Using Proximity and Tag Weights for Focused Retrieval in Structured Documents

Michel Beigbeder, Mathias Géry, Christine Largeron

KAIS manuscript No.
(will be inserted by the editor)

Using Proximity and Tag Weights for Focused Retrieval in Structured Documents

Michel Beigbeder · Mathias Géry · Christine Largeron

Received: Mar 25, 2013 / Revised: Apr 29, 2014 / Accepted: May 18, 2014

Abstract Focused information retrieval is concerned with the retrieval of small units of information. In this context, the structure of the documents as well as the proximity among query terms have been found useful for improving retrieval effectiveness. In this article, we propose an approach combining the proximity of the terms and the tags which mark these terms. Our approach is based on a *Fetch and Browse* method where the fetch step is performed with BM25 and the browse step with a structure enhanced proximity model. In this way, the ranking of a document depends not only upon the existence of the query terms within the document but also upon the tags which mark these terms. Thus, the document tends to be highly relevant when query terms are close together and are emphasized by tags. The evaluation of this model on a large XML structured collection provided by the INEX 2010 XML IR evaluation campaign shows that the use of term proximity and structure improves the retrieval effectiveness of BM25 in the context of focused information retrieval.

Keywords Focused information retrieval · Structured information retrieval · Proximity · XML · Tags

1 Introduction

Focused information retrieval (IR) was originally introduced to provide more direct access to short passages (Trotman et al 2007). More precisely, focused information retrieval aims to give the user extracts of documents rather than whole documents, as is the case with traditional information retrieval. In fact, focused information retrieval covers structured document retrieval and XML retrieval which are both concerned with the development of models for querying and retrieving

Michel Beigbeder
École Nationale Supérieure des Mines de Saint-Étienne
E-mail: michel.beigbeder@emse.fr

Mathias Géry, Christine Largeron
Université de Lyon, Saint-Étienne, France
E-mail: {Mathias.Gery, Christine.Largeron}@univ-st-etienne.fr

relevant parts from structured documents whose structure is usually encoded with mark-up languages, such as HTML, SGML and now predominantly XML (Lalmas and Baeza-Yates 2009; Lalmas and Trotman 2009). In such languages, the logical structure, defined by the logical tags, is used to mark the boundary of parts of the document which have coherence and which could be returned to the user, if they are considered as relevant by the system. Such elements are called logical elements. This is the case, for example of *article* or *section*. However, the mark-up languages include other tags in addition to the logical tags, for instance formatting tags like *STRONG* or *I* in HTML. Figure 1 presents an XML article from the INEX Wikipedia collection (cf. Section 4.1), containing five logical tags (`article`, `bdy`, `header`, `p`, `sec`), one link tag (`link`) and two formatting tags (`b`, `it`).

```
<article xmlns:xlink="http://www.w3.org/1999/xlink">
  <header>
    <title>Handel House Museum</title>
    <id>1707709</id>
  </header>
  <bdy>
    <image width="150px" src="London_Handel_House.jpg" type="thumb">
      <caption>Handel House. Note the <link xlink:href="310649.xml">blue plaque</link>
      </caption>
    </image>
    <p>
      The <b>Handel House Museum</b> at 25 <link xlink:href="2599649.xml">Brook
Street</link>, in the exclusive central <link xlink:href="17867.xml">London</link>
district of <link xlink:href="94167.xml">Mayfair</link> was the home of the <link
xlink:href="11867.xml">German</link> born <link xlink:href="4500.xml">baroque</link>
composer <link xlink:href="12775.xml">George Frideric Handel</link>
from 1723 until his death at the house in 1759. He composed works such as
<it><link xlink:href="149131.xml">The Messiah</link></it>, <it><link
xlink:href="811987.xml">Zadok the Priest</link></it> and the <it><link
xlink:href="1246814.xml">Fireworks Music</link></it> there.
    </p>
    <sec>
      <st>The museum</st>
      <p>
        The house has been restored to look as it did during Handel's occupancy. A typical
early 18th century London terrace house, it comprises a basement, three main storeys and an
attic, and Handel was the first occupant. The attic was later converted into a fourth full
floor. The ground floor is now a music and gift shop and the upper floors are leased to a
charity called the Handel House Trust, and have been open to the public since 8 November
2001. The interiors have been restored to the somewhat spartan style of Georgian era,
using mostly architectural elements from elsewhere, as other than the staircase, few of
the original interior features survived. The Handel House Collection Trust has assembled a
collection of Handel memorabilia, including the Byrne Collection of several hundred items,
which was acquired in 1998.
      </p>
    </sec>
  </bdy>
</article>
```

Fig. 1 Article example "Handel House Museum", from INEX 2010

In the following, we assume that the non-logical tags do not break the document into logical elements. However, they can be useful for information retrieval. Indeed, focused information retrieval has not only been concerned with the retrieval of shorter units of information, delimited by logical tags, but also with the

exploitation of every tag in order to improve the detection of relevant information in answer to specific user needs. The hypothesis underlying this approach is that the tags are used to emphasize words and consequently they can also be used to find relevant information. For instance, a word is undoubtedly more important if it appears within certain sections of a document (a title, a caption, etc.). In the same way, it does not have the same emphasis if it appears in a particular font (bold, italics, etc.). From this perspective, a solution for taking into account the tags consists of improving the bag-of-words models ($tf \cdot idf$, BM25, etc.) in such a way that the ranking of a document depends not only upon the existence of the query terms within the document but also upon the tags which mark these terms (Lalmas 2009). This approach has already been explored in the context of IR (Boyan et al 1996; Sun et al 2000; Rapela 2001; Robertson et al 2004; Trotman 2005). In the context of focused IR it was also used to improve the vector space model (Wilkinson 1994) as well as the probabilistic models (Wolff et al 2000; Lu et al 2006).

Additionally, other models seem very promising in the context of focused information retrieval, notably those based on the proximity of the query terms in the documents. The hypothesis behind the use of proximity is that the closer the query terms appear in a document, the more likely the document is to be relevant. We can illustrate this intuition by an example corresponding to the information need 2010014 of the INEX 2010 campaign in which the title field is "`composer museum`" (see Figure 2). Figure 3 presents an example of a document that is not relevant and Figure 4 an example of a document that is relevant. In both documents, the two terms 'composer' and 'museum' appear. They are closer in the relevant document.

```
<topic id="2010014" ct_no="329">
<title>composer museum</title>
<description>Documents or parts of documents that describe or identify a museum dedicated
or which has a significant section dedicated to a composer</description>
<narrative>I want to know the museum that are dedicated to or which have a significant
section dedicated to a composer. Other music related museum which are not dedicated to a
composer are not relevant (for instance a museum of musical instruments)</narrative>
</topic>
```

Fig. 2 Topic example 2010014 from the INEX 2010 campaign

The proximity based approach is interesting in the context of focused information retrieval since it would favour small extracts of documents containing a great number of occurrences of the query terms. A previous study showed that, in itself, this method is not sufficient to fulfill the focused retrieval task because it returns a very low number of documents (Beigbeder 2007). Thus, it was suggested that focused retrieval should be performed in two steps: *Fetch and Browse* (Chiaramella et al 1996). The first step (fetch step) aims at identifying full articles, while the second step (browse step) focuses on passages within the retrieved articles (Malik

use of sonorism and dodecaphonism . The style emerged out of the political crisis in 1956, following Stalin's death; that same year saw the Warsaw Autumn music festival inaugurated, from whence came additional popularity for the Polish Composers' School. Composers included Tadeusz Baird, Boguslaw Schaeffer, Włodzimierz Kotoński, Witold Szalonek, Krzysztof Penderecki, Witold Lutosławski, Wojciech Kilar, Kazimierz Serocki and Henryk Mikołaj Górecki.
More modern composers include Krzysztof Meyer, Paweł Szymański, Krzesimir Dębski, Hanna Kulenty, Eugeniusz Knapik and Paweł Mykietyn.

Poland has always been a very open country to new music genres and even before the fall of the communism, music styles like rock, metal, jazz, electronic, and New Wave were well-known. Since 1989, the Polish scene has exploded with new talents and a more diverse style. Contrary to most European countries, pop music is not dominant in Poland.
Every year, a huge gathering of young Poles meet to celebrate the rock and alternative music in Jarocin or Żary. These events often attract more than 250,000 people and are comparable to the gatherings in Woodstock and Roskilde.

Poland has a very active underground extreme metal music scene. Some of the bands that have heralded and helped the cause are Vader, Behemoth, Yattering, Decapitated, Graveland, Baphomets Throne, and Dissenter. This has paved ground for a large underground movement. One of the biggest record labels of death metal in Poland is Empire Records.
In jazz music, polish musicians created a specific style, which was most famous in 60s and 70s. Most famous polish jazz artists are: Krzysztof Komeda, Adam Makowicz, Tomasz Stańko, Michał Urbaniak.
Two contemporary big Polish music festivals are Opole Festival and Sopot Festival .

Museums and festivals

Poland offers a wide spectrum of cultural experience. Those interested in high culture will enjoy the renowned music festivals like Wratislavia Cantans and the Warsaw Autumn . Polish museums exhibit remarkable art collections - masterpieces including Leonardo da Vinci 's Lady with an Ermine at the Czartoryski Museum in Kraków; the Veit Stoss High Altar in St. Mary's Basilica , Kraków; and the Last Judgement by Hans Memling (The National Museum in Gdańsk). Ethnographic museums and open-air site-seeing museums also hold attractive collections. The panorama of Polish culture is completed by a medley of local

Fig. 3 Document example: *Culture in Modern Poland*, non relevant to the topic of Figure 2

Handel House Museum

1707709 237680213 2008-09-11T10:03:06Z Pageturners 7302682
Houses in Westminster
Music museums in the United Kingdom
Biographical museums in London
Grade I listed buildings in London
Grade I listed houses
Buildings with blue plaques
Buildings and structures in Westminster

Handel House. Note the blue plaque

The Handel House Museum at 25 Brook Street , in the exclusive central London district of Mayfair was the home of the German born baroque composer George Frideric Handel from 1723 until his death at the house in 1759. He composed works such as The Messiah , Zadok the Priest and the Fireworks Music there.

