Social Feature Re-ranking in INEX 2013 Social Book Search Track

Wei-Lun Xiao[1], Shih-Hung Wu[1]*, Liang-Pu Chen[2], Hung-Sheng Chiu[2], and Ren-Dar Yang[2]
[1] Chaoyang University of Technology, Taiwan, R.O.C
{ s9927632, shwu(*Contact author)}@cyut.edu.tw
[2]Institute for Information Industry, Taipei, Taiwan, R.O.C
{eit, bbchiu, rdyang}@iii.org.tw

Abstract. The emerging of social community generates huge amount useful information in various areas. The information is generated in the context of social relation between people and their friends and is helpful to applications in the context. In the social book search task, we integrate the social feature into the traditional information retrieval technology to give better recommendation on books. We submitted four runs in the INEX 2013 Social book search track, the paper reports the results and discussions.

1 Introduction

The emerging of social community generates huge amount useful information in various areas. The information is generated in the context of social relation between people and their friends and is helpful to applications in the context. In the book search application, the result of traditional information retrieval technology is not enough for the users who need more personal recommendation. Recommendation from friends are more appealing, it might contain more personal feeling and cover more subtle reasons that traditional information retrieval system cannot cover. To combine the two information source of book recommendation, we integrate the social feature into the traditional information retrieval technology to give better recommendation on books. In this task, user-generated metadata is used as the social feature.

The structure of this paper is as follows. Section 2 is the data set description, section 3 shows our architecture and some details, section 4 is the experiment results, and final section gives conclusions.

2 Dataset

2.1 Collection

The document collection in this task is provided by the INEX 2013 social book search track. The documents are in XML format, about 2.8 million books, and the size is 24GB. These documents were collected from Amazon.com and LibraryThing. Table 1 lists all the XML tag used in Social Book Search Track[1].

Table 1.All the XML tag [1]

tag name			
book	similarproducts	title	imagecategory
dimensions	tags	edition	name
reviews	isbn	dewey	role
editorialreviews	ean	creator	blurber
images	binding	review	dedication
creators	label	rating	epigraph
blurbers	listprice	authorid	firstwordsitem
dedications	manufacturer	totalvotes	lastwordsitem
epigraphs	numberofpages	helpfulvotes	quotation
firstwords	publisher	date	seriesitem
lastwords	height	summary	award
quotations	width	editorialreview	browseNode
series	length	content	character
awards	weight	source	place
browseNodes	readinglevel	image	subject
characters	releasedate	imageCategories	similarproduct
places	publicationdate	url	tag
subjects	studio	data	

2.2 Test Topic

Topic set is also provided by INEX 2013 Social Book Search track, which is collected from LibraryThing. A topic describes the information need of a user. Figure 1 gives an example, the XML tags used are : <topic id>,<query>,<title>,<group>,<member>, and <narrative>.

```
- <topic id="2745">
    <query>classical books for kids</query>
    <title>E. Nesbit &amp;amp; Edward Eager</title>
    <group>British & Irish Children's Fiction</group>
    <member>Ffortiwn</member>
  - <narrative>
      When my daughter was six, we read all the
      <a target="_top" rel="nofollow" href="/author/eageredward">Edward Eager</a>
      books together. As she got older, we read a few
      <a target="_top" rel="nofollow" href="/author/nesbite">E. Nesbit</a>
      titles. Other British/Irish authors we've read include
      <a target="_top" rel="nofollow" href="/author/coopersusan">Susan Cooper</a>
      ,
      <a target="_top" rel="nofollow" href="/author/lewiscs">C.S. Lewis</a>
      and
      <a target="_top" rel="nofollow" href="/author/jrrtolkein">J.R.R. Tolkein</a>
      .
      <br/>
      <br/>
      Can anyone recommend something strongly that might fit more or less into this vein?
    </narrative>
  </topic>
```

Fig. 1. A topic example

3 Method of our system

3.1 System architecture

Figure 2 shows the architecture of our system. The first step is the preprocessing includes stop words filtering and stemming. Our system adopts the stop words filtering and stemming modules provides by Lucene. After the preprocessing, our system builds index for retrieval. The results of content-based retrieval will be re-ranked according to the social feature as the final results.

