens found in left and right columns from the CORPUS-2COL training and test sets

We observe that clinical texts from our corpus are composed of more tokens from the left column (58.2%) than tokens from the right column (41.8%). This observation highlights that the clinical core content (more concentrated) uses less space on the page than the contact information (providing an exhaustive list of all contact).
Table 2 shows the number and percentage of lines from the CORPUS-2COL depending on their composition: physical line composed of tokens from both left and right columns, lines only composed of tokens from the left column, lines only composed of tokens from the right column, and blank lines (i.e., no token on this line).

Corpus	Training		Test	
Left and right columns	304	10.6%	215	10.8%
Only left column	1,358	47.3%	977	48.9%
Only right column	520	18.1%	346	17.3%
Blank lines	692	24.1%	461	23.1%

Table 2 – Composition of physical lines (number and percentage) from the CORPUS-2COL training and test sets

We observe that lines only composed of a left column are almost twice as numerous than the three other types of lines (cf. section 2.1.1.). Lines only composed of right column account for 17 to 18% of all lines in the corpus. Since the clinical core content is more varied than contact information, having less lines of this type could make it more difficult to train a robust statistical model.

2.2. Method

2.2.1. CRF-based approach

Our experiments rely on the WAPITI toolkit (Lavergne et al., 2010) which implements the CRF framework (Lafferty et al., 2001). Following features were used: (i) lexical features: the token itself; (ii) typographical features: token length, typographic case of the token, presence of punctuation marks in the token, presence of digits in the token, and Soundex code of the token; (iii) morpho-syntactic features: part-of-speech tag of the token, provided by the Tree Tagger POS tagger (Schmid, 1994); (iv) cluster ID of each token through an automatic unsupervised clustering of all tokens from the corpus into 60 clusters, using the algorithm designed by Brown et al. (1992) and implemented by Liang (2005); and (v) number of lines since the beginning of the file and number of remaining lines until the end of the file.

2.2.2. Design of experiments

We designed two kinds of experiments: first, to identify the column in which each token from a physical line belongs to, and second, the separator between left and right columns on each line. We applied those two experiments on the two corpora we produced (i.e., the whole corpus of 265 files and the sub-corpus of 134 files formatted into two columns, see section 2.1.2.).

Column identification for each token In this first experiment, each token is assigned a LEFT or RIGHT tag to indicate the column the token belongs to. Each token must be classified into one of these two possible columns (a physical line will be composed of zero or more tokens from the left column first, and zero or more tokens from the right column second; the opposite is impossible). We produced two models for this kind of experiment:

- **COLID-all:** model trained on the training set from the whole corpus (i.e., 162 files formatted into one and two columns);
- **COLID-2:** model trained on the training sub-corpus of 81 files formatted into two columns only.

Column separator identification In this second experiment, we only focus on the separator between both columns. This separator could appear at every position on a physical line (e.g., before the first token in case of line only composed of a right column, elsewhere if line is composed of both columns, etc.). We produced two models for this second kind of experiment:

- **COLSEP-all:** model trained on the training set from the whole corpus (i.e., 162 files formatted into one and two columns);
- **COLSEP-2:** model trained on the training sub-corpus of 81 files formatted into two columns only.

2.2.3. Working hypotheses

Through those experiments, we tested the following hypotheses:

- CORPUS-ALL: models trained on corpora combining files composed of one or two columns would be more robust since those models will tackle distinct types of files;
- CORPUS-2COL: models trained on files for which two columns must be found on every file would achieve better results since the corpus is more homogeneous;
- Column separator identification: models trained to identify the separator between columns (COLSEP-*) would achieve lower results since those models have to identify only few tokens in each file;
- Column identification: models trained to identify the column each token belongs to (COLID-*) would be more accurate since all tokens from the corpus must be classified.

3. Evaluation

3.1. Metrics

We evaluate our results using precision (formula 1), recall (formula 2), and F-measure (formula 3, with β=1).

$$\text{Precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \quad (1)$$

$$\text{Recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \quad (2)$$

$$\text{F-measure} = \frac{(1+\beta^2) \times \text{precision} \times \text{recall}}{\beta^2 \times \text{precision} + \text{recall}} \quad (3)$$

3.2. Results

3.2.1. Column identification for each token

COLID models Table 3 presents the results achieved by our COLID models on the experiment of column identification for each token, whether training and test sets are of the same type (experiments #1 and #2, to evaluate the performance of the models) or not (experiments #3 and #4, to evaluate the robustness of the models).