The museum

The house has been restored to look as it did during Handel's occupancy. A typical early 18th century London terrace house, it comprises a basement, three main storeys and an attic, and Handel was the first occupant. The attic was later converted into a fourth full floor. The ground floor is now a music and gift shop and the upper floors are leased to a charity called the Handel House Trust, and have been open to the public since 8 November 2001. The interiors have been restored to the somewhat spartan style of Georgian era, using mostly architectural elements from elsewhere, as other than the staircase, few of the original interior features survived. The Handel House Collection Trust has assembled a collection of Handel memorabilia, including the

Fig. 4 Document example: *Handel House Museum*, relevant to the topic of Figure 2

et al 2006). Based on this paradigm, we introduce an enhanced version of a proximity based model that we propose to combine with BM25. Thus, BM25 is used for the fetch step and the proximity model for the browse one. Moreover, in this extended version of the proximity model, the proximity score is computed by taking into account the tags. In this way, we assume in our approach firstly that the closer the query terms appear in a document the more likely the document is to be relevant and secondly that the tags allow the relevant terms to be emphasized.

However, the use of some proximity based models remains limited because they require Boolean queries and very few users are able to formulate their needs in this way: usually the queries are expressed with a few keywords (O'Keefe and Trotman 2004). For this reason, we also propose an automatic method to convert a list of keywords into a Boolean query in order to overcome this limitation.

In summary, the main contributions of this article are:

- a focused information retrieval method based on a *Fetch and Browse* approach, where the fetch step is performed with BM25 and the browse step with a novel structure enhanced proximity model;
- the integration of structural hints in the proximity score based on a learning stage to estimate tag ability to distinguish relevant terms from others;
- a method to convert a list of keywords into a Boolean query and a preliminary study on the impact of the type of queries: manually built Boolean queries or automatic Boolean queries;
- an evaluation on a large XML structured collection: The INEX 2010 collection consisting of documents extracted from Wikipedia, information needs and relevance judgments. This collection is presented in Section 4.1.

A more formal presentation of the model appears in Section 3 after a presentation of related work in Section 2. The experiments are detailed in Section 4 and the results are reported in Section 5 before the conclusion.

2 Related Work

Luhn (1958) was the first author to point out the interest of proximity for information retrieval. In information retrieval systems, several approaches have been proposed to take into account the proximity of term occurrences in the document. Among the first ones, we can mention the introduction of some operators in the Boolean query language model, for instance:

- NEAR which takes the value true when the two terms of the query connected by this operator appear in the document within a window smaller than a fixed or variable length;
- SENTENCE which is true when the two connected terms appear within the same sentence;
- PARAGRAPH which is true when the two connected terms appear within the same paragraph.

The first attempts at applying these operators in information retrieval systems were carried out with some success but the experiments were performed on very small collections and the comparison against the *quorum level*[1] measure was not

[1] The *quorum level* is the number of unique query terms that appear in a document. This measure is also called *coordination level* by other authors.

sufficient (Keen 1991, 1992) because of its weak effectiveness. Nevertheless, they confirm the intuition that proximity is useful.

In the following subsections, we present other recent works related to the use of proximity in the retrieval process. The first subsection is dedicated to models completely based on proximity scoring and the second one to research which improve a traditional model (either the vector space model or the Okapi probabilistic model). The third subsection presents models that use an influence function at each occurrence of a query term. Finally, the few approaches which use proximity with the structure of documents are presented in the fourth subsection.

2.1 Proximity based models

After Keen's studies, Clarke et al (1995) and Hawking and Thistlewaite (1995) exploit similar ideas: to be relevant, a document should contain all the query concepts. Moreover, the closer and the more numerous these concepts appear in a document, the higher the score of the document. Their systems differ in the way they define the *spans*[2] that cover all the concepts and the way they compute the score attributed to this span, but the common idea is to select the shortest spans that contain all the concepts. As each concept could be represented by different terms, the queries used by their systems are Boolean queries in conjunctive normal form. In their experiments, these queries were manually built and some of them were quite long. Moreover, we can notice that these systems suffer from a low recall because a document can have a score different from zero only if it contains at least one instance of every concept.

In order to address this limitation, the authors have relaxed the queries in different ways. Clarke and Cormack (1996) used a list of sub-queries where the first one was supposed to have a "high-precision" and the subsequent ones increase the recall. Cormack et al (1997) exploited the title field of the TREC topics as an automatic query. In their system, the documents are first ranked by the quorum level and then by the score computed with their previous method applied to the conjunction of the query terms that appear in the document. Further experiments showed that these methods are more useful with short queries (Clarke et al 2000). This indicates that, with a purely conjunctive interpretation of the queries, too many terms tend to impose too many constraints.

We can note that it is not obvious how to extend these models to take into account the document structure. In our work, we retain from this previous research the idea of the conjunctive interpretation of queries with either manually or automatically built queries.

2.2 Integration of proximity in traditional models

Another way to take proximity into account consists of extending traditional information retrieval models and we distinguish three classes of approaches:

- combination of a proximity score to the usual score (Rasolofo and Savoy 2003; Bai et al 2008; Tao and Zhai 2007; Cummins and O'Riordan 2009);

[2] Many authors use the concept of *span* or *range* or *interval* or *segment* which correspond to contiguous extracts of text in the documents.

- addition of new dimensions into the term space: terms and n-grams (Mishne and de Rijke 2005; He et al 2011);
- modification of the *tf* values in order to reward terms that appear close to the other terms of the query (Büttcher et al 2006; Song et al 2008).

To complement the score used in traditional ranking models, many authors suggest measuring the proximity on every query term pair. Rasolofo and Savoy (2003) enhance the Okapi BM25 model by adding a proximity measure for every query term pair. Their conclusion is that using proximity features "may potentially improve precision after retrieving a few documents" and could be useful for very short answers such as those looked for in question-answering systems, which seems to confirm its usefulness for focused retrieval. Tao and Zhai (2007) add to a BM25 or to a KL-divergence score a proximity score computed on query term pairs using for instance *Minimum Pair Distance*, *Average Pair Distance* or *Maximum Pair Distance* and they report that the first is highly correlated with the document relevance. Cummins and O'Riordan (2009) discover different formulas with Genetic Programming, that combine proximity measures based on query term pairs while Bai et al (2008) use n-grams instead of term pairs and show that 3- to 5-grams improve the precision.

A different proximity based approach consists of adding new dimensions to the term space which is classically only composed of uniterms. Both Mishne and de Rijke (2005) and He et al (2011) consider that every n-gram composed of the query terms is used as a uniterm in the standard vector space model while Metzler and Croft (2005) use a similar approach within the language model framework. The experiments reported in these articles show mixed results depending on the collections and the size of the queries.

The third class of methods modify the term count $tf(t, d)$ of term t in document d to take into account the proximity of other query terms. Büttcher et al (2006) count for each occurrence of term t a value greater than 1 when this occurrence is close to other query terms. Song et al (2008) extend this idea with a pseudo-*tf* which replaces the usual *tf* in the BM25 formula. None of these methods enforce the presence of all query terms in the documents. However this was proven to be effective at the document level by Hearst (1996) and also to obtain high precision by Clarke et al (1995) and Hawking and Thistlewaite (1995). For this reason, in our work, we use a proximity based model which enforces the presence of all query terms.

2.3 Models with influence functions

Besides the works detailed in the Subsections 2.1 and 2.2, another approach consists of assigning to each occurrence of a (query) term an influence over the positions in the document. The idea of influence functions is that the influence on relevance of a term occurrence at a given position reaches its maximum at this precise position and decreases around this position down to zero as the distance to the position increases. This idea was first presented by de Kretser and Moffat (1999) and by Tajima et al (1999) but in the first work the influence functions of the query terms are modulated in height and width according to idf[3] while there

[3] Inverse Document Frequency: measure of whether the term is common or rare across all documents.

is no modulation in the second. Kise et al (2001) proposed a very similar approach to that of de Kretser and Moffat (1999) and they report better effectiveness of their method in comparison to VSM (a vector space model), confirming the results obtained by de Kretser and Moffat, and also to PRF (a pseudo relevance feedback model) and LSI (Latent Semantic Indexing) with long documents.

Beigbeder and Mercier (2005) introduce a model in which the influence functions are aggregated with fuzzy Boolean operators but this entails formulating queries with the Boolean query model. However, this model is the only one that allows enforcement of the presence of all query terms by using a purely conjunctive query. Indeed, in this article, we propose an enhanced version of this last model in which we introduce a modulation of height and width as performed by de Kretser and Moffat (1999) and Kise et al (2001). However, unlike the aforementioned studies, the modulation is not based on *idf* data but on the tag weights.

2.4 Proximity and structure

Among the information retrieval ranking models which incorporate proximity, few of them also consider the structure. To our knowledge, there are only three proposals to this end. In the first one Broschart and Schenkel (2008) extend the proximity score initially introduced by Büttcher et al (2006): the structure is taken into account when computing the distance between the term occurrences by introducing virtual gaps at the border of elements in accordance with the element tags. Experiments performed at the document level (classical IR) showed that proximity scoring improves the precision and secondly that the structure gives an additional improvement, but, at the element level (focused IR), the effect of the structure is not positive. However, in their work, only the logical structure is considered. Moreover, the sizes of the gaps are chosen manually and, as pointed out by the authors, automatic methods should be applied to determine the appropriate gap sizes.

