

Page Segmentation by Web Content Clustering

Sadet Alciç, Stefan Conrad
Institute of Computer Science
Heinrich-Heine-University of Duesseldorf
D-40225 Duesseldorf, Germany
{alcic, conrad}@cs.uni-duesseldorf.de

ABSTRACT

Web page segmentation is an important task with benefits for a variety of applications, reaching from data extraction to accessibility improvement. Focusing on the smallest content units of a web page, page segmentation can be reduced to a clustering of web contents to structural and semantical cohesive groups. To investigate the web page segmentation task from the clustering point of view, we define three possible distance measures for content units based on their DOM, geometric and semantic properties. We combine these distance measures with common clustering techniques and evaluate the web page segmentation accuracy on a labelled collection by applying widely used validity measures.

Categories and Subject Descriptors

H.3 [Information Storage and Retrieval]: Content Analysis and Indexing; H.5.4 [Information Interfaces and Presentation]: Hypertext/Hypermedia

General Terms

web page segmentation

Keywords

web content clustering, web content distance

1. INTRODUCTION

In the recent years, the World Wide Web has expanded to the largest information repository with billions of documents. In order to benefit from this information abundance, popular search engines use indexing techniques borrowed from information retrieval and in this way make the information accessible to users.

Traditional information retrieval assumes a document as the basic information unit and extracts keywords from the whole content when creating an index. However, web documents often consist of differing parts (blocks) like a main

content, navigation bars, headers, footers and advertisements and each of them might additionally include multiple contents. This results in inaccurate indexes and hence to unrequested documents in high rankings. Breaking a web page into its principal parts and indexing only the main content can highly improve the indexing performance [22, 4].

Besides document retrieval, there is a variety of other web applications that would benefit of web page partitioning: annotations for web images are more precise if they are extracted from the paragraph the image belongs to [3, 1]; web pages that are broken into logical parts can be better displayed on small-screen devices [2, 6, 10]; duplicate contents appearing in a document or among different documents can be better detected [5, 12]; web page accessibility can be improved by a clear document structure and clutter elimination [13].

Recognizing the importance of web page partitioning, many researchers have proposed solutions to this problem [4, 10, 16, 11, 6, 14]. Most of these works apply heuristic rules that decide whether to break a segment at a specific position or not. A heuristic rule typically employs structural features of web contents as well as geometric properties gathered by rendering the contents in a browser application. In this way, each rule covers a specific design pattern and proposes an appropriate solution. However, the design patterns used to present contents on the Web are unlimited and thus cannot be covered by a finite set of rules. Further, while most of the existing approaches use visual and structural content features, (to our knowledge) there are hardly attempts to integrate semantical properties in partitioning task. Moreover, while we know that structural and geometric features are useful for web page partitioning, there is neither evaluation nor comparison of the features on their principal usability for this task.

Our contribution. In this work, we fill these gaps by presenting a more general approach to web page partitioning. In particular, our contributions are as follows: (1) We first indicate that a web page consists of many atomic content units (web contents) which can be grouped to page blocks. Thus the page segmentation problem is reduced to clustering of web contents. (2) To be able to cluster web contents, it is inevitable to define suitable cost functions that estimate the commonness of two web contents. We present three different distance measures that compute a DOM-based, a geometric and a semantic-based dissimilarity of web contents. (3) We further adapt common clustering algorithms to fit the special requirements of web contents and distance measures. (4) Finally, we evaluate different com-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

WIMS'11 May 25-27, 2011 Sogndal, Norway

Copyright 2011 ACM 978-1-4503-0148-0/11/05 ...\$10.00.

binations of our proposed distance measures with clustering algorithms using common metrics for measuring clustering validity and discuss the results.

Organization. This paper is organized as follows: Section 2 will give a brief overview to related work. In Section 3, we formulate the page segmentation problem as a clustering task. Then we introduce three distance measures for web contents based on different features and describe the clustering approaches that were employed in our experiments. Section 5 summarizes the evaluation studies and discusses the obtained results. The paper is concluded and a short overview to future work is given in Section 6.

2. RELATED WORK

In recent years, a variety of approaches to web page partitioning has been proposed. Related work can be roughly classified into top-down and bottom-up methods.

Top-down page segmentation. Top-down approaches start with the complete web page as a block and partition this block iteratively into smaller blocks using different features obtained from the content of the web page. The Vision-based Page Segmentation algorithm proposed by Cai et al. [4] computes for each block a *Degree-of-Coherence* (DoC) utilizing heuristic rules based on DOM as well as visual features. The DoC ranges from 1 to 10, while 10 identifies maximum coherence. A manually specified threshold value for DoC considers whether a particular block has to be partitioned. Partitioning decision is made by DOM and vision-based rules. Hattori et al. [10] combine two different methods for page segmentation. In the first method, a content-distance based on the order of HTML tags is defined and the initial block is iteratively separated at positions where the content-distance exceeds a dynamically estimated threshold. The second method applies heuristic rules that are based on HTML tags. Lin and Ho [16] propose a simple partitioning method based on HTML table structures. Afterwards they compute a *content entropy* to estimate the importance of each block. Kao et al. [11] define a term entropy, too. Blocks of DOM subtrees are then separated based on the entropies of the contained terms. Vineel [21] defines a content size and an entropy value that measures the strength of local patterns within the subtree of a node. Threshold values are defined for both measures to perform page segmentation. Chen et al. [6] distinguish five high-level blocks, namely, header, footer, left sidebar, right sidebar and main content. Web contents are classified into one of these blocks using heuristics on DOM as well as geometric level. Baluja’s segmentation system [2] divides a web page into nine segments using a decision-tree which employs an *information gain* measure and geometric features. While top-down approaches to page segmentation are justifiable, most of the approaches employ heuristic rules to estimate appropriate block structures, which – as we mentioned in the introduction – are restricted to a small subset of pages on the Web and cannot handle the variety of web pages in principal. The idea of a content-distance proposed by Hattori is quite similar to our DOM-distance. But the formalization of the content-distance is not as evident as our DOM-distance.

