An XML-based Approach for the Presentation and Exploitation of Extracted Information

Manuela Kunze, Dietmar Rösner
Otto-von-Guericke-Universität Magdeburg
Institut für Wissens- und Sprachverarbeitung
P.O.box 4120, 39016 Magdeburg, Germany
(makunze,roesner)@iws.cs.uni-magdeburg.de

Abstract

We present an approach for exploiting knowledge from documents in the web. It is based on the integration of XML technologies with robust tools for natural language processing. The overall goal is to offer a knowledge engineer as much support as possible for the task of extracting and formalizing knowledge from document collections.

1. Introduction

The WWW is a valuable source of knowledge. Its users are confronted with an increasing number of interesting up-to-date documents (about hot topics). To acquire knowledge from these documents is very time-consuming and costly. With our approach we want to support users through a semi-automatic preselection of potentially interesting positions in documents. For this task we use robust linguistic tools for the analysis of documents. One of our applications are documents about the casting domain (in German). A group of these documents are excerpts from textbooks and contain basic definitions, descriptions and examples about the domain. We will use them for the extraction of fundamental definitions. As another group of documents we analyze written guidelines for the casting production. These rules have a fixed structure. A rule begins with a recommendation, followed by the reason for it. In some cases the rule ends with a number of instructions for those situations when the recommendation cannot be applied directly. Our system shall support a knowledge engineer (KE) who is creating a knowledge based system (KBS) by formalizing the knowledge extracted from the documents. The source documents remain linked with the structures of a KBS and can therefore be used for giving explanations by highlighting the particular relevant positions in the document related to the rules, which the KBS uses for the inference process.

2. Modules of XDOC

2.1 Syntactic and Semantic Analyses

In figure 1 we depict a part of our workbench XDOC[1]. In this workbench we combine a number of separate modules. We start with a morphological and syntactic analysis [3]. For this task we use a morphological lexicon and a grammar for the german language.

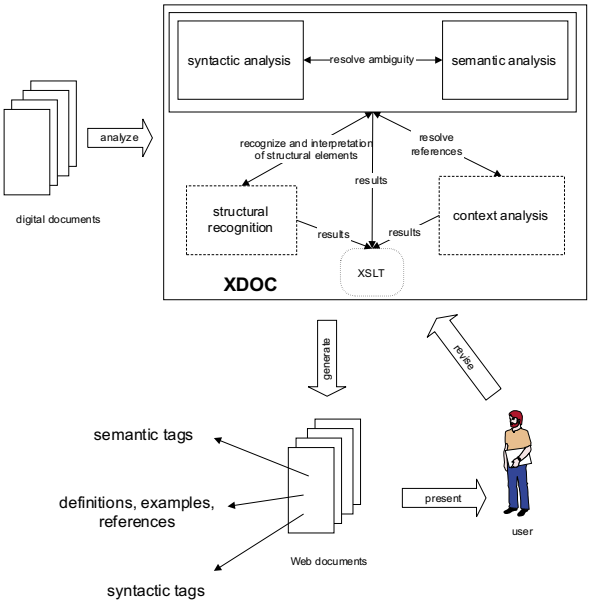

Figure 1. Schema of the Project

The results are used for the semantical analysis as well as a lexicon with the semantic interpretation for the words[2]

[1] XDOC stands for *X*ML based tools for *doc*ument processing

[2] In future we will use also rules to interpret specific and prominent phrases and structures (e.g. list, references for other documents).

and a basic ontology of the domain to complete missing data. The syntactic information is used as input for the semantic interpretation. The type of phrases and the features like case and number are of interest for the following processes. These features are used in the semantic lexicon for the assignment of the correct relations the concepts. The user can interactively verify and revise the automatically derived readings of phrases and sentences. In many cases it is possible that the system finds more than one interpretation of phrases and sentences. In this case the user must decide which reading he judges as correct. A separate module deals with the recognition of structures in the documents. This includes indentification of references to other literature (e.g. *DIN 8580*), or detection of specific identifiers (e.g. name of products: *Gusstueck EN 1982 - CC333G - GS - XXXX*)[2] and so on.

In all modules we need interaction with the user. For this purpose we exploit that all results of the modules are transformed into a uniform representation on the basis of XML. This allows to base the presentation of interesting structures on available tools for the flexible display of XML structures (XSL, xt). At the time of writing the following types of information are displayed:

- syntactic information for the completion of our tools and for semantic analysis (see example 1),
- recognized concepts and relations for the knowledge base (KB) (see example 2),
- positions of relevant definitions or examples as well as references to other documents with necessary information (see example 3).

For the presentation of the syntactic and semantic results we currently use self-explaining XML-tags. Example 1 and 2 show excerpts of both analyses of an example sentence [3] with a definition from our corpus.