Svore et al (2010) extend the Song et al (2008)'s method. Presence or absence of some highlighting formatting tags (bold, italic, etc.) are used in the machine learning process input to score the documents. The evaluation confirms the improvement brought by proximity, but no conclusion can be drawn on the usefulness of using the formatting tags because in the presented results their contribution is mixed with that of linguistic features.

The third and last proposal integrates the structure in the Beigbeder and Mercier's proximity model for flat documents and for two usages: the first one is the definition of logical units to be returned to the user, and the second one is the enlargement of the influence function range over whole sections when query terms appear in the title of the elements. However non logical tags are not taken into account in this work (Beigbeder 2010).

Only Broschart and Schenkel (2008) and Beigbeder (2010) proposed methods to achieve focused retrieval. Moreover, only that of Beigbeder proved its effectiveness in this context. So, the present study is based on this method.

Further investigations were conducted. The first one is the extension to the focused *Relevant in Context* task as defined in the INEX 2010 campaign. The Beigbeder (2010)'s method addressed focused search but without the *Relevant in Context* constraint. In order to take into account this constraint, we use a

Fetch and Browse method such that returned elements are grouped by documents. The second investigation concerns the tags: in Beigbeder's work, logical tags are manually defined and propagation is used for title tags ; in this work, tags are exploited by a learning process. The third investigation is a comparison of manual queries and automatic queries.

3 Scoring with proximity and tag weights

In our retrieval model, we apply a *Fetch and Browse* strategy as proposed by Chiaramella et al (1996). In the fetch step, documents are ranked using a traditional BM25 model. In the browse step, parts of the documents returned in the fetch step are extracted and ranked. This ranking is performed with the proximity model improved with structural hints.

More precisely, the structure is exploited on three levels:

1. Firstly the logical structure allows for the definition of the granularity of the elements which might be returned by the system to the user: the logical elements.
2. Then, the structure (through all the tags: logical tags and other tags) contributes to the relevance of a logical element for a given query. For this purpose, during a training step, we consider, for a given collection and a set of queries, a training set composed of the relevant elements corresponding to each query. These assessments are used to compute a weight for each tag, based upon the probability that the tag is able to distinguish between relevant and non-relevant terms. On the one hand, the larger the relevant passages marked by a tag are, the higher their weight is. On the other hand, the larger the non relevant passages marked by a tag are, the lower their weight is.
3. Finally, during the query step, the weights of the tags which mark the query terms are considered in the ranking function. For this purpose, we define, around each occurrence in the document of a query term, a text area which is influenced by this occurrence. We measure this influence with a function, called influence function. Then, the influence functions of the query terms are combined with the tag weights in order to compute the score of the elements.

A more formal presentation of this model is given in the next sections.

3.1 Notations

Let D be a set of structured XML documents and T the set of terms built from D. In order to illustrate our purpose, we consider a simplified version of the INEX "Handel House Museum" article introduced in Section 1 (cf. Figure 5). We note B the set of tags which appear in the collection. Among these tags, we distinguish B_l, the set of logical tags like *article*, *section*, *p*, etc. and B_t, the set of other tags corresponding for instance to formatting tags like *strong* or *italic*: $B = B_l \cup B_t$. In our example, $B_l = \{article, bdy, p\}$.

The tags define the set E of elements that correspond to parts of documents. These elements are named with their XPath designation. The subset E_l of E

is the set of logical elements: these are the elements that are delimited by logical tags. These elements are the only ones which can be returned to the user. In the example given in Figure 5, $E(d_1)$ denotes the set of elements defined by document d_1 including the logical elements $d1/article[1]$, $d1/article[1]/bdy[1]$ and $d1/article[1]/bdy[1]/p[1]$.

In the proximity based model, introduced in Beigbeder and Mercier (2005), a document d is defined as a function which associates a term $t \in T$ to each position x in the document:

$$\begin{aligned} d : \mathbb{N} &\rightarrow T \\ x &\mapsto d(x) \end{aligned} \tag{1}$$

The set of positions in the document d where one occurrence of term $t \in T$ appears is noted $d^{-1}(t)$ and $|d|$ is the size of the document *i.e.* its number of word occurrences. In the example, $|d_1|$ equals 32 while $d_1^{-1}(museum)$ is the set $\{2, 6\}$ since the word *museum* appears at position 2 and at position 6 in the document d_1. An element $e \in E$ is characterized by the positions of its first and its last term: $x_1(e)$ and $x_2(e)$. For example, the first and the last position of the element $d1/article[1]/header[1]/title[1]$ are respectively 0 and 2. We note $e(x)$, the deepest element (in the XML tree) that surrounds the position x, and $e_l(x)$, the deepest logical element that surrounds this position. For instance, for the sixth position (position 5) of the document d_1, corresponding to the word *House*, the deepest element is $e(5) = \texttt{d1/article[1]/bdy[1]/p[1]/b[1]}$ while the deepest logical element is $e_l(5) = \texttt{d1/article[1]/bdy[1]/p[1]}$. In addition, we say that a position x is marked by tag $b \in B$ if b belongs to the path of $e(x)$ in the XML tree associated to this element. Finally, $M_b(e)$ is the set of the positions in the element e marked by the tag b. In our example, we have $M_{\texttt{title}}(\texttt{d1/article[1]}) = \{0, 1, 2\}$ and $M_{\texttt{b}}(\texttt{d1/article[1]}) = \{4, 5, 6\}$.

In order to compute the score $s(q, e)$ of an element e, given a query q, this model introduces the influence function of a term on a position and the influence of a query on a position. These notions are briefly presented in the following sections. An extended presentation can be found in (Beigbeder 2010).

3.2 Structure enhanced proximity model

3.2.1 Influence of a term on a position

Firstly, given a document d we compute the influence of one occurrence of term t at position i on one position x with an *influence function*. Any function with the three following properties is acceptable:

- symmetric around i,
- decreasing with the distance to i,
- maximum (value 1) reached at i.

The simplest one is a linearly decreasing function centered around i:

$$x \mapsto \max(\frac{k - |x - i|}{k}, 0)$$

where k is a parameter which controls the size of the influence area, *i.e.* the zone where the influence of the occurrence of term t is not zero. The curves of such

Document d_1

```
<article>
  <header>
    <title>Handel House Museum</title>
  </header>
  <bdy>
    <p>
      The <b>Handel House Museum</b> was the home of the German born baroque composer
      George Frideric Handel. He composed works such as <it>The Messiah</it> there.
    </p>
    The house has been restored...
  </bdy>
</article>
```

$T = \{as, baroque, been, born, composed, composer, frideric, george, german, handel, has, he, home, house, messiah, museum, of, restored, such, the, there, was, works\}$
$E(d_1) = \{$d1/article[1], d1/article[1]/header[1], d1/article[1]/header[1]/title[1], d1/article[1]/bdy[1], d1/article[1]/bdy[1]/p[1], d1/article[1]/bdy[1]/p[1]/b[1], d1/article[1]/bdy[1]/p[1]/it[1]$\}$
$B = B_t \cup B_l$
$B_t = \{$b, header, it, title$\}$
$B_l = \{$article, bdy, p$\}$
$d_1(0) = handel$
$d_1(1) = house$
$d_1(2) = museum$
...
$d_1(15) = composer$
...
$|d_1| = 32$
$d_1^{-1}(museum) = \{2, 6\}$
x_1(d1/article[1]/header[1]/title[1]) $= 0$
x_2(d1/article[1]/header[1]/title[1]) $= 2$
$e(5) =$ d1/article[1]/bdy[1]/p[1]/b[1]
$e_l(5) =$ d1/article[1]/bdy[1]/p[1]
M_{title}(d1/article[1]) $= \{0, 1, 2\}$
M_{b}(d1/article[1]) $= \{4, 5, 6\}$

Fig. 5 Collection example with one document (simplified version of the "Handel House Museum" article, cf. Figure 1)

functions have a triangle shape, so we call them *triangle functions*. When the distance between x and i is greater than k, the influence is zero – that is to say that the occurrence of term t at position i is too far from position x to influence it. Moreover, the influence is limited to the logical element $e_l(i)$ that surrounds the position i of the occurrence of the query term t. For this reason, we take the product of the triangle function by the characteristic function $\mathbf{1}_{e_l(i)}$ of the position range that belongs to the logical element $e_l(i)$. Lastly, the influence must be that of the nearest occurrence of the term t, which can be obtained with $\max_{i \in d^{-1}(t)}$ because the influence function is symmetric and decreases with the distance.

So, the influence $p_t^d(x)$ of the term t on the position x in the document d is defined by:

$$p_t^d(x) = \max_{i \in d^{-1}(t)} \left(\mathbf{1}_{e_l(i)} \cdot \max\left(0, \frac{k - |x - i|}{k}\right)\right) \tag{2}$$

Figure 6 shows the influence of the terms `composer` and `museum` (extracted from the topic previously presented in Figure 2) on each position x between 0 and 31, in the illustrative document d_1, given in Figure 5, with $k = 7$. This small value of k leads to readable figures but in the experiments the parameter k is set to 200,

because firstly Hearst (1996) suggests to use windows of 100 to 300 words and secondly that roughly provides influence areas of the size of a paragraph.

