TextSweeper - A System for Content Extraction and Overview Page Detection

Heinz-Peter Lang
MODUL University Vienna
heinz-peter.lang@modul.ac.at

Gerhard Wohlgenannt
Vienna University of Economics and Business
gerhard.wohlgenannt@wu.ac.at

Albert Weichselbraun
University of Applied Sciences Chur
albert.weichselbraun@htwchur.ch

Abstract
Web pages not only contain *main content*, but also other elements such as navigation panels, advertisements and links to related documents. Furthermore, overview pages (summarization pages and entry points) duplicate and aggregate parts of articles and thereby create redundancies. The noise elements in Web pages as well as overview pages affect the performance of downstream processes such as Web-based Information Retrieval. Context Extraction's task is identifying and extracting the main content from a Web page.
In this research-in-progress paper we present an approach which not only identifies and extracts the main content, but also detects overview pages and thereby allows skipping them. The content extraction part of the system is an extension of existing Text-to-Tag ratio methods, overview page detection is accomplished with the *net text length* heuristic. Preliminary results and ad-hoc evaluation indicate a promising system performance. A formal evaluation and comparison to other state-of-the-art approaches is part of future work.

Keywords
content extraction, overview pages, text filtering, natural language processing, contextualized information spaces, Web-based information retrieval

1 Introduction

Besides blocks of relevant content, Web documents also include noise elements such as navigation menus, links to related documents, advertisements and copyright notices that have the potential to considerably reduce the performance of subsequent document processing steps. Gibson et al. (2005) estimate that only about 40–50% of Web data are related to main content. Automatic document processing for tasks such as Web Mining, Web-based Information Retrieval, Natural Language Processing, or consumption of Web pages on mobile devices require automated identification and extraction of the relevant content as a precondition for reliable results.

The presented approach is part of the media monitoring and Web intelligence platform webLyzard (Scharl et al., 2006) and reduces noise by (i) extracting relevant content from Web pages, and (ii) detecting *overview pages*.

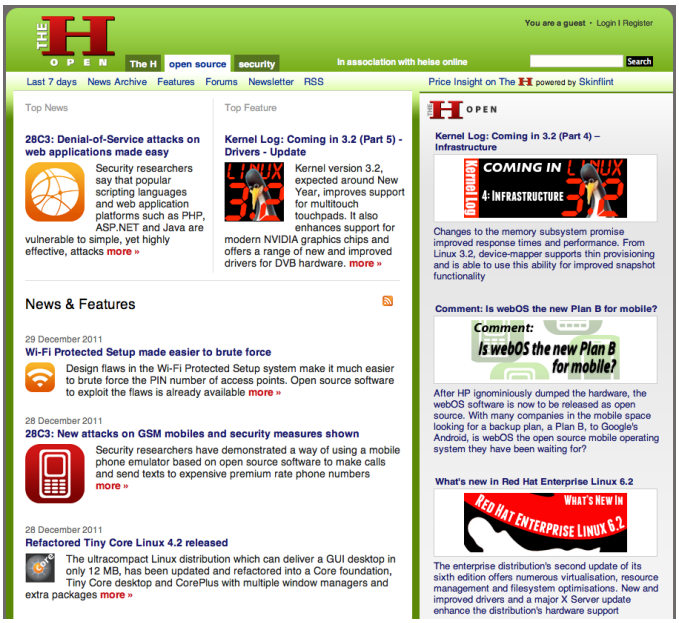

Fig. 1. Example of an overview page (h-online.com/open)

Overview pages (for an example see Figure 1) act as entry points and integrate a number of heterogeneous articles, usually with teasers of article text and “Read more” links to the full articles. All applications that perform tasks such as Web-based information retrieval, or which generate domain specific corpora or portals based on the Web content, benefit from detecting and skipping such pages as they lead to redundancy and biased results in downstream processes. It is hard to detect and eliminate overview pages ex-post with document similarity metrics as overview pages are a mixture of (parts of) a number of documents.