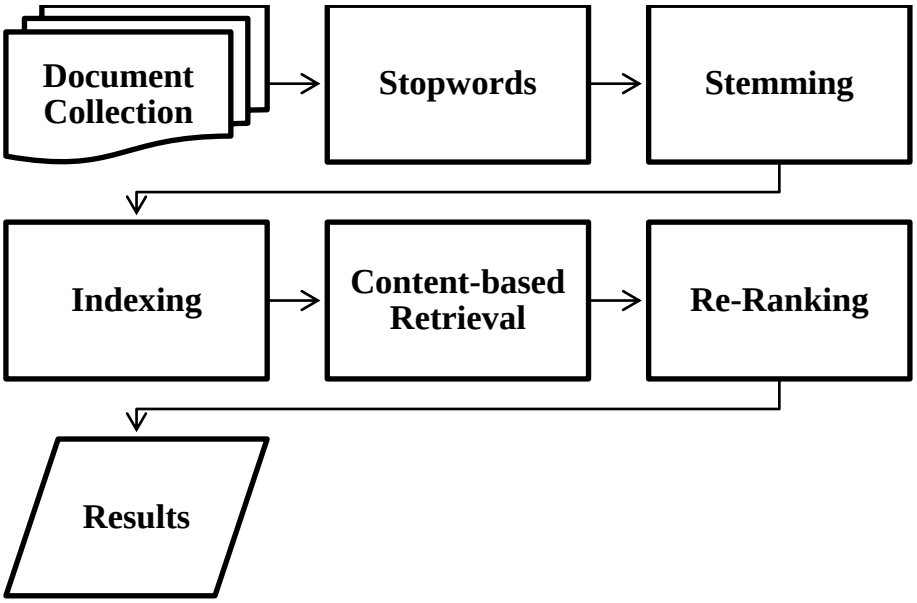

Fig. 2. System architecture

3.2 Indexing

The index and search engine in used is the Lucene system, which is an open source full text search engine provided by Apache software foundation. Lucene is written in JAVA and can be called by JAVA program easily to build various applications [2].

According to (Bogers and Larsen, 2012) [3], 19 tags are more useful in the social book search, they are <isbn>, <title>, <publisher>, <editorial>, <creator>, <series>, <award>, <character>, <place>, <blurber>, <epigraph>, <firstwords>, <lastwords>, <quotation>, <dewey>, <subject>, <browseNode>, <review>, and <tag>. Our system also focused on the 19 tags.

In order to make string matching easier, the content in the <dewey> tag will be restored to strings accordint to the 2003 list of Dewey category descriptions, for example: <dewey>004</dewey> will be restored to <dewey>Data processing Computer science</dewey>. Also, the content of <tag> will be expanded, for example: <tag count="3">fantasy</tag> will be expanded as <tag>fantasy fantasy fantasy</tag>.

The 19 tags were used to build our index file. In additional to the 19 tags, we also index the content of <review> as an independent index

file and named it as reviews.

3.3 Re-ranking

We integrate the user-generated metadata into the traditional content-based search result by re-ranking the results.
The social features are used to give more weight on certain books, for example

- User rating: users might vote a book from 1 to 5, the higher the better.
- Helpful vote: other users might endorse one comment by voting it as helpful.
- Total vote: the total number of helpful or not.

