Overview of the SBS 2016 Mining Track

Bogers, Toine; Hendrickx, Iris; Koolen, Marijn; Verberne, Suzan

Published in:
Working Notes of CLEF 2016 - Conference and Labs of the Evaluation Forum

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Overview of the SBS 2016 Mining Track

Toine Bogers[1], Iris Hendrickx[2], Marijn Koolen[3,4], and Suzan Verberne[2]

[1] Aalborg University Copenhagen, Denmark
toine@hum.aau.dk
[2] CLS/CLST, Radboud University, Nijmegen, the Netherlands
(i.hendrickx|s.verberne)@let.ru.nl
[3] University of Amsterdam, the Netherlands
marijn.koolen@uva.nl
[4] Netherlands Institute for Sound and Vision
mkoolen@beeldengeluid.nl

Abstract. In this paper we present an overview of the mining track in the Social Book Search (SBS) lab 2016. The mining track addressed two tasks: (1) classifying forum posts as book search requests, and (2) linking book title mentions in forum posts to unique book IDs in a database. Both tasks are important steps in the process of solving complex search tasks within online reader communities. We prepared two data collections for the classification task: posts from the LibraryThing (LT) forum and a smaller number of posts from Reddit. For the linking task we used annotated LT threads. We found that the classification task was relatively straightforward, achieving up to 94% classification accuracy. The book linking task on the other hand turned out to be a difficult task: here the best system achieved an accuracy of 41% and F-score of 33.5%. Both the automatic classification of book search requests as the automatic linking of book mentions could next year be part of the pipeline for processing complex book searches.

1 Introduction

The Mining track[1] is a new addition to the Social Book Search (SBS) Lab in 2016. For the past five years, the Suggestion Track has explored techniques to deal with complex information needs that go beyond topical relevance and can include other aspects, such as genre, recency, engagement, interestingness, and quality of writing. In addition, it has investigated the value of complex information sources, such as user profiles, personal catalogues, and book descriptions containing both professional metadata and user-generated content.

So far, examples of such complex search tasks have been taken from the LibraryThing (LT) discussion fora. Book search requests were manually separated from other book-related discussion threads by human annotators, and the suggestions provided by other LT users were used as relevance judgments in the automatic evaluation of retrieval algorithms that were applied to the book

[1] See http://social-book-search.humanities.uva.nl/#/mining

search requests. If we wish to move further towards fully supporting complex book search behavior, then we should not just support the retrieval and recommendation stage of the process, but also the *automatic* detection of complex search needs and the analysis of these needs and the books and authors contained therein. This is the goal of the Mining Track.

The SBS 2016 Mining Track focuses on automating two text mining tasks in particular:

1. **Book search request classification**, in which the goal is to identify which threads on online forums are book search requests. That is, given a forum thread, the system should determine whether the opening post contains a request for book suggestions (i.e., binary classification of opening posts)
2. **Book linking**, in which the goal is to recognize book titles in forum posts and link them to the corresponding metadata record through their unique book ID. The task is not to mark each entity mention in the post text, but to label the post as a whole with the IDs of the mentioned books. That is, the system does not have to identify the exact phrase that refers to book, but only has to identify which book is mentioned on a per-post basis.

The suggestions that LT users provide in response to book search requests are often linked to official book metadata records using so-called *Touchstones*. Touchstones offer a wiki-like syntax for linking books (and authors) mentioned in LT threads to their official LT pages (and thereby the books' metadata records). All books mentioned in a thread are shown in a sidebar, so other LT users can see at a glance which books have already been suggested. Or, to quote a LT user:

> *"The main reason I like Touchstones to work is that they allow me to scan the sidebar to see what books have already been discussed in a thread. This is particularly useful in a thread like this (in which somebody is asking for recommendations) because I can take care to mention something new without reading all previous threads (which I won't necessarily do if the thread gets really really long)."*

However, not every book mentioned in LT threads is marked up using Touchstones; previous preliminary work has shown that around 16% of all books are not linked by LT users [3], which has an as-of-yet unknown effect on their use as relevance assessments in the Suggestion Track.

In this paper, we report on the setup and the results of the 2016 Mining Track as part of the SBS Lab at CLEF 2016. First, in Section 2, we give a brief summary of the participating organisations. Section 3 describes the two tasks in the Mining Track in more detail, along with the data used and the evaluation process. Section 4 presents the results of the participating organisations on the two tasks. We close in Section 5 with a summary and plans for 2017.

