

# Measuring the Structural Similarity of Web-based Documents: A novel Approach

Matthias Dehmer, Frank Emmert Streib, Alexander Mehler and Jürgen Kilian

**Abstract**—Most known methods for measuring the structural similarity of document structures are based on, e.g., tag measures, path metrics and tree measures in terms of their DOM-Trees. Other methods measure the similarity in the framework of the well known vector space model. In contrast to these we present a new approach to measuring the structural similarity of web-based documents represented by so called generalized trees which are more general than DOM-Trees which represent only directed rooted trees. We will design a new similarity measure for graphs representing web-based hypertext structures. Our similarity measure is mainly based on a novel representation of a graph as strings of linear integers, whose components represent structural properties of the graph. The similarity of two graphs is then defined as the optimal alignment of the underlying property strings. In this paper we apply the well known technique of sequence alignments to solve a novel and challenging problem: Measuring the structural similarity of generalized trees. More precisely, we first transform our graphs considered as high dimensional objects in linear structures. Then we derive similarity values from the alignments of the property strings in order to measure the structural similarity of generalized trees. Hence, we transform a graph similarity problem to a string similarity problem. We demonstrate that our similarity measure captures important structural information by applying it to two different test sets consisting of graphs representing web-based documents.

**Keywords**—Graph similarity, hierarchical and directed graphs, hypertext, generalized trees, web structure mining.

## I. INTRODUCTION

THE application of *Data Mining* methods [3] on web-based hypertext data is referred to as *Web Mining* [3]. In view of the vast amount of information available online Web Mining is an important and fruitful research field. Thereby Web Mining can be divided into three major fields (see Fig. (1)): *Web Structure Mining*, *Web Usage Mining* and *Web Content Mining*. In the area of Web Structure Mining we investigate graph based patterns by extracting web-based hypertext structures for measuring the structural similarity of those patterns. In this sense the paper sheds light on the task of the automatic analysis of web genre data. To measure the similarity between graphs is a challenging problem, especially for graphs of large orders the computation of similarity measures is quite involved [17], [23]. Many similarity measures of graphs are based on isomorphic relations and subgraph isomorphism [17], [23], respectively. Because it is well known

Matthias Dehmer is with the Technische Universität Darmstadt, 64289 Darmstadt, Germany, e-mail: dehmer@informatik.tu-darmstadt.de. Frank Emmert-Streib is with the Stowers Institute for Medical Research, 1000 E. 50th Street, Kansas City, MO 64110, USA, e-mail: fes@stowers-institute.org. Alexander Mehler is with the Universität Bielefeld, 33501 Bielefeld, Germany, e-mail: Alexander.Mehler@uni-bielefeld.de. Jürgen Kilian is with the Technische Universität Darmstadt, 64289 Darmstadt, Germany, e-mail: kilian@noteserver.org.

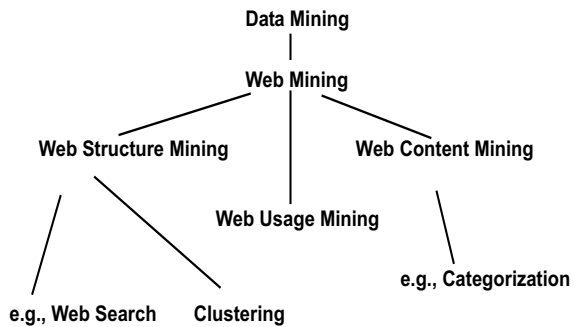


Fig. 1. Major research fields [3] of Web Mining.

that the subgraph isomorphism problem is NP-complete, the complexity of the underlying graph similarity measure is considered to be unacceptable for practical use [18]. Apart from measures that are based on isomorphic relations, there are methods in order to measure the structural similarity of trees. For example SELKOW [15] generalized the well known string edit distance [9], [16] to trees.

