Classifying XML Tags through "Reading Contexts"

Xavier Tannier, Jean-Jacques Girardot and Mihaela Mathieu
École Nationale Supérieure des Mines
158 Cours Fauriel
42023 Saint-Etienne, France
{tannier,girardot,mathieu}@emse.fr

ABSTRACT

Some tags used in XML documents create arbitrary breaks in the natural flow of the text. This may constitute an impediment to the application of some methods of document engineering. This article introduces the concept of "reading contexts", and gives clues to handle it theorically and in practice. This work should notably allow to recognize emphasis tags in a text, to define a new concept of term proximity in structured documents, to improve indexing techniques, and also to open up the way to advanced linguistic analyses of XML corpora.

1. INTRODUCTION

XML (eXtensible Markup Language [5]) is more and more widely used to store and exchange information. Here we focus on a *document-centric* view on XML documents. In this view, mark-up serves for giving information about logical structure and/or about form of a traditional document. This is the case of all texts intended for human people, such as manuals, books, articles or static web pages. This view is opposed to *data-centric* view, used for more database-oriented applications (flight schedules, catalogues, etc.).
XML content analysis is confronted with a particular problem: on the one hand, document structure is described by human experts in a meaningful and flexible way. On the other hand, for any XML processor, tags are all equally and totally meaningless. This does not only raise "top-level" problems (as semantic relations between tags or information extraction), but also, as we will see, more "basic" issues such as original text preserving or indexing.
This paper proposes to use a simple and intuitive division between three classes of tags [2] and to introduce the concept of "reading contexts" in order to solve this kind of issues automatically, considering the tag usage (and not their names). Some experimentations are described, and different uses that can be found and prospects opened by our work are discussed.

2. THREE TYPES OF XML MARKUP

An Information Retrieval oriented division of tags has been proposed by [2], in order to identify different categories that it would be important to distinguish within the framework of XML document retrieval. The original idea was to allow different treatments while searching for a pattern (sequence of characters). We will see (section 4) that, in our opinion, the advantages go far beyond this.

2.1 Hard, soft and jump tags

- *"Hard" tags* are the most frequent, they interrupt the "linearity" of a text, they generally contribute to the structuring of the document. Examples of this type are titles, chapters, paragraphs. Tags `'news'` and `'item'` are both "hard" in the following example:

(1) <news>
<item>A new study about evolution of tourism in the United States</item>
<item>Elections in Ukraine: the Central Commission published the results</item>
</news>

- *"Soft" tags* identify significant parts of a text, like quotations, appearance effects, but become "transparent" while reading the text. This is the case of tags `'bold'`, `'italics'`, `'underlined'` and `'sc'` (small capitals) in examples 2.a, 2.b, and 2.c.

(2) a. <par>
<bold>United States elections</bold> are administered at the state and local levels.
</par>
b. <title>
Noam Chomski's comments about <italics>United States</italics> <underlined>elections</underlined>.
</title>
c. <title>
U<sc>nited</sc> S<sc>tates</sc> E<sc>lections</sc>.
</title>

- *"Jump" tags* are used to represent particular elements, like margin notes, references to bibliography, or glosses. They are detached from the surrounding text. Elements `'comment'` and `'footnote'` in the following examples are "jump" elements.

(3) a. <oral_transcription>
I heard the news today about United States elec<comment>a door snaps</comment>tions.
</oral_transcription>

b. <paragraph>
The 2004 United States<footnote>See an article about the United States of America on page 142</footnote> elections caused less controversy than in 2000.
</paragraph>

c. <abstract>
This document deals with 1995 and 2002 Jacques Chirac <footnote>J. Chirac, the french president, is, by the way, not a really good friend of the president of the United States</footnote> elections.
</abstract>

2.2 Reading Context

We think that this classification introduces a notion that we could call "reading context".
A *reading context* is a small part of text, syntactically and semantically self-sufficient, that a person can read in a go, without any interruption. Reading contexts do not necessarily respect the linearity of the textual document.
For example, a paragraph or a list item (hard elements) change reading contexts. A footnote or a comment (jump elements) are inserted into an existing reading context and compose a new one. Finally, a bold or underlined text (soft element) lies within the current reading context and does not interrupt it.