2 Participating organizations

A total of 28 organisations registered for the Mining Track and 4 organisations ended up submitting a total of 34 runs. The active organisations are listed in Table 1.

3 Mining Track setup and data

In the following sections we describe the data collection and annotation process for both tasks in the 2016 text mining track, as well as the evaluation procedures.

3.1 Task 1: Book search request classification

Data collection For the task of classifying forum threads we created two data sets for training: one based on the LibraryThing (LT) forums and one based on Reddit. For the LT forums, we randomly sampled 4,000 threads and extracted their opening posts. We split them into a training and a test set, each containing 2,000 threads. These threads contained both positive and negative examples of book requests.

The Reddit training data was sampled from three months of Reddit threads collected in September, October, and November 2014. The set of positive book request examples comprises all threads from the suggestmeabook subreddit, whereas the negative examples comprises all threads from the books subreddit. The training set contained 248 threads in total. The Reddit test data was sampled from December 2014 and comprises 89 threads in total. Figure 1 shows an example of the training data format for the classification task.

Annotation The labels of the Reddit training data were not annotated manually, as they were already categorized as positive and negative by virtue of the subreddit they originated from. In the annotation process for the LT threads, positive examples of book requests consisted of all posts where the user described an explicit foreground or background information need and was searching for books to read. Examples include *known-item* requests, where a user is looking for a specific book by describing plot elements, but cannot remember the title;

Table 1. Active participants of the Mining Track of the CLEF 2016 Social Book Search Lab and number of contributed runs or users.

Institute	Acronym	Runs
Aix-Marseille Université CNRS	LSIS	8
Tunis EL Manar University	LIPAH	6
Know-Center	Know	8
Radboud University Nijmegen	RUN	12

Fig. 1. Example of the training data format for the Book search request classification task.

```
<thread id="2nw0um">
  <category>suggestmeabook</category>
  <title>can anyone suggest a modern fantasy series....?</title>
  <posts>
    <post id="2nw0um">
      <author>blackbonbon</author>
      <timestamp>1417392344</timestamp>
      <parentid></parentid>
      <body>
        ....  where the baddy turns good, or a series similar to
        the broken empire trilogy. I thoroughly enjoyed reading it along
        with skullduggery pleasant, the saga of darren shan, the saga of
        lartern crepsley and the inheritance cycle. So whatever you got
        helps :D cheers lads, and lassses.
      </body>
      <upvotes>8</upvotes>
      <downvotes>0</downvotes>
    </post>
    ...
  </posts>
</thread>
```

users asking for books covering a specific topic; and users asking for books that are similar to another book they mention. Posts where users ask for new authors to explore or where they list their favorite books and ask others to do the same were *not* classified as explicit book requests.

The manual annotation of the LT data was performed by the four organizers of the task. To get an impression of the inter-annotator agreement, a small sample of 432 posts was labeled by two annotators. Average agreement according to Cohen's κ was 0.84, averaged over the pairs of annotators, which represents almost perfect agreement according to Cohen [1].

For evaluation, 1,974 out of the 2,000 threads in the LibraryThing test set were used. For the 26 remaining threads, judges were unsure whether the first post was a request or not. The Reddit test set consisted of 89 threads with the subreddit names (books and suggestmeabook) as labels. In order to create a ground truth for the test set, two judges (track organizers) manually classified the 89 test threads. They discussed all disagreements and reached consensus on all 89 threads. 81 of the labels were the same as the original Reddit label; the other 8 were different. We used the manual labels as ground truth. Table 2 shows the proportion of positive and negative examples in the training and test sets of both data sets.

Table 2. Overview of number of positive and negative instances in the training and test sets for the book request classification task.

		LibraryThing	Reddit
Training	Positive	272	43
	Negative	1688	205
Test	Positive	245	13
	Negative	1729	76

3.2 Task 2: Book linking

Data collection Book linking through the use of Touchstones is an striking characteristic of the LT forum, and an important feature for the forum community. A Touchstone is a link created by a forum member between a book mention in a forum post and a unique LT work ID in the LT database. A single post can have zero or more different touchstones linked to it. Touchstones allow readers of a forum thread to quickly see which books are mentioned in the thread.

