SBS 2016 Track mining: Classification with linguistic features for book search requests classification

Mohamed Ettaleb[1], Chiraz Latiri[1], Brahim Douar[1], and Patrice Bellot[2]

[1] Tunis EL Manar University, Faculty of Sciences of Tunis, LIPAH research Laboratory, Campus Universitaire Farhat Hached, Tunis, Tunisia
[2] Aix-Marseille Université, CNRS, LSIS UMR 7296, 13397, Marseille, France
mouhamed.taleb@hotmail.fr, chiraz.latiri@gnet.tn, b.douar@gmail.com, patrice.bellot@univ-amu.fr

Abstract. In this paper, we describe text mining approaches dedicated to the classification track in Social Book Search Track Lab 2016. This track aims to exploit social knowledge extracted from LibraryThing and Reddit collections to identify which threads on online forums are book search requests. Our proposed classification model is based on combination of different textual features, namely : (*i*) basic linguistic features such as nouns and verbs; and, (*ii*) composed features such term sequences and noun phrases generated. Then, we applied a NaiveBayes classifier to specify the user's intentions in the requests.

Keywords: classification, noun phrases extraction, sequences mining.

1 Introduction

The Social Book Search (SBS) Lab investigates book search where the users information needs are complex, looking for more than objective metadata. In this respect, SBS Lab aims to research and develop techniques in order to support users in complex book search tasks. It consists of three tracks:

1. *Interactive Track*: a user-oriented interactive task investigating systems that support users in each of multiple stages of a complex search tasks. The track offers participants a complete experimental interactive IR setup and an exciting new multistage search interface to investigate how users move through search stages.
2. *Suggestion Track*: a system-oriented task for systems to suggest books based on rich search requests combining several topical and contextual relevance signals, as well as user profiles and real-world relevance judgements.
3. *Mining Track*: an NLP/Text Mining track focusing on detecting and linking book titles in online book discussion forums, as well as detecting book search request in forum posts for automatic book recommendation.

In this paper, we only consider the mining track which is a new one in SBS 2016 edition and investigates two tasks : (i) Classification task : how Information Retrieval Systems can automatically identify book search requests in online forums, and; (ii) Linking task : how to detect and link books mentioned in online book discussions.

Our contribution deals only with the classification task. The final objective of this task is to identify which threads on online forums are book search requests. Thereby, given a forum thread with one or more posts, the system should determine whether the opening post contains a request for book suggestions (*i.e.*, binary classification of opening posts).

In this respect, we propose to use two types of approaches, namely : an approach based on textual sequences mining, and an NLP method which relies on nouns, verbs and noun phrases extraction (*i.e.*, compound nouns), to improve the classification efficiency. Then, we use the NaiveBayes classifier with WEKA to specify the user's intentions in the requests.

The remainder of this paper is organized as follows: Section 2 describes the mining track and the test data. Then, section 3 recalls the basic definition for textual sequences mining and details our proposed approaches for book search requests classification. Next, Section 4 details our different submitted runs for the mining track as the official obtained results. The conclusion is given in Section 5.

2 SBS 2016 mining Track

The SBS 2016 mining Track investigates how systems can automatically identify book search requests in online forums and how to detect and link books mentioned in online book discussions. Often, users can have information needs that are difficult to express while considering a classical search engine and they rely in this case to online forums, in order to get recommendations from others users.

2.1 SBS requests classification task

Classification task identifies which threads on online forums are book search requests. That is, given a forum thread with one or more posts, the system should determine whether the opening post contains a request for book suggestions.

2.2 Description of Data collections

The test SBS 2016 collections contains:

1. A collection of 2 780 300 book records from Amazon, extended with social metadata from LibraryThing. This set represents the books available through Amazon. The records contain title information as well as a Dewey Decimal Classification (DDC) code (for 61% of the books) and category and subject information supplied by Amazon. Each book is identified by an ISBN. Note

that since different editions of the same work have different ISBNs, there can be multiple records for a single intellectual work. Each book record is an XML file with fields like ISBN, title, author, publisher, dimensions, number of pages and publication date. Curated metadata comes in the form of a Dewey Decimal Classification in the dewey field, Amazon subject headings in the subject field, and Amazon category labels in the browseNode fields. The social metadata from Amazon and LibraryThing is stored in the tag, rating, and review fields.

2. Two data collections for the classification task: LibraryThing and Reddit:
   - *Reddit training data*: the training data contains threads from the *suggestmeabook* subreddit as positive examples and threads from the books subreddit as negative examples. In the test data, the subreddit has been removed (*cf.* Table 1).
   - *LibraryThing*: 2,000 labelled threads for training, and 2,000 labelled threads for testing.

