Query Type Recognition and Result Filtering in INEX 2014 Social Book Search Track

Shih-Hung Wu[1]*, Pei-Kai Liao[1], Hua-Wei Lin[1], Li-Jen Hsu[1], Wei-Lun Xiao[1], Liang-Pu Chen[2], Tsun Ku[3], and Gwo-Dong Chen[3]

[1] Chaoyang University of Technology, Taiwan, R.O.C
{ shwu(*Contact author), s10027024, s10027072, s10027042}@cyut.edu.tw
[2]Institute for Information Industry, Taiwan, R.O.C
eit@iii.org.tw,
[3]National Central University, Taiwan, R.O.C
cujing@gmail, chen@csie.ncu.edu.tw

Abstract. The paper reports our system in INEX 2014 Social Book Search (SBS) track. This is the second time that we attend the SBS track. Based on our social feature re-ranking system [1], we improve our system by involving some knowledge on understanding the queries. Our baseline system is built on Lucene [6], an open source information retrieval system. The new modification is a set of rules that can filter out unnecessary books from the recommendation list. The official run results show that the system performance is much improved than the 2013 system.

Keywords: Query type recognition, social features, social book search

1 Introduction

The paper reports how we build a system to attend the INEX 2014 Social Book Search (SBS) track [10]. This is the second time that we attend the SBS track [7]. Based on our social feature re-ranking system [1], we improve our system by involving some knowledge on understanding the queries.

In the book search application, we believe that the result of traditional information retrieval technology is not enough for the users who need more personal recommendation. Recommendation from experienced users are more appealing; it might contain more personal feelings and cover more subtle reasons that traditional information retrieval system cannot cover. Our system integrates 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.

According to our observation on the topics in INEX 2012 SBS Track, we find that there are some queries that are different from others. Simply treat the keywords in the topic as search terms will not get good result. Some of them require higher level of knowledge to deal with. System needs to understand the information need behind the keyword, i.e. the knowledge on the types of literature. We analysis the topics and find several types in them. Due to the time limitation, we only implement a module to

recognize one special type of topics and a filtering module to modify the recommendation result.

The structure of this paper is as follows. Section 2 is the data set description, section 3 shows our architecture and the details of our method, 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 2014 social book search track. The documents are in XML format, about 2.8 million books, and the size is 25.9GB. These documents are collected from Amazon.com and LibraryThing. [2]

Table 1.All the XML tag [2]

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

Topics provided by INEX 2014 Social Book Search track are collected from LibraryThing. A topic describes the information needed for a user. Figure 1 and Figure 2 give partial view of an example, the XML tags used are：<topic id>, <title>, <mediated_query>, <group>, <narrative>, <catalog>, <book>, <LT_id>, <entry_date>, and <rating>.

<topics>
<topic id="1116">
<title>Which LISP?</title>
<mediated_query>introduction book to Lisp</mediated_query>
<group>Purely Programmers</group>
<narrative> It'll be time for me to shake things up and learn a new language soon. I had started on Erlang a while back and getting back to it might be fun. But I'm starting to lean toward Lisp--probably Common Lisp rather than Scheme. Anyone care to recommend a good first Lisp book? Would I be crazy to hope that there's one out there with an emphasis on using Lisp in a web development and/or system administration context? Not that I'm unhappy with PHP and Perl, but the best way for me to find the time to learn a new language is to use it for my work... </narrative>
</topic>

Fig. 1. A topic example in INEX 2014 social book search track

<entry_date>2005-10</entry_date>
<rating>0.0</rating>
<tags>audio, language, japanese</tags>
</book>
<book>
<LT_id>28112</LT_id>
<entry_date>2005-10</entry_date>
<rating>10.0</rating>
<tags>community, systems, reference, technology</tags>
</book>
<book>
<LT_id>289716</LT_id>
<entry_date>2005-10</entry_date>
<rating>0.0</rating>
<tags>poetry</tags>
</book>
</catalog>
</topic>

Fig. 2. A topic example in INEX 2014 social book search track

3 CYUT System Methodology

3.1 System Architecture

Figure 3 shows our basic system architecture. The pre-processing includes stop words filtering, and stemming, both modules are provided by Lucene. After the preprocessing, our system builds index for retrieval. The results of content-based retrieval will be re-ranked as the final results according to the social features.