For the book linking task we created a data set based on the touchstones in the LT forum. The training data consisted of 200 threads with 3619 posts in total. The training data contains only those touchstones that had been added by the LT authors; we did not enrich the posts with more annotations. Figure 2 shows an example of the training data format for the linking task. In the example, Insomnia is the title of a book. The task is to identify the LT work ID of the corresponding book and link it to that specific post ID.

Participants used the Amazon/LT collection for linking the book mentions to a database record. This collection originated with the Suggestion track and contains 2.8 million book metadata records along with their LT work IDs. The test data for the linking task comprised 200 LT threads. As opposed to the training data, we did make the annotations in the test data more complete by manually annotating book mentions and linking them to the book database.

Annotation of test data In the annotation process, we linked books manually at the post level by their unique LT work ID. Many books are published in different editions throughout the years with different unique ISBNs, but all of these versions are connected to the same unique LT work ID. If a book occurred multiple times in the same post, only the first occurrence was linked, so participants only need to specify each of the work IDs found in a post once. If a post mentioned a series of books, we linked this series to the first book in the series, e.g., the "*Harry Potter series*" was linked to "*Harry Potter and the Philosopher's Stone*". In some cases, a book title was mentioned, but no suitable work ID was found in the Amazon/LT collection. In this cases, we labeled that book title as UNKNOWN.

We did not link book authors. When a book was referred to as "*the Stephen King book*", we did not mark this as a book title. Similarly, if a series was referred

Fig. 2. Example of the training data thread format for the Book title linking task. The corresponding label file contains three columns: threadid, postid, LT work id. In this case: 122992, 1, 5812.

```
<thread>
  <message>
    <date>Sep 1, 2011, 9:56am </date>
    <text>This month's read is Insomnia. Odd that I'm posting this I am
    yawning and blinking my eyes because I didn't sleep well last night.
    I remember not really caring for this one on my first read. The synopsis
    sounded excellent. But I was disappointed to find that it was basically
    a Dark tower spin-off. We'll see how it goes this time I guess.
    </text>
    <postid>1</postid>
    <username>jseger9000</username>
    <threadid>122992</threadid>
  </message>
  ...
</thread>
```

to by the name of the author, e.g., "*the Stieg Larson trilogy*", then the series was not labeled. We do consider these cases where the author is mentioned as borderline cases, because they point to both the author and the books that they wrote at the same time. In this data set we decided not to include them in the annotation, but we are aware that they fall in the 'grey' area of unclear cases.

Another source of annotation confusion were the forum threads about short stories and collections of stories. In these cases we did not label the individual short stories (they also do not have existing LT work IDs), but only the actual book with the collection.

Other difficult cases for the manual annotation were the cases where it was not immediately clear where the book title begins and ends. For example, in (1) below, the alternative book title could have been "*Bujold's Sharing Knife*" instead of "*Sharing Knife*". Vague or partial matches were also difficult to annotate sometimes. For example, the post containing fragment (2) was not linked to a work ID because it deviated significantly from the actual title of the book that was mentioned (and linked) correctly in the follow up post as being "*Fifteen Decisive Battles of the World from 1851*".

(1) have you read Lois McMaster Bujold's Sharing Knife books?

(2) I think there was a book called something like Ten Decisive Battles by a General Creasey

During manual annotation, 3 of the 220 threads were removed from the test set because of long lists of titles without context. The final test set consists of 217 threads comprising 5097 book titles identified in 2117 posts.

In order to assess the difficulty and subjectivity of the book linking annotation task we had 28 threads (155 posts) annotated by 2 assessors and we analyzed the differences in annotation. We found that there was quite some disagreement between the assessors: 71 books were linked by both assessors, and 247 by only one of the two. This implies that absolute agreement is only 22%.[2] There are two types of disagreement: (a) a book mention was linked by one assessor and missed/skipped by the other, and (b) a book mention was linked by both, but to different work ids. The most difficult were the mentions of book series. These should be linked to the first book of the series, which is not always trivial. For example, consider this post text:

> *Well, I could recommend some great Batman graphic novels, only one problem. They're written for adults, and are pretty dark. Year One is an amazing version of his origin story, but it isn't exactly appropriate for a second grader. You might try some of the Tintin graphic novels. There are dozens of them, and they're great stories. I second Louis Sachar as well. You might want to try Holes. Its a great, inventive story. Plus, you can watch the movie together once he finishes the book.*

Both assessors linked two series in this post. These were linked by assessor 1:

- David Mazzuchelli – Batman: Year One - Deluxe Edition: Year One
- Herge – Tintin in America (Tintin)

and these were linked by assessor 2:

- Lewis Richmond – Batman: Year One (Batman)
- Herge – Tintin in the Land of the Soviets

The difficulty of the annotation for the linking task is a topic that should be addressed in future editions of the SBS lab. One recommendation would be to write more explicit annotation guidelines, and share those with the participants.s

3.3 Evaluation

For the book request classification task, we computed and report only accuracy, as these are binary decisions. For the linking task, we computed accuracy, precision, recall, and F-score.