#	model name	training set	test set	category	TP	FP	FN	P	R	F
1	COLID-all	CORPUS-ALL (162 files)	CORPUS-ALL (105 files)	Left column	16,975	1,144	82	0.937	0.995	0.965
				Right column	3,137	82	1,144	0.975	0.733	0.837
				Overall	20,112	1,226	1,226	0.943	0.943	0.943
2	COLID-2	CORPUS-2COL (81 files)	CORPUS-2COL (53 files)	Left column	5,980	207	124	0.967	0.980	0.973
				Right column	4,074	124	207	0.971	0.952	0.961
				Overall	10,054	331	331	0.968	0.968	0.968
3	COLID-all	CORPUS-ALL (162 files)	CORPUS-2COL (53 files)	Left column	6,063	1,144	41	0.841	0.993	0.911
				Right column	3,137	41	1,144	0.987	0.733	0.841
				Overall	9,200	1,185	1,185	0.886	0.886	0.886
4	COLID-2	CORPUS-2COL (81 files)	CORPUS-ALL (105 files)	Left column	8,338	207	8,719	0.976	0.489	0.651
				Right column	4,074	8,719	207	0.319	0.952	0.477
				Overall	12,412	8,926	8,926	0.582	0.582	0.582

Table 3 – Evaluation of COLID models to identify the column of each token on the test set (TP=True positive, FP=False positive, FN=False negative, P=Precision, R=Recall, F=F-measure)

#	model	training set	test set	category	TP	FP	FN	P	R	F
1'	COLSEP-all	CORPUS-ALL (162 files)	CORPUS-ALL (105 files)	Left column	16,917	1,396	140	0.924	0.992	0.957
				Right column	2,969	142	1,312	0.954	0.694	0.803
				Overall	19,886	1,538	1,452	0.928	0.932	0.930
2'	COLSEP-2	CORPUS-2COL (81 files)	CORPUS-2COL (53 files)	Left column	6,022	930	82	0.866	0.987	0.923
				Right column	3,379	91	902	0.974	0.789	0.872
				Overall	9,401	1,021	984	0.902	0.905	0.904
3'	COLSEP-all	CORPUS-ALL (162 files)	CORPUS-2COL (53 files)	Left column	6,054	1,344	50	0.818	0.992	0.897
				Right column	2,969	57	1,312	0.981	0.694	0.813
				Overall	9,023	1,401	1,362	0.866	0.869	0.867
4'	COLSEP-2	CORPUS-2COL (81 files)	CORPUS-ALL (105 files)	Left column	15,255	982	1,802	0.936	0.894	0.916
				Right column	3,379	1,798	902	0.653	0.789	0.715
				Overall	18,634	2,780	2,704	0.870	0.873	0.872

Table 4 – Evaluation of COLSEP models to identify the column of each token on the test set (TP=True positive, FP=False positive, FN=False negative, P=Precision, R=Recall, F=F-measure)

COLSEP models Similarly, we evaluated the performances of the COLSEP models when they are used to infer the column of each token. First, we applied the COLSEP models on the test set. Second, we gave the LEFT and RIGHT tag to each token from a line depending on their position w.r.t. the column separator identified by those models (i.e., all tokens at the left position of the separator are given the LEFT tag while all tokens at the right position of the separator are given the RIGHT tag). Table 4 presents the results we achieved on this experiment.

3.2.2. Column separator identification

COLSEP models Table 5 presents the results achieved by our COLSEP models on the experiment of column separator identification, whether training and test sets are of the same type (experiments #5 and #6 to evaluate the performance of these models) or not (experiments #7 and #8 to evaluate the robustness of these models).

COLID models We also evaluated the performances of the COLID models when the predictions made are used to infer the column separator. First, we applied the COLID models on the test set. Second, we identified the column separator at the frontier between the two columns, based on the two sets of tokens the models identified as belonging either to the left or to the right column. Table 6 presents the performances of COLID models on the test set when they are used to identify the column separator.