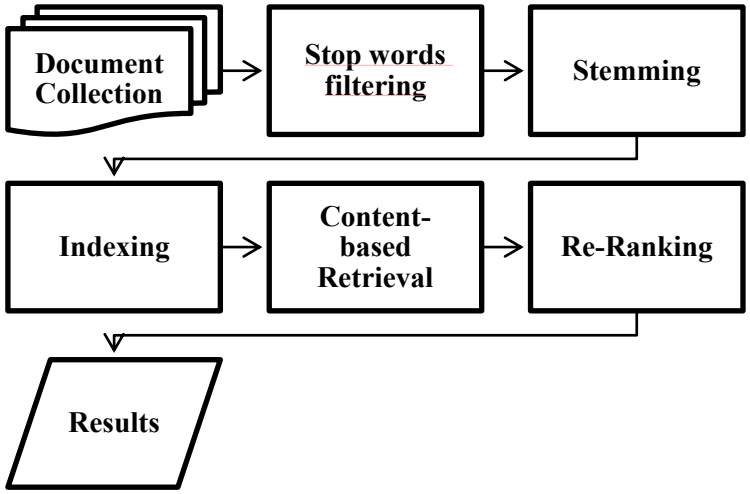

Fig. 3. Basic system architecture [1]

3.2 Indexing and Query

The index and search engine in use is the Lucene system [6], which is an open source full text search engine provided by Apache software foundation. Lucene is written in JAVA and can be called easily by JAVA program to build various applications.

According to Bogers and Larsen [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.

The content in the <dewey> tag is restored to strings accordint to the 2003 list of Dewey category descriptions [9] to make string matching easier. For example: <dewey>004</dewey> will be restored to <dewey>Data processing Computer science</dewey>. The content of <tag> is also expanded according to the count number to emphasize its importance. For example: <tag count="3">fantasy</tag> will be expanded as <tag>fantasy fantasy fantasy</tag>. In additional to the 19 tags, our system also indexes the content of <review> as independent indexes files and names it as

reviews.

According to Koolen et al. [4], an Indri [5] based system using all the contents of <Title>, <Query>, <Group>, and <Narrative> as query terms will give better result.

```
- <topic id="4875">
    <title>Dutch colonies & trading posts</title>
    <group>History Readers: Clio's (Pleasure?) Palace</group>
    <user>myshelves</user>
  - <narrative>
      I'm reading
      <a href="/work/161870">The Island at the Center of the World</a>
      . Can anyone recommend a good book that goes into detail about the 17th century Dutch colonies/trading posts
      around the world?
    </narrative>
  - <types>
      <type>subject</type>
    </types>
  - <genres>
      <genre>non-fiction</genre>
    </genres>
</topic>
```

Fig. 4. A type2 query example that we defined in INEX 2013 social book search track

3.3 Type2 Query Recognition and Result Filtering

According to our observation on the topics in INEX 2012 SBS Track, we find that there are some queries that are different from others, we call them the Type2 queries. Type2 queries are the queries that contain the names of some books that the original users want to find similar ones. Therefore, the books in the topics should not be part of the recommendation. Since the book names are given explicitly, our system originally will find exactly the same books as the top recommendation. To filter out these ones, we define a list of phrases to identify such queries and filter out the books in the queries from the recommendation lists. The phrases are listed in the appendix in the rear of the paper. Figure 4 gives an example of Type2 queries taken from INEX 2013 SBS topics, in which contains a key phrase "I'm reading". We find that there are 174 queries in the INEX 2013 SBS track that can be classified as Type2 queries. Therefore, this year, we add a module in our system to identify the Type2 queries and filtering out the books mentioned in the topics. The modified system flow is shown in Figure 5.

3.4 Re-ranking

The Re-ranking part is similar to that in our previous work [1]. 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 evaluate 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.