We designed 3 different ways to use these social features in re-ranking.
1) User Rating method

Increase the weight of content-based retrieval result by adding the summation of user rating. As shown in formula (1):

$$\text{Score}_{\text{re-ranked}}(\text{i}) = \alpha * \text{Score}_{\text{org}}(\text{i}) + (1-\alpha) * \text{Score}_{\text{user rating}}(\text{i}) \quad (1)$$

2) Average User Rating method
Increase the weight of content-based retrieval result by adding the average of user rating. As shown in formula (2):

$$\text{Score}_{\text{re-ranked}}(\text{i}) = \text{Score}_{\text{org}}(\text{i}) + \text{Score}_{\text{average user rating}}(\text{i}) \quad (2)$$

3) Weights User Rating method

Increase the weight of content-based retrieval result by adding the book which gets more helpful votes. As shown in formula (3) and (4):

$$\text{Score}_{\text{Weights User Rating}} = \text{User rating} * \frac{\text{helpfulvote}}{\text{totalvote}} \quad (3)$$

$$\text{Score}_{\text{re-ranked}}(\text{i}) = \alpha * \text{Score}_{\text{org}}(\text{i}) + (1-\alpha) * \text{Score}_{\text{Weights User Rating}}(\text{i}) \quad (4)$$

4 Experimental results

In our experiment, the content of <query> tag is used as the query. The α is set as 0.9. We sent 6 runs; the results are shown in Table 2. The setting of each run is as follows.

Run1.query.content-base
Search the index file build from 19 tags with content-based search.

Run2.query.Rating
Search the index file build from 19 tags with content-based search and User Rating re-ranking.

Run3.query.RA
Search the index file build from 19 tags with content-based search and Average User Rating re-ranking.

Run4.query.RW
Search the index file build from 19 tags with content-based search and Weights User Rating re-ranking.

Run5.query.reviwes.content-base
Search the index file build from the review tag with content-based search.

Run6.query.reviews.RW
Search the index file build from the review tag with content-based search and Weights User Rating re-ranking.

Table 2. Experiiment results

Run	*nDCG@10*	*P@10*	*MRR*	*MAP*
Run1.query.content-base	0.0265	0.0147	0.0418	0.0153
Run2.query.Rating	0.0376	0.0284	0.0792	0.0178
Run3.query.RA	0.0170	0.0087	0.0352	0.0107
Run4.query.RW	0.0392	0.0287	0.0796	0.0201
Run5.query.reviwes.content-base	0.0254	0.0153	0.0359	0.0137
Run6.query.reviews.RW	0.0378	0.0284	0.0772	0.0165

5 Conclusions

This paper reports our system and result in INEX 2013 Social Book Search track. We sent 6 runs and the results are list in Table 2. In the six runs, run4 give best nDCG@10. Run4 is searching with content-based search and re-ranking with weights user rating, which shows that helpful review is more useful than average user rating. In the future, we will

expand the query with the content in more tags. In our experiment, α=0.9 is a tentative trial, more experiment will be necessary to get the best parameter.

6 Acknowledgement

This study is conducted under the "Digital Convergence Service Open Platform" of the Institute for Information Industry which is subsidized by the Ministry of Economy Affairs of the Republic of China.

References

1. Marijn Koolen, Gabriella Kazai, Jaap Kamps, Michael Preminger, Antoine Doucet, and Monica Landoni, Overview of the INEX 2012 Social Book Search Track, INEX'12 Workshop Pre-proceedings,P.77-P.96,2012.
2. Lucene.http://zh.wikipedia.org/wiki/Lucene
3. Toine Bogers and Birger Larsen. RSLIS at INEX 2012: Social Book Search Track, Overview of the INEX 2012 Social Book Search Track, INEX'12 Workshop Pre-proceedings,P.97-P.108,2012.
4. Kazai, G., Koolen, M., Kamps, J., Doucet, A., Landoni, M.: Overview of the INEX 2011Book and Social Search Track. In: INEX 2011 Workshop pre-proceedings. INEX WorkingNotes Series (2011) 11–36
5. Ludovic Bonnefoy, Romain Deveaud and Patrice Bellot, Do Social Information Help Book Search?, INEX'12 Workshop Pre-proceedings,P.109-P.113,2012.