4. Discussion

4.1. Corpus variety in training and test sets

A first basic observation concerns the fact that models are more efficient when training and test sets are of the same type, either corpora combining documents formatted into one and two columns, F=0.943 (#1) and F=0.813 (#5), or corpora only composed of documents formatted into two columns, F=0.968 (#2) and F=0.875 (#6). This observation is true for both COLID and COLSEP models. Conversely, we observed that models trained on one type of corpus and applied on the other type always produce lower results. For the COLID models, we achieved a global F-measure of 0.886 (#3) vs. F=0.986 (#2) on the CORPUS-2COL test set, and a global F-measure of 0.582 (#4) vs. F=0.943 (#1) on the CORPUS-ALL test set. For the COLSEP models, we achieved a F-measure of 0.820 (#7) vs. F=0.875 (#6) on the CORPUS-2COL test set, and a F-measure of 0.708 (#8) vs. F=0.813 (#5) on the CORPUS-ALL test set.

#	model name	training set	test set	TP	FP	FN	P	R	F
5	COLSEP-all	CORPUS-ALL (162 files)	CORPUS-ALL (105 files)	405	22	164	0.949	0.712	0.813
6	COLSEP-2	CORPUS-2COL (81 files)	CORPUS-2COL (53 files)	459	21	110	0.956	0.807	0.875
7	COLSEP-all	CORPUS-ALL (162 files)	CORPUS-2COL (53 files)	405	14	164	0.967	0.712	0.820
8	COLSEP-2	CORPUS-2COL (81 files)	CORPUS-ALL (105 files)	459	268	110	0.631	0.807	0.708

Table 5 – Evaluation of COLSEP models to identify the separator of columns on the test set (TP=True positive, FP=False positive, FN=False negative, P=Precision, R=Recall, F=F-measure)

#	model name	training set	test set	TP	FP	FN	P	R	F
5'	COLID-all	CORPUS-ALL (162 files)	CORPUS-ALL (105 files)	414	31	155	0.930	0.728	0.817
6'	COLID-2	CORPUS-2COL (81 files)	CORPUS-2COL (53 files)	488	58	81	0.894	0.858	0.875
7'	COLID-all	CORPUS-ALL (162 files)	CORPUS-2COL (53 files)	414	25	155	0.943	0.728	0.821
8'	COLID-2	CORPUS-2COL (81 files)	CORPUS-ALL (105 files)	488	1,108	81	0.306	0.858	0.451

Table 6 – Evaluation of COLID models to identify the separator of columns on the test set (TP=True positive, FP=False positive, FN=False negative, P=Precision, R=Recall, F=F-measure)

4.2. Document type variety in corpora

In all our experiments, we observed that results are higher when working on CORPUS-2COL (experiments #2 and #6) than CORPUS-ALL (experiments #1 and #5). Indeed, CORPUS-2COL includes files which are all composed of two columns (i.e., two columns must be found in every file) while CORPUS-ALL integrates 50% of two columns files and 50% of one column files, making it more difficult to process (i.e., half of the files is formatted into only one column, the other half into two columns). Conversely, the identification of the column separator produced lower results (F-measures of 0.813 and 0.875 for experiments #5 and #6). Having less annotations makes it more difficult to identify the correct separator of columns. For those experiments, the main errors (false negatives as well as false positives) concern two difficult cases. First, to identify whether ambiguous tokens (namely, titles "Dr" and "Pr") refer to a contact information from the left column or a signature from the right column (at the end of the clinical text). Second, to identify lines only composed of tokens from the right column.

4.3. Models performances

We achieved better results using the COLID models than using the COLSEP models, as using training and test sets composed of documents formatted into both one-column and two-columns layout (experiment #1, F=0.943 vs. experiment #5, F=0.813) as using training and test sets only composed of documents formatted into two columns (experiment #2, F=0.968 vs. experiment #6, F=0.875). We assume that results are better for COLID models because tokens are more well balanced between the two categories to predict (left and right columns, cf. Table 1) than predictions of column separators (at most, only one token per line is a separator while other tokens are not).

Moreover, we noticed the COLID models achieved higher recall values on the left column category than the right column. Since more predictions are made for the left column category, we increase the probability to obtain higher recall values for this category.

4.4. Models use and misuse

A last observation concerns the use and the "misuse" of COLID and COLSEP models. We consider a misuse exists when a model is used for a different task than the one it has been designed for (i.e., to predict the column of each token for a column separator identification model, or to predict the column separator for a column identification model). Nevertheless, this misuse is possible since we can infer the expected type of value for the evaluation from all predictions (more details in section 3.2.1. for COLSEP models and section 3.2.2. for COLID models).

