Extracting Structure, Text and Entities from PDF Documents of the Portuguese Legislation

Conference Paper · October 2012

CITATIONS
2

READS
1,289

2 authors:

Nuno Moniz
University of Porto
33 PUBLICATIONS **121** CITATIONS
SEE PROFILE

Fátima Rodrigues
Institute of Engineering of Porto – Polytechnic of Porto (ISEP/IPP)
60 PUBLICATIONS **849** CITATIONS
SEE PROFILE

Extracting Structure, Text and Entities from PDF Documents of the Portuguese Legislation

Nuno Moniz
Institute of Engineering, Polytechnic of Porto, Rua Dr. António Bernardino de Almeida, 4715-357, Porto, Portugal

Fátima Rodrigues
GECAD – Knowledge Engineering and Decision Support Research Center / Computer Engineering Department, Institute of Engineering, Polytechnic of Porto, Rua Dr. António Bernardino de Almeida, 4715-357, Porto, Portugal

Keywords: Information Retrieval, Text Extraction, PDF.

Abstract: This paper presents an approach for text processing of PDF documents with well-defined layout structure. The scope of the approach is to explore the font's structure of PDF documents, using perceptual grouping. It consists on the extraction of text objects from the content stream of the documents and its grouping according to a set criterion, making also use of geometric-based regions in order to achieve the correct reading order. The developed approach processes the PDF documents using logical and structural rules to extract the entities present in them, and returns an optimized XML representation of the PDF document, useful for re-use, for example in text categorization. The system was trained and tested with Portuguese Legislation PDF documents extracted from the electronic Republic's Diary. Evaluation results show that our approach presents good results.

1 INTRODUCTION

The daily increase of information available in the Internet creates the need for tools that are capable of extracting and processing it.

Important sources of information are originally created in the form of text documents. Although stored in computers, these documents do not contain a formal indication about the data types they contain or its own structure. This lack of formal indication prevents the information from being manipulated to meet user's specific needs when accessing/querying/searching it. To make that knowledge computer processable it is necessary to understand the structure of documents, to encode their knowledge and to develop algorithms to bridge the gap between text documents and computer processable representations.

Extracting text from a PDF document is not a direct and simple task. In our research we conclude that OCR is the technology used in most cases (Taylor et al., 1994; Klink and Kieneger, 2001; Todoran et al., 2001; Hollingsworth et al., 2005) due to the attempt to perform text extraction on documents where there is no knowledge of its document's structure. However, in most of the cases mentioned it was concluded that OCR is time consuming and had issues in error recognition.

We can state that a considerable number of public and private organizations that issue official documents regularly adopt well-defined layout structures. These standards include not only the geometric position of text but also its hierarchical structure - differenced fonts, styles and positioning. Using a combination of hereditary and acquired knowledge, we can understand the structure of complex documents without significant effort (Hassan, 2010).

Today there is technology available to parse directly information from PDF documents. We chose to use a free and open source library, iText, described as "a library that allows you to create and manipulate PDF documents."

We found this approach and similar approaches of directly parsing information from PDF documents to be used or described in some of our research (Hassan and Baumgartner, 2005; Antonacopoulos and Coenen, 1999; Rosenfeld et al., 2008; Siefkes, 2003). For grouping text objects these approaches mainly use the font size as criterion for grouping text objects.

In this paper we present an approach for text

Moniz N. and Rodrigues F..
Extracting Structure, Text and Entities from PDF Documents of the Portuguese Legislation.
DOI: 10.5220/0004103501230131
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2012), pages 123-131
ISBN: 978-989-8565-29-7

processing of PDF documents with well-defined layout structures; we used the Portuguese Republic's Diary documents. This approach uses two different extraction methods, according to the two stages of document processing - document analysis and document understanding (Hassan, 2010). The criterion used for grouping the text objects was the font style used in text objects.

The next section presents a general description of the system; it is followed by a section that describes the system implementation and its functionalities as well as the general process; furthermore we present an evaluation of the system performance and the last sections present discussion, future work and conclusions.

