Improving Topic Exploration in the Blogosphere by Detecting Relevant Segments

Renato Domínguez García
(Multimedia Communications Lab, TU Darmstadt, Germany
Renato.Dominguez.Garcia@kom.tu-darmstadt.de)

Alexandru Berlea
(SAP Research, Germany
alexandru.berlea@sap.com)

Philipp Scholl
(Multimedia Communications Lab, TU Darmstadt, Germany
Philipp.Scholl@kom.tu-darmstadt.de)

Doreen Böhnstedt
(Multimedia Communications Lab, TU Darmstadt, Germany
Doreen.Boehnstedt@kom.tu-darmstadt.de)

Christoph Rensing
(Multimedia Communications Lab, TU Darmstadt, Germany
Christoph.Rensing@kom.tu-darmstadt.de)

Ralf Steinmetz
(Multimedia Communications Lab, TU Darmstadt, Germany
Ralf.Steinmetz@kom.tu-darmstadt.de)

Abstract: With the accelerated growth of the blogosphere, automatically analyzing blogs (specifically extracting information) becomes increasingly important. Here, we focus on the fundamental task of automatically detecting blog topics in order to support users to explore a collection of blogs by focusing on different particular topics according to their interests. We show that topic exploration can be significantly improved (by up to 33%) by using a novel approach to detecting blog page segments that contain relevant information for the blogs' topic.

Key Words: Blogs, web page segmentation, segment classification, machine learning, Topic Exploration
Category: M.7, H.3.3, M.0

1 Introduction

It is estimated that there are more than 50 million weblogs on the internet and this number is growing daily. They are forming a network —the blogosphere—

by means of comments and trackbacks to other blogs. Usually, blogs' contents reflect (and sometimes even influence) the public opinion [Ulicny 08].

Dealing with the huge amount of content available on the blogosphere requires the ability to filter it based on one's topics of interest. Keyword search, as typically offered by (blog) search engines, merely returns a list of pages containing the keywords. Most of the pages in the result list are not relevant for the user, the user being left alone with the task of manually filtering the list in order to find his topics of interest. The user can be relieved from this burden if appropriate methods for automatic topic detection are used. We show that this is possible by identifying one such approach to topic detection, i.e., finding out topic(s) represented in text fragments —here: blog entries and comments— and applying it to a real application scenario: we query the blogosphere for answers to the question: "Who is responsible for the current financial crisis?".

In particular, we show that our approach to topic exploration is feasible in practice, especially if we filter out irrelevant content by first segmenting the page and then *classifying these segments*. In principle, any web mining application can benefit from the detection and classification of blogs' functional segments. We illustrate our approach and show this can significantly improve automatic topic exploration in blogs in Section 2. Section 3 presents our evaluation method and reports the evaluation's results. A conclusion will be given in Section 4.

2 Our Approach

We restrict ourselves to the challenge of recognizing relevant segments in blogs, as they are a widely adopted and used as the most present medium for diffusion of user generated content. We show that this task is feasible and provide a proof-of-concept evaluation by exploring topics in the blogosphere. Weblog authoring software supports different languages, therefore use of language in our scenario is not a discerning feature for filtering of relevant content and should be neglected in favour of a truly language agnostic approach. The task described in our application scenario can be done by analyzing either directly the whole blog page or only relevant segments. The second option is computationally more expensive due to the costs of extracting relevant segments, but we will show that the results are significantly better. These two different options are illustrated in Figure 1.

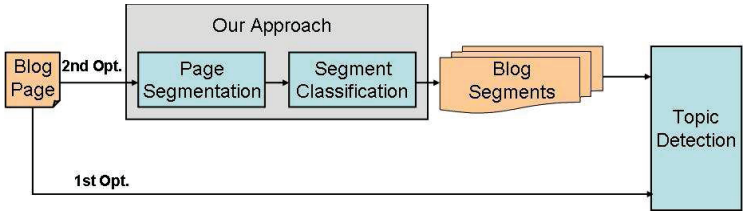

Figure 1: Topic Detection on the whole page (Option 2) vs. only on selected segments (Option 1)

Additionally, Figure 1 shows a process overview of our approach. It consists of two steps that will be explained in this section: segmentation of blogs (Section 2.1) and segment classification using machine learning methods (Section 2.2). Eventually, we explain the applied topic detection approach in Section 2.3.

