Web Content Classification with Topic and Sentiment Analysis

Shuhua Liu, Thomas Forss, Kaj-Mikael Bjork

Web Content Classification with Topic and Sentiment Analysis

Shuhua Liu, Thomas Forss and Kaj-Mikael Björk

Department of Business Administration and Analytics
Arcada University of Applied Sciences
Jan-Magnus Janssonin aukio 1 00560 Helsinki Finland
{shuhua.liu, thomas.forss, kaj-mikael.bjork}@arcada.fi

Abstract. Automatic classification of web content has been studied extensively, using different learning methods and tools, investigating different datasets to serve different purposes. Most of the studies have made use of content and structural features of web pages. In this study we present a new approach for automatically classifying web pages into pre-defined topic categories. We apply text summarization and sentiment analysis techniques to extract topic and sentiment indicators of web pages. We then build classifiers based on the extracted topic and sentiment features. Our results offer valuable insights and inputs to the development of web detection systems.

Keywords: web content classification, sentiment analysis, text summarization, online safety solutions

1 Introduction

Web content classification, also known as web content categorization, is the process of assigning one or more predefined category labels to a web page. It is often formulated as a supervised learning problem where classifiers are built through training and validating using a set of labeled data. The classifiers can then be applied to label new web pages, or in other words, to detect if a new webpage falls into certain predefined categories.

Automatic classification of web pages has been studied extensively, using different learning methods and tools, investigating different datasets to serve different purposes [13]. Chakrabarti et al [3] studied hypertext categorization using hyperlinks. Chen and Dumais [4, 7] explored the use of hierarchical structure for classifying a large, heterogeneous collection of web content. They applied SVM classifiers in the context of hierarchical classification and found small advantages in accuracy for hierarchical models over flat (non-hierarchical) models. They also found the same accuracy using a sequential Boolean decision rule and a multiplicative decision rule, with much more efficiency.

Our research concerns the development of classification systems for online safety and security solutions. Our work is motivated by the fact that certain groups of web pages such as those carry hate and violence content have proved to be much harder to classify with good accuracy when both content and structural features are already taken into consideration. There is a need for better detection systems that utilize enriched features coupled with good classification methods for identifying excessively offensive and harmful websites.

Hate and violence web pages often carry strong negative sentiment while their topics may vary a lot. Based on this observation, in this study we explored the effectiveness of combined topic and sentiment features for improving automatic classification of web content. We first apply text summarization and sentiment analysis techniques to extract topic and sentiment indicators of web pages. We then build classifiers based on the extracted topic and sentiment features. Large amount of experiments and analysis were carried out. Our results offer valuable insights and inputs to the development of web detection systems and online safety solutions.

2 Data and Methods

Our dataset is a collection of over 165,000 single labeled web pages in 20 categories. Each webpage is represented by a total of 31 attributes including full page, URL, Title and other meta-content, plus structural info and link information. The experiments reported in this paper mainly concern a number of specific web categories (described in Section 3 and Section 4).

In this study we only take into consideration the page attributes that are text-related. Our focus is on added value to web classification that can be gained from textual content analysis. Taking into account of missing entries for different attributes, we selected a subset of the content features as the raw data for our study: full page free text content, URL words, title words (TextTitle), meta-description terms (CobraText, CobraMetaDescription, CobraMetaKeywords, TagTextA and TagTextMetaContent).

We should point out that, structural features and hyperlink information capture the design elements of web pages that may also serve as effective indicators of their content nature and category [2, 6]. They contain very useful information for web classification. In addition, analysis of images contained in a web page would provide another source of useful information for web classification [5]. However, these topics are studied in other projects.

Our approach to web content classification is illustrated in Figure 1. Exploring the textual information, we applied word weighting, text summarization and sentiment analysis techniques to extract topic features, content similarity features and sentiment indicators of web pages to build classifiers.