2 GENERAL DESCRIPTION

PDF uses a structured binary file format described by a derivation of PostScript page description language. Objects are the basic data structure in a PDF file. For the purposes of this paper we elaborate some of the elements. The content stream is a stream object that contains the sequence of instructions that describe the graphical elements of the page. A dictionary object is an associative table containing key/value pairs of objects. A name object is an atomic symbol uniquely defined by a sequence of characters (Adobe Systems Incorporated, 2008).

PDF document processing can be divided into two phases referring to the two structures in a document: document analysis in order to extract the layout structure and document understanding for mapping the layout structure into a logical structure (Klink et al., 2000). Our approach is divided in three phases: the previous described phases and a third that combines the outputs from the previous phases.

2.1 Document Analysis

The first step in document analysis is layout analysis or segmentation. It consists on parsing a document into atomic blocks. We found in our research two approaches for segmentation: top-down and bottom-up.

The top-down approach is an OCR simulation that usually makes use of whitespace density graphs or similar. This consists on parsing the documents along the x and y axis in order to find whitespace areas. We found reports (Hassan and Baumgartner, 2005) of block recognition problems in certain layouts.

The bottom-up approach can be described as a parsing and grouping process of the smallest segments that share a group of common characteristics such as font size (Hassan and Baumgartner, 2005).

In terms of region comparison, we based our discussion in research made by Antonacopoulos and Coenen (1999), where two categories of methods for region comparison are described: pixel-based and geometric. The geometric is described as the best approach, but the authors openly state their reservations of this approach due to the need of accurate descriptions of the regions.

Regarding segmentation our intended output is not a hierarchical structure but only the coarse-grained regions of each page of the documents, representing in our approach the two halves of the document page, as shown in Figure 1. We will elaborate this option in section 3.3. In the given example, the graphic regions are defined by vertical ruling.

As stated, our approach is destined for known and fixed-structured documents. Therefore, we consider that the top-down approach and the geometric region comparison method is the most proper for this step.

The second step is to extract text from the regions resulting from segmentation. Using the iText library mentioned before, we are able to determine areas to extract text within.

Note that the layout objects extracted are solely for the purpose of extracting text from the PDF file. The output of this phase is an array of text segments, according to the reading order but without any explicit logical structure.

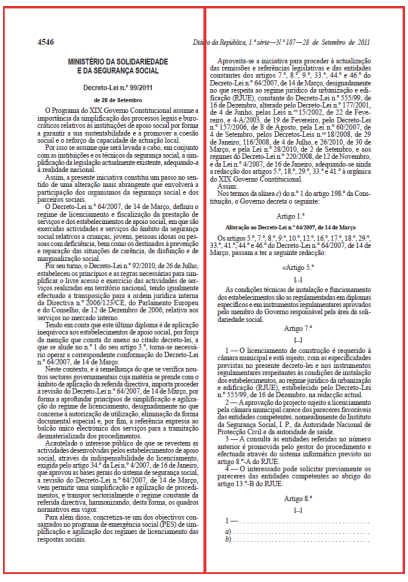


Figure 1: Resulting regions of segmentation process.

2.2 Document Understanding

According to Todoran et al. (2001), document understanding can be divided into two other phases: the process of grouping the layout document objects in order to classify the logical objects; and the process of determining their logical relations.

In order to complete the first phase the best criteria for grouping the layout objects is similar to the perceptual grouping referred by Rosenfeld et al. (2008). Rosenfeld used spatial knowledge to aggregate primitive text objects and create groups of text (line, paragraphs and columns). In our approach we used the font used in the text segments as criteria.

As mentioned before, in the structure of a PDF we are able to find the fonts dictionary in the resources dictionary. An example from Adobe Systems Incorporated (2008) is in Figure 2.

```
3 0 obj
   << /Type /Page
      /Parent 4 0 R
      /MediaBox [0 0 612 792]
      /Resources << /Font << /F3 7 0 R
                             /F5 9 0 R
                             /F7 11 0 R
                          >>
                    /ProcSet [/PDF]
                 >>
      /Contents 12 0 R
      /Thumb 14 0 R
      /Annots [ 23 0 R
                24 0 R
              ]
   >>
endobj
```

Figure 2: Example of Font Dictionary.

Like the work of Giuffrida et al. (2000), Hu et al. (2005) and Hassan and Baumgartner (2005) the use of fonts is present in our approach, although the criterion for grouping objects is different. We implemented a similar approach, but defined the criteria as the font itself, as defined by the content stream of a document.