2.1 Web Page Segmentation

Web pages usually consist of several functional building blocks like menus, headers, blog posts or comments. Such block types are called *segment genres*. *Segmentation* of web pages tries to identify these fragments. There are many different approaches for segmenting web pages. Ye and Chua [Ye 04] propose a segmentation based on structural differences between two pages (considering the same page at different times or different pages having similar structure), but this approach needs many samples to work accurately. *Document Object Model* (DOM) based methods as proposed by Debnath et al. [Debnath 05] suggest heuristic rules for dividing a page into blocks that also work in case of absence of tables by also considering other tags. For example, a heuristic that is applied is that header-tags are starts of segments. Ramaswamy et al. [Ramaswamy 04] use the shingling algorithm, which calculates a fingerprint of a block that only changes little if the content of the block changes little. Miloi [Miloi 05] evaluates the Levenshtein distance based on letter alteration operations and a simple distance based on word counts. Yi et al. [Yi 03b] use the internal tag structure of a block and do not consider the textual content. An approach with good results but high computational complexity is proposed in [Cai 03]. Features for this segmentation method are computed based on the visual representation of the page, similar to a human perceiving coherent blocks. A limitation of this approach is the fact that segments tend to be recognized in a very coarse granularity and it is difficult to define appropriate margin thresholds.

As different segmentation approaches have shown to have strengths and weaknesses, a combination of multiple segmentation processes seems to be a promis-

ing approach. Our approach combines ideas from pure DOM-based approaches, visual approaches, detection of re-occurring patterns and common class and ID attributes within a web page. An important goal of the novel approach is to work for a wide variety of web pages. Therefore it cannot depend on a certain design or structure of a web page. Nevertheless, the implementation is flexible enough to add specific algorithms for certain page types. Our approach for segmentation is shown in Figure 2.

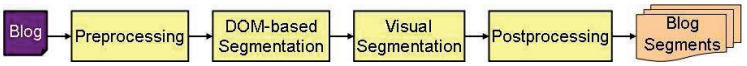

Figure 2: Segmentation Workflow

It is modularized into several steps with a pre- and a post-processing step wrapping the actual segmentation steps. The preprocessing step itself performs no segmentation, but the HTML source and the CSS source(s) are parsed and the DOM tree is built. In terms of CPU time, this is an expensive step but indispensable for visual segmentation. The DOM-based step uses block-level elements (i.e. elements that denote the start of a new visual block in the rendered HTML, e.g. div, table and ul) as basis and detects patterns in the web page. Repeating fragments in blogs are usually either blog posts or comments. The visual segmentation step uses information like position of an element in the layout or the background colors. The segmentation steps are applied in descending order by their reliability. The advantage of this order is that the latter segmentation steps can access and use the results of the previous steps. The post-processing step is used to remove segments that have been found but that are considered to be superfluous. For example, this is the case when two or more segments are nested and the inner segment group contains the same elements as the outer segment without adding textual content.

2.2 Segment Classification

There are numerous different approaches for detecting relevant content in web pages based on segmentation. For example, content-based approaches make use of the fact that noisy fragments usually share a common (tree) structure [Yi 03a]. Other approaches segment a collection of pages in "pagelets" and then try to identify common duplicates [Bar-Yossef 02] or use DOM-based heuristics [Gupta 03].

Our approach adopts methods for *document classification* to achieve our goal. Document classification can be made along different criteria like author or genre

(i.e., kind of document) [Santini 07, Scholl 09]. The goal of document classification is to automatically determine a category for a document using its properties as features. Machine learning algorithms in a supervised approach are a common method to achieve this. For detecting the genre of a segment, the segment is represented by the values of features that are relevant for distinguishing among genres. A classifier learns the characteristics of the different segment genres from a set of pre-classified examples called *training corpus* and guesses the genre of a given segment based on these values.