Bottom-up page segmentation. In the bottom-up approach to web page segmentation mostly the (leaf) nodes of the DOM representation are taken as atomic content units.

In the next step, these units are grouped into segments. Li et al. [14] propose improving web search quality by segmenting web pages into cohesive *micro-units*. The segmentation procedure involves creating a *tag tree* which is similar to DOM and then applying two heuristics to aggregate tree nodes into segments. In particular the heuristics are: merge headings with the following content, and merge adjacent text paragraphs. Chakrabarti et al. [5] approached the page segmentation problem from a graph-theoretic point of view. DOM nodes are supposed to be nodes in a complete weighted graph and the edge weights estimate the costs needed to put the connected nodes in one block. Thus the partitioning task could be reduced to an well known optimization problem. Kohlschuetter et al. [12] represent atomic content units of a web page by a quantitative linguistic measure of *text-density* and reduce the segmentation problem to solving a 1D-partitioning task. An iterative block fusion algorithm applying methods adapted from computer vision is presented as a solution. While Li et al.’s algorithm [14] relies on heuristics rules that we declared as insufficient due to reasonable limitations, Kohlschuetter et al.’s as well as Chakrabarti et al.’s approaches can handle a broader set of documents. However, the graph-based approach needs test data to compute edge weights that might be expensive in creation, and the densiometric-based approach is limited to textual contents while images are not considered. Nonetheless, the last two approaches are most similar to our clustering approach.

3. CLUSTERING WEB CONTENTS

In this work, we use the term *web content* to refer to the smallest indivisible content of a web page which is usually represented as a leaf node in the DOM tree. Further, by the term *block* we define a group of web contents that share a coherent topic, style and/or structure. With this terminology, the web page segmentation can be described as a grouping of web contents to blocks. As already mentioned in the introduction, this problem formulation is closely related to the well known problem of clustering objects with similar properties into groups (or clusters). But how can we decide if two web contents are similar or not?

Web contents own different properties that represent them in a different context. First, they are a part of a complex DOM tree that can be directly derived from the HTML of their hosting page. This tree partially describes the hierarchical structure in which the contents are placed. Secondly, once the contents are rendered in a browser application, they possess geometrical properties like an exact position in the output panel and horizontal and vertical dimensions. The third representation of web contents is based on the included contents and their semantics. While this representation can be easily extracted from text nodes, we can extract the alternative text, the title text or at least the url for images and other objects.

Recognizing these different features of web contents, we present corresponding distance measures that are able to estimate a dissimilarity based on different criteria. These distance measures can then be employed with common clustering techniques to find appropriate page blocks.

3.1 Distances between Web Contents

In this section, we will introduce three distance measures for web contents that are based on different content repre-

sentations.

3.1.1 DOM-based Distance

In the DOM tree of a web document the web contents are the leaves that are grouped by tags which are the inner nodes of the tree. Our aim is to provide a distance measure that will exactly map this tree structure into one distance value. We make following preconditions:

- Adjacent sibling leaf nodes have all the same distance.
- The minimal distance of leaves belonging to different parents must be greater than the maximum distance of these leafs to their siblings.

A formalization of a distance function that satisfies these preconditions is given below.

Definition: DOM-based distance. Let d be a web document and $D_d(N, E)$ the corresponding DOM tree, with N as a set of nodes and E as a set of edges. Further let $P = (p_1, \dots, p_n)$ be the sequence of all nodes in N ordered by traversing D_d in preorder traversal. The index $i \in \{1, \dots, n = |N|\}$ gives the order of $p_i \in N$ in the sequence P . Based on this formulation, we define the *DOM Distance* $d : N \times N \rightarrow \mathbb{R}$ for two nodes $p_a, p_b \in N, 1 < a < b < n$ (if $b < a$, wlog. switch p_a, p_b) as follows:

$$d(p_a, p_b) = \sum_{i=a+1}^b w_{p_i}, \quad (1)$$

where w_{p_i} is the *block weight* of element p_i , which has to be further specified. In general, the metric d is a sum of block weights.