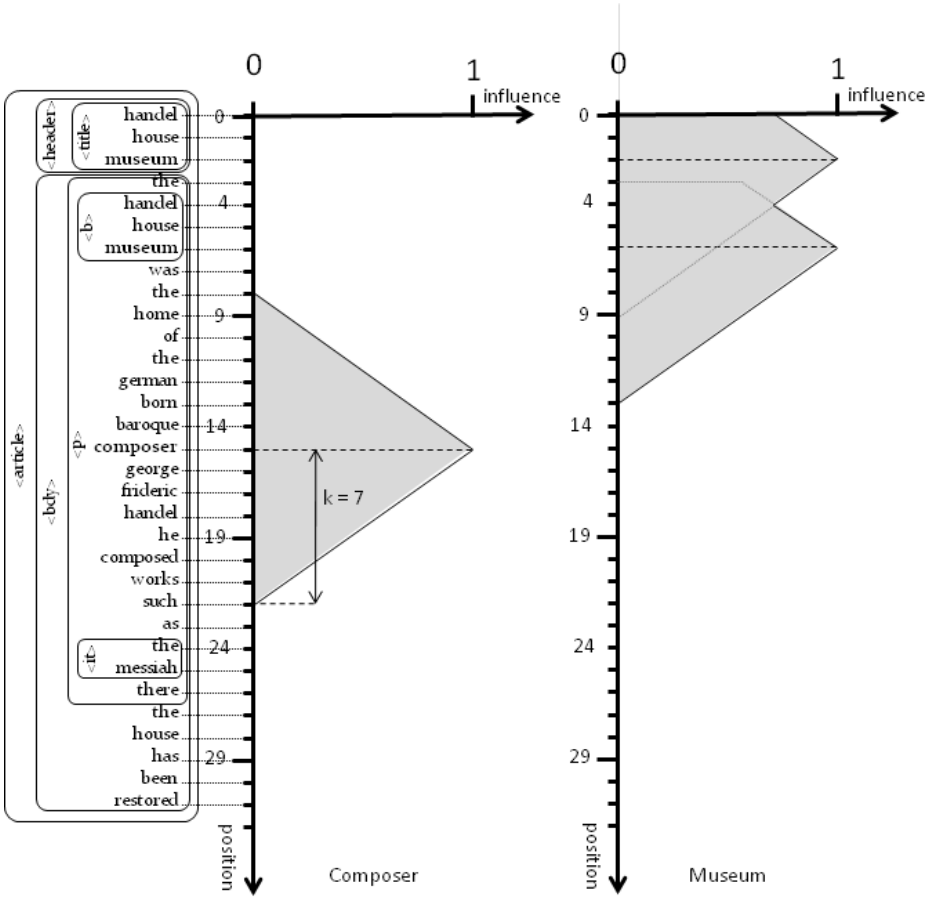

Fig. 6 Influence of the terms `composer` (on the left) and `museum` (on the right)

3.2.2 Influence of a query on a position

As previously mentioned, the influences of the query terms on a position are used to compute the influence of a query on a position, which is used itself to compute the score of the elements for this query. This influence of a query on a position is defined as follows. In the simplest case where a query q contains only one term $t \in T$, the influence of the query on a position x equals the influence of the term t on the position x:

$$p_q^d(x) = p_t^d(x) \tag{3}$$

In the other cases, the query q is defined, as in the Boolean model, by a tree with conjunctive and disjunctive nodes. Formula 3 is used on the leaves of the tree and the two following equations are used on the other nodes. The influence of a conjunctive query "q_1 AND q_2" is the minimum of the influence functions of its sub-queries:

$$p^d_{q_1 \text{ AND } q_2}(x) = \min(p^d_{q_1}(x), p^d_{q_2}(x)) \tag{4}$$

Similarly, the influence of a disjunctive query "q_1 OR q_2" is the maximum of the influence functions of its sub-queries:

$$p^d_{q_1 \text{ OR } q_2}(x) = \max(p^d_{q_1}(x), p^d_{q_2}(x)) \tag{5}$$

These formulas are used recursively during a post-order traversal of the query tree to compute the influence on the root of the tree, that is to say the influence of the query itself.

Figure 7 shows the influence of the queries q_{ex1} = "`composer` AND `museum`" and q_{ex2} = "`composer` OR `museum`" to the positions of document d_1.

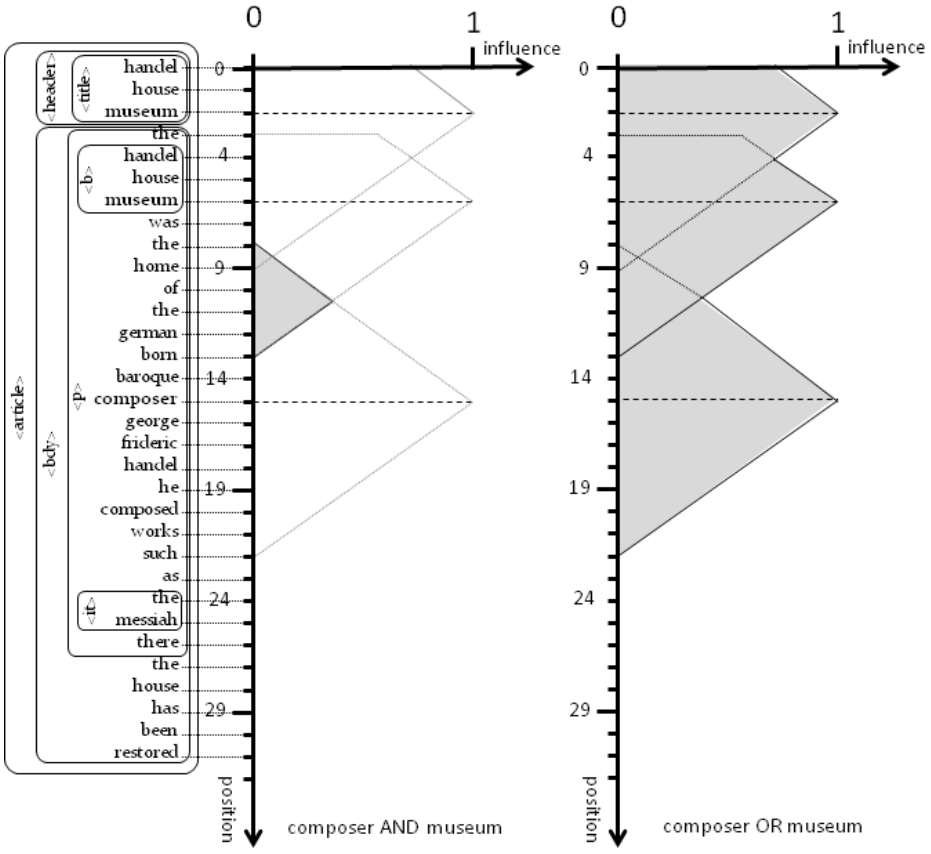

Fig. 7 Influence of the queries q_{ex1} = "`composer` AND `museum`" and q_{ex2} = "`composer` OR `museum`"

3.2.3 Score of an element

Given a query q and a logical element e characterized by the positions of its first and its last terms ($x_1(e)$ and $x_2(e)$), the proximity score $s_p(q,e)$ of e for q equals the sum of the influence of the query q on each position of e, normalized by the size of e:

$$s_p(q,e) = \frac{\sum_{x_1(e)\leq x\leq x_2(e)} p_q^d(x)}{x_2(e) - x_1(e) + 1} \tag{6}$$

We can note that the normalization by the number of occurrences of terms in the element favours the short elements; which is advantageous in the context of focused information retrieval.

In our example, the score of the logical element `d1/article[1]/bdy[1]/p[1]` is given in Figure 8 for the queries q_{ex1} and q_{ex2}.

$$s_p(q_{ex1}, \texttt{d1/article[1]/bdy[1]/p[1]}) = \frac{0+\ldots+0+\frac{1}{7}+\frac{2}{7}+\frac{2}{7}+\frac{1}{7}+0+\ldots+0}{26-3+1} \approx 0.0357$$

$$s_p(q_{ex2}, \texttt{d1/article[1]/bdy[1]/p[1]}) = \frac{\frac{4}{7}+\frac{5}{7}+\frac{6}{7}+\frac{7}{7}+\frac{6}{7}+\frac{5}{7}+\frac{4}{7}+\frac{3}{7}+\frac{3}{7}+\frac{4}{7}+\frac{5}{7}+\frac{6}{7}+\frac{7}{7}+\frac{6}{7}+\frac{5}{7}+\frac{4}{7}+\frac{3}{7}+\frac{2}{7}+\frac{1}{7}+0+0+0+0+0}{26-3+1} \approx 0.5119$$

Fig. 8 Scores of the logical element `d1/article[1]/bdy[1]/p[1]` for two queries q_{ex1} and q_{ex2}

3.3 Weighting the tags

As we suppose that the tags may be used to emphasize words, they can be exploited to improve the detection of relevant information. In order to measure the capacity of a tag to emphasize terms in relevant passages, a weight is estimated for each tag using a training set. For each tag $b \in B$, this weight is computed following the learning method proposed by Géry and Largeron (2012): it is based on the probability that b marks either a relevant position[4] or an irrelevant one. This weight is afterwards used to modulate the influence function of the term occurrences.

A first set of queries with assessments is used as a training set. Given this learning set, a contingency table (Table 1) is built. In this contingency table, $R_q(e)$ is the set of the relevant positions in the element $e \in E$ for the topic $q \in Q$, and $M_b(e)$ is the set of the positions of e marked by the tag $b \in B$.

The weight $w_b(q)$ of a tag b for a query q is defined by:

$$w_b(q) = \frac{\frac{t_{rm}(b,q)+s}{t_{rm}(b,q)+t_{r\overline{m}}(b,q)+s}}{\frac{t_{\overline{r}m}(b,q)+s}{t_{\overline{r}m}(b,q)+t_{\overline{r}\overline{m}}(b,q)+s}} \tag{7}$$

[4] A relevant position is a position marked as relevant by the assessor.