Table 1. Example of data format Reddit

```
<?xml version="1.0"?>
<forum type="reddit">
<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 inhe ritance cycle. So whatever you got helps :D
cheers lads, and lassses.</body>
<upvotes>8</upvotes>
<downvotes>0</downvotes>
</post>

</posts>
</thread>
</forum>
```

3 Approaches for book search requests classification

In this work, as depicted in Figure 1, we present two approaches for book search requests classification. The first one is based on the sequences mining technique to extract frequent sequences from textual content requests. While the second

one is based on NLP techniques. It consists in exploring textual content requests, and extracting verbs, nouns and compound nouns.

3.1 linguistic feature extraction

In the linguistic feature model, we begin with making the simplifying assumption about a text in the request that it can be represented as collections of words in which syntactic information a negligible and even the word order is unimportant. Text features extraction is the process of transforming what is essentially a bag of terms into a feature set that is usable by a classifier. We employed TreeTagger for annotating text with part-of-speech and lemma information [3]. We notice that the linguistic feature model is the simplest method; it constructs a word presence feature set from all the words of an instance. This method doesn't care about the order of the words, or how many times a word occurs, all that matters is whether the word is present in a list of words. In our approach, we chose to keep only the nouns and verbs for each request of the collection.

3.2 Compound nouns feature extraction

Earlier works in the literature proved that the use of simple terms features in classification is not accurate enough to represent the documents contents due to the words ambiguity. A solution to this problem is to use compound nouns[3] instead of simple words. The assumption is that compound nouns are more likely to identify semantic entities than simple words. We propose to perform a linguistic approach to extract compound nouns from the request content of the mining track 2016. The goal is to identify the dependencies and relationships between words through language phenomena. The linguistic approach for compound nouns extraction is based on two steps:

1. A complex−syntactic with a tagger (*i.e.*, Treetagger). Each word is associated to a tag corresponding to the syntactic category of the word, example: noun, adjective, preposition, proper noun, determiner, etc.
2. The tagged corpus is used to extract a set of compound nouns by the identification of syntactic patterns as detailed in [1].
   We adopt the definition of syntactic patterns given in [1], where a pattern is a syntactic rule on the order of concatenation of grammatical categories which form a noun phrase, *i.e.*, a compound noun.
   For the English language, We choose to define 12 syntactic patterns: 4 syntactic patterns of size two (for example: Noun Noun, Adjective Noun, etc.), 6 syntactic patterns of size three (for example: Adjective Noun Noun, Adjective Noun Gerundive, etc.) and 2 syntactic patterns of size 4.

[3] By compound nouns, we refer to complex terms and noun phrases.

3.3 Sequences feature mining

Most methods in text classification rely on contiguous sequences of words as features. Indeed, if we want to take non−contiguous (gappy) patterns into account, the number of features increases exponentially with the size of the text. Furthermore, most of these patterns will be more noisy. To overcome both issues, sequential pattern mining can be used to efficiently extract a smaller number of the most frequent features.

Sequential pattern mining problem was first proposed in [4], and then improved in [5]. It is worth noting that many methods used to discover sequential patterns are usually extension of approaches dedicated to mining frequent itemsets. Most of these approaches proceed on a bottom-up way. First, the frequent sets, or sequences, of size 1 are found, then longer frequent sequences are iteratively obtained starting from the shorter ones [5]. Finally, all the sequences fulfilling the required conditions are found. In our work, we use the LCM_SEQ algorithm [2][4] which is a variation of LCM[5] for sequences mining. The algorithm follows the scheme so called PREFIX SPAN, but the data structures and processing method are LCM based.

We adapt to our purpose the basic definitions of the theoretical framework for frequent sequential patterns discovery introduced in [4].

Definition 1. *A sequence $S = \langle t_1, \ldots, t_j, \ldots, t_n \rangle$, such that $t_k \in$ vacabulary V and n is its length, is a n-termset for which the position of each term in the sentence is maintained. S is called a n-sequence.*

Definition 2. *Given S a sequence discovered from the collection. The support of S is the number of sentences in $\mathcal{P}$ that contain S, S is said to be frequent if and only if its support is greater than or equal to the minimum support threshold minsupp.*

Interestingly enough, to address book search requests classification in an efficient and effective manner, we claim that a synergy with some advanced text mining methods, especially sequence mining [4], is particularly appropriate. However, applying the frequent sequences of terms in the context of requests classification can help select good features and improve classification accuracy, mostly because of the huge number of potentially interesting frequent sequences that can be drawn from a request collection.