The block weights w_{p_i} can be explained as the cost needed to reach the content block of p_i from its preorder predecessor p_{i-1} . Therefore w_{p_i} corresponds to the distance $d(p_{i-1}, p_i)$ of two neighbored contents p_{i-1}, p_i . For nodes on the same tree level l , the block weights are all equal. On a lower level $l-1$ the block weight has to be at least greater than the maximal distance of two sibling nodes at level l . The maximal distance of two sibling nodes on level l corresponds to the degree of the nodes on level $l-1$ which is the maximal count of children of the nodes at level l .

With these considerations, the block weight function $w : N \rightarrow \mathbb{R}$ can be formulated by following recurrence:

$$w(l) = \begin{cases} c & : d_l = 0 \\ d_l \cdot w(l+1) & : d_l > 0, \end{cases} \quad (2)$$

where d_l refers to the maximal degree of nodes at level l and $c > 0$ is the smallest distance between two leafs on the deepest level (can be set to 1 and does not affect the behaviour of the measure). To apply w in Equation 1, we define function $l : N \rightarrow \mathbb{N}$ that delivers the tree level of a node. Thus the block weight w_p for a node $p \in N$ can be expressed by $w_p = w(l(p))$.

For clearance, consider the following example DOM tree in Figure 1.

To compute the distance of node 4 to node 8, we have to sum up all weights of the preorder successors of node 4 until node 8 is reached. This can be written as $d(4, 8) = w_5 + w_3 + w_6 + w_7 + w_8 = 1 + 3 + 1 + 1 + 1 = 7$

3.1.2 Geometric Distance

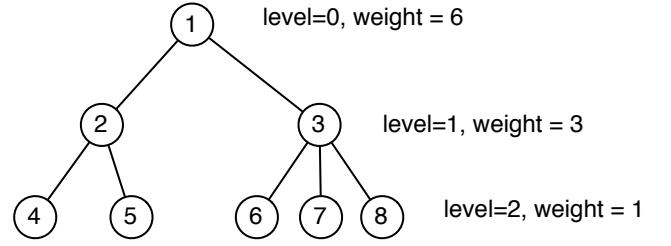


Figure 1: A simplified example of DOM tree. Weights are computed using formula in Equation 2 with $c=1$.

In the rendered output of a web document, contents are presented to web users in a two dimensional plane. In particular, each content is placed within a bounding rectangle with specified *width* and *height* and further the rectangle has a *position* within the general browser plane. In the rest of this paper, the bounding rectangle of a web content a will be represented by two points, referring to its two diagonal edges, e.g., $[(left, top), (right, bottom)]$. There are different ways to define a distance between rectangles. We can compute the minimal or the maximal distance between the rectangles, or even the distance between the center points of the rectangles. In this work, the minimal distance has been considered as the geometric distance.

Definition: Geometric Distance. Let R and S be two web contents and $r = [(r_x, r_y), (r_x', r_y')]$ be the bounding rectangle of R and respectively $s = [(s_x, s_y), (s_x', s_y')]$ the bounding rectangle of S . The visual distance of two contents R and S can now be computed as the minimum distance between the corresponding rectangles $r = [(r_x, r_y), (r_x', r_y')]$ and $s = [(s_x, s_y), (s_x', s_y')]$. In general, this distance can be defined as

$$mindist(R, S) = \sum_{i \in x, y} t_i^2$$

where

$$t_i = \begin{cases} r_i - s_i' & \text{if } r_i > s_i' \\ s_i - r_i' & \text{if } r_i' < s_i \\ 0 & \text{if otherwise.} \end{cases} \quad (3)$$

In Figure 2 we show an example with two rectangles and the minimum distance between the rectangles.

3.1.3 Semantic Distance

Web contents can be of different media types as text, images, or flash objects. To estimate the semantic distance (or dissimilarity) of web contents, it is necessary to determine a common representation for the different media types. Due to the fact that text is the most used media object in web documents and furthermore there already exist approved measures to compare the similarity of text contents, it is obvious to choose text as common representation. The other media types can be represented by their alternative or title text, or – if such not available – by their url string. Based on the textual representation, we can apply common metrics to compute a semantic distance of contents.

There is a variety of similarity metrics for texts in the literature. The simpler methods adopted from information retrieval rely on some kind of lexical matching between units of one text to the others. While these approaches can be

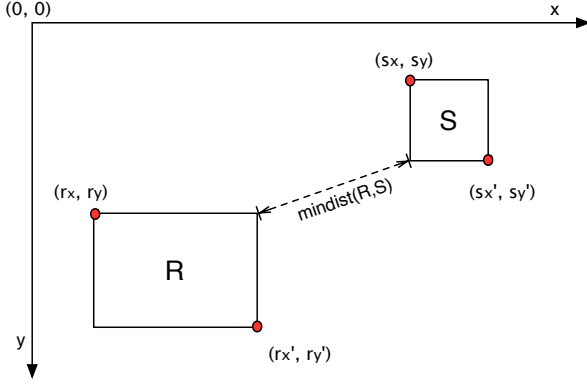


Figure 2: The (minimal) geometric distance between two rects R and S .

