Методы поиска в структурированных файлах

функции ранжирования

Мытрова Марина, 425 гр.

Содержание

- Векторная модель
- TF-IDF
- Косинусная мера
- Структурированный файл на примере XML
  - Лексические поддеревья
  - Структурные термы
  - Расширение векторной модели на случай структурированных файлов
  - Схожесть контекстов
- Okapi BM25
  - BM25F
  - BM25E

Векторная модель

- **Векторная модель (англ. vector space model)** — представление коллекции документов векторами из одного общего для всей коллекции векторного пространства.

- **Коллекция** - неупорядоченное множество документов.

- **Документ** - неупорядоченное множество термов.

- **Термы (словарные термы)** - слова, из которых состоит текст (определение терма зависит от приложения)

- В векторной модели **термы – это измерения**.
  Вес терма – координата в данном измерении.

Векторная модель

- Более формально

$$\mathbf{d_j = (w_{1j}, w_{2j}, \ldots, w_{nj})}, \text{ где}$$

$\mathbf{d_j}$ — векторное представление j-го документа,
$\mathbf{w_{ij}}$ — вес i-го терма в j-м документе,
n — общее количество различных термов во всех документах коллекции.

- Запросы представляются в той же форме, что и документы. Т.е.

$$\mathbf{q = (w_{1q}, w_{2q}, \ldots, w_{tq})}, \text{ где}$$

q – векторное представление запроса,
$\mathbf{w_{iq}}$ - вес i-го терма в запросе

TF-IDF

- **TF-IDF (от англ. TF — term frequency, IDF — inverse document frequency)** — статистическая мера, используемая для оценки важности слова в контексте документа, являющегося частью коллекции документов. **TF-IDF = TF*IDF**