Column identification for each token The use of COLID models allows us to obtain better results (experiments #1 to #4) than the misuse of COLSEP models (experiments #1' to #4') which were not designed to predict the column each token belongs to. One noticeable exception concerns the experiment #4' (F=0.872) which strongly outperforms the experiment #4 (F=0.582). We observed that the COLID-2 model achieved the lowest results of all COLID models in both experiments #4 (F=0.582) and #8' (F=0.451). An explanation would be the lack of robustness of this specific model (training set only composed of documents formatted in two columns vs. test set composed of documents formatted in one and two columns), used to predict the column of each token. In the same situation, the COLSEP model seems to be more robust (experiments #4' and #8).

Column separator identification We obtained similar results, as using COLSEP models (experiments #5 to #8) as misusing COLID models (experiments #5' to #8') which were not designed to identify the column separator. Since the COLID models predict the column each token belongs to, we can use those predictions to easily infer the column separator (i.e., the frontier between both left and right columns). An exception concerns the experiment #8' (F=0.451) which obtained lower results than the experiment #8 (F=0.708). Those lower results are due to the corpus variety between training set (CORPUS-2COL, more specific) and test set (CORPUS-ALL, more varied), making it difficult to predict correct values.

#	Output	Left column	Right column
1	Expected:	TroGENEnQ | r:	Reguested bj
	Produced:	TroGENEnQ |	r: Reguested bj
2	Expected:	CtîO0EnEnQUn	Rcguested by Df LAST-3483
	Produced:	CtîO0EnEnQUn Rcguested by Df LAST-3483	
3	Expected:	Allergist:	ventricular dilatation
	Produced:		Allergist: ventricular dilatation
4	Expected:	1. a small craniofacial dysmorphia	
	Produced:	1.	a small craniofacial dysmorphia
5	Expected:		FIRST-2495 LAST-4360
	Produced:	FIRST-2495 LAST-4360	

Table 7 – Distinct types of errors of segmentation produced by our COLID models

#	Output	Left column	Right column
6	Expected:		produced following results
	Produced:	produced following results	
7	Expected:		Chrornosomalfornrula: 47, XY, + 2l
	Produced:	Chrornosomalfornrula: 47, XY, + 2l	
8	Expected:	TEL-4014	Mrs FIRST-4013 LAST-1034
	Produced:	TEL-4014 Mrs FIRST-4013 LAST-1034	

Table 8 – Distinct types of errors of segmentation produced by our COLSEP models

4.5. Error analysis

Table 7 presents the principle types of errors produced while classifying each token from a physical line into left or right columns (COLID models). A few errors of segmentation involve digitization errors, making it difficult to correctly identify the frontier between left and right columns. Produced outputs include errors of classification for one token (Case #1) or empty columns (Case #2). Another kinds of errors concern punctuation marks and numbered lists, producing wrong segmentation (Case #3) and unexpected segmentation (Case #4). At last, confusions occur between signatures from the right column and contact information generally found in the left column (Case #5).

Table 8 presents the principle types of errors produced while identifying the separator of columns for a physical line (COLSEP models). We observed that errors are of similar type than those produced using our COLID models. They involve empty left columns (Case #6), digitization errors (Case #7), punctuation marks, etc. Moreover, PHI tags—mainly found in left columns—also caused errors when first name and last name of patients are found in right columns (Case #8).

5. Conclusions

In this paper, we presented the CRF-based experiments we made to recover the original page decomposition into two columns from layout damaged digitized files. In this aim, we compared two approaches: first, to classify each token from each physical line according to the column it belongs to, and second, to identify the separator between left and right columns. Additionally, we compared the impact of the corpus used as training set (either only files formatted into two columns or a balanced combination of files formatted into one or two columns) on the outputs. We also evaluated the performance of those models when they are used for a different task than the one they have been originally designed for.

We achieved better results when classifying each token into left or right columns (COLID models, F-measure ranging from 0.943 to 0.968) rather than identifying the column separator (COLSEP models, F-measure ranging from 0.813 to 0.875). Moreover, models trained and applied on sub-corpora of files only formatted into two columns (COLID-2 and COLSEP-2 models) outperformed models trained and applied on the whole corpus combining files of one and two columns (COLID-all and COLSEP-all models): we obtained F-measure of 0.968 (COLID-2) and 0.875 (COLSEP-2) vs. F-measure of 0.943 (COLID-all) and 0.813 (COLSEP-all). Nevertheless, differences are not so high between results obtained with models *-2 and *-all. Our experiments show it is possible to recover the original layout in columns of digitized documents with results of quality. Next steps consist to apply NLP tools on those outputs and to evaluate the impact of this recovering on further processes.