successful to a certain degree, they fail to identify the *semantic* similarity of texts. For example the synonyms *Plane* and *Aircraft* have a high semantic correlation which cannot be detected without background knowledge. To overcome these limitations a variety of more sophisticated metrics were proposed in the literature. Instead of estimating a word-to-word lexical matching degree, the words are mapped to corresponding concepts in a knowledge base and a concept-to-concept accordance is computed. An overview of different semantic metrics which uses the WordNet taxonomy [17] as background knowledge is given in Patwardhan et al. [18]. We decided to apply the similarity metric introduced by Lin [15] to compute the accordance of concepts due to its eligible performance and low computational costs. The definition of the similarity function is as follows:

$$\text{sim}_{\text{Lin}} = \frac{2 \cdot \text{IC}(\text{LCS})}{\text{IC}(\text{concept}_1) + \text{IC}(\text{concept}_2)} \quad (4)$$

where LCS is the least common subsumer of the two concepts in the WordNet taxonomy, and IC returns the information content [19] which is defined as:

$$\text{IC}(c) = -\log P(c)$$

and $P(c)$ is the probability of encountering an instance of concept c in a large corpus.

This metric computes the semantic similarity on the concept level and should now be extended to a text paragraph level. But before this can be done, some preprocessing is needed to bring the paragraph in the right format. We first apply a sentence detector followed by a word level tokenizer and finally do part of speech (POS) tagging to identify the word class for each word using the Open NLP Library. Afterwards, the terms are stemmed using the WordNet stemmer and mapped to suitable WordNet concepts if possible. The text similarity can now be computed as proposed by Coley et al. [7]. Their approach is as follows: for each noun (verb) in the set of nouns (verbs) belonging to one text, they identify the noun (verb) in the other text with the highest semantic similarity (maxSim) according to the concept similarity sim_{Lin} . For the other word classes a lexical matching to their counterparts in the other text is performed. The resulting similarity function between two texts T_1, T_2 is de-

fined as follows:

$$\text{sim}(T_1, T_2)_{T_1} = \frac{\sum_{w_i \in T_1} \text{maxSim}(w_i, T_2) \cdot \text{idf}(w_i)}{\sum_{w_i \in T_1} \text{idf}(w_i)} \quad (5)$$

with $\text{idf}(w_i)$ identifying the inverse document frequency of the word w_i in a large corpus. Sparck-Jones introduced the *idf* in [20] to estimate the *specificity* of a term. With the defined similarity metric in Equation 5 a directional similarity score is computed with respect to T_1 . The score from both directions can be combined into a bidirectional similarity as proposed in [7]:

$$\text{sim}(T_1, T_2) = \text{sim}(T_1, T_2)_{T_1} \cdot \text{sim}(T_1, T_2)_{T_2} \quad (6)$$

This similarity score has a value between 0 and 1, and thus easily can be converted to a normalized distance function:

$$\text{dist}_{\text{sem}}(T_1, T_2) = 1 - \text{sim}(T_1, T_2) \quad (7)$$

4. CLUSTERING METHODS

Cluster analysis is a primary method for discovering groups of similar objects in various data repositories. In this section, we will briefly describe three basic clustering approaches that can be applied to group web contents.

4.1 Partitioning Clustering

Partitioning clustering is a classical approach to data clustering. The most popular representatives are k-means and k-medoid. The k-means algorithm starts by randomly partitioning the data points to k clusters and computing the initial cluster centroids. In the following step each data point is assigned to its next centroid and afterwards the centroids are recomputed. This step is repeated until the centroids do not change in a run. Since web contents are non numerical, the cluster centroid should be represented by a web content which has the minimum distance to all other contents of the cluster. This variant of partitioning clustering techniques is named *k-medoid*.

One critical issue with partitioning clustering is the parameter k , since the number of clusters varies among different web pages. Nevertheless, to estimate a good performing k , the algorithm can be run several times with different k and clustering validity can be measured by computing the *sum of squared errors* or the *silhouette coefficient*. In our experiments, we used labelled web pages. As a consequence the number of blocks (or clusters) was implicitly given and did not need further estimation.

Partitioning clustering can be run with all proposed distance functions with no need to be adapted to the properties of the functions.

4.2 Hierarchical agglomerative clustering

Unlike the partitioning methods, hierarchical clustering approaches do not simply separate the data into clusters, but they generate a hierarchical representation of the data which then can be used to derive a clustering. The clustering result is given in a dendrogram, a tree-like diagram which displays both the cluster-subcluster relationships and the order in which the clusters were merged.

The basic algorithm to compute the dendrogram is as follows: starting with individual web contents as clusters, the

inter-cluster distance is computed and the two closest clusters are merged to a single cluster. This merging step is successively repeated until only one cluster remains that comprises all clusters.

To compute the distance between sets of web contents, we apply the single link strategy, which corresponds to selecting the minimum distance of a web content from the first set to a content of the second set. Once the dendrogram is computed, the final clustering can be derived by cutting the dendrogram at a given level.

The clusters are derived by cutting the dendrogram at an appropriate level.