$$\mathrm{TF} = \frac{n_i}{\sum_k n_k}$$

n_i - число вхождений терма в документ

k – общее число термов в документе

$$\mathrm{IDF} = \log \frac{|D|}{|(d_i \supset t_i)|}$$

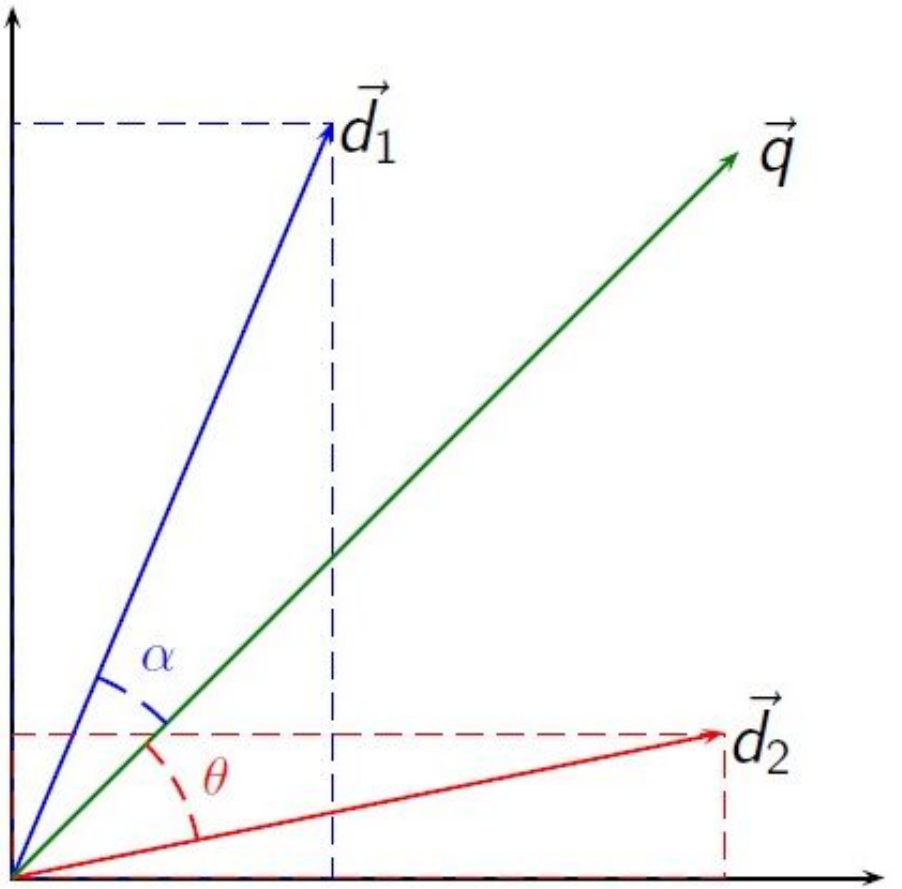
|D| — количество документов в коллекции

$|(d_i \supset t_i)|$ — количество документов, в которых встречается терм t_i (когда $n_i \neq 0$)

Косинусная мера

$$\cos\theta = \frac{\mathbf{d_2} \cdot \mathbf{q}}{\|\mathbf{d_2}\| \, \|\mathbf{q}\|}$$

$$\|\mathbf{v}\| = \sqrt{\sum_{i=1}^{n} v_i^2}$$



Косинусная мера

$$\rho(Q,D) = \frac{\sum_{ti \in Q \cap D} w_Q(t_i) * w_D(t_i)}{\|Q\| * \|D\|}$$

ρ(Q,D) – соответствие запроса Q документу D

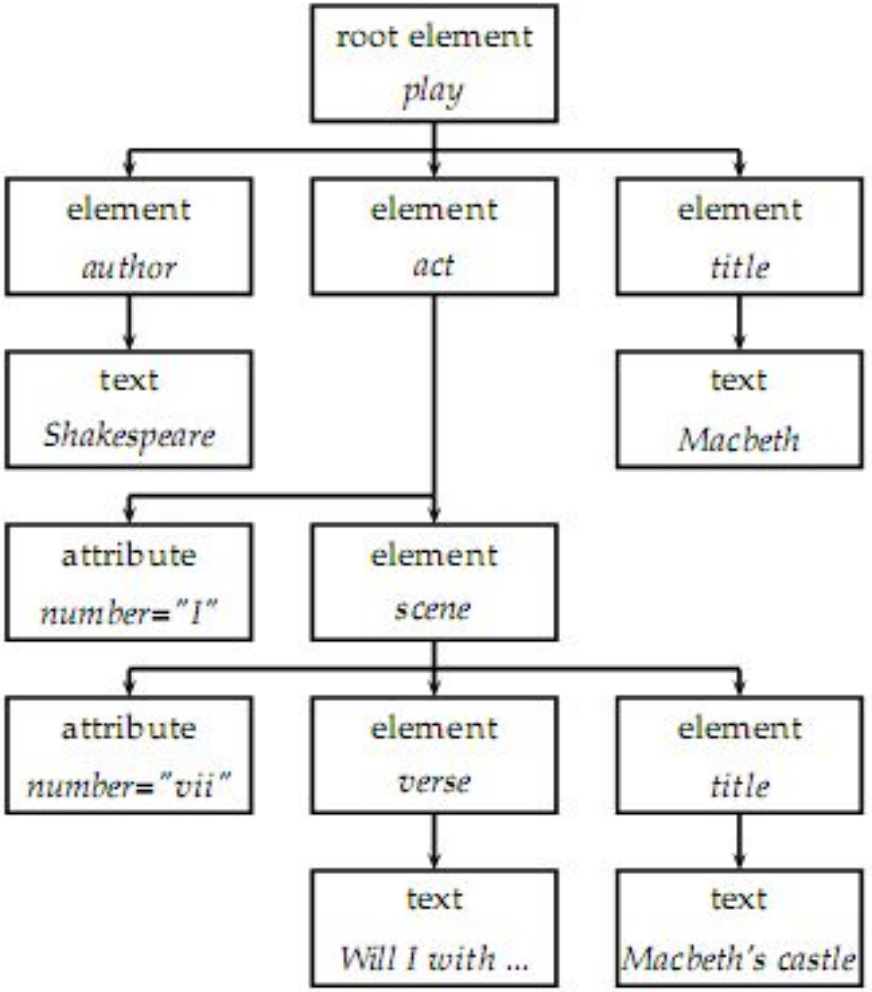
t_i – терм (измерение)