We propose a pre-processing framework that applies an enhanced Text-to-Tag ratio method which takes the text length within links into account in the content extraction process. The system also facilitates the detection and rejection of overview pages based on *net text length* of most the relevant text blocks. To our knowledge there exist no content extraction systems which address the task of overview page detection.

The remainder of this paper is organized as follows: Section 2 briefly reviews related work, Section 3 describes the proposed method for content extraction and the detection of overview pages, and Section 4 concludes the paper with a summarization, the roundup of the main contributions, and future work.

2 Related Work

The term *content extraction* was first used by Rahman et al. (2001), some authors also refer to *boilerplate detection* (Kohlschütter et al., 2010) or Web page segmentation (Kohlschütter & Nejdl, 2008; Yu et al., 2003).
Vision-based approaches (e.g. Yu et al., 2003) rely on partially rendering Web pages to identify visually grouped blocks. Song et al. (2004) apply machine learning to detect important blocks with the help of spatial (e.g. visual location) and content features.
Densiometric methods use lower level properties of text, especially the number of tokens in text fragments. Kohlschütter and Nejdl (2008) utilize text-density derived from vision-based methods to detect text segments. Sun et al. (2011) present a method that uses DOM node text density and preserves the original structure. For languages not based on the Latin alphabet the density of non-ASCII versus ASCII characters helps to identify the main content (Mohammadzadeh et al., 2011). Finn et al. (2001) introduced *statistical approaches*, they extracted the continuous region with the highest ratio of words vs. HTML tags. Many content extraction methods rely on DOM-level features applied in handcrafted-rules or by trained classifiers (Kohlschütter et al., 2010). Chen et al. (2001) try to understand the function of DOM objects in Web pages and intention of Web site authors.
Weninger et al. (2010)'s method of context extraction via Text-to-Tag ratios (which is based on Weninger and Hsu (2008)) computes tag ratios in a line by line fashion and clusters the resulting histograms into content and noise areas. Before clustering they apply smoothing on the histograms in order not to lose content-lines such as the page title. Kohlschütter et al. (2010) present a low-cost but effective boilerplate detection model based on shallow text features, such as *number of words* and *link density*.
Some authors try to learn the common structure of pages stemming from a Web page collection (template detection), e.g. Yi et al. (2003) remove unrelated content with the help of a *site style tree*. Such methods remove identical parts from Web pages, therefore the model requires to be built for each Website.

3 Method

Based on ideas and concepts from related work, we developed *TextSweeper*: An HTML pre-processing component which extracts the main content of a Web page by applying a modified version of the Text-to-Tag ratio and *link density*. Figure 2 presents an activity diagram of *TextSweeper*'s architecture. The figure neglects error handling to keep the model simple.
In the first step, *TextSweeper* parses the plain HTML content and removes known non-content nodes from the generated document object model (DOM) tree. Such nodes are for instance comments in the source code, which adulterate the calculation of the Text-to-Tag ratio, or containers, which never contain any content (like <form> or <script>).
Afterwards the component analyses the parsed content, detects the text in every node and calculates the per node text length. Equation 1 shows that *TextSweeper* is not limited to the number of characters, but also considers a text weight in its calculation. This weight reflects the likelihood of a particular node to be relevant content, it is computed