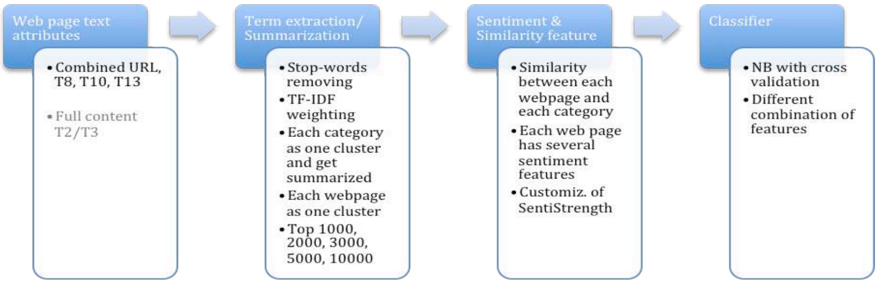
Web page text attributes
• Combined URL, T8, T10, T13
• Full content T2/T3
Term extraction/ Summarization
• Stop-words removing
• TF-IDF weighting
• Each category as one cluster and get summarized
• Each webpage as one cluster
• Top 1000, 2000, 3000, 5000, 10000
Sentiment & Similarity feature
• Similarity between each webpage and each category
• Each web page has several sentiment features
• Customiz. of SentiStrength
Classifier
• NB with cross validation
• Different combination of features

Fig. 1. Web content classification based on topic and sentiment analysis

2.1 Topic Extraction

We start with extracting topics from each web page and the collections of web pages belonging to same categories. The extracted topics hopefully give a good representation of the core content of a web page or a web category.

Topic extraction is based on automatic identification of important and informative terms from a text. The goal is to select a set of words or phrases that are related to the main topics discussed in the given text. Topic extraction has been the subject of study for a long time, and there exists a large body of literature on it and many proposed methods. Hasan and Ng [20] presented a recent survey of the state of the art in key phrase extraction, and grouped topic extraction methods into two broad categories: supervised and unsupervised. Some early studies on key phrase extraction took a supervised approach and formulated the task as a classification problem [17, 22]. Later on Jiang et al. [18] proposed a pairwise ranking approach, to learn a ranker that rate two candidate key phrases and has been shown to significantly outperform the classification methods. These supervised methods make use of statistical features, structural features, syntactic features of the corpus, as well as external resource based features (e.g. Wikipedia). Unsupervised approach on the other hand applied graph-based ranking methods [21, 23, 24], clustering methods [21, 25, 26] (Grineva et al., 2009; Liu et al., 2009b; Liu et al., 2010) and language modeling methods [19].

The various different methods, with their pros and cons, have brought continuous development in the field. However, as pointed out in Liu et al [26] and Hasan and Ng [20], topic extraction is still a task far from being accomplished when we look at the state-of-the-art performance level. To make further improvements it is more important to incorporate background knowledge than solely focus on algorithmic development.

For our study, we choose to make use of results from text summarization research. Text summarization tools have the capability to distill the most important content from text documents. However, most of the text summarization systems are concerned with sentence extraction targeted for human users. To help web content classification, we believe simple term extraction could be a sufficiently effective and more efficient approach, as topic extraction is just one of the many middle-steps towards facilitating open domain text classification, we want to keep it generic, simple and efficient. So we applied the time-tested tf-idf weighting method to extract topic terms from web pages [16]. Terms in this study are still limited to individual words (experiments with n-grams in our later work). We will compare the effect of using n-grams and language models in another article.

For each webpage, we make use of its different content attributes as input for term weighting. Applying different compression rate, we obtained different sets of topic words (e.g. top 50, top 100, top 20%, 35%, 50%, 100%).