3. DETERMINING TAG CATEGORY

We propose a method to determine the category of a tag name automatically. This is done through a procedure based on linguistic definitions of the classes.

3.1 Syntactic analysis

A syntactic linguistic analysis is performed in two steps. First, a *part-of-speech (POS) tagging*: the recognition of the grammatical category of each word, done with the free tool TreeTagger [3]. Then, the application of grammatical rules. The analysis is performed with a set of context-free rules describing some grammatical constructions. As an example, figure 1 shows a *syntactic tree* obtained after a POS tagging and the application of syntactic rules.

3.2 Algorithms

Let tn be the tag name that we want to determine the class of; e an element of name tn.

Soft tags. To recognize whether e is soft or not, we select the text before, within and after it (mark-up only is removed). We perform a syntactic analysis of this text.

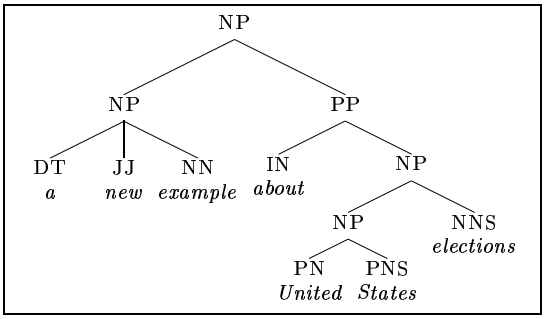
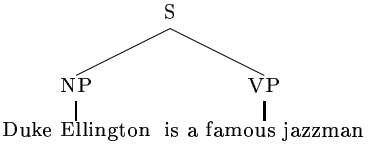


Figure 1: Example of syntactic tree.

If the text still "means" something, *i.e.* if a same syntactic tree groups together the content of e and some surrounding words together, then e is soft:

(4) <it>Duke Ellington</it> is a famous jazzman.



Jump tags. To detect jump tags, we select the text before and after the element e (the entire element is removed). We perform a syntactic analysis of this text. If the text still "means" something without e, *i.e.* if a same syntactic tree groups together some words on both sides of the location of e, then e is a jump element.

Hard tags. The safest and simplest definition of hard tags is: *a* hard tag *is neither a soft tag nor a jump tag.*

Obviously we cannot consider separately each element of the corpus. In particular, some elements have no text before or after them; in this case we cannot conclude. Moreover automatic parsing of a text is far from being a solved problem for common language. Erroneous analyses, careful though the grammar was designed, stay very numerous. Furthermore, in scientific publications, many tags contain abbreviated expressions or mathematical notations that our system fails to analyse. But we want a tag class to be attributed to a **tag name**, and not to each occurrence of this tag. Hence, we do not need a 100 % precision, but only statistically significant results, *i.e.* allowing to associate a tag name with its category with no doubt. Our set of grammatical rules does only contain very simple rules. Then we can be pretty sure that each syntactic tree is correct; on the other hand, many structures can be missed. For all these reasons, we consider that most of our results, higher than 40 or 50 % of precision, are significant. Indeed they should be compared to the rate of incorrect phrases that are likely to be analyzed as correct by the system. This rate is expected to be very low (< 5 %).

3.3 General Results

We performed our experimentations on the INEX [1] collection, consisting of 12107 scientific articles from various

IEEE journals. The corpus structure depicts both logical structuring, like *sections*, *paragraphs*, *titles*, and presentation tags. The DTD describes 192 different content models. We call a *successful analysis* an analysis that meets the conditions of our definitions for soft and jump tags: syntactic connection between element content and surrounding text (soft tags), or syntactic connection between words around the element (jump tags). A *percentage* of successful analyses is, for one tag name *tn* and one tag class, the rate of occurrences of *tn* that lead to a successful analyses.
We chose a threshold of 20 %, which means that if more than 20 % of *tn* occurrences are successfully analysed, then *tn* is considered to be a soft (resp. jump) tag name. This leads to the following classification: All emphasis tags (fonts, bold, etc.) have been recognized as soft tags, as well as links to urls. Bibliographic citations are considered as jump tags. As an “official” classification of these tags does not exist we can hardly evaluate these results in terms of recall and precision. However we consider that all soft tags have been correctly found. Tags `'a'`, `'url'` and `'ref'` can be discussed.