4.3 Density-based clustering

In density-based clustering approaches clusters are data regions of a higher density that are bordered by data regions of lower density. In DBSCAN [8] the minimal desired density is defined by two input parameters **Eps** and **MinPts**. Starting with an arbitrary data point, the point is checked on being a *core object*. A data point is called a core point, provided the number of data points within its **Eps**-environment is at least **MinPts**. If the actual data point is a core object, a new cluster is created with the actual point as its first member and the cluster is expanded by all points that are density-reachable by the actual data point. A data point b is *density-reachable* by a data point a iff there is a sequence of connected core objects starting with a and ending with a data point in the **Eps** environment of b . Data points that are neither core objects nor density-reachable by any core object are marked as "noise". The algorithm finishes when all data points are either clustered or marked as noise.

Regarding the DOM-based distance measure, we recognize that clusters should be groups of cohesive densities while different densities should be considered, since web content blocks can be distributed over different tree levels. Thus the common DBSCAN algorithm would fail with such a metric. To consider the special requirements of the DOM-based distance, a variation of DBSCAN [9] has been applied and adopted to work without input parameters. This extended DBSCAN algorithm clusters points of regions with a coherent density. This can be very useful in combination with the DOM distance, which spreads coherent web contents over a similar density region.

The other distance measures do not require any special treatments and work better with the common DBSCAN algorithm. However, the input parameters have to be estimated here.

5. EVALUATION

5.1 Distance measures

In the first part of the evaluation, we will focus on properties of the proposed distance measures and show some empiric results. As example web page, the start page of the WIMS 2011 conference has been employed, which is depicted in Figure 4. Preliminary, we have extracted all text and content image nodes (too small images or images with a width-height-ratio exceeding a threshold were ignored) of the example page and they were put in a vector V in the order as they appear in HTML representation. Figure 4 further contains blocks with index ranges of contents within the specified block.

Computational Time. The graphs in Figure 3 show

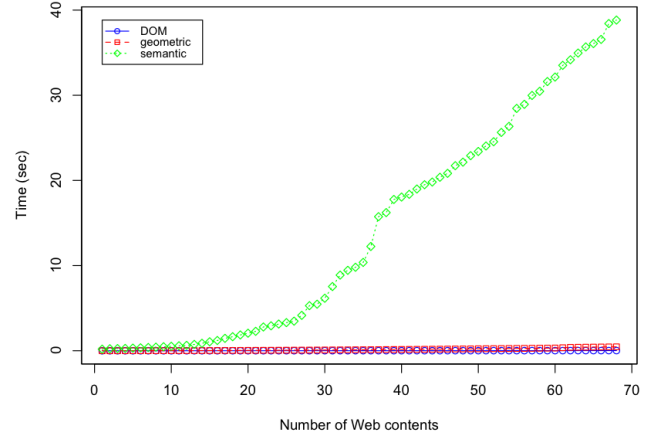


Figure 3: Computational times for the distance-matrices of different distance measures.

the computational times in seconds needed to compute a distance-matrix with different distance measures depending on the number of web contents. While the DOM-based distance and the geometric distance can be computed reasonably fast, the computation of the semantic distance is more time consuming. Instead of computing distance-matrices for different web pages and averaging over the computed times, we decided to show the results for only one document in order to show that, e.g., the semantic distance does not only depend on the number of contents, but also on the length of the contents. On the other side, the DOM and the geometric distance need are not affected by any node properties and thus produce smooth graphs.

Distance-Matrix Visualization. The computed distance matrices of contents of the WIMS start-page are represented in Figure 5 in a graphical way. The web contents are arranged in the same order from left to right in the columns and top down in the rows. Further, this order also corresponds to the order in which the contents appear in the HTML. Each pixel in an image represents the distance of two contents. A bright pixel color corresponds to a smaller distance, while darker pixels represent high distances. Since the range of the DOM distance can reach extraordinary high values (depending on the DOM tree depth), the matrix-values were transformed into logarithmic scale for visualization. The matrices of the geometric and the semantic distance were not preprocessed. Obviously, all matrices have a white diagonal because the distance of a content to itself is 0. To start with the DOM distance, the graphic shows clear rectangles of nearly constant brightness in the matrix diagonal which are further hierarchically organized: less brighter rectangles surround brighter ones. We can clearly recognize the rectangles which correspond to the blocks depicted in the WIMS start page. Similar patterns can also be found in the diagonal of the geometric distance matrix. However, there are many other bright regions in the geometric distance matrix, e.g., $(1 - 19, 29 - 43)$ because the corresponding blocks are next to each other in the browser output. The semantic distance matrix depicts only partially the contained structures. The rectangle $(8 - 19, 8 - 19)$ has a pattern similar to a chessboard. This is because each label content is followed by a date content; dates have a strong similarity to each

Home 57-65

Organisation
Call for Papers
Submission
Registration
Program
Important Dates
Accommodation
Venue

66-68

ORGANISED BY 44-54

WESTERN NORWAY RESEARCH INSTITUTE
VESTLANDSFORSKING

Proceedings will be published by

Association for Computing Machinery

Selected papers from WIMS'11, after further revisions, will be published in the special issues of the following journals:

International Journal of Metadata, Semantics and Ontologies

International Journal of Web Services Practices

International Journal of Information Retrieval Research

International Journal of Computer Science & Applications

Some selected extended papers from WIMS'11 will be considered for Elsevier (Morgan Kaufmann) book [pending approval].