In text objects from the content stream of a PDF file we can find both objects: name and string. An example from Adobe Systems Incorporated (2008) is presented in Figure 3.

Therefore, we have the objects that are required for grouping according to our criteria. The operators BT and ET represent the beginning and the end of the text object. Using Figure 3 as example, the second line sets the font and the fourth line prints the string.

Based on this we are able to state that by extracting the text objects of a PDF file we are able to group strings by font used. The result is then translated into a XML. Note that this result has no guarantee of being in the correct reading order.

The second phase of document understanding is integrated in the third and final phase of document processing, described as follows.

```
BT
   /F13 12 Tf
   288 720 Td
   (ABC) Tj
ET
```

Figure 3: Text Object.

2.3 Merging Phase

At this point we have two outputs from the previous phases: a complete text description in correct reading order and a XML file with strings tagged and grouped by font used.

Therefore, we are missing two processes: we need to join the two outputs in order to have a XML file that contains the tagged string groups in the correct reading order and, it is necessary to apply the second phase in document understanding, described as the process of determining the logical relations between the groups of objects.

In this approach one logical relation that is dealt from the start, as stated above, is the reading order. Other logical relations have to be inputted by the user of the system, such as the structural relationships between segments (e.g., a paragraph contains lines). Our approach is based on two sets of rules: structural and logical rules. Structural rules are mainly applied in order to classify and create new groups or to re-label the existing ones; syntactical rules are used. Logical rules are applied in order to establish logical relations between groups. Both structural rules and logical rules have their own specific syntax. In Section 3 we will explain them in detail.

The expected output of our approach is a XML file containing the text description of the PDF file, in correct reading order, tagged accordingly and containing logical relations set out by the user.

3 SYSTEM DESCRIPTION

In the previous chapter we presented the general description of our approach. In this chapter we will describe its implementation.

We would like to state that despite the previously presented division of phases, our approach doesn't implement them in the same order.

The implementation has two phases: extraction and analysis.

The extraction phase contains three processes: extraction of information from the PDF's content stream, extraction of text using geometric positioning and merging the output of the two previous processes into a XML file.

The analysis phase contains two processes: application of structural rules and application of logical rules. The system output is a XML file that contains the mapping of the layout structure to a logical structure of the PDF document.

The extraction of text within tables and the extraction of images were not implemented but they are on our future work objectives.

Before the description of the phases and processes, we would like to map our processes with previous research.

3.1 Background

In order to be clear about the influence of the studied approaches, Table 1 represents the mapping of our processes with what we consider to be correspondent to both following descriptions.

Niyogi (1994) presents a description of a computational model for extracting the logical structure of a document, described as follows:

1. a procedure for classifying all the distinct blocks in an image;
2. a procedure for grouping these blocks into logical units;
3. a procedure for determining the read-order of the text blocks within each logical unit;
4. a control mechanism that monitors the above processes and creates the logical representation of the document;
5. a knowledge base containing knowledge about document layout and structure and;
6. a global data structure that maintains the domain and controls data.

Taylor et al. (1994) presents four phases in his implementation:

1. Physical Analysis
2. Logical Analysis
3. Functional Analysis
4. Topical Analysis

We assume that this mapping is not an exact match, but it presents a general idea of the correspondence of processes present in our approach and previous research.

Giuffrida et al. (2000) used spatial knowledge of a given domain knowledge to encode a rule-based system for automatically extracting metadata from research papers; they used spatial knowledge to create a rule; the metadata was extracted from PostScript files and formatting information was used.

Table 1: Mapping of implemented processes with previous research.

Processes	Niyogi (1994)	Taylor et al (1994)
Extraction from Content Stream	1) and 2)	1)
Extraction from Layout	3)	1)
XML output	3)	2)
Application of structural rules	4)	3)
Application of logical rules	4)	4)

Hu et al. (2005) proposed a machine learning approach to title extraction from general documents; tests were made with Word and PowerPoint documents. This method mainly utilizes formatting information such as font size in the models.

Both approaches use formatting information, such as the font used. We use the font as declared in the content stream of PDF documents as criteria for perceptual grouping.