Example: 1 *Excerpt from syntactic analysis:*

```
<PP CAS="AKK">
  <PRP CAS="AKK">durch</PRP>
  <NP TYPE="COMPLEX" RULE="NPC1" GEN="NTR" NUM="SG"
                                 CAS="AKK">
    <NP TYPE="FULL" RULE="NP1" CAS="AKK" NUM="SG"
                               GEN="NTR">
      <N>Schaffen</N>
    </NP>
    <NP TYPE="FULL" RULE="NP2" CAS="GEN" NUM="SG"
                               GEN="MAS">
      <DETD>des</DETD>
      <N>Zusammenhalts</N>
    </NP>
  </NP>
</PP>
```

[3] Example: Nach DIN 8580 ist Urformen Fertigen fester Koerper aus formlosem Stoff durch Schaffen des Zusammenhalts. In English: According to DIN 8580 primary shaping is the production of solid objects from formless matter by creating cohesion.

The tags of the result of the syntactic analysis contain information about the type of the structure (e.g. PP, NP, ...), marked as tag , as well as details on features (e.g. cases) and the matching rule of the grammar (both presented through an attribute of the tag). Figure 2 shows the graphical presentation of the results of example 1.

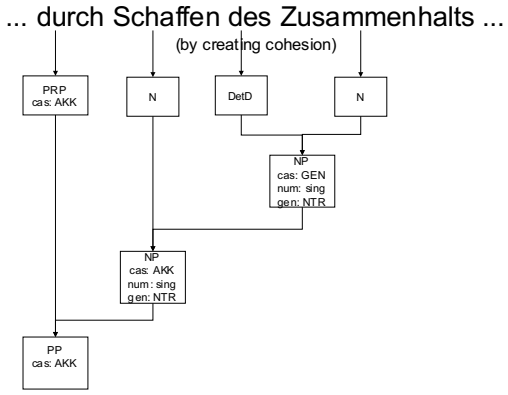

Figure 2. Schema of Example 1

The user can decide, which information he wants to be displayed through XSL Transformations [1] (see figure 4).

The follow example shows an excerpt of the semantic analyses:

Example: 2 *Excerpt from semantic interpretation:*

```
<CONCEPT TYPE=Prozess>
   <WORD>Fertigen</WORD>
   <DESC>Schaffung von etwas</DESC>
   <SLOT>
       <RESULT FORM="N(gen, fak) P(akk, fak, von)">
               fester Koerper</result>
       <SOURCE FORM="P(dat, fak, aus)">aus formlosem
               Stoff </source>
       <INSTRUMENT FORM="P(akk, fak, durch)">durch
               Schaffen des Zusammenhalts</instrument>
   </SLOT>
</CONCEPT>
```

The tags of the semantic analysis contain details about the type of recognized concepts and possible relations to other concepts. So far we also use attributes to show the description of the concepts and we annotate the relevant relations between the concepts through nested tags.

Figure 2 shows a possible presentation of the results of the semantic analysis, which is shown in example 2, through the use of

XSL transformations (see figure 4).

2.2 Structural Analysis

In many documents there are references to other literature (e.g. see example sentence). If this referenced literature is contained in our corpus, we want to set a link to this document and ideally to the referenced position in the document.

