Document Layout Analysis for Semantic Information Extraction

W. T. Adrian[1,2], N. Leone[1], M. Manna[1], C. Marte[1]

[1]Department of Mathematics and Computer Science, University of Calabria, Italy

[2]AGH University of Science and Technology, Krakow, Poland

{w.adrian,leone,manna,marte}@mat.unical.it

AI*IA 2017

UNIVERSITÀ DELLA CALABRIA

BACKGROUND

- Due to the increasing volume of electronic documents, it is important to devise and refine techniques to automatically analyze and understand their content
- ***Document Layout Analysis (DLA)*** is the process of decomposing a document into its component regions and understanding their functional roles and relationships
- The field of DLA is concerned with geometrical and logical labeling of document content
- Among the phases of DLA, *page decomposition* is a key phase that aims to identify maximal "homogeneous" regions of the document
- Existing approaches to page decomposition exploit different approaches/techniques :
  - *Bottom-up* (or aggregating method)
  - *Top-down* (or divisive method)

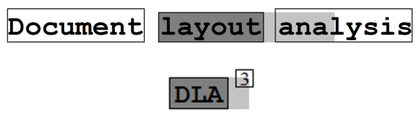
MOTIVATION AND OBJECTIVES

- Despite the progress observed over the last years, **challenges are still open**: they range from accurately detecting content boxes to classifying them into semantically meaningful classes
- The variety of document styles and formats makes non-trivial the task of automatically recognize the structure of arbitrary documents
- Our interest in document layout analysis mainly derives from actual needs of **improving Information Extraction** performances over complex real-world documents sharing some "key information"
- Curriculum Vitae represent one possible interesting use-case

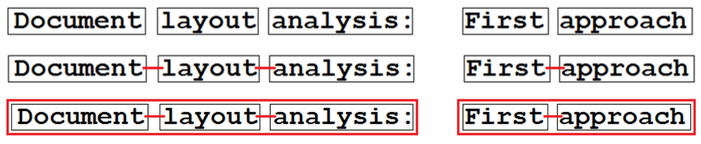
DOCUMENT CHARACTERIZATION

- From a logical viewpoint each word can be characterized as a tuple $\mathbf{w} = (\boldsymbol{id}, \boldsymbol{val}, \boldsymbol{x_1}, \boldsymbol{y_1}, \boldsymbol{x_2}, \boldsymbol{y_2}, \boldsymbol{\kappa})$

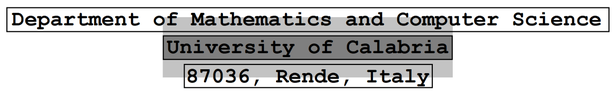
- **Definition 1**. Give to words w and w', we say that $\boldsymbol{w'}$ **follows** $\boldsymbol{w}$ if $x_2 < x_1'$ and if at least one of the following conditions is satisfied:
  1. $y_2' \leq y_1 \leq y_1' \in x_1' - x_1 < \kappa$;
  2. $y_2' \leq y_2 \leq y_1' \in x_1' - x_2 < \kappa$;
  3. $y_2 < y_2' < y_1' < y_1 \in x_1' - x_2 < \kappa$;



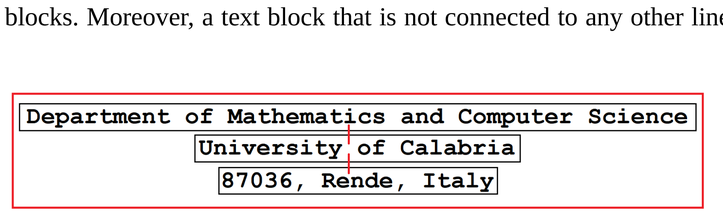
- **Definition 2**. A **line block** is formally defined as any maximal connected component of the graph associated to the document. By definition, a word that is not followed by and does not follows any other word is trivially a line block.



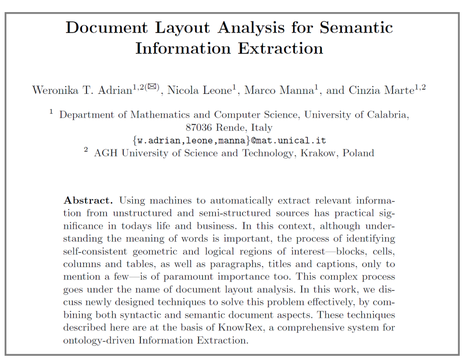
- **Definition 3**. Give two line blocks LB_{Down} and LB'_{Up}, we say that LB_{Down} and LB'_{Up} are **linked lines** if $y_T < y'_B$ and if at least one of the following conditions is satisfied :
  1. $x'_L \leq x_L \leq x'_R$;
  2. $x'_L \leq x_R \leq x'_R$;
  3. $x_L < x'_L < x'_R < x_R$.