$w_Q(t_i)$ – вес терма t_i в запросе Q

$w_D(t_i)$ – вес терма t_i в документе D

Структурированный файл на примере XML*

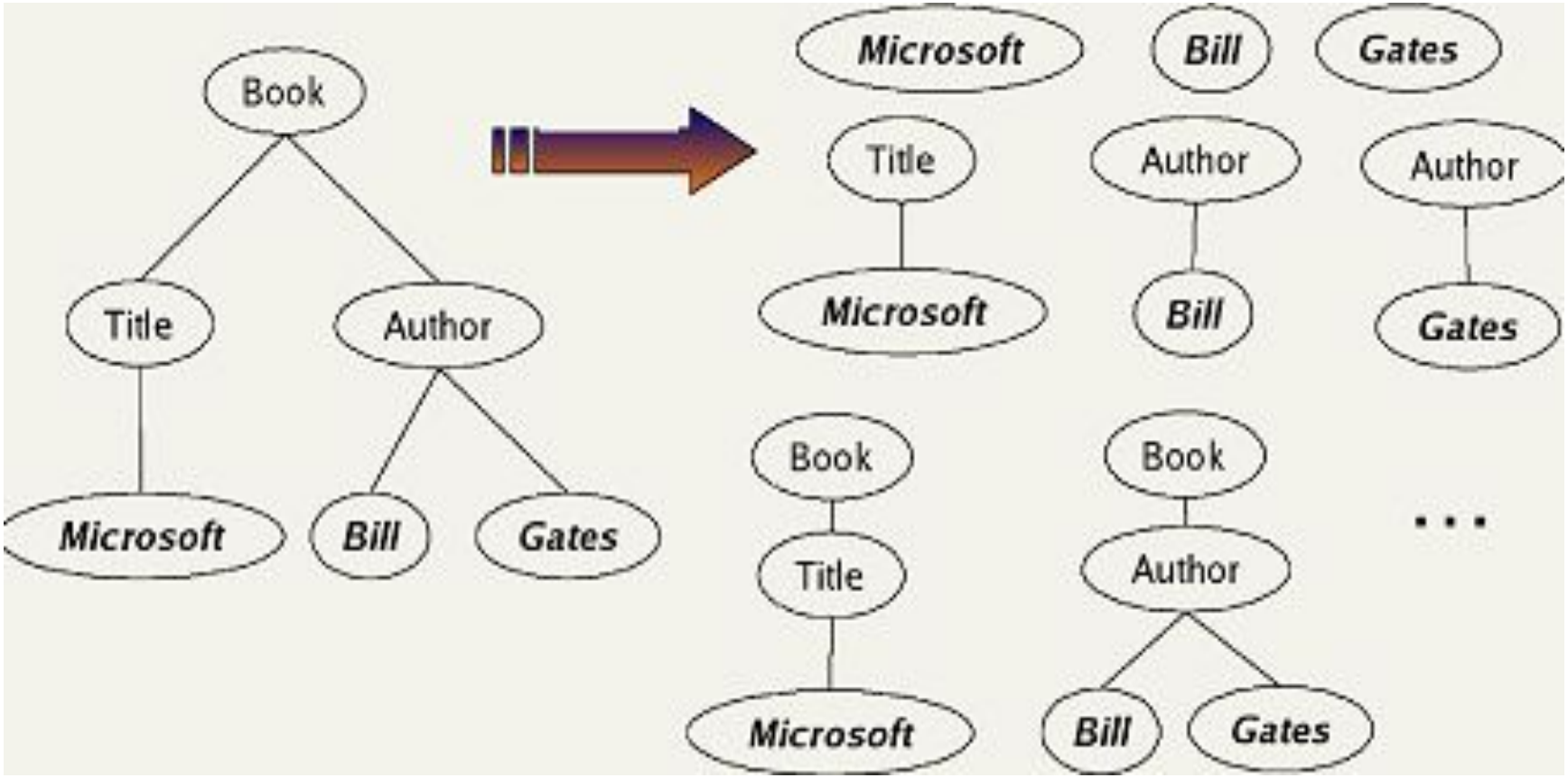
```
<play>
<author>Shakespeare</author>
<title>Macbeth</title>
<act number="I">
<scene number="vii">
<title>Macbeth's castle</title>
<verse>Will I with wine and
    wassail ...</verse>
</scene>
</act>
</play>
```



* Здесь и далее под структурированным файлом подразумевается XML-файл

Лексические поддеревья

- Деревья, содержащие хотя бы один словарный терм



Лексические поддеревья

С увеличением количества узлов в дереве растет число лексических поддеревьев.