The content of a web category is obtained through summarization of all the web pages in the same category. For each web page collection, we apply the Centroid method of the MEAD summarization tool to make summaries of the document collection [14, 15]. Through this we try to extract topics that are a good representation of a specific web category. MEAD is applied here instead of simply tf-idf weighting to facilitate processing of large collection of web pages and reducing redundancy. MEAD offers a benchmarking text summarization method. Given a document or a collection of documents to be summarized, it creates a cluster and all sentences in the cluster are represented using tf-idf weighted vector space model. A pseudo sentence, which is the average of all the sentences in the cluster, is then calculated. This pseudo sentence is regarded as the centroid of the document (cluster). A centroid represents a set of the most important/informative words of the whole cluster, thus can be regarded as the best representation of the entire document collection.

2.2 Extracting Sentiment Features

Sentiment analysis is the process of automatic extraction and assessment of sentiment-related information from text. Sentiment analysis has been applied widely in extracting opinions from product reviews, discovering affective dimension of the social web [8, 11].

Sentiment analysis methods generally fall into two categories: (1) the lexical approach - unsupervised, use direct indicators of sentiment, i.e. sentiment bearing words; (2) the learning approach - supervised, classification based algorithms, exploit indirect indicators of sentiment that can reflect genre or topic specific sentiment patterns. Performance of supervised methods and unsupervised methods vary depending on text types [12].

SentiStrength [11, 12] takes a lexical approach to sentiment analysis, making use of a combination of sentiment lexical resources, semantic rules, heuristic rules and additional rules. It contains a EmotionLookupTable of 2310 sentiment words and wordstems taken from Linguistic Inquiry and Word Count (LIWC) program [9], the General Inquirer list of sentiment terms [10] and ad-hoc additions made during testing of the system. The SentiStrength algorithm has been tested on several social web data sets such as MySpace, Twitter, YouTube, Digg, Runners World, BBC Forums. It was found to be robust enough to be applied to a wide variety of social web contexts.

While most opinion mining algorithms attempt to identify the polarity of sentiment in text - positive, negative or neutral, SentiStrength gives sentiment measurement on both positive and negative direction with the strength of sentiment expressed on different scales. To help web content classification, we use sentiment features to get a grasp of the sentiment tone of a web page. This is different from the sentiment of opinions concerning a specific entity, as in traditional opinion mining literature.

As a starting point, we apply unsupervised method to the original SentiStrength system [11, 12]. Sentiment features are extracted by using the key topic terms extracted from the topic extraction process as input to SentiStrength. This gives sentiment strength value for each web page in the range of -5 to +5, with -5 indicating strong negative sentiment and +5 indicating strong positive sentiment. We found that negative sentiment strength value a better discriminator of web content than positive sentiment strength value at least for the three web categories Hate, Violence and Racism. Thus, in our first set of experiments we only uses negative sentiment strength value as data for learning and prediction. Corresponding to the six sets of topic words for each web page, six sentiment features are obtained.

2.3 Extracting Topic Similarity Features

We use topic similarity to measure the content similarity between a web page and a web category. Topic similarity is implemented as the cosine similarity between topic terms of a web page and topic terms of each web category. Topic terms are a set of top-weighted individual words. We set a length for the topic vectors based on testing of several options. A set of similarity features is extracted for each web page, considering different compression rates.

3 Sentiment based Classifier for Detecting Hate, Violence and Racism Web Pages

Three datasets are sampled from the full database. The datasets contain training data with balanced positive and negative examples for the three web categories: Hate, Violence and Racism. Each dataset makes maximal use of positive examples available, resulting in a dataset of 3635 web pages for Violence, 9040 for Hate and 6155 for

Racism. Features for learning include a number of negative sentiment strength values of each web page, based on different sets of topic terms.

We built classification model using NäiveBayes (NB) method with cross validation, as three binary classifiers: c = 1, belong to the category, (Violence, Hate, Jew-Racism), c = 0 (not belong to the category). NB Classifier is a simple but highly effective text classification algorithm that has been shown to perform very well on language data. It uses the joint probabilities of features and categories to estimate the probabilities of categories given a document. Support Vector Machines (SVM) is another most commonly used algorithms in classification and foundation for building highly effective classifiers to achieve impressive accuracy in text classification. We experimented with both NB and SVM methods, found that they achieved similar results, while SVM training takes much longer time in training.