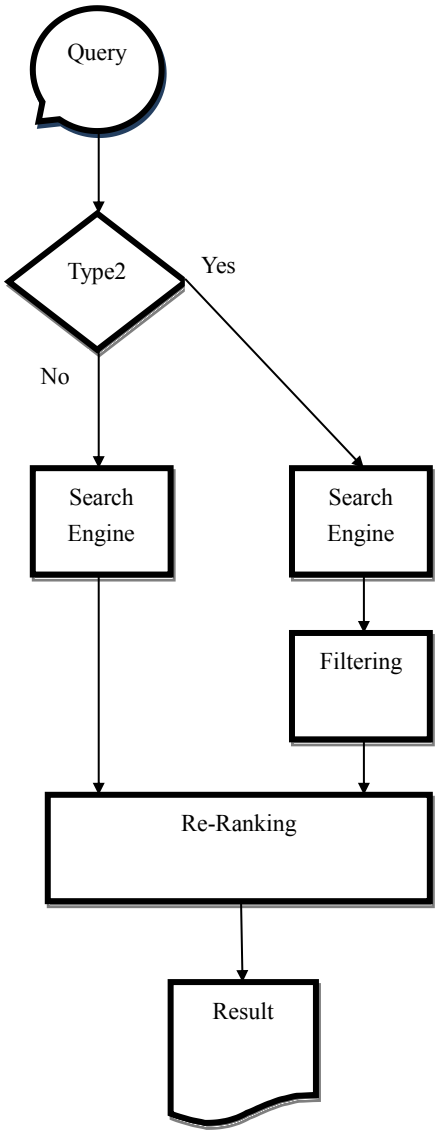

Fig. 5. The modified system flow of our system

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 us-

er 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):

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

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

3.5 Find the Best α Value by Experiment

Since there is no theoretical reference on how to set the α value, in our official runs, the value is selected via a series experiments that we conduct on the 2013 dataset. Table 2 shows the results, we find that the system gets the best result when α is 0.95.

Table 2. Experimental Result for different α on 2013 data set

α	P@10	MAP
0.50	0.0221	0.0193
0.60	0.0221	0.0193
0.70	0.0224	0.0195
0.80	0.0226	0.0196
0.90	0.0237	0.0204
0.95	**0.0245**	**0.0220**

4 Experimental results

In the official evaluation, we sent four runs. This year, we use four fields in the topics as query terms, and we filter out some book candidates for all the type2 queries. The configuration of each run is as follows. Run 1, the CYUT - Type2QTGN: without re-ranking. Run 2, the CYUT - 0.95AverageType2QTGN, re-ranking with Average User Rating. Run 3, the CYUT - 0.95RatingType2QTGN, re-ranking with User Rating. Run 4, CYUT - 0.95WRType2QTGN, Re-ranking with Weights User Rating.

Table 3 shows the official evaluation results of our four runs. Among them the CYUT - Type2QTGN run gives the best NDCG@10 [8] result, while the re-ranking run CYUT - 0.95AverageType2QTGN gives similar result. The other two runs give poor results due to technical errors; the system searches the document in 2013 index file. The last two runs should be better result if the system searches the document in 2014 index file. Comparing to the 2013 INEX SBS results in Table 4, our system performance improved significantly.

Table 3. Official evaluation results in 2014 INEX SBS

Run	*nDCG@10*	*MRR*	*MAP*	*R@1000*
CYUT - Type2QTGN	**0.119**	**0.246**	**0.086**	**0.340**
CYUT - 0.95AverageType2QTGN	**0.119**	0.243	0.085	0.332
CYUT - 0.95RatingType2QTGN	0.034	0.101	0.021	0.200
CYUT - 0.95WRType2QTGN	0.028	0.084	0.018	0.213

Table 4. Official evaluation results in 2013 INEX SBS

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 2014 Social Book Search track. We sent four runs and the results are list in Table 3. In the four runs, the CYUT - Type2QTGN run gives best nDCG@10, which is searching with content-based search and applying a set of filtering rules based on a list of key phrase. In the future, we will implement more modules with literature knowledge on the writers, genre of books, geometric categories of the publishers, and temporal categories of the authors that can deal with the special cases in the topics.

Acknowledgement

This study was conducted under the "Online and Offline Integrated Smart Commerce Platform (1/4)" of the Institute for Information Industry, which is subsidized by the Ministry of Economic Affairs of the Republic of China.

References

1. Wei-Lun Xiao, Shih-Hung Wu, Liang-Pu Chen, Hung-Sheng Chiu, and Ren-Dar Yang, “Social Feature Re-ranking in INEX 2013 Social Book Search Track”, CLEF 2013 Evaluation Labs and Workshop Online Working Notes, 23 - 26 September, Valencia, Spain.
2. 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.
3. Toine Bogers and Birger Larsen, “RSLIS at INEX 2012: Social Book Search Track”,