	$R_q(e)$	$\overline{R_q(e)}$
$M_b(e)$	$t_{rm}(b,q)$	$t_{\overline{r}m}(b,q)$
$\overline{M_b(e)}$	$t_{r\overline{m}}(b,q)$	$t_{\overline{rm}}(b,q)$
Total	$t_r^{coll}(q)$	$t_{\overline{r}}^{coll}(q)$

Table 1 Contingency table for the query q and for the tag b

with:

- $t_{rm}(b,q) = \sum_{e \in E} |R_q(e) \cap M_b(e)|$: number of relevant positions for the query q marked by the tag b;
- $t_{r\overline{m}}(b,q) = \sum_{e \in E} |R_q(e) \cap \overline{M_b(e)}|$: number of relevant positions for the query q not marked by the tag b;
- $t_{\overline{r}m}(b,q) = \sum_{e \in E} |\overline{R_q(e)} \cap M_b(e)|$: number of irrelevant positions for the query q marked by the tag b;
- $t_{\overline{rm}}(b,q) = \sum_{e \in E} |\overline{R_q(e)} \cap \overline{M_b(e)}|$: number of irrelevant positions for the query q not marked by the tag b.

The parameter s is a smoothing parameter, which was fixed at 0.5 in our experiments.

In fact, we believe that the capacity of a tag to highlight relevant terms (or on the contrary to reduce their visibility) is intrinsic to the tag itself and is not dependent on the query. Thus, we estimate the weight w_b for each tag b instead of a weight for each pair (tag b, query q). This weight w_b of a tag b is defined as the average on the learning set of queries, according to the formula:

$$w_b = \frac{1}{|Q|} \sum_{q \in Q} w_b(q) \tag{8}$$

3.4 Modulating influence function shapes

Then the weights of the tags are integrated into the score of an element. More precisely, the weights of the tags are used to modulate the influence function of the query term occurrences with two methods. In the first one, the height of the triangle is modified and the resulting influence function of a term is:

$$ph_t^d(x) = \max_{i \in d^{-1}(t)} \left(\mathbf{1}_{e_l(i)} \cdot \max \left(0, w_{b(i)} \cdot \frac{k - |x-i|}{k} \right) \right) \tag{9}$$

and in the second one, both the height and the width of the triangle are modified and the resulting influence function of a term is:

$$phw_t^d(x) = \max_{i \in d^{-1}(t)} \left(\mathbf{1}_{e_l(i)} \cdot \max \left(0, \frac{w_{b(i)} \cdot k - |x-i|}{k} \right) \right) \tag{10}$$

Figure 9 shows the modulation of the influence function of the query q_{ex1} = "`composer` AND `museum`" on the positions of the document d_1 using the following tag weights: $w_{\texttt{title}} = 1.5$, $w_{\texttt{b}} = 1.4$, $w_{\texttt{p}} = 0.9$, according to the strategies ph (on the left) and phw (on the right). The score of the logical element `d1/article[1]/bdy[1]/p[1]` using these tag weights is given in Figure 10 for the queries q_{ex1} and q_{ex2}.

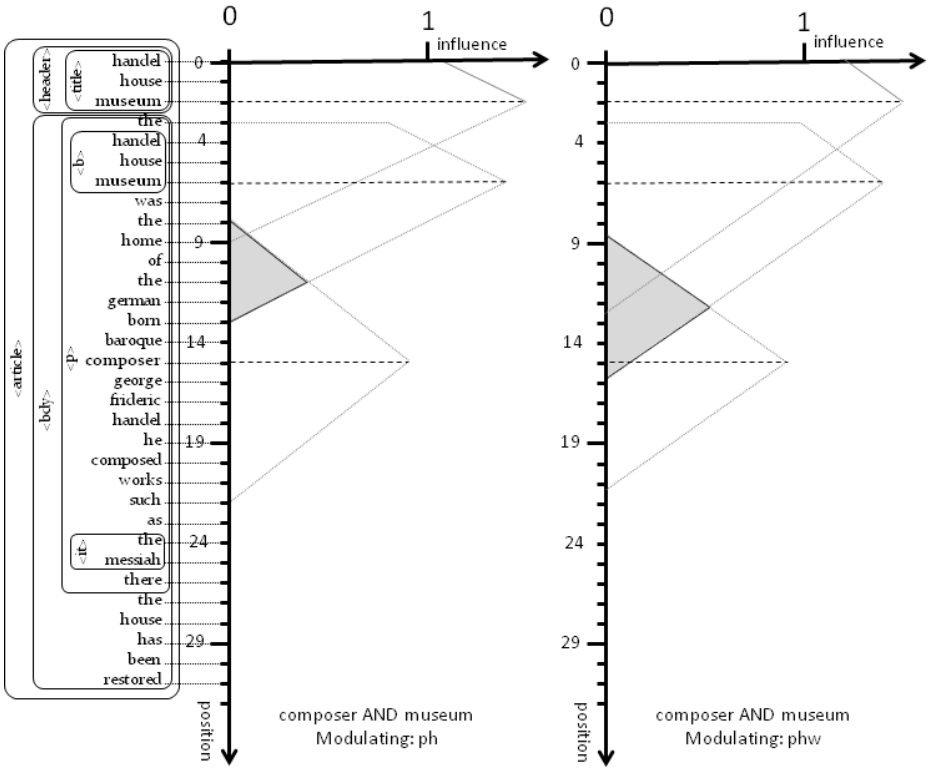

Fig. 9 Modulation of the influence function of the query q_{ex1} = "`composer` AND `museum`" using the three tag weights: $w_{\texttt{title}} = 1.5$, $w_{\texttt{b}} = 1.4$, $w_{\texttt{p}} = 0.9$

$s_p(q_{ex1}, \texttt{d1/article[1]/bdy[1]/p[1]})$	≈ 0.0357
$s_{ph}(q_{ex1}, \texttt{d1/article[1]/bdy[1]/p[1]})$	≈ 0.0405
$s_{phw}(q_{ex1}, \texttt{d1/article[1]/bdy[1]/p[1]})$	≈ 0.0750

$s_p(q_{ex2}, \texttt{d1/article[1]/bdy[1]/p[1]})$	≈ 0.5119
$s_{ph}(q_{ex2}, \texttt{d1/article[1]/bdy[1]/p[1]})$	≈ 0.5804
$s_{phw}(q_{ex2}, \texttt{d1/article[1]/bdy[1]/p[1]})$	≈ 0.6173

Fig. 10 Scores of the logical element `d1/article[1]/bdy[1]/p[1]` for two queries q_{ex1} and q_{ex2}, using the three tag weights: $w_{\texttt{title}} = 1.5$, $w_{\texttt{b}} = 1.4$, $w_{\texttt{p}} = 0.9$

4 Experiments

The framework for our experiments is INEX[5], the international XML IR campaign which is presented in the next section with the related evaluation measures (Section 4.2) and experimental protocol (Section 4.3). Then we describe our query building method in Section 4.4, and finally the selection and weighting of the tags in Section 4.5.

4.1 The INEX collection

For our experiments, we used the INEX Ad-Hoc 2009 & 2010 collections, extracted in October 2008 from the English Wikipedia[6] (Schenkel et al 2007). This collection contains a significant amount of structured XML data. It also contains relevance assessments measured at the character granularity[7], which allows evaluation of the quality of Focused XML IR systems.

The corpus includes 2,666,190 articles from the Wikipedia encyclopaedia, and 120 topics with the relevance judgments (68 of them were used during the 2009 INEX edition, and the remaining 52 topics were used during the 2010 edition). The original Wiki syntax was converted by the organizers into XML, using tags for the logical structure (*e.g. article*, *sec*, *p*, etc.), formatting tags (*e.g. b*, *it*, etc.), link tags (*e.g. link*, *weblink*, etc.) and semantic tags (*e.g. company*, *song*, *writer*, etc.). Some of the tags belonging to this last category were extracted from the textual content by YAGO (Schenkel et al 2007).

The documents are strongly structured since they are composed of more than 2 billion XML elements and 101,917,424 of them contain at least 50 characters. There is no DTD defining the available tags. Consequently, there are 32,311 different tags in the collection, although most of them appear in very few articles. Each XML article can be viewed as a tree containing on average 750 elements (with 38 of them containing at least 50 characters). Moreover, the whole articles (textual content + XML structure) represent 50.7 GB of data whereas the textual content represents only 12 GB. Thus, the structural information (tags and attributes) is four times as large as the textual information.

Moreover, we chose not to use a stemmer or stopword removal, because many experiments have been conducted on the INEX collections, by ourselves and by other INEX participants, and with these collections, stemming has not always proven to be very effective. More precisely, Jia et al (2011) have studied the effect of various stemming algorithms: they have shown that indexing without stemming gives better results than the well known Porter stemmer, and that their refined stemmer improves the results slightly.

[5] Initiative for the Evaluation of XML Retrieval: `http://inex.mmci.uni-saarland.de/`

[6] Wikipedia: `http://wikipedia.org`

[7] In the INEX 2010 assessments, the assessors had to highlight the relevant passages, and this is performed at the character level.

4.2 Evaluation measures and baseline

We have evaluated our model in the context of the "Relevant in Context Task" (RIC) of the INEX campaign. The scenario underlying this task is the return of a ranked list of articles and within those articles the relevant information captured by a set of non-overlapping elements or passages (Arvola et al 2011). Thus, the ranked list of XML elements should be grouped per article.

The evaluation of the Relevant in Context Task is based on the measures of generalized precision and recall over articles (Kekäläinen and Järvelin 2002). INEX is most interested in overall performances, so the main INEX measure is the *mean average generalized precision*, $MAgP$, introduced together with the *generalized precision*, gP, at INEX 2006 (Lalmas et al 2007). The score per document is the harmonic mean of precision and recall in terms of the fractions of retrieved and relevant text in the document. It reflects how well the retrieved text matches the relevant text in the document.