3.4 Mining and learning process

The thread classification system serves to identify which threads on online forums are book search requests. Our proposed text mining based approaches are depicted in Figure 1. The classification threads process is performed on the following steps:

[4] http://research.nii.ac.jp/∼uno/code/lcm_seq.html
[5] LCM : Linear time Closed itemset Miner

1. Annotating the selected threads with part-of-speech and lemma information using TreeTagger.
2. Extracting linguistic features, *i.e.*, verbs and compound nouns from the annotated threads.
3. Generating the term sequence features using the efficient algorithm LCM_seq.
4. Generation of the classification model using the NaiveBayes classifier[6] under Weka[7].
5. Applying the classification model to the supplied test set.

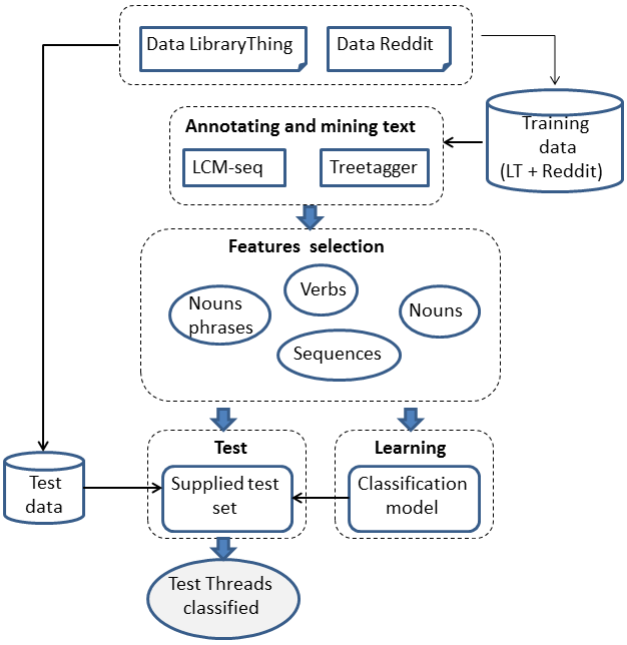

Fig. 1. The proposed approaches steps for book search requests classification

4 Experiments and results

4.1 Runs description

We conducted six runs according to the approaches described in Section 3, namely: four runs on the LibraryThing data collection and two runs on the Reddit data collection.

[6] The Bayesian Classification represents a supervised learning method as well as a statistical method for classification.

[7] http://www.cs.waikato.ac.nz/ml/weka/

Runs on the LibraryThing data collection

1. **Run1 (ID = Classification-NV)**: We used in this run, only Bag of linguistic features (*i.e.*, nouns and verbs) to generate the classification model, using the NaiveBayes classifier under WEKA using the default configurations[8].
2. **Run2 (ID = Classification-NVC)**: We extracted first, Bag of linguistic features (*i.e.*, nouns and verbs) and compound nouns from a set of 2000 threads. Then, we used these features to generate the classification model, using the NaiveBayes classifier.
3. **Run3 (ID = Classification-NVSeq)**: We used the nouns and verbs as in Run1, then, we extracted the sequences of words using LCM_SEQ algorithm with a threshold of *minsupp* =5, we noticed after series of experiments with differents threshold values that the *minsupp* =5 give the best results and had abvious clear impact on this features extraction. Finally, we combined all features to extract the classification model, using the NaiveBayes classifier.
4. **Run4 (ID = Classification-CSeq)**: In this run, we combined the compound nouns with sequences, using the NaiveBayes classifier.

Runs on the Runs Reddit data collection

1. **Run5 (ID = Classification-V)**: In this run, we used only the verbs as features to extract the classification model, using the NaiveBayes classifier.
2. **Run6 (ID = Classification-VSeq)**: In the second run on post Reddit, we extracted the sequences of words and the verbs as features using LCM_SEQ algorithm with a threshold of *minsupp* =3, we chose a low value of *minsupp* due to the limited number of sequence extracted from the collection Reddit. Finally, we generated the classification model with the NaiveBayes classifier.

4.2 Evaluation metric and results

The results obtained by our runs conducted for the classification task requests are evaluated in a single metric, which is the *Accuracy*. It simply measures how often the classifier makes the correct prediction. It is the ratio between the number of correct predictions and the total number of predictions (the number of test data points), thus :

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

where :

- TP: Number of True Positives
- FP: Number of False Positives
- TN: Number of True Negatives

[8] We used in all experiments the NaiveBayes classifier with WEKA using default configurations.

– FN: Number of False Negative

In the 2016 SBS Mining Track, a total of 3 teams submitted 20 runs, 2 teams submitted 14 runs for the Classification task and 2 teams submitted 6 runs for the Linking task.