INEX'12 Workshop Pre-proceedings,P.97-P.108,2012.
4. Marijn Koolen, Hugo Huurdeman and Jaap Kamps, “Comparing Topic Representations for Social Book Search”, CLEF 2013 Evaluation Labs and Workshop Online Working Notes, 23 - 26 September, Valencia – Spain.
5. T. Strohman, D. Metzler, H. Turtle, and W. B. Croft, “Indri: a language-model based search engine for complex queries”, In Proceedings of the International Conference on Intelligent Analysis, 2005.
6. Lucene, https://lucene.apache.org
7. Marijn Koolen, Gabriella Kazai, Michael Preminger, and Antoine Doucet, “Overview of the INEX 2013 Social Book Search Track”, CLEF 2013 Evaluation Labs and Workshop Online Working Notes, 23 - 26 September, Valencia – Spain.
8. Järvelin, K., Kekäläinen, “J.: Cumulated Gain-based Evaluation of IR Techniques”, ACM Transactions on Information Systems 20(4) (2002) 422–446.
9. 2003 list of Dewey category descriptions, https://www.library.illininois.edu/ugl/about/dewey.html
10. INEX 2013 Social Book Search Track, https://inex.mmci.uni-saarland.de/tracks/books

Appendix: The key phrases for recognizing Type2 queries.

```
<TotalKeyWord>
        <keyWord>I've just finished</keyWord>
        <keyWord>I'm now reading</keyWord>
        <keyWord>I'm reading</keyWord>
        <keyWord>I've read</keyWord>
        <keyWord>I read</keyWord>
        <keyWord>I've ever read</keyWord>
        <keyWord>Any book as good as</keyWord>
        <keyWord>I'm not interested</keyWord>
        <keyWord>I already own</keyWord>
        <keyWord>I own</keyWord>
        <keyWord>picked up</keyWord>
        <keyWord>I can find</keyWord>
        <keyWord>I read</keyWord>
        <keyWord>I've looked through</keyWord>
        <keyWord>I've just found</keyWord>
        <keyWord>I have already read</keyWord>
        <keyWord>I was reading</keyWord>
        <keyWord>I had read</keyWord>
        <keyWord>to read</keyWord>
        <keyWord>what other</keyWord>
        <keyWord>I'm already completely</keyWord>
        <keyWord>I have already read</keyWord>
        <keyWord>I've started on</keyWord>
        <keyWord>I just finished</keyWord>
        <keyWord>I did enjoy</keyWord>
        <keyWord>something like</keyWord>
        <keyWord>without</keyWord>
```

<keyWord>I am reading</keyWord>
<keyWord>starting with</keyWord>
<keyWord>I already have</keyWord>
<keyWord>I'm thinking of</keyWord>
<keyWord>I just finished reading</keyWord>
<keyWord>similar</keyWord>
<keyWord>I adore</keyWord>
<keyWord>I tried reading</keyWord>
<keyWord>I also have</keyWord>
<keyWord>I've seen</keyWord>
<keyWord>I recently read</keyWord>
<keyWord>I discovered</keyWord>
<keyWord>I have recently read</keyWord>
<keyWord>have been suggested</keyWord>
<keyWord>has been suggested</keyWord>
<keyWord>I've enjoyed</keyWord>
<keyWord>I've just completed</keyWord>
<keyWord>I haven't yet read</keyWord>
<keyWord>I have only found</keyWord>
<keyWord>I have found</keyWord>
<keyWord>I have read</keyWord>
<keyWord>I am re-reading</keyWord>
<keyWord>I also recently started</keyWord>
<keyWord>I recently started</keyWord>
<keyWord>I just re-read</keyWord>
<keyWord>I've compiled</keyWord>
<keyWord>I'd really like to read</keyWord>
<keyWord>I've already enjoyed</keyWord>
<keyWord>I can think of</keyWord>
<keyWord>I was considering</keyWord>
<keyWord>Currently reading</keyWord>
<keyWord>Apart from</keyWord>
<keyWord>I'm nearly finished</keyWord>
<keyWord>have been recommended</keyWord>
<keyWord>other recommendations</keyWord>
<keyWord>having read</keyWord>
<keyWord>on my list</keyWord>
<keyWord>I've been reading</keyWord>
<keyWord>I have just received</keyWord>
<keyWord>finishing</keyWord>
<keyWord>also read</keyWord>
<keyWord>recent readings</keyWord>
<keyWord>I have been reading</keyWord>
<keyWord>I've recently finished</keyWord>
<keyWord>other books</keyWord>
<keyWord>additional resources</keyWord>
<keyWord>The most recent book I haved</keyWord>

```
        <keyWord>I saw a book</keyWord>
        <keyWord>Thus far I</keyWord>
        <keyWord>what else should I</keyWord>
        <keyWord>Can anyone think of any more</keyWord>
        <keyWord>books like</keyWord>
        <keyWord>something elsee</keyWord>
        <keyWord>I thoroughly enjoyed</keyWord>
        <keyWord>My reading suggestions</keyWord>
</TotalKeyWord>
```