We assumed this option due to the often presence of different styles within text segments of the same font size. Usually this represents an entity; therefore, using the content stream font description as criteria instead of the font size, we enable a better information extraction process.

In the following sections we will describe the processes of our system's implementation.

3.2 Extraction from Content Stream

As explained in Section 2 a PDF document is composed by objects. Regarding this section we will refer only to text objects.

The objective of this process is to extract strings labelled with the font resource declared for its use. This is done by parsing sequentially the content stream extracting each text object and parsing its font and string. Sequential strings that have the same font are grouped. As the results are obtained, they are appended in a XML structure.

After this, two procedures are called: one to extract explicit entities and another to clean the XML.

In the first procedure, as explained before, we use a single criterion of font used. By analysing the fonts used in the Portuguese Republic's Diary, we found that the italic style is most often used to refer to an entity. Therefore, this process consists on the extraction of these explicit entities and its relabeling.

In the second procedure cleaning operations are

made e.g. cleaning empty tags, joining two consecutive objects with the same tag. Also, in this procedure tables are removed. However, before this operation, a regular expression for entity recognition is applied in the text within, in order to extract the entities present in the documents tables.

In Figure 4 we present an excerpt of the auxiliary XML file created to store this information and the respective PDF document.

3.3 Extraction from Layout

In this process we extract text from the PDF document using region filters present in the iText library. We therefore extract text from a known location. The documents we refer are organized in double columns. Therefore, we extract text by setting a vertical ruling in the middle of the page.

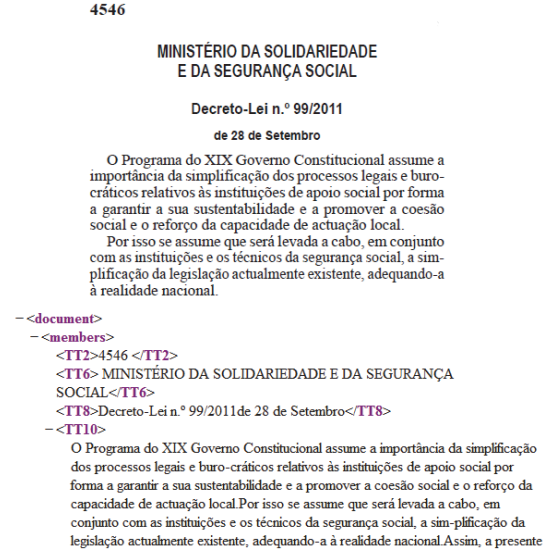
The output of this operation consists on arrays of strings which are joined in order to produce a unique array. This array will be a sequential list, according to the reading order of the text. Each string of the array contains a line of a column.

The sole purpose of this process is to extract text in the correct reading order.

3.4 XML Output

This is the final process after both extraction phases. It consists of sequential comparisons between the previous extractions.

For each line obtained from the Extraction from Layout, a lookup in the auxiliary XML of the Extraction from content stream process is made. The resulting matches are appended into a XML file.


4546

MINISTÉRIO DA SOLIDARIEDADE
E DA SEGURANÇA SOCIAL

Decreto-Lei n.º 99/2011

de 28 de Setembro

O Programa do XIX Governo Constitucional assume a importância da simplificação dos processos legais e burocráticos relativos às instituições de apoio social por forma a garantir a sua sustentabilidade e a promover a coesão social e o reforço da capacidade de actuação local.

Por isso se assume que será levada a cabo, em conjunto com as instituições e os técnicos da segurança social, a simplificação da legislação actualmente existente, adequando-a à realidade nacional.


```
−<document>
 −<members>
    <TT2>4546 </TT2>
    <TT6> MINISTÉRIO DA SOLIDARIEDADE E DA SEGURANÇA SOCIAL</TT6>
    <TT8>Decreto-Lei n.º 99/2011de 28 de Setembro</TT8>
   −<TT10>
      O Programa do XIX Governo Constitucional assume a importância da simplificação dos processos legais e buro-cráticos relativos às instituições de apoio social por forma a garantir a sua sustentabilidade e a promover a coesão social e o reforço da capacidade de actuação local.Por isso se assume que será levada a cabo, em conjunto com as instituições e os técnicos da segurança social, a sim-plificação da legislação actualmente existente, adequando-a à realidade nacional.Assim, a presente
```


Figure 4: Excerpt of the auxiliary XML.