A web resource most often consists of several building blocks that are determined by its web genre. For example, a blog page has the building blocks *blog post* and *comment*, a forum consists of posts in a thread and an overview of all available topics or threads on the start page. Such blocks are called *segment genres*. Certain segment genres are common in regard to their occurrence in several web genres. For example, nearly all web resources exhibit some form of navigational elements (like menus) in order to access different parts of a web site. Similarly, headers usually impose an identity of the web resource's source onto the page in order to show a surfer where she is, including a logo and some tag phrase. Finally, footers often serve to present a copyright notice and secondary navigation like imprint and contact. These are not our targeted segment genres as they do not convey the content of a page but serve rather administrative and presentational purposes. Based on the works described in [Meyer zu Eissen 04], [Santini 07], and others, we extracted the features that showed to be most promising for our approach. As we are developing a language-agnostic approach for blogs we only considered (syntactical) features that can be extracted from the HTML documents' mark-up and structure. Features that we deemed appropriate were tag frequency, URL depth, punctuation frequencies, document length, ratio of plain text vs. mark-up and link ratios (outgoing vs. intra-site vs. intra-page links). Additionally to these features, we developed new features that we deemed profitable for classification of segments like structural features [Dominguez 08], coverage features (calculation how much of the content of the segment a HTML tag wraps) or URL-similarity features (similarity between all their hyperlink targets) leading to a total number of 215 features for training and testing a machine learning algorithm.

2.3 Topic Detection with Probabilistic Models

An emerging approach to topic detection are probabilistic topic models (PTMs), which tend to perform well for large text collections and lead to improvements in text retrieval tasks. PTMs detect topics as patterns of co-occurrences of statistically relevant words at a document collection level. PTMs view documents as a mixture of topics whereas each topic is a probability distribution over words. Therewith a document can be seen as the result of a generative probabilistic

procedure as follows: 1) Choose a distribution over topics; 2) For each word position in the document randomly chooses a topic from the topic distribution and draw a word from that topic. For a compact formal description of PTMs we refer to [Dietz 06].

The strengths of the PTM approach to topic detection that we leverage on for our application scenario of topic exploration are derived by intrinsically accounting for words' context. Natural language ambiguities are implicitly accounted for by the detected patterns of word co-occurrences; for example, different occurrences of a polysem (an ambiguous word) in different text fragments are associated to different topics due to the different words occurring in the same text. The actual meaning of an occurrence is (implicitly) denoted by the corresponding topic. Therewith we are able to fully automatically group blogs dealing with similar topics, without the large costs incurred by other approaches to semantic disambiguation. This facilitates the topic exploration considerably, as confirmed by our application scenario presented in Section 1.

3 Evaluation of our Approach

The evaluation of our approach consists of two steps: First, we show that it is feasible to detect relevant segments using machine learning methods and then we use our approach to improve the topic detection.

3.1 Experimental Setting and Results

In order to validate our classification approach, a corpus has been built and manually labelled with the relevant segment genres (*blogpost* and *comment*) and the genre *outliers* representing all other segments. For segment classification we applied J48, a decision tree learner. This machine learning algorithm generates an (un-balanced) binary decision tree based on the most significant feature values from the training data. J48 has shown to be robust, provide good results using few features and very time-efficient, which is especially important because classification is executed for each segment. Our corpus consists of 42 randomly selected blog pages (among those are 17 blog start pages and 25 blog post pages). After segmenting and manually labelling these pages, we obtained 194 blog posts, 46 blog comments and 867 other segments (in total 1107 segments).

blogpost	comment	other	← classified as
166	1	27	blogpost
0	37	9	comment
23	6	838	other
0.87	0.84	0.95	Precision
0.85	0.80	0.96	Recall
0.86	0.82	0.95	F–Measure
		94.03%	Accuracy

Table 1: Summarized results of testing blog segment corpus

Table 1 shows the result of testing the blog segment corpus using a J48 classifier with 10-fold cross-validation. The overall performance is very good with 94% correctly classified instances. Precision and recall rank about 80%-87% for the blog posts and comments classes. The used features were ranked using Information Gain [Mitchell 97]. Information Gain evaluates the significance of features. The tag frequencies and coverage of div, h3, p, images and hyperlinks were rated best. Additionally, especially the length of the mark-up and the plain text length are deemed significant for the segment genre "blog post". This is because blog posts are usually longer than arbitrary segments. Links to the same domain are meaningful as well for the segment "other", because menus contain a lot of those. A major source of misclassification (nearly all of the incorrectly classified segments) is the differentiation between blog posts and arbitrary segments. The same applies to the comment segments. We think that this is due to several reasons:

1. Blog posts mainly consist of the input of an author and allow nearly all of the HTML element subset that is available. Therefore, they tend to be very heterogeneously structured and written. The other category is pretty heterogeneous in itself, thus segments belonging to that category are easily misclassified. For example, consider a side bar of a blog containing a widget that is providing information about a blog. It will be longer than e.g., a menu and will expose similar properties as a short blog post in regards to length of text and paragraph structure.

2. In general, blogs have two different feedback mechanisms, one being comments. Additionally, most modern blog applications support trackbacks. Trackbacks are incoming references (hyperlinks) from other blog posts and are a way to relate to other blogs. In short, "When someone links to one of my posts, my post links back to them". In our blog segment corpus we labelled trackbacks as being a different class than comments, adding it to

the category "other". However, in some blog applications trackbacks are displayed in the same way as comments and are thus recognized by the classifier as such.

This confusion between the classes is clearly reflected in precision and recall. Nevertheless, the results are very encouraging and make it possible to apply this blog segment classification to real-world scenarios.

3.2 Applying our Approach to Topic Exploration

In this section we show that our approach can significantly improve results of Topic Exploration in blogs. We used automatic topic detection as discussed in Section 2.3 to get an overview of the various particular topics discussed in blogs obtained by retrieving the result pages returned by a popular blog search engine for the keywords "financial crisis" and "blame". This resulted in a number of 645 documents on which we run the previously mentioned topic detection algorithm. We experimented with different levels of granularity of the topics: we set the number of topics to be detected to 30, respectively 75. In each case, we first run the topic detection on the whole textual content of the HTML files, then re-run them only on the textual content of the blog segments as detected by the algorithm presented in Section 2. For the sake of conciseness we denote each setting by either the word `WPage` or `Segments`, depending on how the textual content has been obtained, followed by the number of topics used.

Four of the topics automatically detected for `WPage_30` are shown in Figure. 3. Each pie diagram denotes the words that were most frequently assigned to the corresponding topic. We note that (a) is mainly about *Freddie Mac* and *Fannie Mae* while (b) denotes a topic related with the financial crisis that has religious connotations. Other detected topics are less meaningful. For example, the month's names in (c) are due to taking into consideration blog archive segments, where previous blog entries are listed together with their publishing dates. We will henceforth call this kind of topics and the tokens associated with them *noise*. Another example of a noise segment is denoted by topic (d) which is due to considering segments which are part of the blog infrastructure rather than of the actual content of blog entries, as denoted by words such as *join* or *forum*.