This website is licensed under a Creative Commons Attribution-Share Alike 3.0 License 55-56

Welcome 29-43

The International Conference on Web Intelligence, Mining and Semantics (WIMS'11) will be organised under the auspices of [Western Norway Research Institute](#).

This is the first in a new series of conferences concerned with intelligent approaches to transform the World Wide Web into a global reasoning and semantics-driven computing machine. Next conferences in this series, WIMS'12 and WIMS'13, will take place in Craiova (Romania) and Madrid (Spain) respectively.

The conference will provide an excellent international forum for sharing knowledge and results in theory, methodology and applications of Web intelligence, Web mining and Web semantics. The program will feature several keynote and invited talks, from academia and the industry.

The purpose of the WIMS'11 is:

- To provide a forum for established researchers and practitioners to present past and current research contributing to the state of the art of Web technology research and applications.
- To give doctoral students an opportunity to present their research to a friendly and knowledgeable audience and receive valuable feedback.
- To provide an informal social event where Web technology researchers and practitioners can meet.

Scientific American article by Tim Berners-Lee, Ora Lassila and James Hendler was published in May 2001. The WIMS'11 conference is an appropriate event to reflect on the 10 years - successes and misses >>>

Call for Breaking News Abstracts

WIMS'11 Flyer

Important Dates 1-19

Electronic submission of papers/posters (Extended): **November 20, 2010**

Tutorial and Workshop proposals due: **December 15, 2010** 🌟

Notification of paper acceptance: **January 15, 2011**

Registration opens: **February 1, 2011**

Camera-ready of accepted papers: **February 15, 2011**

Breaking news - Abstract submission: **March 1, 2011** 🌟

Registration closes: **May 10, 2011**

Conference: **May 25 - 27, 2011**

CONTACT: 20-28

Vestlandsforskning,
PO Box 163,
NO-6851 SOGNDAL, NORWAY

Phone: +47 916 85 807
Fax: +47 947 63 727
E-mail: wims11@vestforsk.no

SPONSOR WIMS'11!

Figure 4: The start-page of WIMS 2011 and possible blocks in which the page could be segmented. The numbers are the id's of the contents arranged in the order as they appear in HTML.

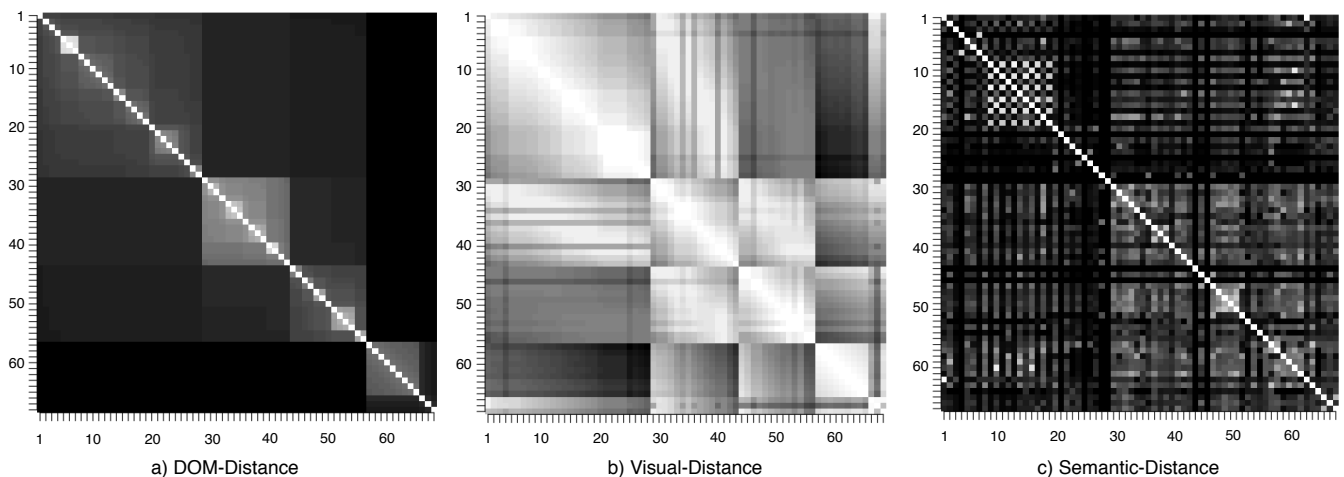


Figure 5: Visualized distance matrices for a) DOM distance, b) geometric distance and c) semantic distance.

other, while the labels are unrelated to dates. It is also interesting, that contents distributed over the whole web page can be very similar, as e.g., (1, 63), but comparing the corresponding labels which is *Important Dates* in both cases, shows the reason.

5.2 Clustering performance

In this part of the evaluation, we investigate the performance of different clustering approaches combined with the proposed distance metrics.

Dataset. We collected a sample of 78 web documents from 8 different categories of the Yahoo! directory resulting in 23,819 web contents (text and content images) as data collection. The web contents of the documents were grouped manually by 3 volunteers. They got the instructions to create structurally and semantically coherent groups. Although this process lacks of objectivity, we suppose that the volunteers at least have a good idea of what "coherent" blocks are.