The INEX 2010 Relevant in Context task is viewed as a form of "snippet" retrieval, and the evaluation takes length and reading effort into account, including a "Tolerance to Irrelevance" (T2I) score per document into the generalized precision and recall measures (Arvola et al 2011). The reading stops when the user's tolerance to irrelevance is met (e.g. 300 irrelevant characters with T2I(300)).

All the results presented here, including those of INEX systems, were computed using the INEX 2010 evaluation programs: *inex_eval*, version 3.0, including the T2I(300) score per document.

The results obtained with our models are compared with those of the **Reference run** provided by the organizers. This run, based on a tuned BM25 method, is considered as the baseline in our article. It is important to note that this method, designed for classical information retrieval, has also proved its effectiveness in the context of focused information retrieval, especially during the INEX campaign (Arvola et al 2011).

The significance of the improvements against the baseline has been checked by using statistical tests based on Wilcoxon matched-pairs signed-rank test at the 0.05 level, *i.e.* the improvement is significant when the p-value is less than 0.05.

4.3 Experimental protocol

In the learning stage, the 2009 INEX set of 68 queries and the associated relevance judgments related to the collection composed of 2,666,190 articles, were firstly used as a training set in order to estimate the tag weights w_b. Following this step, our indexing and querying experiments were carried out on the same 2,666,190 articles but using the 52 new queries from the 2010 edition of the INEX Ad-Hoc track. The 2010 set of queries is thus used as a testing set. Therefore, even if the same collection of documents is used in both stages: when estimating tag-weights (*i.e.* the training stage), and during IR experiments (*i.e.* testing stage), it represents in fact two distinct collections from a IR point of view, thanks to the two different sets of queries. The problem of overfitting is thus avoided.

4.4 Manual and automatic queries

As explained in Section 3.2.2, the proximity model requires Boolean queries. In the first series of experiments, we used a set of manually built Boolean queries based on the official set of 52 topics of INEX 2010. These Boolean queries were mainly built with the terms of the title field of the topic connected with the AND operator, and sometimes the OR operator was also used. For some queries we also used some variations (synonyms, close concepts) from the description and narrative fields or flexional variants, and in this case these terms were combined with the OR operator with the original terms of the title field terms. We did not use the field *castitle* (structured part of the query). For example, the query built for the topic presented in Figure 2 is: `(composer OR composers) AND (dedicated OR dedicate) AND museum`.

Then, in the second series of experiments, we used an automatic method to build the query. The main idea is to connect the terms of the topic fields with AND operators but as the INEX topic titles use two operators ('+' and '-') we have to take them into account. These operators are given as hints to search engines and do not have strict semantics: '+' is used to emphasize an important concept, and '-' is used to denote an unwanted concept. So, the following rules were applied at the lexical level:

- removing of the '+' operator;
- replacement of '-' operator by the NOT operator;
- the remaining items (simple terms or phrases) are connected by the AND operator.

For example, the automatic query extracted from the topic presented in Figure 2 is composed of only two terms and one operator: "`composer AND museum`".

4.5 Tag selection and weighting

Another important parameter is the set B_l of logical tags that defines what elements are returnable by the system. For the experiments, we selected 12 logical tags considering their frequency in the whole INEX 2009 collection (*i.e.* the training set) and in the relevant passages. The following criteria were used:

- coverage $\geq 1\%$: the ratio between the number of term occurrences marked by the tag and the number of term occurrences in the collection;
- top 25 "relevant" tags: the 25 tags having the higher number of occurrences in the relevance judgments (*i.e.* including at least one relevant character).

It leads to the following set of logical tags:

$$B_l = \{\texttt{article}, \texttt{bdy}, \texttt{col}, \texttt{entry}, \texttt{list}, \texttt{p}, \texttt{reflist}, \texttt{row}, \texttt{sec}, \texttt{ss1}, \texttt{ss2}, \texttt{table}\}$$

The set $B = B_l \cup B_t$ used in our model was chosen amongst the 32,311 different tags appearing in the 2,666,190 documents, using the following criteria:

- Select frequent tags: number of occurrences $\geq$ 1,000;
- Select high coverage tags: coverage $\geq 0.01\%$;
- Select non-semantic tags: tags without `wordnetid` attribute (this attribute appears in the semantic tags added by YAGO).

Tag	Weight	Coverage	Logical tag	#occs
direction	31.35	0,04%	-	1'533
format	16.60	0,07%	-	1'436
mission	15.89	0,06%	-	1'195
residence	12.37	0,07%	-	2'838
engine	11.49	0,09%	-	4'053
shape	7.62	0,09%	-	2'761
genre	7.58	0,10%	-	4'075
code	7.55	0,05%	-	1'034
branch	6.19	0,04%	-	1'752
subject	6.13	0,12%	-	1'737
...	...	...	...	...
event	1.45	4,65%	-	137'438
ss1	*1.23*	*17,17%*	X	*1'701'799*
ss2	*1.23*	*2,31%*	X	*264'936*
artist	1.07	2,38%	-	57'958
bdy	*1.03*	*95,51%*	X	*2'649'102*
article	*1.00*	*100,00%*	X	*2'666'958*
sec	*0.86*	*69,86%*	X	*6'468'391*
p	*0.77*	*87,22%*	X	*19'745'936*
entry	*0.72*	*13,77%*	X	*12'923'052*
writer	0.69	2,37%	-	70'317
list	*0.69*	*14,41%*	X	*4'458'003*
group	0.61	3,27%	-	91'361
reflist	*0.58*	*3,95%*	X	*743'283*
location	0.54	5,57%	-	184'472
region	0.37	5,36%	-	180'189
template	0.36	2,56%	-	900'226
parameters	0.35	2,37%	-	887'629
district	0.33	3,91%	-	100'109
header	0.27	4,49%	-	3'066'317
table	0.22	38,94%	-	4012'236
row	*0.21*	*36,04%*	X	*11'720'914*
leader	0.20	2,60%	-	92'596
col	*0.20*	*31,17%*	X	*7'256'813*
city	0.18	2,26%	-	43'407
title	0.14	0,04%	-	42'227
commune	0.12	2,14%	-	38'206

Table 2 Tag weights, sample of $B = B_l \cup B_t$: top 10 tags, or coverage > 2%

The resulting set was composed of 201 tags.

We note that even if this selection eliminates 99.4% of the 32,311 different tags, most of the tag occurrences are still considered. Indeed, the 0.64% remaining tags (201 / 32,311) correspond to more than 99% of tag occurrences.

The weights of the 201 tags of $B = B_l \cup B_t$, including the 12 logical tags B_l, were computed according to equation 7. Table 2 presents a sample of $B = B_l \cup B_t$: the ten highest weighted tags together with the tags having a coverage ratio greater than 2%.

We note that most of the top tags have a very low coverage ratio (*e.g.*: `direction`, `format`, `mission`, etc.). Their impact on the XML elements scoring is thus very low. Most of these tags belong to the category of the semantic tags (anterior to YAGO).

We also note that most of the tags having an important coverage ratio have a structural function in the document (*e.g.*: `ss1`, `ss2`, `sec`, `p`, `list`, `table`, `row`, `col`, etc.). Most of them belong to our set B_l of logical tags.

4.6 Fetch and browse implementation

One objective of our work is to evaluate the effectiveness of the *Fetch and Browse* approach in the context of focused information retrieval. The INEX campaign uses a tuned BM25 as a baseline. For this reason we also use it as a baseline in order to allow comparison with the other participants submissions. Moreover this choice for the Fetch method allows for the analysis of the improvements brought by the Browse method.

Then, the structure enhanced proximity model is used in the browse step in order to choose some elements within a document. The score for each logical element is computed according to formula 6 and the influence functions that take into account tag weights as explained in Section 3.4.

More precisely, the idea is to sort the logical elements of the document by decreasing proximity score, and to return the top ranked elements. Moreover, the elements that overlap with already returned element are eliminated so that there is no duplication of returned text in the final list.

5 Results

We will now present the results obtained by our model on the INEX Ad-Hoc 2010 collection. Our objective was firstly to compare our *Fetch and Browse* approach with the BM25 baseline then to evaluate the use of automatic Boolean queries against the use of manual queries, to study the impact of tag-weights in the structure enhanced proximity model, and finally to put the results in the context of the INEX 2010 campaign. Three different models have been evaluated: **prox, prox-h, prox-hw**. BM25 is used at the fetch step for each model.

- **prox**: our structure enhanced proximity model is used at the browse step (cf. equation 2).
- **prox-h**: our structure enhanced proximity model is used at the browse step, and the tag weights are used to modulate the height of the influence function (cf. equation 9).
- **prox-hw**: our structure enhanced proximity model is used at the browse step, and the tag weights are used to modulate the height and the width of the influence function (cf. equation 10).

With our model two series of experiments were performed, using either manually built queries or automatic queries (cf. Section 4.4) and the results are respectively presented in Tables 3 and 4.

5.1 Fetch and browse approach

In the first series of experiments, performed with manually built queries, we compare the *Fetch and Browse* approach (fetch = **BM25**, browse = **prox**, **prox-h** or **prox-hw**) with the BM25 method alone (noted **Reference**) using $MAgP$ and $gP[10]$ measures. Table 3 shows the results. Qualitatively, we note that the browse step, based on the structure enhanced proximity model, is useful for focused information retrieval. Quantitatively the improvements are above 25% for the two

evaluation measures. Indeed, the *MAgP* is equal to 0.1436 for the BM25 model when it is higher or equal to 0.1799 for the other systems. Similarly, the *gP*[*10*] is at 0.2322 for BM25 while it reaches at least 0.2952 for the *Fetch and Browse* approach. These results confirm the effectiveness of a *Fetch and Browse* approach with a browse step based on a structure enhanced proximity model.