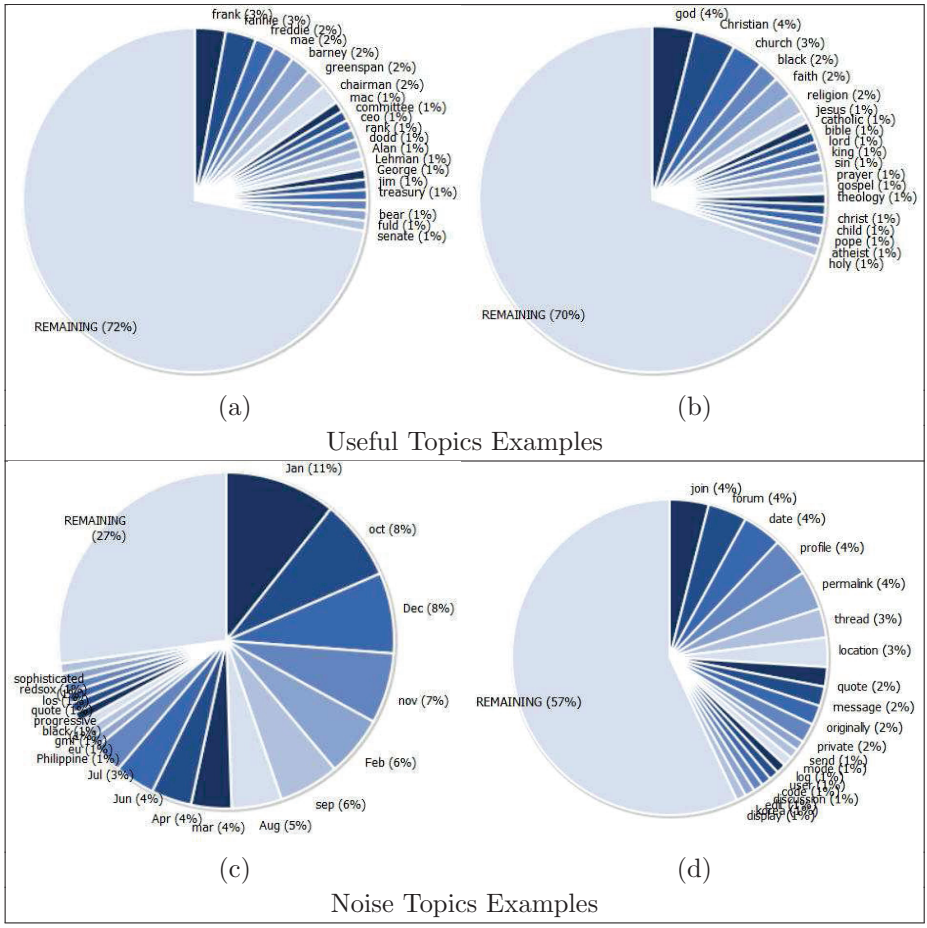

Figure 3: Some topics derived in `WPage_30`

The noise topics as defined above are exactly the type of noise information that can be filtered out by page segmentation and segment classification. We indeed found that our segmentation significantly improves the topic detection results as summarized in Table 2. For example, for `WPage_30`, we found 9 topics to be noise, whereas for `Segments_30` only 6 out of 30 topics were noise, leading thus to a relative improvement of 33%. Taken at the token level, the number of noise tokens was reduced by more than 37%. The good results are confirmed when considering a different number of topics as can be derived from the second half of Table 2. We obtained similar results using different corpora (searching for "financial crisis" or "Schwarzenegger") and different number of topics (im-

provements between 15% and 40%). Altogether, the results suggest that our segmentation approach leads to a significant improvement independently of the exact setting of the topic detection algorithm used. Generalizing, this validates our hypothesis that our segmentation tailored for blogs can significantly improve information retrieval tasks on the blogosphere.

Corpus	Noise topics(%)	Rel. Impr.(%)	Noise Tokens(%)	Rel. Impr(%)
WPage_30	30.00	n/a	24.40	n/a
Segments_30	20.00	33.33	15.20	37.70
WPage_75	17.33	n/a	17.51	n/a
Segments_75	14.67	15.35	13.45	23.19

Table 2: Summary of results

The reason, why still noise topics exist after segment detection is that it is very difficult to eliminate *all* irrelevant content. In particular some limitations of our approach are as follows: 1) An automatic segmentation tool is not perfect. Some segments are too big, others too small. We tuned our algorithm using our training corpus to obtain meaningful segments. 2) Our corpus for segment classification had 1100 segments, but we used only 42 blog pages as source. This is not representative for the diverse types of blogs. Still, we showed that even 1000 training segments can significantly improve Topic Exploration.

4 Conclusions and Future Work

In this paper we examined the possibilities of improving blogosphere exploration by filtering out irrelevant content. Therefore, we segment blogs in coherent page fragments and then classify the resulting segments. Our main contributions in this paper consist of developing this approach and providing a proof-of-concept evaluation. We showed that automatically differentiating between these segment genres is feasible and explained the limitations of our approach. Still, there is much space for improvement: segmentation can be enhanced by using heuristics, for example taking into account that each blog post page has exactly one post. Our classification algorithm can be improved by building a more heterogeneous and larger corpus. However, we showed that even a small training corpus can be used to substantially improve a search query.