In Figure 5 a bit from the auxiliary XML (not in correct reading order) is presented. In Figure 6 a bit from the XML output is presented. It is possible to denote that the numbers in the TT8 tag are not sequential. We do not detain the necessary information to specify why the iText library is unable to parse the text objects from the content stream in correct reading-order. However, we assume this could be either due to the content stream not having all of its text objects in a sequential manner or due to the use of misleading character recognition because of the use of vectors in that process.


```
<TT6> MINISTÉRIO DOS NEGÓCIOS ESTRANGEIROS</TT6>
<TT8>Aviso n.º 19/2011</TT8>
−<TT10>
  Por ordem superior se torna público que, em 22 de Janeiro de 2009 e e
  dos Negócios Estrangeiros da Ucrânia e pelo Ministério dos Negócios E
  respectivas formalidades constitucionais internas de aprovação do Acor
  Criminalidade, assinado em Lisboa em 24 de Junho de 2008.Pela Parte
  República n.º 75/2010 e ratificado pelo Decreto do Presidente da Repú
  Julho de 2010.Nos termos do artigo 13.º do Acordo, este entrará em vi
  notificação.Direcção-Geral de Politica Externa, 15 de Dezembro de 201
</TT10>
<TT8> Aviso n.º 21/2011</TT8>
```


Figure 5: Bit of auxiliary XML.


```
<TT6>MINISTÉRIO DOS NEGÓCIOS ESTRANGEIROS</TT6>
<TT8>Aviso n.º 19/2011</TT8>
−<TT10>
  Por ordem superior se torna público que, em 22 de Janeiro de 2009 e
  dos Negócios Estrangeiros da Ucrânia e pelo Ministério dos Negócio
  respectivas formalidades constitucionais internas de aprovação do Ac
  Criminalidade, assinado em Lisboa em 24 de Junho de 2008.Pela Par
  República n.º 75/2010 e ratificado pelo Decreto do Presidente da Re
  artigo 13.º do Acordo, este entrará em vigor em 7 de Março de 2011
  Politica Externa, 15 de Dezembro de 2010. O Director-Geral, Nuno
</TT10>
<TT8>Aviso n.º 20/2011</TT8>
```


Figure 6: Bit of XML output.

3.5 Application of Structural and Logical Rules

These are the processes of the analysis phase.

Although these processes are separated, they are implemented using the same paradigm. It consists on a rule based system that is applied according to the syntax defined for each set of rules (structural and logical).

In order to apply these rules it is necessary a user input. This input is done by the declaration of rules in four text files containing the respective structural and logical rules.

Our system embeds operations that enable the application of these rules. The operations are the result of the knowledge acquired from the analysis of the auxiliary XML file – the output of the previous phase.

3.5.1 Structural Rules

The application of structural rules obeys pre-defined types of operations. The structural rules are defined in two separate files.

The first file contains rules relating to operations that include recognition of structure entities (articles, lines, chapters, sections and others), deletion of structure entities and recognition of entities. The second file contains rules to alter original XML tag names to tag names with a meaning.

This list of operations is not static and we believe it will grow according with different PDF documents processed.

The syntax for the specification of rules in the first file is as follows: '`RegExp:::operation`'.

The operation bit represents an internal process encoded in our system. As mentioned, our domain knowledge is based uniquely in the Portuguese Republic's Diary documents. Some examples of these internal processes are insertion after or before the present tag and recognition of structural elements within text objects.

In Figure 7 we present a XML output without the application of any rule. In Figure 8 the same case is presented with the application of an example rule from the first structural rules file, related to the recognition of chapter elements: '`CAPÍTULO\s[IVXLCM]+):::chapter`'.

As an important remark, the extraction of entities is processed at this point with the application of a structural rule. The successful results obtained have no implication in the structure of the text or document; they are stored separately.

The second file of structural rules consists on a list of rules that deal solely with altering the initial XML tag names into the user specified desired tag names.

The syntax for the specification of rules in the second file is as follows: '`RegExp:::previoustag:::newtag`'. The rule may or may not contain regular expressions. In Figure 9 an example rule from the second file is applied to the bit previously presented in Figure 5 where tag <TT6> is replaced by tag <govEntity>: '`^\s?[A-ZÁ-Ü]{2}.+$:::TT6:::govEntity`'.