Parameter estimation. The parameter k in k -means clustering can be set to the number of classes, the user has specified for a given web page. Similar, for single link clustering, the appropriate level at which the dendrogram is cut, can be chosen by considering the number of clusters to produce. However, in DBSCAN the parameters **Eps** and **MinPts** had been considered empirically. For this purpose, we run DBSCAN iteratively with different parameter configurations, and the best parameter settings in regard to the Rand statistics were applied.

Cluster Separation. In order to compare the results of different clustering techniques in different configurations, we compute the Dunn index for each run. The Dunn index expresses, how well the web contents are separated in a given clustering result. This measure is defined as the ratio of the smallest distance between two web contents that are not in the same cluster to the highest distance of contents that are in the same cluster. The higher the Dunn index value, the better the clusters are separated in regard to a distance measure. Values over 1 indicate a good separation.

Table 1 summarizes the average Dunn index over the complete collection for the different clustering approaches with different distance measures employed.

	DOM-based	geometric	semantic
k-medians	1.03	0.34	1.05
single link	1.12	0.43	1.02
DBSCAN	-	0.24	1.18
DBSCAN (ext.)	0.35	-	-

Table 1: Average Dunn index for different distance measures and different clustering techniques.

The first remarkable observation is, that the semantic-based distance has Dunn index values over 1 which indicates that in the metric space derived by the semantics of contents, the contents are well distributed. In our experimental studies, we further recognized, that the Dunn index could be even better for less clusters when semantically coherent blocks had to be separated due to the specified number of clusters. The DOM-based distance gets similar results for k-medoid and single link clustering, however in the extended DBSCAN algorithm, the Dunn index is very low.

This happens due to an important property of the extended DBSCAN: we start first by putting together contents with a high density in a first cluster and then continue with the next cluster of a higher density, and repeat this step until in the last cluster we reach the highest density. This yields to a very high intra-cluster distance in the cluster with highest density and on the other hand a relatively small inter-cluster distance for clusters of small density. However, we have measured the Dunn index for the extended DBSCAN in combination with the DOM-distance because of an important distribution property of DOM-distance: contents are on different levels of the DOM tree and thus clusters of different densities are expected. The geometric distance measure reaches low separation values. This observation can be justified with the following simple example depicted in Figure 6.

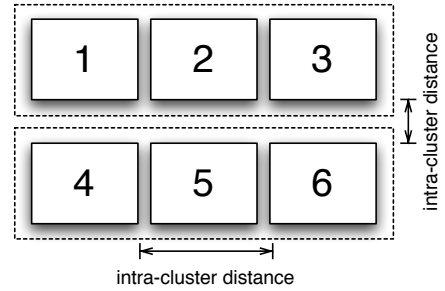


Figure 6: Partitioning six contents in two clusters using geometric distance.

In this example, six web contents (here presented by their visual boxes) have to be grouped in two clusters. A good clustering is given by the dotted rectangles. However, in this case the inter-cluster distance is much smaller than the intra-cluster distance, which results in Dunn index lower than 1.

Rand statistics. To compute the correspondence between computed and manually defined clusters, the following contingency table shows four relations that can occur between pairs of contents:

	Same cluster	Different cluster
Same class	f_{11}	f_{10}
Different class	f_{01}	f_{00}

In this table, we distinguish between content pairs that are classified by volunteers to belong together or not, or respectively computed by a clustering method to belong together or not. Based on this values, we can compute the Rand statistic, which is defined as follows:

$$Rand\ statistic = \frac{f_{00} + f_{11}}{f_{00} + f_{01} + f_{10} + f_{11}}.$$

As for the Dunn index, the higher the Rand statistic value, the better the clustering performance. Table 2 summarizes the average Rand statistic results for different clustering approaches with different distance measures.

For the single link and extended DBSCAN method the DOM-based distance gets the highest accuracy while the for k-medoid clustering the Rand statistic gets lower. This might be a consequence of the hierarchical distribution of the contents by the DOM-distance. This hierarchy is not considered by the k-medoid algorithm. The geometric distance

	DOM-based	geometric	semantic
k-medoid	0.45	0.47	0.25
single link	0.52	0.41	0.24
DBSCAN	-	0.43	0.27
DBSCAN (ext.)	0.61	-	-

Table 2: Average Rand statistic for different distance measures and different clustering techniques.

reaches the highest accuracy combined with the k-medoid method which might be justified by the property of the geometric distance to spread the web contents over a geometric space. In tests with single link and traditional DBSCAN the accuracy values were lower. Finally, a last interesting observation is that the Rand statistics of the semantic distance were low independent of which clustering method was used. A possible explanation might be given by the distance-matrix visualization example from Figure 5. The semantic distance can assign a high value to web contents that, e.g., in the browser representation are next to each other because they do not share common semantics. Respectively, contents that are spread over the complete document can get very low distances because their semantics are almost equal.