Table 2 shows 2016 SBS track mining official results for our 4 runs conducted on the LibraryThing collection. Our runs are (Classification-NVC, Classification-NVSeq, Classification-CSeq, Classification-NV) ranked sixth, seventh, eighth and tenth, respectively, for the classification task. These results highlight that the combination of Bag of linguistic features (*i.e.*, nouns and verbs) and compound nouns performs the best in term of accuracy, *i.e.*, Classification-NVC. We note also that the combination of nouns, verbs and sequences of words, *i.e.*, Classification-NVSeq increases accuracy compared to the use of only Bag of linguistic features (*i.e.*, nouns and verbs). This is mainly due to the difference between users' descriptions of their needs.

Table 3 describes 2016 SBS track mining official results for our 2 runs conducted on the Reddit collection (Classification-VSeq and Classification-V), which are ranked first and third, respectively, in the classification task. The best run is performed with the sequences of words and the verbs as features for classification. This result confirms that mining sequences is useful for classification task.

It's worth noting that the obtained classification evaluation results shed light that our proposed approaches, based on NLP techniques, offer interesting results and helps to identify book search requests in online forums .

Table 2. Classification of the LibraryThing Threads

Rank	Team	Run	posts	Accuracy
1	**baseline**	**character_4-grams.LinearSVC (Best run)**	**1974**	**94.17**
2	baseline	Words.LinearSVC	1974	93.92
3	Know	Classification-Naive-Results	1974	91.59
4	baseline	character_4-grams.KNeighborsClassifier	1974	91.54
5	baseline	Words.KNeighborsClassifier	1974	91.39
6	**LIPAH**	**Classification-NVC**	**1974**	**90.98**
7	**LIPAH**	**Classification-NVSeq**	**1974**	**90.93**
8	**LIPAH**	**Classification-CSeq**	**1974**	**90.83**
9	Know	Classification-Veto-Resutls	1974	90.63
10	**LIPAH**	**Classification-NV**	**1974**	**90.53**
11	baseline	character_4-grams.MultinomialNB	1974	87.59
12	baseline	Words.MultinomialNB	1974	87.59
13	Know	Classification-Tree-Resutls	1974	83.38
14	Know	Classification-Forest-Resutls	1974	74.82

Table 3. Classification of the Reddit posts

Rank	Team	Run	posts	Accuracy
1	**LIPAH**	**Classification-VSeq (Best run)**	**89**	**82.02**
2	know	Classification-Naive-Resutls	89	82.02
3	**LIPAH**	**Classification-V**	**89**	**80.90**
4	baseline	Words.KNeighborsClassifier	89	78.65
5	baseline	Words.LinearSVC	89	78.65
6	baseline	character_4-grams.LinearSVC	89	78.65
7	baseline	character_4-grams.KNeighborsClassifier	89	78.65
8	know	Classification-Tree-Resutls	89	76.40
9	Know	Classification-Veto-Resutls	89	76.40
10	baseline	Words.MultinomialNB	89	76.40
11	baseline	character_4-grams.MultinomialNB	89	76.40
12	know	Classification-Forest-Resutls	89	74.16

5 Conclusion

In this paper, we presented our contribution for the 2016 Social Book Search Track, especially for the SBS Mining track. In the 6 submitted runs dedicated for book search requests classification, we tested three approaches for features selection, namely : Bag of linguistic features (*i.e.*, nouns and verbs), compound nouns and sequences, and their combination. We performed classification with Weka with NaiveBayes classifier. We showed that combining Bag of linguistic features (*i.e.*, nouns and verbs) and compound nouns improves accuracy, and integrating sequences in classification process enhances the performance. So, the results confirmed that the synergy between the NLP techniques (textual sequences mining and nouns phrases extraction) and the classification system is fruitful.

References

1. Hatem Haddad. French noun phrase indexing and mining for an information retrieval system. In *String Processing and Information Retrieval, 10th International Symposium, SPIRE 2003, Manaus, Brazil, October 8-10, 2003, Proceedings*, pages 277–286, 2003.
2. Takanobu Nakahara, Takeaki Uno, and Katsutoshi Yada. *Knowledge-Based and Intelligent Information and Engineering Systems: 14th International Conference, KES 2010, Cardiff, UK, September 8-10, 2010, Proceedings, Part III*, chapter Extracting Promising Sequential Patterns from RFID Data Using the LCM Sequence, pages 244–253. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
3. Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK, 1994.
4. R. Srikant and R. Agrawal. Mining generalised associations rules. In *Proceedings of the* 21^{th} *International Conference on Very Large Databases, VLDB'95*, pages 407–419, Zurich, Switzerland, September 1995.

5. R. Srikant and R. Agrawal. Mining sequential patterns : Generalizations and performance improvements. In *Proceedings of the* 5^{th} *International Conference on Extending Database Technology, EDBT'96*, volume 1057 of *LNCS*, pages 3–17, Avignon, France, March 1996. Springer-Verlag.