<TT2>Regulamentação especifica</TT2>
−<TT10>
As condições técnicas de instalação e funcionamento dos estabelecimentos são as regulamentadas em diplomas específicos e em instrumentos regulamentares aprovados pelo membro do Governo responsável pela área da soli-dariedade social.CAPÍTULO II

Figure 7: XML output without structural rules.

<TT2>Regulamentação especifica</TT2>
−<TT10>
As condições técnicas de instalação e funcionamento dos estabelecimentos são as regulamentadas em diplomas específicos e em instrumentos regulamentares aprovados pelo membro do Governo responsável pela área da soli-dariedade social.
</TT10>
<chapter>CAPÍTULO II</chapter>

Figure 8: XML output with structural rules.

<govEntity>MINISTÉRIO DOS NEGÓCIOS ESTRANGEIROS</govEntity>
<TT8>Aviso n.º 19/2011</TT8>
−<TT10>
Por ordem superior se torna público que, em 22 de Janeiro de 2009 e em 8 de Setembro de 2010, foram rece-bidas notas, respectivamente pelo Ministério dos Negócios Estrangeiros da Ucrânia e pelo Ministério dos Negócios Estrangeiros da República Portuguesa, em que se comu-nica terem sido cumpridas as respectivas formalidades constitucionais internas de aprovação do Acordo entre a República Portuguesa e a Ucrânia no Domínio do Combate à Criminalidade, assinado em Lisboa em 24 de Junho de 2008.Pela Parte portuguesa, o presente Acordo foi aprovado pela Resolução da Assembleia da República n.º 75/2010 e ratificado pelo Decreto do Presidente da República n.º 77/2010, publicados no n.º 141, de 22 de

Figure 9: XML output with tag structure rules.

3.5.2 Logical Rules

These rules are defined in two separate files as well.

The logical rules intend to structure the final XML file in order to replicate the information hierarchy present in the original PDF document. This process requires a previous user analysis in order to specify the correct options. For our example, in terms of information hierarchy we find that the Legislation Entity is the most important element in the Portuguese Republic's Diary; each Legislation Entity may or may not have a Sub-Entity; these Entities issue Legislation Documents; a Legislation Document may or may not have a Description; a Legislation Document may or may not be organized by Articles, etc.

In order to reproduce that hierarchy we require two types of processing: a first process where a specific tag appends all the following objects until a similar tag is found; a second process that appends the objects of a specific tag onto another preceding it.

The first logical rules file represents the rules applied for the first process; the second file contains the rules that are applied in order to perform the second process.

The first logical rules file represents a top-down approach of aggregation. It appends every tag onto a specific user defined tag. The syntax for these rules is as follows: '`firstTag:::aggregationTag`'.

The firstTag field represents the parent tag, and the aggregationTag represents the tag to which the following will be appended.

This process is used primarily with the objects that have higher importance in the structure or contains most of the text (for example Legislation Entities and Legislative Documents).

In Figure 10 we present a bit of a XML output with the application of an example rule '`LexEntity:::LexDocument`', from the first logical rules file.

In the previously mentioned figure we can observe the application of a rule that follows what was stated concerning information hierarchy.

The second logical rules file contains rules that have the objective of appending objects with a specific tag onto another user defined tag. The syntax for these rules is as follows: '`parentTag:::tagToAppend`'.

The parentTag field represents the tag onto which the objects will be appended; the second field represents the tag to be appended.

In Figure 11 we present a bit of a XML output with the application of the rule '`line:::paragraph`', from the second logical rules file.

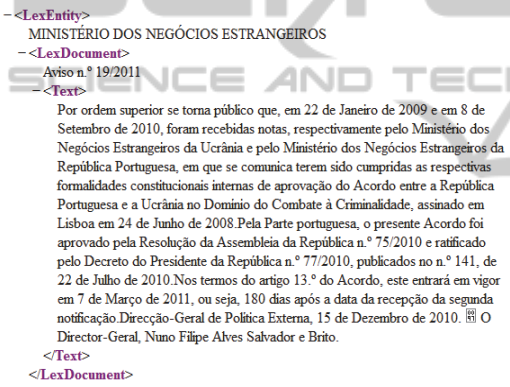


Figure 10: XML output with logical rules.