Структурные термы

- Будем рассматривать только такие лексические поддеревья, которые оканчиваются единственным словарным термом
- Такие поддеревья называются **структурными термами** и обозначаются парой **(t,c)**, где **t** – это терм, **c** - его XML-контекст.

Расширение векторной модели на случай структурированных файлов

$$\rho(Q,D) = \frac{\sum_{(ti,ci)\in Q}\sum_{(ti,ck)\in D} w_Q(t_i,c_i) * w_D(t_i,c_k) * cr(c_i,c_k)}{\|Q\| * \|D\|}$$

ρ(Q,D) – соответствие запроса Q документу D

(t_i,c_i) – структурный терм (измерение)

$w_Q(t_i,c_i)$ – вес структурного терма (t_i,c_i) в запросе Q

$w_D(t_i,c_i)$ – вес структурного терма (t_i,c_i) в документе D

$cr(c_i,c_k)$ – схожесть контекстов (context resemblance) c_i и c_k, **$0 \le cr(c_i,c_k) \le 1$**

Схожесть контекстов

- 1 способ

$$\mathrm{CR}(c_q, c_d) = \begin{cases} \frac{1+|c_q|}{1+|c_d|} & \text{если } c_q \text{ соответствует } c_d \\ 0 & \text{иначе} \end{cases}$$

$|c_q|$ - число узлов в контексте, соответствующем терму из запроса

$|c_d|$ - то же, но для документа

Схожесть контекстов

- 2 способ

  Рассмотрим запрос в форме **<q1><q2><q3>T</q3></q2></q1>**

  Q = q1q2q3 – контекст появления T в запросе
  A = a1a2…a8 – контекст появления T в произвольном XML документе

  Пример:
  Q = language/book/title
  A = language/media/book/chapter/section/subsection/title/number

a1	a2	a3	a4	a5	a6	a7	a8
language	media	book	chapter	section	subsection	title	number
q_1		q_2				q_3	
language		book				title	

Схожесть контекстов

- **LCS(Q,A)**

  Longest Common Subsequence

  LCS(Q,A) = lcs(Q,A)/|Q|, где

  lcs(Q,A) – длина наибольшей общей подпоследовательности Q и A

  0 ≤ LCS(Q,A) ≤ 1

- Критерии оценки

1. Контекст A включает больше элементов qi в правильном порядке. (В примере - 3)

2. Элементы qi появляются ближе к началу A, чем к концу. (В примере – совпадение q1q2q3 с a1a3a7 предпочтительнее, чем с a1a3a8)

3. Элементы qi появляются в A ближе друг к другу. (В примере – совпадение q1q2q3 с a2a3a4 предпочтительнее, чем с a1a3a5)

4. Из двух контекстов документа, одинаково совпадающих с контекстом запроса, выше оценивается тот, который имеет меньшую длину.

Схожесть контекстов

- **POS(Q,A)**

  POS(Q,A) = 1-((AP-AverOptimalPosition)/(|A|-2*AverOptimalPosition+1))

  AverOptimalPosition - среднее положение оптимального совпадения Q и A (если совпадение начинается с первого элемента и продолжается без пробелов)

  AP - фактическое среднее положение совпадения Q и A

  0 ≤ POS(Q,A) ≤ 1
  (0 – в случае полного несовпадения, 1 – в случае «самого левого» совпадения)

Схожесть контекстов

- **GAPS(Q,A)**

$$GAPS(Q,A) = gaps/(gaps + lcs(Q,A))$$

gaps - число «пробелов» (в примере gaps = 4)

0 ≤ GAPS ≤ 1
(0 – полное совпадение)

Схожесть контекстов

- **LD(Q,A)**

$$LD(Q,A) = (|A| - lcs(Q,A))/|A|$$

$0 \leq LD \leq 1$
(0 – полное совпадение)