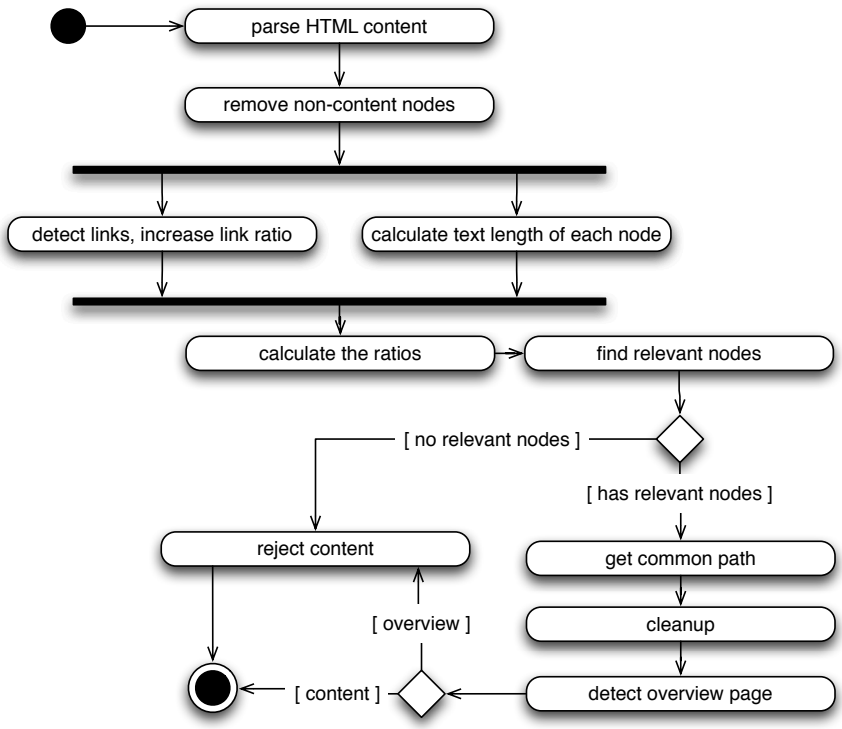

Fig. 2. Activity diagram of the *TextSweeper* HTML pre-processing component

based on impact of the surrounding layout tags (for instance, headings and bold text are considered more important than other elements) and the text length (the component has a bias towards long text passages because they are a good indicator for an article's content).

$$text\_length = actual\_text\_length * text\_weight * parent\_text\_weight \quad (1)$$

In addition to the plain text length *TextSweeper* also checks if the element contains a link. At this step the system does not only rely on the occurrence of an anchor tag, but also takes attributes such as "onclick" and the actual text into account. The text in overview pages very often only presents a summary of an article and ends with a phrase like "Read more" or "...". If *TextSweeper* finds one of these indicators, it considers the element as well as its parent as a link element.

In the next step the system calculates a modified version of the Text-to-Tag ratio (Weninger & Hsu, 2008). Equation 2 shows the calculation of this ratio: In contrast to Weninger and Hsu (2008) our method uses the *net text length* (plain text minus link text) and divides them by the number of tags, ignoring inline text tags, and the level of the node.

$$ratio = \frac{text\_length - link\_text\_length}{tag\_count + node\_level} \quad (2)$$

While calculating the ratio, the *TextSweeper* system also detects the node with the highest Text-to-Tag ratio which is considered to contain most of the main content. Manual evaluations indicate that this node alone does not always include all the relevant content (main content) of a Web page. Therefore, we adopted a simple smoothing approach (see also Weninger et al. (2010)) to return all nodes for which their ratio is within a range – this range is set by a predefined tolerance ratio. For example, if the tolerance ratio is set to 0.9 and the node with the highest Text-to-Tag ratio has a value of 1,000, then all nodes with a ratio greater than 900 will be considered relevant. For all relevant nodes *TextSweeper* detects their common path. This approach reduces the risk of loosing relevant content, although it increases the likelihood of including noise elements.
To remove smaller blocks of unrelated content, such as advertisements, *TextSweeper* performs a cleanup step to delete all nodes which fall below a predefined minimum text length or which mainly contain links. Finally, in order to detect overview pages, *TextSweeper* examines the *net text length* of the returned node and rejects all documents which are below a given minimum threshold.