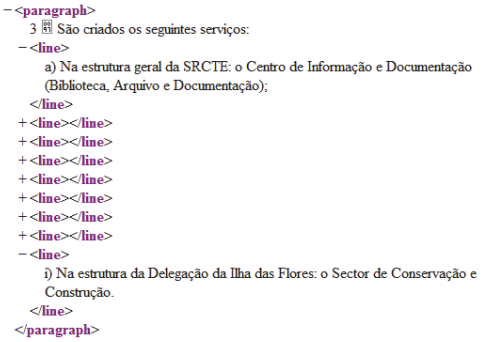


Figure 11: XML output with logical rules.

This is the final process of the analysis phase. The output of this phase is the final XML file that contains the mapping of the layout structure to logical structure of a PDF document.

4 PERFORMANCE

We tested our system with a group of 40 Portuguese Republic's Diary PDF documents. We chose the documents randomly in terms of size and date. For this performance test we did not include the Diaries supplements. Regarding the timeline of the documents, it stands between the 1st of January 2009 and 19th of March 2012. The access to the documents of our sample was done in an online environment – remote access.

For each document in our sample we confirmed if the text extraction was done in a correct and successful manner. The confirmation was based on a manual comparison between the original text in the PDF documents and the XML output. We also confirmed the extraction of entities; it was based on a one-by-one evaluation of each entity extracted.

The documents were graded, in terms of percentage, according to its accuracy in both processes: extracting text and extracting entities. We searched for unsuccessful text extractions and non-entities that were flagged as correct entities.

In our experiments we used the two measures: Text Extraction Accuracy (TEA) and Entity Extraction Accuracy (EEA). The measures were defined as follows:

$$TEA = 1 - (UTE / TTE) \quad (1)$$

$$EEA = 1 - (UEE / TEE) \quad (2)$$

Here, UTE and UEE are defined as Unsuccessful Text Extractions and Unsuccessful Entity Extractions; TTE and TEE are defined as Total of Text Extractions and Total of Entity Extractions. In the following table the results are presented.

Table 2: Results of evaluation.

Period	TEA	EEA
Jan 2009 – Dec 2009	99,82%	93,55%
Jan 2010 – Dec 2010	99,53%	92,55%
Jan 2011 – Dec 2011	99,68%	94,31%
Jan 2009 – Mar 2012	99,73%	93,61%

For both confirmations, partial results were considered as wrong. As for the first confirmation (TEA), the incorrect extractions were promptly pointed by the system. Nonetheless, some results pointed out as incorrect were accepted due to the previous stated expectations: relating to text inside a table, we expect the system to ignore it. As such, these results were considered correct. However, in the second confirmation (EEA), we had to observe and classify one-by-one, each entity. Entities that were incomplete; had incorrect phrasing or minor

errors were considered as wrong.

In the development of this evaluation, despite the well-defined layout structure, we found the use of different and unique combinations of fonts. This caused some of the text extraction errors. Most of the text extraction errors were due to minor incompatibilities (a space character misplaced, for example) between the content stream extraction and the layout extraction. At this point we are improving this situation through trial-and-errors. We are also considering different approaches in order to extract the text from the PDF documents, in the correct reading-order using only its content stream.

To complete this performance evaluation we would like to point out some global indicators that were obtained during this process. They are presented in the following table.

Table 3: Additional evaluation indicators.

Indicator	Result
Average PDF size	696,5 Kb
Average Final XML size	101,5 Kb
Average page number per PDF	23
Average processing time per PDF	12 s
Average processing time per PDF page	0,5 s

5 DISCUSSION AND FUTURE WORK

The main objective of our work was to achieve a structure, text and entities extraction system from PDF documents that would be simple, fast and able to receive inputs from the user. Simple because we still need a solution that is flexible; fast because the volume of PDF documents used requires a system with the ability to process a large number of documents; and a user-guided system, because this is directed for cases where there is more specific knowledge than general knowledge (Klink and Kieneger, 2001), and that specific knowledge is static throughout every document of that type.

There are some immediate subjects to improve or develop in order to achieve a more enthusiastic result.