Both tasks were performed and evaluated at the level of forum posts. We detected whether a forum post was a book request in the classification task, and whether a certain book title occurred in a post. In case the same book title was mentioned multiple times in the same post, we only counted and evaluated on one occurrence of this particular book title. Each book title is mapped to a LibraryThing work ID that links together different editions of the same book (with different ISBNs).

[2] Note that Cohen's κ is undefined for these data because we the number of book titles for which the assessors agree that they should *not* be linked is infinite.

During manual annotation, we came across several book titles for which we were unable to find the correct LT work ID (labeled as UNKNOWN). These cases were problematic in the evaluation: just because the annotator could not find the correct work ID does not mean that it does not exist. For that reason, we decided to discard these examples in the evaluation of the test set results. In total, 180 out of the 5097 book titles in the test set were discarded for this reason.

Similarly, during the book request classification task, we also found some cases in the LT data where we were unsure about categorizing them as book search requests or not. We discarded 26 such cases from the test set in the evaluation.

4 Results

A total of 3 teams submitted 15 runs, 2 teams submitted 9 runs for the Classification task and 2 teams submitted 6 runs for the Linking task.

4.1 Task 1: Classifying forum threads

Baselines For the baseline system of the classification task, we trained separate classifiers for the two data sets (LT and Reddit) using scikit-learn.[3] We extracted bag-of-words-features (either words or character 4-grams) from the title and the body of the first post, and for LT also from the category (for Reddit, the category was the label). We used tf-idf weights for the words and the character 4-grams from these fields. We ran 3 classifiers on these data: Multinonial Naive Bayes (MNB), Linear Support Vector Classification (LinearSVC) and KNN, all with their default hyperparameter settings in scikit-learn. The results are in Table 3.

Evaluation of submitted runs The *Know* team reported an interesting experiment on the LT training data of the classification task [5]. A Naive Bayes classifier trained on a single feature, namely the quantified presence of question marks within the post, already achieved an accuracy of 80% on the LT training material. This gives us some insight into the skewed nature of this domain specific data set from a dedicated book forum: a post containing a question is likely to express a question with a book search request.

The LIPAH team compared two types of features for the classification task: (a) all nouns and verbs in the posts, and (b) compound nouns and phrases extracted using syntactic patterns. They found that the addition of syntactic phrases improves the classification accuracy [2].

Table 3 shows that for the LT data, the submitted runs did not beat the LinearSVC baselines. For the Reddit data however, runs by both teams were able to beat the best baseline system by a large margin. Since the Reddit dataset was much smaller than the LT dataset, the best strategy seems to be to add the LT

[3] http://scikit-learn.org/

Table 3. Results for the classification task for the two datasets in terms of accuracy on the 1974 LibraryThing and 89 Reddit posts.

LibraryThing			
Rank	**Team**	**Run**	**Accuracy**
1	baseline	character_4-grams.LinearSVC	94.17
2	baseline	Words.LinearSVC	93.92
3	Know	Classification-Naive-Results	91.59
4	baseline	character_4-grams.KNeighborsClassifier	91.54
5	baseline	Words.KNeighborsClassifier	91.39
6	LIPAH	submission2-librarything	90.98
7	LIPAH	submission3-librarything	90.93
8	LIPAH	submission4-librarything	90.83
9	Know	Classification-Veto-Results	90.63
10	LIPAH	submission1-librarything	90.53
11	baseline	character_4-grams.MultinomialNB	87.59
12	baseline	Words.MultinomialNB	87.59
13	Know	Classification-Tree-Results	83.38
14	Know	Classification-Forest-Results	74.82

Reddit			
Rank	**Team**	**Run**	**Accuracy**
1	LIPAH	submission6-reddit	82.02
2	Know	Classification-Naive-Results	82.02
3	LIPAH	submission5-reddit	80.90
4	baseline	Words.KNeighborsClassifier	78.65
5	baseline	Words.LinearSVC	78.65
6	baseline	character_4-grams.LinearSVC	78.65
7	baseline	character_4-grams.KNeighborsClassifier	78.65
8	Know	Classification-Tree-Results	76.40
9	Know	Classification-Veto-Results	76.40
10	baseline	Words.MultinomialNB	76.40
11	baseline	character_4-grams.MultinomialNB	76.40
12	Know	Classification-Forest-Results	74.16

Table 4. Results for the linking task for the LibraryThing data set in terms of accuracy.

