Evaluating the Informative Quality of Web Documents Using Fuzzy Linguistic Techniques

E. Herrera-Viedma
School of Library
Science Studies, Univ. of
Granada,Spain
viedma@decsai.ugr.es

E. Peis,
School of Library
Science Studies, Univ. of
Granada,Spain
epeis@ugr.es

J.C. Herrera
School of Library
Science Studies, Univ. of
Granada,Spain
jcht@correo.de

K. Anaya
School of Library
Science Studies, Univ. of
Granada,Spain
akarima@ugr.es

Abstract

Recommender systems evaluate and filter the great amount of information available on the Web to assist people in their search processes. A fuzzy linguistic evaluation method of Web documents is presented to generate recommendations. Given an XML document type (e.g. scientific article), we consider that its components are not equally informative. This is indicated by defining linguistic importance attributes to the more meaningful elements of the XML Schema designed for Web documents. The evaluation method generates linguistic recommendations according to linguistic evaluation judgements provided by different recommenders on meaningful elements.

Keywords: Quality evaluation, XML, recommender system, fuzzy linguistic modelling.

1 Introduction

Since its conception in the early 90's, the World Wide Web (WWW) has become a critical component in the strategic thinking of content providers around the world [7]. The WWW contains a vast amount of data. A large debate on the quality information available on the WWW exists, and how to recognize useful and quality information becomes a critical problem. Therefore, users are in need of tools to help them cope with the mass of content available on the WWW [10,13].

The development of standard formats for the representation of Web documents improves substantially the quality of information retrieved by search engines. The logic structure of the Web documents can be expressed with metalanguages like XML [2]. The *eXtensible Markup Language* (XML) is a simplified subset of the *Standard Generalized Markup Language* (SGML) intended to make it more usable for distributing materials on the Web. SGML introduces the notion of document type and, consequently, a *document type definition* (DTD). XML has emerged as an important specification for the interchange of structured documents and data. It is believed that it will become a universal format not only for business-to business applications but also for knowledge and information managements [12]. A recent advance is the use of *XML Schema* to define the Web document's structure. The main advantages of this new technology with respect to the definition of Web documents based on DTDs are two: the possibilities of information interchange on the Web are increased and the data definition language is more expressive (different data types, user data types, …). Furthermore, XML allows to define the documents' output in other format using XSL stylesheets (e.g. XML, HTML, XHTML, …) [9].

Another promising direction to improve the effectiveness of search engines concerns the topic *collaborative filtering systems* or *recommender systems* [11]. In these systems the people collaborate to help one another and to perform filtering by recording their reactions to documents they read. In a typical recommender system people provide evaluation judgements or annotations on documents as inputs, which the system then aggregates obtaining recommendations that directs to appropriate recipients. Later, these recommendations can be reused to assist another people in their search processes. In this sense, recommendations are a kind of plausible measure of the informative quality of Web documents. Usually, the evaluation judgements are expressed by means of numerical values. Sometimes, however a person could have a vague knowledge about judgement valuations, and cannot express his/her judgements with an exact numerical value. Then, a more realistic approach may be to use linguistic assessments to express the evaluation judgements instead of numerical values, i.e., to suppose that the variables which participate in the evaluation process are assessed by linguistic terms [3,8].

In [4] we present a quality evaluation methodology for SGML documents based on computing with words. The advance in storage and communications enable users to store massive amounts of data, and to share it seamlessly with their peers. With the advent of XML, we expect a significant portion of this data to be in XML format. Therefore, users will require appropriate mechanisms for locating quality XML data.

The main aim of the paper is to present a fuzzy linguistic computing method for evaluating the informative quality of Web documents in XML format to generate recommendations. We assume XML documents represented by means of XML Schema. Given a kind of web document (e.g. “scientific article”), we establish a evaluation model composed by a subset of set of elements that define XML Schema (e.g. “title, authors, abstract, introduction, body, conclusions, bibliography”). We assume that each component of that subset has a distinct informative role, i.e., each one affects the overall evaluation of a document in a different way. This peculiarity is added in the Schema by defining an attribute for each meaningful component that contains a linguistic relative importance degree. Then, given an area of interest (e.g. “fuzzy information retrieval”), the recommendation for a Web document is obtained by combining the linguistic evaluation judgements provided by different recommenders on the meaningful components of the document structure. The recommendations obtained are linguistic values that express qualitatively the informative quality of Web documents with respect to an interest topic.