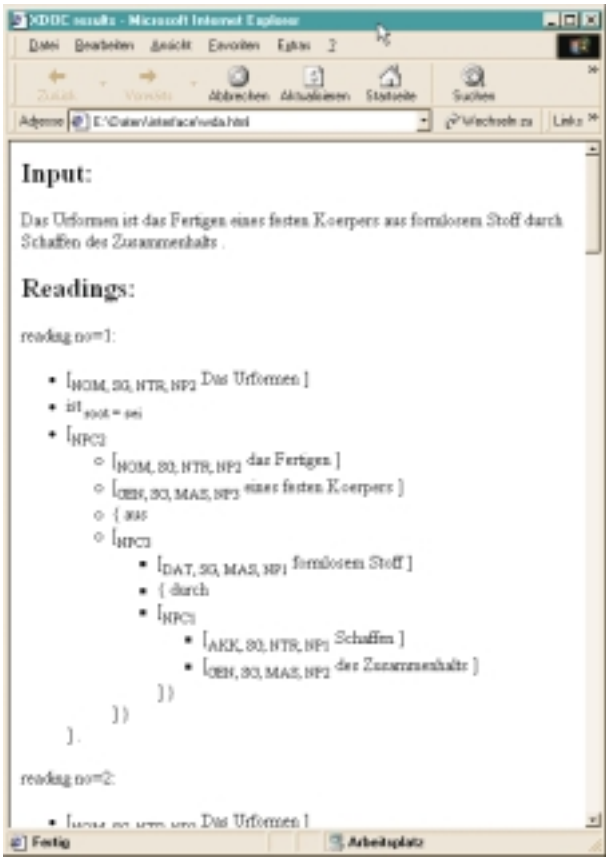

Figure 3. Presentation of the Syntatic Results

Figure 4. Presentation of the Semantic Results

Links and pointers are also used to relate knowledge representation structures with corresponding documents: linkage of detected concepts with the position of their definitions in the document as well as with describing examples, linkage of the enumeration of elements of a set with the definition in form of a list. Inside the structural recognizer we also use a lexicon, combined with information for the interpretation of these structures. In example 3 an excerpt of the lexicon [4] for structural analysis is shown.

Example: 3

Rule	*Description*	*Type*	*Function*
abbNR	*standard*	*Lit-Ref*	*Link*
MObject	*meta-object*	*MO-Ref*	*Link*
SAls1	*enumeration of elements*	*classification*	*position of definition*
SAls2	*definition of concepts*	*definition*	*position of definition*

The use of the rule *abbNR* on our example sentence leads to the follow results:

Example: 4

```
 ----------  search for specific structure ------------
"<REFERENCE TYPE=\"LITERATURE\">DIN 8580</REFERENCE> "
```

Inside the document the phrase *DIN 8580* (DIN is the German Institute for Standardization) is described as reference literature. If this literature is contained in our corpus, it will be set a link to this document. The user has the possibility to move between documents via link.
Other relevant structures in documents can be e.g. sentences like this: *Als formlose Stoffe werden Gase, Fluessigkeiten und Pulver bezeichnet.*[5]. In this example (match to rule SAls1) or rather in cases like last example, the phrase can be interpreted as an enumeration of instances of a concept. Through this structure of a sentence are realized an automatic classification of concepts (see figure 3). Other phrases in the documents like *Figure 3* or *Table 3.2* are handled as metaobjects, which we cannot further analyze with XDOC.

3. Discussion

With the approach described we want to support the user to extract information from a large collection of documents

[4] **Rule** stands for the name of the grammar rule, **Description** is a comment for the developer, **Type** describes the kind of reference and **Function** explains the uses of the reference

[5] In English: Formless substances are named gases, liquidis and powders.

about a domain. For this purpose it is necessary, that the various results of our modules are made available for the user. For the effective presentation of our results we use XSL in combination with the tool xt of J. Clark.

Using an interface with a selection of specific XSL transformations, every user can decide which information and in which form will be presented.

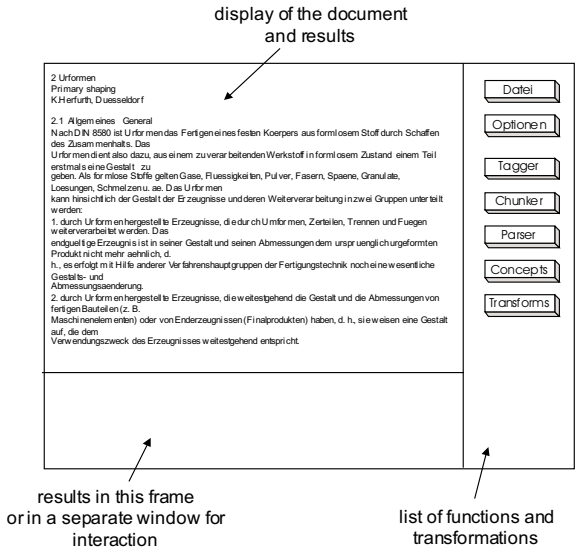

Figure 5. XDOC-Interface

In the future we will use a WWWbrowser as basis for the interface, like shown in figure 5. The advantage is that the users are already acquainted with the functionality of a browser and we are relatively independent of the operating system. To sum up: Using XML and XSL as a basic technology is in many respects advantageous for our purposes. We can offer results of the system in variations for different types of user (e.g. grammar and lexicon developer vs. knowledge engineer). Furthermore it is also easy to merge several documents through links between the documents and we can mark relevant position inside a document (e.g. definitions). Next steps in our work will include to extent the functionality of the structure detection module and - more important - to enhance the mapping of analysis results into formal knowledge representation structures.

References

[1] http://www.w3.org/Style/XSL.

[2] M. Kunze and D. Roesner. Eine XML-basierte Werkbank fuer das Document Mining. *Proceedings der GLDV-Fruehjahrstagung 2001*, pages 131–140, March 2001.

[3] D. Roesner. Combining robust parsing and lexical acquisition in the xdoc system. *KONVENS 2000: 5. Konferenz zur Verarbeitung natuerlicher Sprache*, pages 75–80, 2000.