Rank	Team	Run	# posts	Accuracy	Recall	Precision	F-score
1	Know	sbs16classificationlinking	4917	41.14	41.14	28.26	33.50
2	LSIS	BA_V2bis	4917	26.99	26.99	38.23	31.64
3	LSIS	BA_V1bis	4917	26.54	26.54	37.58	31.11
4	LSIS	B_V2bis	4917	26.01	26.01	35.39	29.98
5	LSIS	BUbis	4917	26.34	26.34	34.50	29.87
6	LSIS	Bbis	4917	25.54	25.54	34.80	29.46

training data to the Reddit training data for classifying the Reddit test threads. The best run for the Reddit data is LIPAH-submission6, which uses sequences of words and verbs as features.

4.2 Task 2: Book linking

Evaluation of submitted runs The results of the book linking task can be found in Table 4. The *Know* team used a list look-up system combined with a weighting threshold in their sbs16classificationlinking run to prevent the over-generation of potential book titles [5].

The LSIS team [4] first tried to detect book titles and author names at the phrase level (using SVM and CRF) inside posts and used Levenshtein distance to match titles to LT work IDs. Each detected unique book title was assigned to the larger post unit. They submitted 5 runs that varied in the way work IDs were matched against the potential book titles and the feature representation.

Both teams investigated the usage of author names in the proximity of potential book titles to disambiguate between potential titles and show that this is indeed a helpful feature. Both teams use complementary strategies for the book linking task as the *Know* systems has a higher recall while the LSIS run all achieves a better precision (as well as the highest F-score).

5 Conclusions and Plans

This was the first year of the Social Book Search Mining Track. Our goal was create a benchmark data set for text mining of book related discussion forum. In this first edition we focused on two tasks. The first task was to automatically identify which posts in a book forum tread are actual book search requests, and the second task was to detect which book titles are mentioned in a forum post and link the correct unique book ID to the post. We had three active participants who submitted a total of 15 runs. The book search classification task turned out to be a relatively straightforward task, both in manual annotation and in automatic prediction. A rather simple bag-of-words baseline classifier achieved an accuracy up to 94% on the LibraryThing data. The book linking task on the

other hand turned out to be a difficult task and here the best system achieved an accuracy of 41% and F-score of 33.5%.

Developing effective algorithms for automatically detecting and linking these book mentions would be a boon to the process of supporting complex search needs. Moreover, other book discussion websites, such as GoodReads or even dedicated Reddit threads may not have Touchstone-like functionality. Here, the need for automatic book linking algorithms is even more pressing.

Bibliography

1. J. Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20:37–46, 1960.
2. M. Ettaleb, C. Latiri, B. Douar, and P. Bellot2. In *Proceedings of the 7th International Conference of the CLEF Association, CLEF 2016*, Lecture Notes in Computer Science.
3. M. Koolen, T. Bogers, M. Gäde, M. A. Hall, H. C. Huurdeman, J. Kamps, M. Skov, E. Toms, and D. Walsh. Overview of the CLEF 2015 social book search lab. In J. Mothe, J. Savoy, J. Kamps, K. Pinel-Sauvagnat, G. J. F. Jones, E. SanJuan, L. Cappellato, and N. Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 6th International Conference of the CLEF Association, CLEF 2015, Toulouse, France, September 8-11, 2015, Proceedings*, volume 9283 of *Lecture Notes in Computer Science*, pages 545–564. Springer, 2015.
4. A. Ollagnier, S. Fournier, and P. Bellot. Linking task: Identifying authors and book titles in verbose queries. In *Proceedings of the 7th International Conference of the CLEF Association, CLEF 2016*, Lecture Notes in Computer Science.
5. H. Ziak, A. Rexha, and R. Kern. In *Proceedings of the 7th International Conference of the CLEF Association, CLEF 2016*, Lecture Notes in Computer Science.