We tested with different combination of the sentiment features. The best results show good precision and recall levels for all three categories.

Table 1. Sentiment based NB classifiers

Model Performance		
Category	Precision	Recall
Hate	71.38%	77.16%
Racism	63.29%	72.79%
Violence	81.91%	73.92%

4 Combining Topic Similarity and Sentiment Analysis in Web Content Classification

Following our first batch experiments, we extend our study from 3 to 8 web categories (dataset size 3282, 3479, 5105, 400, 4667, 5438, 1919, 3432). We first developed classifiers based on topic similarity features, and the results were very disappointing for most categories, low on both precision and recall measures in general. We thus conclude that topic similarity based classifiers alone do not perform well.

Next we seek to improve the classification performance through combined use of topic similarity features and sentiment features. The results are very encouraging and the classification performance is significantly improved for most categories.

4.1 Extracting New Sentiment Features

In this second round of experiments we made use of combined metadata of web pages as raw data, extracted topic terms and then the sentiment features again for each web page. We tried different ways to customize the SentiStrength algorithm: (1) Counts of the amount of positives and negative sentiment words in a web page; (2) Sum of word sentiment value weighted by word frequency, normalized on total word counts, value between -5 and 5; (3) update the EmotionLookupTable. We found only few novel terms comparing with the original EmotionLookupTable, so we didn't pursue it further as the effect would be minor.

We tested new sentiment feature based NB classifiers for a few web categories. They do not necessarily perform better than the earlier sentiment based classifier. The performance varies from category to category, some slightly better, some not.

4.2 Classification using Combined Features

Next, we built NäiveBayes classification models (with cross validation) for eight web categories, using combined topic similarity features and sentiment features. The model performances are significantly improved for almost all categories when compared with solely sentiment based or solely topic similarity based classifiers, as is shown in Table 2. Recall levels are especially good, except the Violence category, which has a bit lower recall level but very good precision level.

Table 2. Classifiers making use of combined sentiment and topic similarity features

Model Performance (combined features)					
Category	Precision	Recall	Category	Precision	Recall
Cults	75.8%	90.55%	RacismWh	98.26%	96.30%
Occults	87.08%	91.84%	RacistGr	69.96%	91.82%
Violence	93.69%	82.75%	JewRel	64.43%	96.28%
Unknown	89.59%	93.31%	Religion	67.01%	92.81%

5 Conclusions and Future Work

In this study we set out to build sentiment aware web content detection systems. We developed different models for automatically classifying web pages into pre-defined topic categories. Word weighting, text summarization and sentiment analysis techniques are applied to extract topic and sentiment indicators of web pages. Large amount of experiments were carried out and classifiers are built based on topic similarity and sentiment features. Our results indicate that sentiment based classifiers bring much added value in the classification of Violence, Hate and Racism webpages. Topic similarity based classifiers solely do not perform well, but when topic similarity and sentiment features are combined, the classification model performance is significantly improved for most of the eight selected web categories.

Our future work would include the incorporation of LDA topic models [1] and its variations, n-grams, word ontology, domain knowledge and structural features. We will also look into new topic similarity measures. We believe there is still much room for improvements and some of these methods will hopefully help to enhance the classification performance to a new level. Our goal will be on improving precision and reducing false positives.