Tests have shown that due to the often use of unexpected fonts in the text, results can be misleading. However, it showed that although it reduces the ability for classification of the text through a rule based approach, the system still generally recognizes it as valid text strings.

We did not ponder the use of an ontology based component instead of the developed rule based. Nonetheless, this presents an inevitable question for the future, due to the present growth of Semantic Web (Hendler et al., 2002).

We think it will be necessary for a wider and diverse evaluation of the system using different types of documents; this should be critical in order to develop the user-inputs operability and also to increase the error-solving capability.

The application of rules and the extraction of entities are still matters for improvement. Although we obtained good results, we observed certain recurrent errors that we should address. At this point we're dismissing the processing of images and tables. However, the entities inside the tables are processed.

6 CONCLUSIONS

We presented the problem of text extraction in PDF documents with known and fixed layout structures. We presented a grouping-based approach as a possible solution. Furthermore, this solution presents a capability to extract entities present in the text. This approach enables the creation of XML files containing the text and a representation of the PDF documents structure. The main contribution of our work is the development of a user-guided system for text and entities extraction using methods based on our research. By not using OCR technologies and by using geometric-based region representations for segmentation it requires low storage space and low processing time.

We consider we've been able to show that this goal was achieved with some success. Although some improvements have to be made, our preliminary results we're enthusiastic. Nonetheless we reckon the system still requires an extended period of experiments in order to evolve with the processing of more sets of documents.

ACKNOWLEDGEMENTS

The authors would like to thank all the support provided by Knowledge Engineering and Decision Support Research Center.

REFERENCES

Hassan, T. 2010. User-Guided Information Extraction from Print-Oriented Documents. Dissertation. Vienna University of Technology

Taylor, S., Dahl, D., Lipshitz, M. et al. 1994. Integrated Text and Image Understanding for Document Understanding. Unisys Corporation

Klink, S, Kieninger, T. 2001. Rule-based Document Structure Understanding with a Fuzzy Combination of Layout and Textual Features. German Research Center for Artificial Intelligence

Todoran, L., Worring, M., Aiello, M., Monz, C. 2001. Document Understanding for a Broad Class of Documents. ISIS technical report series, Vol. 2001-15

Hollingsworth, B., Lewin, I., Tidhar, D. 2005. Retrieving Hierarchical Text Structure from Typeset Scientific Articles - a Prerequisite for E-Science Text Mining. University of Cambridge Computer Laboratory

Hassan, T., Baumgartner, R. 2005. Intelligent Text Extraction from PDF. Database & Artificial Intelligence Group, Vienna University of Technology, Austria

Antonacopoulos, A., Coenen, F. P. 1999. Region Description and Comparative Analysis Using a Tesseral Representation. Department of Computer Science, University of Liverpool.

Rosenfeld, B., Feldman, R., Aumann, Y. et al. 2008. Structural Extraction from Visual Layout of Documents. *CIKM '02*

Siefkes, C. 2003. Learning to Extract Information for the Semantic Web. Berlin-Brandenburg Graduate School in Distributed Information Systems. Database and Information Systems Group, Freie Universität Berlin

Adobe Systems Incorporated. 2008. Document management — Portable document format — Part 1: PDF 1.7

Klink, S., Dengel, A., Kieninger, T. 2000. Document Structure Analysis Based on Layout and Textual Features. *DAS 2000: Proceedings of the International Workshop of Document Analysis Systems*

Adobe Systems Incorporated. 2008. Document management – Portable document format – Part 1: 1.7.

Niyogi, D. 1994. A Knowledge-Based Approach to Deriving Logical Structure from Document Images. PhD thesis, State University of New York at Buffalo

Hendler, J., Berners-Lee, T., Miller, E., 2002. Integrating Applications on the Semantic Web. Journal of the Institute of Electrical Engineers of Japan, Vol 122(10), October 2002, p.676-680

Hu, Y., Li, H., Cao, Y., Meyerzon, D., Zheng, Q., 2005. Automatic Extraction of Titles from General Documents using Machine Learning. *JCDL '05*

Giuffrida, G., Shek, E., Yang, J., 2000. Knowledge-Based Metadata Extraction from Post-Script Files. *DL'00*