Схожесть контекстов

$$\mathbf{cr(Q,A) = \alpha LCS(Q,A) + \beta POS(Q,A) - \gamma GAPS(Q,A) - \delta LD(Q,A)}$$

$0 \le \alpha \le 1,\ 0 \le \beta \le 1,\ 0 \le \gamma \le 1,\ 0 \le \delta \le 1$

$\alpha + \beta = 1$ (т.к. $cr(Q,A) = 1$ в случае полного совпадения)

Примеры

Показывают, как влияют оценки **LCS(Q,A) , POS(Q,A), GAPS(Q,A), LD(Q,A)** на **cr(Q,A)**

Q = q1q2q3 = book/chapter/title

Положим **α = 0.75, β = 0.25, γ = 0.25, δ = 0.2**

Для простоты будем рассматривать lcs(Q,A) вместо LCS(Q,A),
AP вместо POS, gaps вместо GAPS, ld вместо LD

Пример A1. Влияние lcs(Q,A) на cr(Q,A)

A	lcs	AP	gaps	ld	cr
media/**book/chapter/title**/number	3	3	0	2	0.84
media/**chapter/ book/title**/number	2	3	0	3	0.53
media/**title/chapter/book**/number	1	2	0	4	0.29
magazine/volume/article/**title**/number	1	4	0	4	0.19

Пример A2. Влияние AP(Q,A) на cr(Q,A)

A	lcs	AP	gaps	ld	cr
book/chapter/title/subtitle/number	3	2	0	2	0.92
media/**book/chapter/title/** number	3	3	0	2	0.84
media/catalog/ **book/chapter/title**	3	4	0	2	0.75

Пример A3. Влияние gaps(Q,A) на cr(Q,A)

A	lcs	AP	gaps	ld	cr
media/catalog/ **book/chapter/title**/subtitle/number	3	4	0	4	0.78
catalog/**book**/chapters/**chapter**/section/ **title**/number	3	4	2	4	0.68

Пример A4. Влияние ld(Q,A) на cr(Q,A)

A	lcs	AP	gaps	ld	cr
book/chapter/title/subtitle/subtitle/ number/bullet	3	2	0	4	0.88
book/chapter/title/subtitle	3	2	0	1	0.95

Пример B1. Влияние AP(Q,A) на cr(Q,A) при меньшем lcs (Q,A)

A	lcs	AP	gaps	ld	cr
book/section/**title**/subtitle/number	2	2	1	3	0.51
media/**book**/section/**title**/number	2	3	1	3	0.45
media/catalog, **book/s**ection/**title**	2	4	1	3	0.39

Okapi BM25

$$W(\overline{d}, q, c) = \sum_j w_j(\overline{d}, C) \cdot q_j$$

d - документ

C – коллекция документов

W(d,q,C) – релевантность документа d из коллекции C запросу q

$\mathbf{w_j(d,C)}$ – вес j-го терма в документе d коллекции C

$\mathbf{q_j}$ – совпадание терма j из документа с термом запроса

Okapi BM25

$$w_j(\overline{d},C) = \frac{(k_1+1)tf_j}{k_1((1-b)+b\frac{dl}{avdl})+tf_j}\log\frac{N-df_j+0.5}{df_j+0.5}$$

d - документ

C – коллекция документов

w_j(d,C) – вес j-го терма в документе d коллекции C

tf_j – частота j-го терма в документе d коллекции C (TF)

df_j – количество документов коллекции, содержащих j-й терм

dl – длина документа

avdl – средняя длина документов в коллекции

k_1, b – коэффициенты (обычно k_1 = 2, b = 0.75)

BM25F

- модификация BM25, в которой документ рассматривается как совокупность нескольких полей (таких как, например, заголовки, основной текст, ссылочный текст), длины которых независимо нормализуются, и каждому из которых может быть назначена своя степень значимости в итоговой функции ранжирования.