The paper is set out as follows. The fuzzy linguistic approach is discussed in Section 2. The Web documents in XML format are studied in Section 3. The evaluation method is defined in Section 4. Finally, Section 5 includes our conclusions.

2 Fuzzy linguistic approach

The *fuzzy linguistic approach* is a *soft computing tool* to manage linguistic information, which is based on the concept of *linguistic variable* [8]. It allows us to model in the problems qualitative values typical of human communication for representing qualitative concepts such as “importance” or “significance”. In our linguistic molelling we assume as in [4]: i) *a finite and totally ordered label set*, $S = \{s_i, i \in H = 0, \dots, T\}, s_i \geq s_j, if\ i \leq j,$

with odd cardinality (7 or 9 labels) to express the assessments, and ii) two aggregation operators of ordinal fuzzy linguistic information, the LOWA and LWA operators.

Definition 1 [6]. *Let A = {a_1, . . . , a_m } be a set of labels to be aggregated, then the LOWA operator, F, is defined as F (a_1, . . . , a_m) = W · B^T = C^m{ w_k, b_k, k = 1, . . . , m } = w_1 Qb_1 Å (1 - w_1) QC^{m-1}{b_h, b_h, h = 2, . . . , m }, where W = [w_1, . . . , w_m], is a weighting vector, such that, w_i Î [0, 1] and å$_i$$w_i$ =1. b_h = w_h/(å$_2^m$ w_k), h =2, . . . , m, and B = { b_1, . . . , b_m } is a vector associated to A, such that, B = s(A) = { $a_{s(1)}$, . . . , $a_{s(m)}$}, where, $a_{s(j)}$ £ $a_{s(i)}$ " i £ j, with s being a permutation over the set of labels A. C^m is the convex combination operator of m labels and if m=2, then it is defined as C^2{ w_i, b_i, i = 1, 2 } = w_1 Q s_j Å (1 - w_1) Q s_i = sk, such that k = min { T, i + round (w_1 · (j - i)) } s_j , s_i Î S, (j ³ i), being ”round” the usual round operation, and b_1 = s_j , b_2 = s_i. If w_j = 1 and w_i = 0 with i ¹ j " i, then C^m{ w_i, b_i, i = 1, . . . , m } = b_j .*

Definition 2 [5]**.** *The aggregation of a set of weighted linguistic opinions, {(c_1, a_1), . . . ,(c_m, a_m) } c_i , a_i Î S, according to the LWA operator P is defined as P [(c_1, a_1), . . . ,(c_m, a_m)]=F(h(c_1, a_1), . . . ,h(c_m, a_m)), where a_i represents the weighted opinion, , c_i the importance degree of a_i , and h is the transformation function defined depending on the weighting vector W assumed for the LOWA operator F, such that, h= MIN(c_i , a_i) if orness(W) ³ 0.5, and h= MAX(Neg(c_i), a_i) if orness(W) < 0.5.*

3 Web Documents in XML Format

Standard Generalized Markup Language (SGML) is a metalanguage, that is, a means of formally describing a language. Specifically, SGML provides the rules for defining a markup language based on tags [1]. Each instante of SGML correspond to a description of the document structure called a *document type definition* (DTD). SGML is a protocol devised to articulate structures of contents instead of the appearance of documents. Hence, an SGML document is defined by: 1) a description of the structure of the document and 2) the text itself marked with tags which describe the structure.

Example 1. The following DTD involved by SGML represents the structure of a document that is a scientific article:

```
<!DOCTYPE article [

<!ELEMENT article (title, authors, abstract?, introduction,body,conclusions,bibliography)>

<!ELEMENT title (#PCDATA)>

<!ELEMENT authors (author+)>

<!ELEMENT (author | abstract | introduction) (#PCDATA)>

<!ELEMENT body (section+)>

<!ELEMENT section (titleS, #PCDATA)>

<!ELEMENT titleS (#PCDATA)>

<!ELEMENT conclusions (#PCDATA)>

<!ELEMENT bibliography (bibitem+)>

<!ELEMENT bibitem (#PCDATA)> ]
```