- **Definition 4**. A **text block** is formally defined as any maximal connected component of the graph associated to line blocks. Moreover, a text block that is not connected to any other line block is, trivially, a text block.



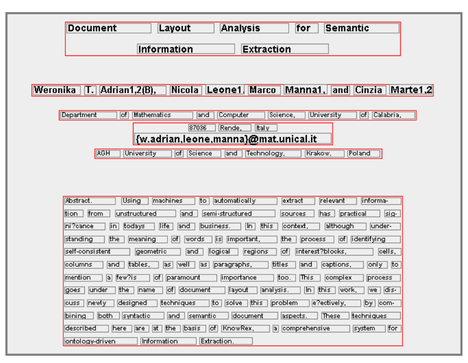
Real result on a PDF document

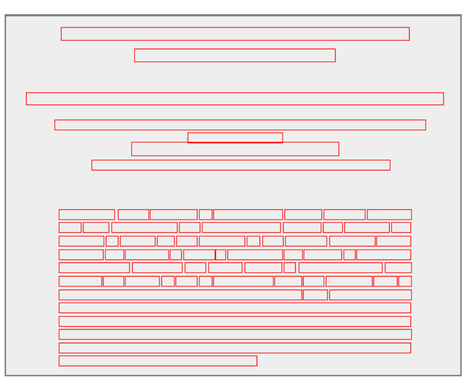

Document Layout Analysis for Semantic Information Extraction

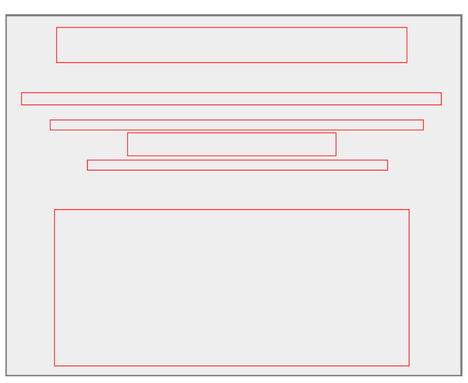
Weronika T. Adrian[1,2(✉)], Nicola Leone[1], Marco Manna[1], and Cinzia Marte[1,2]

[1] Department of Mathematics and Computer Science, University of Calabria, 87036 Rende, Italy
{w.adrian,leone,manna}@mat.unical.it
[2] AGH University of Science and Technology, Krakow, Poland

Abstract. Using machines to automatically extract relevant information from unstructured and semi-structured sources has practical significance in todays life and business. In this context, although understanding the meaning of words is important, the process of identifying self-consistent geometric and logical regions of interest—blocks, cells, columns and tables, as well as paragraphs, titles and captions, only to mention a few—is of paramount importance too. This complex process goes under the name of document layout analysis. In this work, we discuss newly designed techniques to solve this problem effectively, by combining both syntactic and semantic document aspects. These techniques described here are at the basis of KnowRex, a comprehensive system for ontology-driven Information Extraction.







LOGIC-BASED ENCODING

- Our input consists of a set of words that can be represent by a fact of the form

```
word(Id, Val, X1, Y1, X2, Y2, K)
```

- To identify pairs of words connected by the relation *follows* we use a logic rule for each condition e.g.:

```
follows(IdL,IdR):-word(IdL,ValL,X1L,Y1L,X2L,Y2L,K1),
                  word(IdR,ValR,X1R,Y1R,X2R,Y2R,K2),
                  Y2R<=Y1L, Y1L<=Y1R, X2L<X1R, Delta=X1R-X2L, Delta<=K1.
```

- To identify the first element of a text line, we use a concept of an "ancestor", which is inferred from the *follows* relation, and then keep it in `inLine` predicate:

```
linkedWord(IdL):-follows(IdL,IdR).  linkedWord(IdR):-follows(IdL,IdR).
isolateWord(Id):-word(Id,_,X1,Y1,X2,Y2,K), not linkedWord(Id).
ancestorOf(IdA,IdB):-follows(IdA,IdB), IdA<IdB.
ancestorOf(IdA,IdC):-ancestorOf(IdA,IdB), follows(IdB,IdC), IdA<IdC.
child(Id):-ancestorOf(_,Id). inLine(Id,Id):-isolateWord(Id).
inLine(IdA,IdA):-ancestorOf(IdA,IdB), not child(IdA).
inLine(IdA,IdB):-ancestorOf(IdA,IdB), not child(IdA).
```