$$wf_j(\bar{d}, C) = \frac{(k'_1 + 1)tf'_j}{k'_1((1-b) + b\frac{dl'}{avdl'} + tf'_j} \log \frac{N - df_j + 0.5}{df_j - 0.5}$$

tf'$_j$ – взвешенная частота j-го терма в документе d

dl' – взвешенная длина документа

avdl' – взвешенная средняя длина документа

k'$_1$ – взвешенный параметр

BM25F

Пусть имеется nF полей **f = 1, …, nF**

В данном поле f документа d терм t имеет частоту $\mathbf{tf_{d,t,f}}$

Пусть **V** – это словарь (набор термов). Тогда

Длина поля f в документе d

$$dl_f = \sum_{t \in v} tf_{d,t,f}$$

Частота терма t в документе d

$$tf_{d,t} = \sum_{f} tf_{d,t,f}$$

BM25F

Пусть имеется nF полей **f = 1, …, nF**

В данном поле f документа d терм t имеет частоту $\mathbf{tf_{d,t,f}}$

Пусть **V** – это словарь (набор термов). Тогда

Длина документа d

$$dl = \sum_f dl_f = \sum_f \sum_t tf_{d,t,f} = \sum_t tf_{d,t}$$

Средняя длина документа

$$avdl = \frac{1}{N} \sum dl$$

BM25F

Если считать, что полю f присвоен вес $\mathbf{w_f}$, получим:

$$tf'_{d,t} = \sum_f w_f tf_{d,t,f}$$

$$dl' = \sum_f w_f dl_f = \sum_f \sum_t w_f tf_{d,t,f} = \sum_t tf'_{d,t}$$

$$avdl' = \frac{1}{N} \sum dl'$$

$$k'_1 = k_1 \frac{atf_{weighted}}{atf_{unweighted}} = k_1 \frac{avdl'}{avdl}$$

N – мощность коллекции

atf – средняя частота терма

BM25E

- В BM25F вместо частоты терма в документе используется линейная комбинация взвешенных частот терма в полях
- Этот метод можно применить к поиску элементов.
- Элементы можно обрабатывать так же, как и документы. Но каждый элемент может иметь ещё и дополнительные, унаследованные поля

BM25E

Пусть имеется nE элементов **e = 1, …, nE** в коллекции **C**

В элементе **e** терм **t** имеет частоту $\mathbf{tf_{d,t,e}}$

el – длина элемента

avel – средняя длина элемента

Тогда расширение BM25 на случай поиска элементов:

$$w_j(e,\bar{d},C) = \frac{(k_1+1)tf_{e,j}}{k_1((1-b)+b\frac{el}{avel})+tf_{e,j}} \log\frac{N-df_j+0.5}{df_j+0.5}$$

BM25E

Соответственно, функция BM25E:

$$wf_j(e,\bar{d},C) = \frac{(k'_1+1)tf'_{e,j}}{k'_1((1-b)+b\frac{el'}{avel'})+tf'_{e,j}} \log\frac{N-df_j+0.5}{df_j+0.5}$$

tf'$_{e,j}$ – взвешенная частота j-го терма в элементе e

el' – взвешенная длина элемента

avel' – взвешенная средняя длина элемента в коллекции

k'$_1$ – взвешенный параметр

BM25E

Соответственно,

$$tf'_{f,t} = \sum_{f \in e} w_f tf_{f,t}$$

$$k'_1 = k_1 \frac{atf_{weighted}}{atf_{unweighted}} = k_1 \frac{avel'}{avel}$$

$$el' = \sum_{f \in e} w_f el = \sum_{f \in e} \sum_{t} w_f tf_{f,t} = \sum_{f,t} tf'_{f,t}$$

$$k'_1 = k_1 \frac{atf_{weighted}}{atf_{unweighted}} = k_1 \frac{avel'}{avel}$$

M – мощность коллекции

atf – средняя частота терма

Литература

- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press. 2008.

- David Carmel, Nadav Efraty, Gad M. Landau, Yoelle S. Maarek, Yosi Mass, An Extension of the Vector Space Model for Querying XML Documents via XML Fragments, ACM SIGIR'2002 Workshop on XML and IR, Tampere, Finland , Aug 2002

- Wei Lu, Stephen Robertson, Andrew Macfarlane, Advances in XML Information Retrieval and Evaluation (INEX 2005). LNCS 3977, Springer 2006 (pp 161-171).

Спасибо за внимание!