	Queries	Fetch	Browse	*MAgP*		*gP*[*10*]	
Reference	INEX	BM25	-	0.1436	0.0%	0.2322	0.0%
Prox	INEX+Manual	BM25	prox	0.1835*	27.8%	0.3025*	30.3%
Prox-h	INEX+Manual	BM25	prox-h	0.1834*	27.7%	0.3023*	30.2%
Prox-hw	INEX+Manual	BM25	prox-hw	0.1799*	25.3%	0.2952*	27.1%

Table 3 Results with manual queries. The star indicates the results are statistically better than the Reference run (Wilcoxon matched-pairs signed-rank test at the 0.05 level)

5.2 Manual versus automatic queries

In the second series of experiments, we use the same settings for the different systems but with automatic queries built with the title field of the topics (cf. Section 4.4). Table 4 displays *MAgP* and *gP*[*10*] measures obtained with the baseline (**Reference**) and with the structure enhanced proximity models (**prox**, **prox-h** or **prox-hw**).

	Queries	Fetch	Browse	*MAgP*		*gP*[*10*]	
Reference	INEX	BM25	-	0.1436	0.0%	0.2322	0.0%
Prox	INEX+Auto	BM25	prox	0.1605*	11.7%	0.2670*	15.0%
Prox-h	INEX+Auto	BM25	prox-h	0.1629*	13.5%	0.2713*	16.9%
Prox-hw	INEX+Auto	BM25	prox-hw	0.1593*	10.9%	0.2662	14.7%

Table 4 Results with automatic queries. The star indicates the results statistically better than the Reference run (Wilcoxon matched-pairs signed-rank test at the 0.05 level)

As previously, we obtain consistent improvements relative to the baseline, though their magnitude is lower, around 10%. From a statistical point of view, both the *MAgP* and the *gP*[*10*] are significantly improved compared to the baseline except for *gP*[*10*] with **prox-hw**, as indicated by stars in Table 4.

In Section 4.4 an example of a manually built query was given. In the automatic set of queries there is an average of 3.15 words per query with a standard deviation of 1.30 and in the manual set of queries the average is 6.19 and the standard deviation is 2.92. As expected, the results are better when the queries are built manually but the use of manual queries can be seen as a drawback of the proximity models. In fact, our experiments show that this is not a limitation since Boolean queries can be automatically built from the users queries and the results obtained with these automatic queries remain better than those of the baseline. Thus, the proximity based model can be helpful in the context of focused information retrieval even with an automatic transformation of the queries provided by the users.

5.3 Tag weighting

Finally, the experiments also aimed at evaluating the impact of the integration of structural hints into the proximity model, in other words the aim is to compare the proximity based model **prox** with its variants (**prox-h** and **prox-hw**) in which the weights of the tags are integrated in order to modulate the influence function. The results show that the improvement against the baseline varies depending on the way the structure is taken into account. More precisely, the models **prox-h** and **prox** give better results than the model **prox-hw** for the *MAgP* criterion as well as for $gP[10]$.

Indeed, with manual queries, in Table 3, the improvement against the baseline in terms of *MAgP*, statistically significant, is equal to 27.7% for **prox-h** and 27.8% for **prox** when it is equal to 25.3% for **prox-hw**. The improvement is also statistically significant in terms of $gP[10]$ with 30.2% for **prox-h**, 30.3% for **prox** and only 27.1% for **prox-hw**.

In the same way, with automatic queries, the results of Table 4 confirm the advantage of the models **prox** and **prox-h** in comparison with **prox-hw** with an improvement of 13.5% for **prox-h**, 11.7% for **prox**, 10.9% for **prox-hw** in terms of *MAgP*, and of 16.9% for **prox-h**, 15% for **prox** and 14.7% for **prox-hw** in terms of $gP[10]$.

These results show that taking structural information into account (model **prox-h**) gives better results than the proximity based model **prox**, especially with automatic queries. Given that the promixity model **prox** improves the strong baseline based on a well-tuned BM25 weigthing function (**Reference**), the improvement between **prox** and **prox-h**, despite being not statistically significant, is very encouraging and confirms the interest of taking structural information into account for focused information retrieval.

Moreover, the structural information helps to improve the results when it is used to modulate the height of the influence function (**prox-h**), but not when it is used to modulate its width (**prox-hw**). Thus, we conclude that the tag weights can be helpful to enhance the relevant passages of a document (*i.e.* improve their relevance score), but not to broaden them (*i.e.* returning larger passages).

5.4 Comparison with the INEX 2010 campaign

To put our results in context, we reproduce in Table 5 the results of the top ten participants in the INEX 2010 campaign AdHoc track Relevant in Context Task. According to (Arvola et al 2011), 18 teams submitted 213 runs. Among them the run labeled p4-Reference is the reference run we previously used. Among the top ten participants, six[8] are at the document level which means that good document retrieval models perform quite well in the context of focused retrieval. The four others in the top ten use a fetch and browse approach. Three of them (p22-Emse303R (Beigbeder et al 2011), p167-36p167 (Gao et al 2011), p5-reference (Arvola et al 2011, p. 18)) are uniquely or strongly based on BM25 for the fetch step, the fourth one (p98-I10LIA1FTri (Arvola et al 2011, p. 17)) uses a language model. For the browse step p98-I10LIA1FTri uses a very crude approach: it returns

[8] They are marked with *no* in the Browse column of Table 5.

	Browse	*MAgP*	*gP[10]*	
p22-Emse303R	yes	0.1977	0.3273	ENSM-SE
Manual Prox	*yes*	*0.1835*	*0.3025*	
Manual Prox-h	*yes*	*0.1834*	*0.3023*	
Manual Prox-hw	*yes*	*0.1799*	*0.2952*	
Auto Prox-h	*yes*	*0.1629*	*0.2713*	
p167-36p167	yes	0.1615	0.2536	Peking University
Auto Prox	*yes*	*0.1605*	*0.2670*	
Auto Prox-hw	*yes*	*0.1593*	*0.2662*	
p98-I10LIA1FTri	yes	0.1588	0.2607	LIA - University of Avignon
p5-reference	yes	0.1521	0.2372	Queensland University of Technology
p4-reference	no	0.1436	0.2322	University of Otago
p65-runRiCORef	no	0.1377	0.2310	Radboud University Nijmegen
p25-ruc-2010-base2	no	0.1372	0.2198	Renmin University of China
p62-RMIT10titleO	no	0.1335	0.2487	RMIT University
p55-DUR10atcl	no	0.1014	0.1484	Doshisha University
p6-0	no	0.0695	0.1614	University of Amsterdam

Table 5 Results of our model with manual and automatic queries (in italic) mixed with the Top 10 participants in the AdHoc Track Relevant in Context Task evaluated with the INEX 2010 T2I-score.

one element, the first section of the document. The p5-reference run is rather less crude as it returns the first section containing at least one of the search terms. Only p167-36p167 and p22-Emse303R use non trivial browse approaches. p167-36p167 is based on the selection of the elements that contain all the query words and a ranking with four features, two of them based on the proximity of the query words.

Finally the run p22-Emse303R was submitted by us and the model is very similar to the one described here. There are three differences: (i) the fetch was done with a mixture between the BM25 Reference run and the proximity model used at the article level, (ii) the list of logical tags was different from the one used here because these logical tags were manually selected, (iii) and propagation of title terms was used as in Beigbeder (2010) (details can be found in Beigbeder et al (2011)). Concerning the first difference, we explained in Section 4.6 why we wanted to use the actual BM25 Reference run for the fetch step. Concerning the two other differences, we wanted to use an automatic method to select, weight and exploit the logical tags in the focused retrieval model so that the same method could be used with another collection. So we choose here to focus on two components: proximity use and tag weights in the browse step.

As can be seen in Table 5, of our six runs, four of them would have obtained the second rank in the official ranking of the INEX 2010 campaign, and the two others would have obtained the third rank.

6 Conclusion and future work

Several editions of the XML IR evaluation campaign INEX have shown the difficulty of retrieving small units of information using the traditional IR models.

In order to handle this focused retrieval task, the approach presented in this article is based on a *Fetch and Browse* method. During the fetch step, documents are ranked using a traditional BM25 model, then during the browse step, parts

of the documents returned in the fetch step are extracted and ranked, using our model based on proximity enhanced with structural hints. As the proximity model requires Boolean queries, in our experiments, we used a first set of manually built Boolean queries and a second one automatically built.

The evaluation of this model on a large XML structured collection (INEX 2010 Wikipedia collection) shows that the use of term proximity and structure enhances the effectiveness of the traditional model in the context of focused information retrieval. Particularly, compared to the BM25 "fetch only" approach or to the *Fetch and Browse* method with the basic proximity model, the experiments confirm the effectiveness of taking structural information into account for focused information retrieval, especially when the tag weights are used to modulate the height of the influence function.

Moreover, a second series of experiments shows that it is possible to build the set of Boolean queries automatically, using an initial set of queries provided by users. The results obtained by our model with these automatic queries are not as good as with the manual queries, but remain significantly better than the baseline results. The structure enhanced proximity model can thus be easily implemented for focused information retrieval with user's queries.

We believe that combining proximity and structure is a promising approach for focused information retrieval. We should now refine our approach. Firstly, we have to study the parameters of the influence function, *e.g.* the parameter k to adjust the width of triangles, the modulation of the influence function. We also plan to explore on the one hand, other approaches for combining tag weights and, on the other hand, modulation strategies with different influence function shapes.

7 Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions, which were helpful in improving the paper.