4 Outlook and Conclusions

In this article we introduced *TextSweeper*, a framework which extracts the main content from Web pages. *TextSweeper* uses a modified version of the Text-to-Tag (text length versus the number of tags) ratio relying on the *net text length* (plain text length minus link text length) of particular DOM elements to remove noise from Web pages. In contrast to related work (see Chapter 2), *TextSweeper* also supports the detection and removal of overview pages.
The main contributions of this work are the presentation of (i) a modified and enhanced version of the Text-to-Tag ratio method that is enhanced by considering the link density and (ii) a simple, efficient and unsupervised approach for the detection and elimination of overview pages. It considerably improves the performance of downstream processes such as Web Information Retrieval.
As this is a research in progress paper, an essential part of future work will lie in extensive evaluation of the framework's performance in terms of precision and recall. Furthermore, we shall test the framework on the CleanEval benchmark and provide a performance comparison to competitive systems.
Currently, we use a very simple approach of smoothing Text-to-Tag ratios over neighboring DOM elements. This method will benefit from more sophisticated techniques that will be implemented in our future work.

References

Chen, J., Zhou, B., Shi, J., Zhang, H., & Fengwu, Q. (2001). Function-based object model towards website adaptation. In *Www '01: Proceedings of the 10th international conference on world wide web* (pp. 587–596). New York, NY, USA: ACM Press.

Finn, A., Kushmerick, N., & Smyth, B. (2001). Fact or fiction: Content classification for digital libraries. In *Delos workshop: Personalisation and recommender systems in digital libraries.*

Gibson, D., Punera, K., & Tomkins, A. (2005). The volume and evolution of web page templates. In *Special interest tracks and posters of the 14th international conference on world wide web (www2005)* (p. 830-839).

Kohlschütter, C., Fankhauser, P., & Nejdl, W. (2010). Boilerplate detection using shallow text features. In B. D. Davison, T. Suel, N. Craswell, & B. Liu (Eds.), *Wsdm* (p. 441-450). ACM.

Kohlschütter, C., & Nejdl, W. (2008). A densitometric approach to web page segmentation. In J. G. Shanahan et al. (Eds.), *Cikm* (p. 1173-1182). ACM.

Mohammadzadeh, H., Gottron, T., Schweiggert, F., & Nakhaeizadeh, G. (2011). A fast and accurate approach for main content extraction based on character encoding. In *Tir'11: Proccedings of the 8th workshop on text-based information retrieval.*

Rahman, A. F. R., Alam, H., & Hartono, R. (2001). Content extraction from html documents. In *In 1st int. workshop on web document analysis (wda2001)* (pp. 7–10).

Scharl, A., Weichselbraun, A., & Liu, W. (2006, January). An ontology-based architecture for tracking information across interactive electronic environments. In *Proceedings of the 39th hawaii international conference on system sciences (hicss-39).* Kauai, Hawaii: IEEE Computer Society Press.

Song, R., Liu, H., Wen, J.-R., & Ma, W.-Y. (2004). Learning block importance models for web pages. In *Www '04: Proceedings of the 13th international conference on world wide web* (pp. 203–211). New York, NY, USA: ACM Press.

Sun, F., Song, D., & Liao, L. (2011). Dom based content extraction via text density. In *Proceedings of the 34th international acm sigir conference on research and development in information retrieval* (pp. 245–254). New York, NY, USA: ACM. Available from `http://doi.acm.org/10.1145/2009916.2009952`

Weninger, T., & Hsu, W. H. (2008). Text extraction from the web via text-to-tag ratio. In *Dexa workshops* (p. 23-28). IEEE Computer Society.

Weninger, T., Hsu, W. H., & Han, J. (2010). Cetr: content extraction via tag ratios. In M. Rappa, P. Jones, J. Freire, & S. Chakrabarti (Eds.), *Www* (p. 971-980). ACM.

Yi, L., Liu, B., & Li, X. (2003). 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* (pp. 296–305). New York, NY, USA: ACM Press.

Yu, S., Cai, D., Wen, J.-R., & Ma, W.-Y. (2003). Improving pseudo-relevance feedback in web information retrieval using web page segmentation. In *Www '03: Proceedings of the 12th international conference on world wide web* (pp. 11–18). New York, NY, USA: ACM Press.