6. CONCLUSIONS AND FUTURE WORKS

In this work, we have reduced the page segmentation task to web content clustering. We have defined possible distance functions which were combined with different clustering techniques. The DOM-distance measure combined with the extended DBSCAN algorithm reached the best Rand statistic value in average. Otherwise, the best separation values have been achieved by the semantic distance. As a next step, we plan to further investigate distance measures that are derived by combining the proposed distance measures. Also dividing the clustering process in more steps could be suitable. In the first step a coarse segmentation could be done by a clustering technique combined with a distance measure, and in the next step the resulting clusters can be partitioned into finer segments by another clustering technique and distance measure. We further intend to confirm the results on a larger scale, with more labeled documents and more users involved and compare the results to the other Page segmentation approaches from the literature.

7. REFERENCES

- [1] S. Alcic and S. Conrad. 2-DOM: A 2-Dimensional Object Model towards Web Image Annotation. *Semantic Media Adaptation and Personalization, International Workshop on*, pages 3–8, 2008.
- [2] S. Baluja. Browsing on small screens: recasting web-page segmentation into an efficient machine learning framework. In *Proceedings of the 15th international conference on World Wide Web, WWW '06*, pages 33–42, New York, NY, USA, 2006. ACM.
- [3] D. Cai, X. He, Z. Li, W.-Y. Ma, and J.-R. Wen. Hierarchical clustering of WWW image search results using visual, textual and link information. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 952–959, New York, USA, 2004.
- [4] D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma. VIPS: a Vision-based Page Segmentation Algorithm. Technical report, Microsoft Research (MSR-TR-2003-79), 2003.
- [5] D. Chakrabarti, R. Kumar, and K. Punera. A graph-theoretic approach to webpage segmentation. In *Proceeding of the 17th international conference on World Wide Web, WWW '08*, pages 377–386, New York, NY, USA, 2008.
- [6] Y. Chen. Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices. In *In Intl. World Wide Web Conf. (WWW)*, pages 225–233. ACM Press, 2003.
- [7] C. Corley and R. Mihalcea. Measuring the semantic similarity of texts. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, EMSEE '05*, pages 13–18, Morristown, NJ, USA, 2005. Association for Computational Linguistics.
- [8] M. Ester, H. Peter Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In , pages 226–231. AAAI Press, 1996.
- [9] A. M. Fahim, G. Saake, A.-B. M. Salem, F. A. Torkey, and M. A. Ramadan. An Enhanced Density Based Spatial clustering of Applications with Noise. In *DMIN*, pages 517–523, 2009.
- [10] G. Hattori, K. Hoashi, K. Matsumoto, and F. Sugaya. Robust web page segmentation for mobile terminal using content-distances and page layout information. In *Proceedings of the 16th international conference on World Wide Web, WWW '07*, pages 361–370, New York, NY, USA, 2007. ACM.
- [11] H.-Y. Kao, J.-M. Ho, and M.-S. Chen. WISDOM: Web Intrapage Informative Structure Mining Based on Document Object Model. *IEEE Trans. on Knowl. and Data Eng.*, 17:614–627, May 2005.
- [12] C. Kohlschütter and W. Nejdl. A densitometric approach to web page segmentation. In *Proceeding of the 17th ACM conference on Information and knowledge management, CIKM '08*, pages 1173–1182, New York, NY, USA, 2008. ACM.
- [13] A. Lee and V. Hanson. Enhancing web accessibility. In *Proceedings of the eleventh ACM international conference on Multimedia, MULTIMEDIA '03*, pages 456–457, New York, NY, USA, 2003. ACM.
- [14] X. Li, B. Liu, T. heng Phang, and M. Hu. Using Micro Information Units for Internet Search. In *Proc. of the ACM 11th International Conf. on Information and Knowledge Management (CIKM-02)*, 2002.
- [15] D. Lin. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2, ACL-36*, pages 768–774, Morristown, NJ, USA, 1998. Association for Computational Linguistics.
- [16] S.-H. Lin and J.-M. Ho. Discovering Informative Content Blocks from Web Documents. In *In Proceedings of ACM SIGKDD'02*, pages 588–593, 2002.
- [17] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. WordNet: An on-line lexical database.

International Journal of Lexicography, 3:235–244, 1990.

- [18] S. Patwardhan, S. Banerjee, and T. Pedersen. Using measures of semantic relatedness for word sense disambiguation. In *Proceedings of the 4th international conference on Computational linguistics and intelligent text processing, CICLing'03*, pages 241–257, Berlin, Heidelberg, 2003. Springer-Verlag.
- [19] P. Resnik. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *International Joint Conference for Artificial Intelligence (IJCAI-95)*, pages 448–453, 1995.
- [20] K. Sparck Jones. *A statistical interpretation of term specificity and its application in retrieval*, pages 132–142. Taylor Graham Publishing, London, UK, UK, 1988.
- [21] G. Vineel. Web page DOM node characterization and its application to page segmentation. In *Internet Multimedia Services Architecture and Applications (IMSAA), 2009 IEEE International Conference on*, pages 1 –6, 2009.
- [22] S. Yu, D. Cai, J.-R. Wen, and W.-Y. Ma. Improving pseudo-relevance feedback in web information retrieval using web page segmentation. In *Proceedings of the 12th international conference on World Wide Web, WWW '03*, pages 11–18, New York, NY, USA, 2003. ACM.