- Soft tags: `'tt'` (typewriter font), `'ss'` (sans serif); `'b'` (bold), `'ub'` (medium bold); `'it'` (italics), `'rm'` (roman), `'scp'` (small caps), `'u'` (underlined), `'large'`; `'ariel'`, `'bi'`, `'bu'`, `'bui'` (emphasis); `'a'` and `'url'` (links to url)
- Jump tag: `'ref'` (hyperlink)
- Hard tags: all the others...

More technical details about analysis, threshold and misclassification, as well as a study of the importance of corpus size, can be found in [4].

4. COMMENTS AND PERSPECTIVES

Soft/jump/hard classification of XML tag names, either automatically obtained or not, opens up the way to many uses; The following are some trails that we intend to follow in our further works. Aside from the first one, they are all based on our new concept of “reading context”:

- In almost all cases, soft tags are ***emphasis tags*** (bold, italics, underlined). These tags generally express the importance of some words in the text. This information is quite important, in Information Retrieval especially, where many systems use emphasis tags to give more weight to the terms that they contain.
- An indexing issue can be solved. In the following example 5.a, indexed words should be *’Tom’* and *’Sawyer’* in spite of `'sc'` tags (small capitals) cutting them. In this case tags are transparent. But if we apply the same method to examples 5.b and 5.c, then *’MarkTwain’* and *’1876The’* are recognized as unique terms. Here `'fn'` and `'ln'` tags must be replaced by blank characters. Finally *’Clemens’* is a single term in 5.d.

(5) a. <title>
        T<sc>om</sc> S<sc>awyer</sc>
    </title>
b. <author>
        <fn>Mark</fn><ln>Twain</ln>
    </author>
c. Book written in 1876<note>The author was born in 1835.</note>.
d. <transcription>
        His real name was Samuel Langhorne Clem<correction>the first transcriptor wrote a double ’m’ here</correction>ens.
    </transcription>

- This categorization allows to distinguish *physical proximity*, in the XML file, from what we could call ***logical proximity***. Logical proximity depends on the arrangement of the terms in the structure. it is closely related to the concept of “reading context”. Two words separated by a start/end soft tag are *logically* adjacent because they are consecutive in the same reading context. It is not the case when mark-up represents jump or hard tags, because the words can belong to different reading contexts.
  This is particularly interesting in Information Retrieval. Suppose that one wants some information about U.S. elections. The set of examples proposed in section 2 proves that the physical proximity of terms “United States” and “Elections” is not a guarantee of relevance. Relevance should rather be related to logical proximity. Thus examples 2.a, 2.b, 3.a and 3.b are relevant, while 1 and 3.c are not.
- Document-centric semi-structured documents are, as well as flat (non structured) documents, a good playing field for natural language processing researchers. But in the case of XML, an additional issue is the necessity and the difficulty to preserve reading contexts (what a human actually read). Yet this is the condition for performing a correct part-of-speech tagging, which is often the first step of a NLP work, but also for any kind of advanced syntactic/semantic linguistic analyses.

5. REFERENCES

[1] N. Fuhr, M. Lalmas, S. Malik, and Z. Szlàvik, editors. *Advances in XML Information Retrieval. Third Workshop of the Initiative for the Evaluation of XML retrieval (INEX)*, volume 3493 of *Lecture Notes in Computer Science*, Schloss Dagstuhl, Germany, Dec. 2005. Springer-Verlag.

[2] L. Lini, D. Lombardini, M. Paoli, D. Colazzo, and C. Sartiani. XTReSy: A Text Retrieval System for XML documents. In D. Buzzetti, H. Short, and G. Pancalddella, editors, *Augmenting Comprehension: Digital Tools for the History of Ideas*. Office for Humanities Communication Publications, King’s College, London, 2001.

[3] H. Schmid. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *International Conference on New Methods in Language Processing*, Sept. 1994.

[4] X. Tannier. Dealing with XML structure through "Reading Contexts". Technical Report 2005-400-007, Ecole Nationale Supérieure des Mines de Saint-Etienne, Apr. 2005. http://www.emse.fr/~tannier/publications.html.

[5] Extensible Markup Language (XML). World Wide Web Consortium (W3C) Recommandation, 2004. http://www.w3.org/TR/2004/REC-xml-20040204/.