The research presented here provides a foundation on which further applications and research (e.g., in the field of Community Mining [Berlea 09]) can be based. Generally, our approach enables to handle blogs in a more fine-granular

manner, i.e., separating and targeting the parts of a blog directly. Hence, examination of semantic information in segments (e.g., name of the author or creation date) could be enhanced by restricting the analysis only on the respective segments. In information retrieval, genre-based queries are possible with availability of the segment structure, e.g., "show me all blog posts containing Java Code", thus filtering all boilerplate template contents that are irrelevant to the actual query.

Acknowledgements
The results presented here have been obtained in projects funded by means of the German Federal Ministry of Economy and Technology under the promotional reference "01MQ07012" for the second author and by a research grant from SAP for the remaining authors.

References

[Bar-Yossef 02] Bar-Yossef, Z. and Rajagopalan, S.: "Template Detection via Data Mining and its Applications". Proc. of 11^{th} World Wide Web Conf. (2002).

[Berlea 09] Berlea, A., Döhring, M. and Reuschling, N.: "Content and Communication based Subcommunity Detection using Probabilistic Topic Models". In International Conference on Intelligent Systems and Agents ISA, 2009.

[Cai 03] Cai, D., Yu, S., Wen, J.R. and Ma, W.Y.: "VIPS – a Vision-based Page Segmentation Algorithm". Technical Report, Microsoft Research (2003).

[Debnath 05] Debnath, S., Mitra, P., Pal, N. and Giles, C. L.: "Automatic Identification of Informative Sections of Web Pages". In: IEEE Transactions on Knowledge and Data Engineering, IEEE Computer Society, 17, 1233-1246 (2005).

[Dietz 06] Dietz, L. and Stewart, A.: "Utilize probabilistic topic models to enrich knowledge base" In Proc. of the ESWC 2006 Workshop on Mastering the Gap: From Information Extraction to Semantic Representation (2006).

[Dominguez 08] Dominguez Garcia, R., Scholl, P., Böhnstedt, D., Rensing, C. and Steinmetz, R.: "Automatic Web Genre Classification using Structural Features". Technical Report KOM-TR-2008-06) Multimedia Kommunikation – Technische Universität Darmstadt (2008).

[Gupta 03] Gupta, S. et al.: "DOM-based Content Extraction of HTML Documents". Proc. of the 12^{th} World Wide Web Conf. (2003).

[Meyer zu Eissen 04] Meyer zu Eissen, S. M. and Stein, B.: "Genre Classification of Web Pages – User Study and Feasibility Analysis" In KI 2004: Advances in Artificial Intelligence, Springer Berlin / Heidelberg, 3238, 256-269 (2004).

[Miloi 05] Miloi, T. A. "Ähnlichkeitsbasierte Vorverarbeitung von Webseiten". (Bachelor Thesis). Friedrich-Alexander-Universität Erlangen-Nürnberg, 2005.

[Mitchell 97] Mitchell, T. M. "Machine Learning". THe Mc-Graw-Hill Companies, 1997.

[Ramaswamy 04] Ramaswamy, L., Arun, I., Liu, L. and Douglis, F. "Automatic detection of fragments in dynamically generated web pages". In: Proceedings of the 13th international conference on World Wide Web, pp. 443-454, 2004.

[Santini 07] Santini, M. "Automatic Identification of Genre in Web Pages"(PhD Thesis). University of Brighton (2007)

[Scholl 09] Scholl, P., Domínguez García, R., Böhnstedt, D., Rensing, C. and Steinmetz, R.: "Towards Language-Independent Web Genre Detection". In: Proceedings of the WWW 2009 (2009).

[Ulicny 08] Ulicny, B.: "Modeling Malaysian Public Opinion by Mining the Malaysian Blogosphere"; Social Computing, Behavioral Modeling, and Prediction, Springer, New York (2008).

[Ye 04] Ye, S. and Chua, T. "Detecting and Partitioning Data Objects in Complex Web Pages". In: WI '04: Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence, IEEE Computer Society, 669-672 (2004).

[Yi 03a] Yi, L. and Liu, B. "Web Page Cleaning for Web Mining through Feature Weighting". Proceeding of the 18^{th} International Joint Conferenece on Artificial Intelligence, (2003).

[Yi 03b] Yi, L., Liu, B. and Li, X. "Eliminating noisy information in Web pages for data mining". In: KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM Press, pp. 296-305, 2003.