- Afterwards, we calculate the bounding box of the text line:

```
blockLine(IDln,X1,Y1,X2B,Y2B):-word(IDln,_,X1,Y1,X2,Y2,_),
                               word(IdB,_,X1B,Y1B,X2B,Y2B,_),
                               inLine(IDln,IdB), not overcome(IDln,X2B).
```

- Similarily, we build rules that encode conditions for the linked lines.
- Finally, we find the bounding box of the entire text block:

```
ascissaTL(IDblk,X1):-inBlock(IDblk,IDln), blockLine(IDln,X1,Y1,X2,Y2),
                     not existsLowerThanTL(IDblk,X1).
ordinataTL(IDblk,Y1):-inBlock(IDblk,IDln), blockLine(IDln,X1,Y1,X2,Y2),
                      not overcomeTL(IDblk, Y1).
angleTopLeft(IDblk,X1,Y1):-ascissaTL(IDblk,X1), ordinataTL(IDblk,Y1).
```

CONTENT RECOGNITION

- We divide the document in *homogeneous section* and proceed with the structure recognition technique (Fig.1)
- We use the external tool Quablo (http://quablo.eu) to recognize a table or *draw* one from fixed section of the PDF document, obtaining a grid representation (Fig.2)
- Consecutively, we normalize the grid representation with the use of a *heading ontology*, in order to obtain a **two-dimensional representation** of the document (Fig.3)

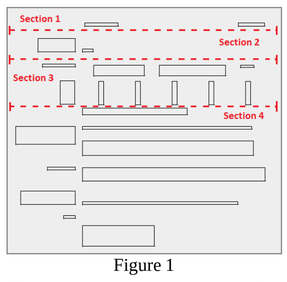


Figure 1

Professional Experience	
Date	1/01/2009 – 30/12/2012
Occupation or position held	Junior Java Programmer
Main activities and responsibilities	Software development in Java. Design and implementation of the user interface.
	Preparing technical documentation of the system.
	Additional technologies used: Ant, Tomcat and Hibernate.
Name and address of employer	IBM, Via Roma 32, Milano. Contatto: contact@ibm.milano.it
Type of business or sector	ICT

Figure 2

eu_cv_workexp_label	
eu_cv_date_label	1/01/2009 fffd 30/12/2012
eu_cv_occupation_label	Junior Java Programmer
eu_cv_activities_label	Software development in Java. Design and implementation of the user interface. Preparing technical
eu_cv_empl_label	IBM, Via Roma 32, Milano. Contatto: contact@ibm.milano.it
eu_cv_business_label	ICT

Figure 3

- This is the base to perform Information Extraction with *extraction rules* that use the spatial context of information

CONTRIBUTION

- We use **Logic Programming** to provide a declarative characterization of the basic geometric properties of text units/block
- We split the document in sections, each containing some "homogeneous" layout
- We exploit external tools for table recognition to detect tables
- We use **Semantic Annotation** to identify/classify atomic text units
- We reason over geometric and semantic information to identify blocks that are "logically connected"

RESEARCH PROPOSAL – FUTURE WORK

Geometric and Semantic Aspects for Document Layout Analysis

Cinzia Marte

For the near future we would like to:

- *dynamically* refine and improve the parameter $\boldsymbol{\kappa}$ deduced in the developed approach, used to provide an upper-bound to the width of a white space placed between consecutive words
- increase the graph representing words and their connection by adding values on the edges, estimating these values both from syntactic and semantic point of view. In particular, we think to define a multi-variable function (one variable for each aspect considered, as semantical, syntactic, etc.), that represents the probability that two words are consecutive
- *determine a reading flow* in order to merge some areas above detected and *automatically* deduce homogeneous section of the document
- use *Disjunctive Logical Programming* to refine the characterization of fixed part of the document and to provide an improved declarative representation of the basic logical properties of the document
- combine efficiently this information, exploiting as much semantic information as possible, in order to obtain a truthful/faithful page segmentation, useful to get better Information Extraction performances over document as Curriculum Vitae