References

Arvola P, Geva S, Kamps J, Schenkel R, Trotman A, Vainio J (2011) Overview of the INEX 2010 Ad Hoc Track. In: Comparative Evaluation of Focused Retrieval, 9th Workshop of the INititative for the Evaluation of XML Retrieval (INEX'10), pp 1–32

Bai J, Chang Y, Cui H, Zheng Z, Sun G, Li X (2008) Investigation of partial query proximity in web search. In: 17th conference on World Wide Web (WWW'08), Beijing, China, pp 1183–1184

Beigbeder M (2007) Structured content-only information retrieval using term proximity and propagation of title terms. In: Comparative Evaluation of Focused Retrieval, 5th Workshop of the INititative for the Evaluation of XML Retrieval (INEX'06), Dagstuhl Castle, Germany, pp 200–212

Beigbeder M (2010) Focused retrieval with proximity scoring. In: ACM Symposium on Applied Computing (SAC'10), Sierre, Switzerland, pp 1755–1759

Beigbeder M, Mercier A (2005) An information retrieval model using the fuzzy proximity degree of term occurences. In: ACM Symposium on Applied Computing (SAC'05), Santa Fe, New Mexico, USA, pp 1018–1022

Beigbeder M, Géry M, Largeron C, Seck H (2011) ENSM-SE and UJM at INEX 2010: Scoring with proximity and tag weights. In: Comparative Evaluation of Focused Retrieval, 9th Workshop of the INititative for the Evaluation of XML Retrieval (INEX'10), Vught, The Netherlands, pp 44–53

Boyan J, Freitag D, Joachims T (1996) A machine learning architecture for optimizing web search engines. In: AAAI Workshop on Internet-based Information Systems, Portland, Oregon, USA

Broschart A, Schenkel R (2008) Proximity-aware scoring for XML retrieval. In: 31st ACM Conference on Research and Development in Information Retrieval (SIGIR'08), Singapore, pp 845–846

Büttcher S, Clarke CLA, Lushman B (2006) Term proximity scoring for ad-hoc retrieval on very large text collections. In: 29th ACM Conference on Research and Development in Information Retrieval (SIGIR'06), Seattle, Washington, USA, pp 621–622

Chiaramella Y, Mulhem P, Fourel F (1996) A model for multimedia information retrieval. Tech. Rep. 4/96, University of Glasgow, report of ESPRIT Project 8134 "FERMI"

Clarke CLA, Cormack GV (1996) Interactive substring retrieval (multitext experiments for TREC-5). In: 5th Text REtrieval Conference (TREC'96), Gaithersburg, MD, USA, pp 267–278

Clarke CLA, Cormack GV, Burkowski FJ (1995) Shortest substring ranking (multitext experiments for TREC-4). In: 4th Text REtrieval Conference (TREC'95), Gaithersburg, MD, USA, pp 295–304

Clarke CLA, Cormack GV, Tudhope EA (2000) Relevance ranking for one to three term queries. Information Processing and Management 36(2):291–311

Cormack GV, Clarke CLA, Palmer CR, To SSL (1997) Passage-based refinement (multitext experiements for TREC-6). In: 6th Text REtrieval Conference (TREC'97), Gaithersburg, MD, USA, pp 303–319

Cummins R, O'Riordan C (2009) Learning in a pairwise term-term proximity framework for information retrieval. In: 32nd ACM Conference on Research and Development in Information Retrieval (SIGIR'09), Boston, MA, USA, pp 251–258

Gao N, Deng ZH, Jiang JJ, Lv SL, Yu H (2011) Combining strategies for XML retrieval. In: Comparative Evaluation of Focused Retrieval, 9th Workshop of the INititative for the Evaluation of XML Retrieval (INEX'10), pp 319–331

Géry M, Largeron C (2012) BM25t: a BM25 extension for focused information retrieval. Knowledge and Information Systems 32(1):217–241

Hawking D, Thistlewaite P (1995) Proximity operators - so near and yet so far. In: 4th Text REtrieval Conference (TREC'95), Gaithersburg, MD, USA, pp 131–143

He B, Huang JX, Zhou X (2011) Modeling term proximity for probabilistic information retrieval models. Information Sciences 181:3017–3031

Hearst MA (1996) Improving full-text precision on short queries using simple constraints. In: Proceedings of the Symposium on Document Analysis and Information Retrieval, pp 217–228

Jia X, Alexander D, Wood V, Trotman A (2011) University of Otago at INEX 2010. In: Comparative Evaluation of Focused Retrieval, 9th Workshop of the INititative for the Evaluation of XML Retrieval (INEX'10), Vught, The Netherlands, pp 250–268

Keen EM (1991) The use of term position devices in ranked output experiments. Journal of Documentation 47(1):1–22

Keen EM (1992) Term position ranking: Some new test results. In: 15th ACM Conference on Research and Development in Information Retrieval (SIGIR'92), Copenhagen, Denmark, pp 66–76

Kekäläinen J, Järvelin K (2002) Using graded relevance assessments in ir evaluation. Journal of the American Society for Information Science and Technology 53:1120–1129

Kise K, Junker M, Dengel A, Matsumoto K (2001) Experimental evaluation of passage-based document retrieval. In: 6th Conference on Document Analysis and Recognition (ICDAR'01), Seattle, WA, USA, pp 592–596

de Kretser O, Moffat A (1999) Effective document presentation with a locality-based similarity heuristic. In: 22nd ACM Conference on Research and Development in Information Retrieval (SIGIR'99), Berkeley, CA, USA, pp 113–120

Lalmas M (2009) Structure weight. In: Encyclopedia of Database Systems, Springer US, p 2862

Lalmas M, Baeza-Yates RA (2009) Structured document retrieval. In: Encyclopedia of Database Systems, Springer US, pp 2867–2868

Lalmas M, Trotman A (2009) XML retrieval. In: Encyclopedia of Database Systems, Springer US, pp 3616–3621

Lalmas M, Kazai G, Kamps J, Pehcevski J, Piwowarski B, Robertson S (2007) INEX 2006 evaluation measures. In: Comparative Evaluation of XML Information Retrieval Systems, 5th Workshop of the INititative for the Evaluation of XML Retrieval (INEX'06), pp 20–34

Lu W, Robertson SE, MacFarlane A (2006) Field-Weighted XML Retrieval Based on BM25. In: Advances in XML Information Retrieval and Evaluation, 4th Workshop of the INititative for the Evaluation of XML Retrieval (INEX'05), Dagstuhl Castle, Germany, pp 161–171

Luhn HP (1958) The automatic creation of literature abstracts. IBM Journal of Research and Development 2:157–165

Malik S, Kazai G, Lalmas M, Fuhr N (2006) Overview of INEX 2005. In: Advances in XML Information Retrieval and Evaluation, 4th Workshop of the INititative for the Evaluation of XML Retrieval (INEX'05), Schloss Dagstuhl, Germany, pp 1–15

Metzler D, Croft WB (2005) A markov random field model for term dependencies. In: 28th ACM Conference on Research and Development in Information Retrieval (SIGIR'05), Salvador, Brazil, pp 472–479

Mishne G, de Rijke M (2005) Boosting web retrieval through query operations. In: 27th European Conference on IR Research (ECIR'05), Santiago de Compostela, Spain, pp 502–516

O'Keefe R, Trotman A (2004) The simplest query language that could possibly work. In: 2nd Workshop of the INititative for the Evaluation of XML Retrieval (INEX'03), Schloss Dagstuhl, Germany, pp 167–174

Rapela J (2001) Automatically combining ranking heuristics for HTML documents. In: 3rd Workshop on Web Information and Data Management (WIDM'01), Atlanta, Georgia, USA, pp 61–67

Rasolofo Y, Savoy J (2003) Term proximity scoring for keyword-based retrieval systems. In: 25th European Conference on IR Research (ECIR'03), Pisa, Italy, pp 207–218

Robertson S, Zaragoza H, Taylor M (2004) Simple BM25 extension to multiple weighted fields. In: 13th ACM Conference on Information and Knowledge Management (CIKM'04), Washington, D.C., USA, pp 42–49

Schenkel R, Suchanek FM, Kasneci G (2007) YAWN: A semantically annotated Wikipedia XML corpus. In: GI-Fachtagung fur Datenbanksysteme in Business, Technologie und Web (BTW'07), 12, pp 277–291

Song R, Taylor MJ, Wen JR, Hon HW, Yu Y (2008) Viewing term proximity from a different perspective. In: 30th European Conference on IR Research (ECIR'08), Glasgow, UK, pp 346–357

Sun YHK, Kim S, hong Eom J, Zhang BT (2000) SCAI experiments on TREC-9. In: 9th Text REtrieval Conference (TREC'00), Gaithersburg, MD, USA, pp 392–399

Svore KM, Kanani PH, Khan N (2010) How good is a span of terms?: exploiting proximity to improve web retrieval. In: 33rd ACM Conference on Research and Development in Information Retrieval (SIGIR'10), Geneva, Switzerland, pp 154–161

Tajima K, Hatano K, Matsukura T, Sano R, Tanaka K (1999) Discovery and retrieval of logical information units in web. In: ACM Digital Library Workshop on Organizing Web Space (WOWS'99), Berkeley, CA, USA, pp 13–23

Tao T, Zhai C (2007) An exploration of proximity measures in information retrieval. In: 30th ACM Conference on Research and Development in Information Retrieval (SIGIR'07), Amsterdam, The Netherlands, pp 295–302

Trotman A (2005) Choosing document structure weights. Information Processing and Management 41(2):243–264

Trotman A, Geva S, Kamps J (2007) Report on the SIGIR 2007 workshop on focused retrieval. SIGIR Forum 41:97–103

Wilkinson R (1994) Effective retrieval of structured documents. In: 17th ACM Conference on Research and Development in Information Retrieval (SIGIR'94), New York, NY, USA, pp 311–317

Wolff JE, Flörke H, Cremers AB (2000) Searching and browsing collections of structural information. In: IEEE Advances in Digital Libraries (ADL'00), Washington, DC, USA, pp 141–150