There is still room for improvement, particularly to ensure the robustness of our models. The models we created took as feature the token found in corpus, which limits their robustness since unknown tokens found in a new corpus to process would be processed with less accuracy. We plan to design new experiments to be less dependent of surface forms, especially using global layout statistics. Moreover, we believe an approach composed of two steps, a first one to identify documents formatted into two columns, and a second one to recover the original layout of document previously identified as being composed of two columns, would be more suitable.

6. Acknowledgments

This work was supported by the French National Agency for Research under grant ACCORDYS[3] (ANR-12-CORD-0007). The clinical records from the corpus have been selected by Dr. Marie Gonzales, Dr. Ferdinand Dhombres, and Pr. Jean-Marie Jouannic. We thank the reviewers for their insightful comments.

7. References

Brown, P. F., Della Pietra, V. J., de Souza, P. V., Lai, J. C., and Mercer, R. L. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–79.

Furrer, L. (2013). *Unsupervised Text Segmentation for Correcting OCR Errors*. Ph.D. thesis, Universität Zürich.

Grouin, C. and Zweigenbaum, P. (2013). Automatic de-identification of French clinical records: Comparison of rule-based and machine-learning approches. In *Stud Health Technol Inform*, volume 192, pages 476–80, Copenhagen, Denmark. MEDINFO.

Grouin, C. (2013). *Anonymisation de documents cliniques : performances et limites des méthodes symboliques et par apprentissage statistique*. Ph.D. thesis, University Pierre and Marie Curie (UPMC), Paris, France.

Ha, J., Haralick, R. M., and Philips, I. T. (1995). Document page decomposition by the bounding-box projection technique. In *Proc of ICDAR*.

Kaur, S., Mann, P. S., and Khurana, S. (2013). Page segmentation in OCR system- a review. *International Journal of Computer Science and Information Technologies*, 4(3):420–2.

Kieninger, T. G. (1998). Table structure recognition based on robust block segmentation. In *Proc of SPIE*, volume 3305, pages 22–32.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. In *Proc of ICML*, pages 282–9, Williamstown, MA.

Lavergne, T., Cappé, O., and Yvon, F. (2010). Practical very large scale CRFs. In *Proc of ACL*, pages 504–13, Uppsala, Sweden, July.

Liang, P. (2005). Semi-supervised learning for natural language. Master's thesis, Massachusetts Institute of Technology.

Ng, H. T., Lim, C. Y., and Koo, J. L. T. (1999). Learning to recognize tables in free text. In *Proc of ACL*, pages 443–50, Stroudsburg, PA, USA. Association for Computational Linguistics.

Nikolaou, N., Makridis, M., Gatos, B., Stamatopoulos, N., and Papamarkos, N. (2010). Segmentation of historical machine-printed documents using adaptive run length smoothing and skeleton segmentation paths. *Image and Vision Computing*, 28:590–604.

Randriamasy, S. and Vincent, L. (1994). Benchmarking page segmentation algorithms. In *Proc of Computer Vision and Pattern Recognition*, pages 411–416, Seattle, WA.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proc of International Conference on New Methods in Language*.

Shafait, F., Keysers, D., and Breuel, T. M. (2007). Performance evaluation and benchmarking of six page segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):941–54.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). BRAT: a web-based tool for NLP-assisted text annotation. In *Proc of EACL Demonstrations*, pages 102–7, Avignon, France. ACL.

Sylwester, D. and Seth, S. (1995). A trainable, single-pass algorithm for column segmentation. In *Proc of Conference on Document Analysis and Recognition*.

Xu, S., McCusker, J., Schultz, M., and Krauthammer, M. (2008). Improving OCR performance in biomedical literature retrieval through preprocessing and postprocessing. In *Proc of International Conference on Intelligent Systems for Molecular Biology*, Toronto, Canada.

[3] ACCORDYS: Agrégation de Contenus et de COnnaissances pour Raisonner à partir de cas de DYSmorphologie fœtale (Content and knowledge aggregation to reason about cases of fetal dysmorphology).