References

1. Blei, D. M., Ng, A. Y., and Jordan, M. I. 2001. *Latent dirichlet allocation*. Advances in neural information processing systems. 601-608.
2. Calado, P., Cristo, M., Goncalves, M. A., de Moura, E. S., Ribeiro-Neto, B., and Ziviani, N. 2006. Link-based similarity measures for the classification of web documents. *Journal of the American Society for Information Science and Technology* (57:2), 208-221.
3. Chakrabarti, S., B. Dom and P. Indyk. 1998. Enhanced hypertext categorization using hyperlinks. *Proceedings of ACM SIGMOD 1998.*
4. Chen, H., and Dumais S. 2000. Bringing order to the Web: automatically categorizing search results. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 145–152. New York, NY, ACM Press.
5. Chen, Z., Wu, O., Zhu, M., and Hu, W. 2006. A novel web page filtering system by combining texts and images. *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, 732–735. Washington, DC IEEE Computer Society.
6. Cohen, W. 2002. Improving a page classifier with anchor extraction and link analysis. In S. Becker, S. Thrun, and K. Obermayer (Eds.), *Advances in Neural Information Processing Systems* (Vol. 15, Cambridge, MA: MIT Press) 1481–1488.
7. Dumais, S. T., and Chen, H. 2000. Hierarchical classification of web content. *Proceedings of SIGIR'00,* 256-263.
8. Pang, B., and Lee, L. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval* 2(1-2), 1-135, July 2008
9. Pennebaker, J., Mehl, M., & Niederhoffer, K. 2003. Psychological aspects of natural language use: Our words, our selves. *Annual review of psychology, 54(1),* 547–577.
10. Stone, P. J., Dunphy, D. C., Smith, M. S., and Ogilvie, D. M. 1966. *The general inquirer: a computer approach to content analysis.* The MIT Press, Cambridge, Massachusetts, 1966. 651
11. Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., and Kappas, A. 2010. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology, 61(12),* 2544–2558.
12. Thelwall, M., Buckley, K., and Paltoglou, G. 2012. Sentiment strength detection for the social Web. *Journal of the American Society for Information Science and Technology, 63(1),* 163-173.

13. Qi, X., and Davidson, B. 2007. *Web Page Classification: Features and Algorithms.* Technical Report LU-CSE-07-010, Department of Computer Science and Engineering, Lehigh University, Bethlehem, PA, 18015
14. Radev, D., Allison, T., Blair-Goldensohn, S., Blitzer, J., Celebi, A., Dimitrov, S., and Zhang, Z. 2004. MEAD-a platform for multidocument multilingual text summarization. *Proceedings of the 4th International Conference on Language Resources and Evaluation 2004* (Lisbon, Portugal, May 2004)
15. Radev D., Jing, H., Styś, M., and Tam, D. 2004. Centroid-based summarization of multiple documents. *Information Processing and Management* (40) 919–938.
16. Salton, G., and Buckley, C. 1988. Term-weighting approaches in automatic text retrieval. *Information processing and management,* 24(5), 513-523.
17. Turney P. 2000. Learning algorithms for keyphrase extraction. Information Retrieval, 2:303–336.
18. Jiang X., Y. Hu, and H. Li. 2009. A ranking approach to keyphrase extraction. In Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 756–757.
19. Tomokiyo T. and M. Hurst. 2003. A language model approach to keyphrase extraction. In Proc. of the ACL Workshop on Multiword Expressions, pp 33–40.
20. Hasan K. and V. Ng. 2014. Automatic Keyphrase Extraction: A Survey of the State of the Art. To appear in the proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL), 2014
21. Grineva M., M. Grinev and D. Lizorkin. 2009. Extracting key terms from noisy and multi-theme documents. In Proceedings of the 18th International Conference on World Wide Web, pages 661–670.
22. Frank E., G. W. Paynter, I. H. Witten, C. Gutwin, and C. G. Nevill-Manning. 1999. Domain-specific keyphrase extraction. In Proceedings of 16th International Joint Conference on Artificial Intelligence, pages 668–673.
23. Matsuo M. and M. Ishizuka. 2004. Keyword extraction from a single document using word co-occurrence statistical information. International Journal on Artificial Intelligence Tools, 13
24. Mihalcea R. and P. Tarau. 2004. TextRank: Bringing order into texts. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pages 404–411.
25. Liu Z., P. Li, Y. Zheng and M. Sun. 2009. Clustering to find exemplar terms for keyphrase extraction. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 257–266.
26. Liu Z., W. Huang, Y. Zheng and M. Sun. 2010. Automatic keyphrase extraction via topic decomposition. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 366–376.