An Automatic Classification of Book Texts to User-Defined Tags

Sharon Givon* and **Theresa Wilson**†
School of Informatics
Edinburgh University
Edinburgh, UK
*S.Givon@sms.ed.ac.uk , †twilson@inf.ed.ac.uk

Abstract

We describe work on automatically assigning labels to books using user-defined tags as the label set. Using supervised learning and exploring both binary and multiclass classification, we train and test classifiers on several sets of features, focusing on the size of the sets, part-of-speech classes and named entities. Results indicate that a binary classifier, trained and tested on a feature space that consists of a limited selection of parts of speech as well as all frequent named entities, achieves a classification precision of 81%, significantly outperforming a baseline which assigns the top-10 most popular tags to each book.

Introduction

The feature of *tagging*, where on-line users associate free-text tags with items and products, has been recentlty gaining great popularity as a source of information from which users as well as e-tailers and online service providers can bennefit. For example, tagging information may be used by consumers to search and identify unknown products. Tags can also be used by companies as a source of information for making recommendations. In this work we focus on book texts. Books make a “classic” product type to be marketed and consumed on-line and major on-line booksellers report consistent sales growth. Bookselling websites get richer with “bookstore-like” features such as the ability to view selected pages, or search inside the book. At the same time, more sources of full texts of books are becoming available on-line. Different websites collect metadata related to books, with tagging being one type, as well as information such as ratings and book reviews. Such sources of information are valuable for marketing tasks. However, not all books have this information associated with them. In such cases, it is hard - if not impossible - to bring the books to the notice of potential buyers. With user-provided tags and book texts becoming increasingly available, a possible solution to this problem is to learn to automatically assign tags based on the full texts, such as would be generated by real users.

Related Work

(Betts, Milosavljevic, & Oberlander 2007) conducted experiments using the full content of books to assign them labels from the Library of Congress Classification (LoCC). They explored the utility of Information Extraction (IE) techniques within a text categorisation (TC) framework, automatically extracting structured information from the full text of books. In our experiments, we adopt their approach by working with slightly similar types of features; however, rather than using a fixed taxonomy we are classifying the texts to user-defined tags. Work on utilizing book texts and metadata towards tasks such as improving recommendations included (Givon & Milosavljevic 2007) (extracting central characters), and (Mooney & Roy 2000) (used structured metadata and unstructured textual content to classify books and improve product recommendations).

Data & Pre-Processing

We used Project Gutenberg[1] as our primary source for book text data and collected a total of 120 books written in English and associated with the fiction/literature domains. We used 90 books for training and the remainder for testing. Each book was passed through an NLP pipeline. This produced an output that consisted of the tokenised text and information such as sentence splits, part-of-speech (POS) tags, and named entities (NEs). Tagging data was collected from LibraryThing[2]. We processed the entire tag set to filter out irrelevant tags and to group similar tags. Our final tag set consisted of 50 tags.

Experiments

Classification Method & Types

Since tag words are not likely to appear in the texts, this task becomes more complex than simple keyword search, and we approach it as one of supervised text classification. As our classifier, we chose BoosTexter (Schapire & Singer 2000), a general purpose machine-learning program that is based on boosting. We investigated the difference in performance between binary and multiclass classification methods. To train a multiclass classifier, all of the top-10 tags for each

Copyright © 2008, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

[1] http://www.gutenberg.org/

[2] http://www.librarything.com

book were given to the classifier in each experiment. During the multi-classification process, for each book, BoosTexter assigned a rank to each of the 50 tags in the set. From the assigned tags, we selected the 10 with the highest rank as the classifier's prediction for the book. For binary classification, we trained 50 different classifiers, one for each tag. For a given book and a given tag the class-label is true if the tag is in the boook's top-10 tags.

The Feature Space

Our feature space combined bags of words from the book, and named entities, which are atomic elements in text representing names of persons, organizations, locations and more. We chose words to be included in the bags according to their POS and tf-idf score[3]. For named entities we selected only the ones appearing more than twice. For both features, we used the POS and NE output of the pipeline. Our experiments varied in the selection of POS, the size of bags of words and the use of NEs.