SGML is a complex technology that requires significant investment. Due to this, in the past few years, work on structured documents has centered on simplifying SGML. Two of these efforts are HTML (HyperText Markup Language) and XML (eXtensible Markup Language). HTML is a simple language well suited for hypertext, multimedia, and display of small and simple documents. However, it presents many limitations, e.g., it does not allow users to specify their own elements or attributes in order to semantically qualify their data. XML, in contrast, is a simplified subset of SGML intended to make it more usable for distributing materials on the Web [4]. XML is not a markup language, as HTML is, but a metalanguage that is capable of containing markup languages in the same way as SGML. XML has not many of the restrictions imposed by HTML, but, however imposes a more rigid syntax [2]. With it you can literally create your own markup language [9]. The designers of XML simply took the best parts of SGML, guided by the experience with HTML, and produced something that is not less powerful than SGML, but vastly more regular and simpler to use. XML is a profile of SGML that eliminates many of the difficulties of implementing things (existence of a DTD, for example), so for the most part it behaves just like SGML. XML includes SGML ability to define new elements. XML is easier to learn and implement. XML removes the requirement of having to validate documents against a DTD, assuming that the tags can be obtained while the parsing of data is done. The main difference between SGML and XML is that many XML based documents don't need a DTD. DTDs define the structure and order of your element types, as well as the rules for using them. The downside is that DTD syntax differs from and can be more complicated. Today, although are not definitive specification, to explain syntax and content of a valid XML based document –i.e. well formed document with correspondence, instead, to a DTD- researchers are employed XML Schema.

The purpose of an XML Schema is to define the legal building blocks of an XML document, just like a DTD. One of the many things XML Schema proposes is the use of XML to describe the structure and rules for using elements. Thus, schemas define: i) elements that can appear in a document, ii) attributes that can appear in a document, iii) which elements are child elements iv) the order of child elements v) the number of child elements vi) whether an element is empty or can include text vii) data types for elements and attributes viii) default and fixed values for elements and attributes. They also define the element types may be child elements of a particular parent element, and the type of content of an element. More importantly, you can create your schemas directly in XML [9].

The use of XML Schemas presents many advantages. XML Schemas are extensible to future additions, are richer and more useful than DTDs, are written in XML, support data types, and support namespaces. One of the greatest strength of XML Schemas is the support for data types. With the support for data types: It is easier to describe permissible document content, validate the correctness of data, work with data from a database, define data facets (restrictions on data) and data patterns (data formats), convert data between different data types. XML Schemas are extensible, just like XML. With an extensible Schema definition you can reuse your Schema in other Schemas, create your own data types derived from standard types, and reference multiple schemas from the same document. Consequently, in the future XML Schemas will be used in most Web applications as a replacement for DTDs.

Example 2. An example of a document instance of DTD defined in Example 1 but using XML Schema is the following:

```
<?xml version="1.0" encoding="UTF-8"?>

<xsd:schema
xmlns:xsd="http://www.w3.org/2001/XMLSchema”
targetNamespace="http://www.ugr.es/~gilrs/schema
s" xmlns="http://www.ugr.es/~gilrs/schemas"

elementFormDefault ="qualified">

<xsd:element name="article"
maxOccurs="unbounded">

<xsd:complexType> <xsd:sequence>

<xsd:element name="title"type="xsd:string"/>

<xsd:element name="authors">

<xsd:complexType> <xsd:sequence>

<xsd:element name="author" type="xsd:string"
maxOccurs="unbounded"/> </xsd:sequence>

</xsd:complexType> </xsd:element>

<xsd:element name="abstract" type="xsd:string"
minOccurs="0"/>

<xsd:element name="introduction"
type="xsd:string"/>

<xsd:element name="body">

<xsd:complexType> <xsd:sequence>

<xsd:element name="section"
maxOccurs="unbounded">

<xsd:complexType><xsd:sequence>

<xsd:element name="titleS" type="xsd:string"/>

<xsd:element name="p" type="xsd:string"
maxOccurs="unbounded"/>

</xsd:sequence> </xsd:complexType>

</xsd:element>  </xsd:sequence>

</xsd:complexType> </xsd:element>

<xsd:element name="conclusions"
type="xsd:string"/> <xsd:element
name="bibliography" minOccurs="0">

<xsd:complexType> <xsd:sequence>

<xsd:element name="bibitem" type="xsd:string"
maxOccurs="unbounded"/>

</xsd:sequence> </xsd:complexType>

</xsd:element> </xsd:sequence>

</xsd:complexType> </xsd:element>

</xsd:schema>
```

An Internet search engine –e.g., Altavista or Infoseek- returns thousands of so-called matched documents from a single query, some of which are relevant and others irrelevant to the query. End users typically have problems with organizing and digesting such vast quantities of information, in which much of the information retrieved is likely to be irrelevant. XML holds the promise that searching can be done more precisely because structural, self-describing information and metadata (e.g., RDF) is available, to allow for context-based and/or Category-based search. XML also holds the promise to model heterogeneous data, generate from databases (DBs) or from word processors, thereby enabling search engines to locate and process heterogeneous documents or records [12].