Results

For each book, we evaluated how many of the top-10 tags assigned by a classifier are contained within the top-10 tag set of the book. This yields an accuracy score for the multiclass classification and precision, recall and F-score scores for the binary classification. The difference between the nature of the scores for the two classification methods is that for multiclass classification, the number of false and true 'positives' remained constant resulting in a single accuracy score, while for binary classification these numbers varied based on the book, resulting in different recall and precision scores. Nevertheless, since the accuracy and precision scores are computed similarly, we analyse the results based on a comparison between the two.

Table 1 shows the results of our experiments. The baseline is a classifier that assigns to all books in the test set the same top-10 most-frequent tags from the books in the training set. The first column shows the features used in each experiment, the second column (MC) shows the accuracy for multiclass classification, and the last three columns show precision, recall and F-score for binary classification. The results for both classification methods proved better than their baselines. The difference was significant[4] with $p \leq 0.0003$ for multiclass accuracy and $p \leq 0.0002$ for the binary F-score. When comparing the results of the multiclass classification to those of the binary method, we found that the binary classification results were consistently higher. This shows that binary classification is a more effective solution to the problem of assigning tags to books. When adding NEs to the feature space, in most cases, the scores were improved or remained the same. In particular, for the binary results we found that in all cases where we added NEs to the feature space, recall improved, and in some of the cases accuracy improved as well. The best NE results yielded a significantly[4] better average F-score of 0.67 (with $p \leq 0.001$).

Table 1: Classification Results

	MC	Binary		
Features	A	P	R	F
* NN, ADJ	0.69	**0.82**	0.52	0.62
* NN, ADJ, NE	0.67	**0.81**	0.56	0.63
NE	0.63	0.72	0.49	0.56
$^{\triangle}$ VB	0.64	0.76	0.4	0.52
$^{\triangle}$ VB, NE	0.67	0.76	0.53	0.58
$^{\triangle}$ NN, ADJ, VB	0.68	0.76	0.4	0.52
$^{\triangle}$ NN, ADJ, VB, NE	0.68	**0.81**	**0.58**	**0.67**
* NN, ADJ, VB	0.69	0.76	0.4	0.52
* NN, ADJ, VB, NE	**0.7**	**0.81**	0.54	0.64
$^{\triangle}$ All POS	**0.71**	0.73	0.57	0.62
$^{\triangle}$ All POS, NE	0.69	**0.81**	0.56	0.65
Baseline	0.5	0.28	0.44	0.33

*No limitation on set size $^{\triangle}$Top 1500 tf-idf

Conclusions & Further Work

Our encouraging results show that it is possible to generate metadata such as tags from book texts. We believe that this will prove to be a highly efficient solution to the new item "ramp-up" problem (Konstan *et al.* 1998) - when a new item is encountered that does not have sufficient metadata and thus cannot be easily recommended. We showed the results for two classification methods for the task of classifying full texts of books to a set of user defined tags. We used a set of fiction book texts to train and test BoosTexter to classify them to a set of the 50 most frequent tags across all of the books in our corpus. We found that the binary classification results outperformed the multiclass classification results in experiments. In terms of NEs, the results support our hypothesis which stated that adding NEs to the feature space significantly improves the results.

References

Betts, T.; Milosavljevic, M.; and Oberlander, J. 2007. The utility of information extraction in the classification of books. In *Proceedings of the European Conference on Information Retrieval*.

Givon, S., and Milosavljevic, M. 2007. Extracting useful information from books. In *Proceedings of Recherche d'Information Assiste par Ordinateur*.

Konstan, J. A.; Riedl, J.; Borchers, A.; and Helocker, J. L. 1998. Recommender systems: A grouplens perspective. In *Recommender Systems: Papers from the 1998 Workshop (AAAI Technical Report WS-98-08)*, 60–64. AAAI Press.

Mooney, R., and Roy, L. 2000. Content-based book recommending using learning for text categorization. In *Proceedings of the 5th ACM Conference on Digital Libraries*.

Schapire, R. E., and Singer, Y. 2000. BoosTexter: A boosting-based system for text categorization. *Machine Learning* 39(2/3):135–168.

[3] Computed using a corpus of 1200 books

[4] Tested with the Wilcoxon Two Sample Test