Moreover, XML is a platform independent language and allows to define in a simple manner the documents' output format (e.g. XML, HTML, XHTML (eXtensible HyperText Markup Language), etc.) just using XSL (eXtensible Stylesheet Language), a language for expressing stylesheets. It consists of two parts: a language for transforming XML documents, and an XML vocabulary for specifying formatting semantics. An XSL stylesheet specifies the presentation of a class of XML documents by describing how an instance of the class is transformed into an XML document that uses the formatting vocabulary. This structuring is used to store the documents in the web server, but not to show this documents on the client's browser, because the browser would not be able to process it. Due to this, when a client requests a document to the server, this will dynamically transform it into a browser readable format using XSL. This format could be XHTML, a XML syntax version of HTML, compatible with most of the existing web browsers.

The linguistic terms set that the “rank” attribute may take as possible values is independently defined in another XML Schema, referred from each different document type's schemas. Then, given a search topic -e.g. ”recommender systems”-, the relevance for an XML document is obtained by combining the linguistic evaluation judgements provided by the visitor on the meaningful components of its XML Schema.

4 Evaluating Web Documents for Generating Recommendations

Suppose that we want to generate a recommendation database for qualifying the information of a set of valid XML based documents, $\{d_1, . . ., d_l\}$, with the same XML Schema. These documents can be evaluated from a set of different areas of interest, $\{A_1,...,A_q\}$. Consider an evaluation scheme composed by a finite number of elements of the XML Schema, $\{p_1,...,p_n\}$, which will be evaluated in each document d_k by a panel of recommenders or visitors $\{e_1 ,..., e_m\}$. We assume that each component of that evaluation scheme presents a distinct informative role. This is modeled by assigning to each p_j a relative linguistic importance degree I(p_j) supported by the linguistic variable "Importance" defined as in [3], i.e., I(p_j) $\in S=\{s_1 ,..., s_T\}$. Each importance degree I(p_j) is a measure of the relative importance of element p_j with respect to others existing in the evaluation scheme. We propose to include these relative linguistic importance degrees in the XML Schema. This can be done easily by defining in the XML Schema an attribute of importance "rank" for each component of evaluation scheme using the XML syntax.

Example 3. Defining an attribute of importance “rank” for the “title” element of XML Schema given in Example 2:

```
<xsd:element name="title">
   <xsd:complexType>
     <xsd:simpleContent>
        <xsd:extension base="xsd:string">
   <xsd:attribute   name="rank"   type="lblRank"
use="optional" default="I(title)"/>
        </xsd:extension>
     </xsd:simpleContent>
   </xsd:complexType>
</xsd:element>
```

The linguistic term set that the "rank" attribute may take as possible values is independently defined in another XML Schema, referred from each different document type's schemas.

Example 4 : The XML Schema "Labels.xsd", associated with a linguistic term set of nine labels is as follows

```
 <?xml version="1.0" encoding="UTF-8"?>
<xsd:schema
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
>elementFormDefault="qualified">
 <xsd:simpleType name="lblRank">
 <xsd:restriction base="xsd:string">
 <xsd:enumeration value="Total"/>
 <xsd:enumeration value="ExtremelyHigh"/>
 <xsd:enumeration value="VeryHigh"/>
 <xsd:enumeration value="High"/>
 <xsd:enumeration value="Medium"/>
 <xsd:enumeration value="Low"/>
 <xsd:enumeration value="VeryLow"/>
 <xsd:enumeration value="ExtremelyLow"/>
 <xsd:enumeration value="None"/>
 </xsd:restriction>
 <"/xsd:simpleType>

</xsd:schema>
```

Let e^{ij}_{kt} be a linguistic evaluation judgement provided by the recommender e_k measuring the informative quality or significance of element p_j of document d_i with respect to the area of interest A_t. Consider that e^{ij}_{kt} is supported by the linguistic variable "Significance", which uses the same label set associated to "Importance", but with a different interpretation, i.e., $e^{ij}_{kt} \in S$. Then, the evaluation procedure of a XML based document d_i obtains a recommendation, $r^{i}_{t} \in S$ (it is also supported by the linguistic variable "Significance") using evaluation method based on the LWA and LOWA operators as follows [4]:

1. Capture the topic of interest A_t, the linguistic importance degrees of evaluation scheme fixed in the XML Schema $\{I(p_1),..., I(p_n)\}$, and all the evaluation judgements provided by the panel of recommenders $\{ e^{ij}_{kt}\}, j=1,...,n, k=1,...,m.$
2. Calculate for each e_k his/her individual recommendation r^{i}_{kt} by means of the LWA operator as

$$r^{i}_{kt} = P\,[(\,I(p_1), e^{i1}_{kt})\ ,...,(\,I(p_n), e^{in}_{kt})] = \,)] = F(h(I(p_1), e^{i1}_{kt}),..., h(I(p_n), e^{in}_{kt})).$$

Therefore, r^{i}_{kt} is a significance measure that represents the informative quality of d_i with respect to topic A_t according to the Q evaluation judgements provided by e_k , being Q the linguistic quantifier used to compute the weighting vector of LOWA operator F [6].

3. Calculate the global recommendation r^i_t by means of F guided by the fuzzy majority concept represented by the linguistic quantifier Q as

$$r^i_t = F(r^i_{1t}, \ldots, r^i_{mt}).$$

Then , r^i_t is a significance measure that represents the informative quality of d_i with respect to A_t according to the Q evaluation judgements provided by the Q recommenders. r^i_t represents the linguistic informative category of d_i with respect to A_t

4. Store the recommendation r^i_t in a recipient in order to assist users in their later search processes.

5 Conclusions

In this paper, we have presented a fuzzy linguistic evaluation method to characterize the information contained in XML based documents. The method generates linguistic recommendations for structured documents by taking into account the fuzzy majority of linguistic evaluation judgements provided by different recommenders to evaluate the informative quality of the more meaningful component of XML Schema. The use of fuzzy linguistic modelling facilitates the activity of the filtering systems due to that the user-system interaction is more user-friendly.

References

[1] C. Goldfarb (1990). *The SGML Handbook*. Oxford University Press, Oxford, 1990.

[2] C. Goldfarb and P. Prescod (1998). *The XML Handbook*. Prentice Hall, Oxford, 1998.

[3] E. Herrera-Viedma (2001). Modeling the retrieval process for an information retrieval system using an ordinal fuzzy linguistic approach. *Journal of the American Society for Information Science and Technology*, 52(6):460-475.

[4] E. Herrera-Viedma and E. Peis (2003). Evaluating the Informative Quality of Documents in SGML-Format Using Fuzzy Linguistic Techniques Based on Computing with Words. *Information Processing & Management*, 39(2):195-213.

[5] F. Herrera and E. Herrera-Viedma (1997). Aggregation operators for linguistic weighted information. *IEEE Transactions on Systems, Man, and Cybernetics, Part A,* 27:646-656.

[6] F. Herrera; E. Herrera-Viedma and J. L. Verdegay (1996). *Direct approach processes in group decision making using linguistic owa operators*. Fuzzy Sets and systems, 79:175-190, 1996.

[7] H. W. Lie and J. Saarela (1999). *Multipurpose Web Publishing using HTML, XML and CSS*. Communications of the ACM, October 1999, vol. 42, nº 10.

[8] L.A. Zadeh (1975). The concept of a linguistic variable and its applications to approximate reasoning. Part i. *Information Sciences,* 8:199-249. Part ii. *Information Sciences*, 8:301-357. Part iii. *Informations Sciences*, 9:43-80.

[9] M. Floyd (2000). *Builiding Websites with XML.* Prentice Hall PTR, New Jersey.

[10] M. Kobayashi and K. Taleda (2000). *Information retrieval on the web*. ACM Computing Surveys, 32(2):144-173, 2000.

[11] P. Reisnick and H.R. Varian (1997). Special Issue: Recommender systems. *Comm. Of the ACM.* 40(3).

[12] R. Baeza-Yates, D. Carmel, Y. Maarek, and A. Soffer, (2002). Special Topic Issue: XML. *Journal of the American Society for Information Science and Technology,* 53(6).

[13] S. Lawrence and C. L. Giles (1999). *Searching the web: General and scientific information access*. IEEE Communications, 37(1):116-122, 1999.