Algorithmic improvements of tree alignment techniques for applications in Bioinformatics were proposed by, e.g., JIANG et al. [10] and ZHANG et al. [22]. But for the most graph similarity problems these methods are not suitable because the topology of the underlying graphs is more complex than the topology of trees. For this reason, we will present a new method for measuring the similarity of *labeled/unlabeled, hierarchical and directed graphs*. The term *hierarchical* means that there is an underlying directed rooted tree which forms a tree hierarchy. The main idea of our new method consists in two steps: (i) derivation of property strings from a graph and (ii) evaluating the alignment of the corresponding strings representing graphs. This is uniquely possible, because we restrict ourselves to hierarchical graphs. In the following we call these graphs *generalized trees*. This graph class has been first introduced by MEHLER et al. [13] and is defined by

**Definition 1.1:** Let  $\mathcal{H} = (\hat{V}, E_1)$  be an directed rooted tree. Let  $m_{\hat{V}} : \hat{V} \rightarrow A_{\hat{V}}$  be a vertex labeling function and  $A_{\hat{V}}$  denote a vertex alphabet. The vertex set  $\hat{V}$  is defined by

$$\hat{V} := \{v_{0,1}, v_{1,1}, v_{1,2}, \dots, v_{1,\sigma_1}, v_{2,1}, v_{2,2}, \dots, v_{2,\sigma_2}, \dots, v_{h,1}, v_{h,2}, \dots, v_{h,\sigma_h}\}$$

where  $v_{i,j}$  denotes the  $j$ -th vertex on the  $i$ -th level,  $0 \leq i \leq h$ ,  $1 \leq j \leq \sigma_i$ .  $h$  denotes the depth of  $\mathcal{H}$  and  $\sigma_i$  is the number of vertices on level  $i$ . The edge set  $\hat{E} := \hat{E}_1 \cup \hat{E}_2 \cup \hat{E}_3 \cup \hat{E}_4$  is defined as [13]:

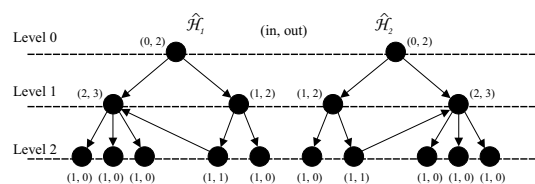


Fig. 3. Two generalized trees with their property strings. As an example the property string in terms of out-degrees of  $\mathcal{H}_1$  on level 1 equals "3  $\circ$  2". The symbol  $\circ$  denotes usual string concatenation.

- If  $\hat{E}_2$ ,  $\hat{E}_3$  and  $\hat{E}_4 \neq \emptyset$  then  $\hat{\mathcal{H}} = (\hat{V}, \hat{E}, m_{\hat{V}}, A_{\hat{V}})$  denotes a generalized tree. If we set  $A_{\hat{V}} = \emptyset$ , the generalized tree is unlabeled (see Fig. (2)).

This paper is organized as follows: In the following section we state shortly some results of *hypertext categorization* [13] in order to intensify our motivation of our new approach for measuring the structural similarity of web-based documents. In Section (III) we repeat some graph theoretic measures for the structural analysis of hypertext structures introduced so far in the context of Web Mining. We present in Section (IV) our similarity measure mathematically and apply it in Section (V) together with agglomerative clustering to two different data sets, each one containing web-based document structures. This paper finishes in Section (VI) with a summary and conclusions.

An important task of hypertext analysis is content-based hypertext categorization [21]. Hypertext categorization is the

For motivating our new method for measuring the similarity of hypertext structures representing generalized trees we first look at known measures for the structural analysis of hypertexts. We will see, that those measures are not suitable for a similarity-based structural analysis of graph-based documents, because they can not capture enough information that allows a meaningful clustering of web-based documents. In the field of structural analysis of hypertexts there are many measures that are called *indices* describing structural properties of hypertexts [2], [19]. The characteristic property of an index is that the described structural property, e.g., connectedness, is mapped on a normalised measured value. For example, BOTAFOGO et al. [2] defined the well known graph theoretic measure *Compactness* of a directed hypertext graph  $\mathcal{H}$  as

which expresses how well connected a hypertext graph is. Here,

$$(c_{ij})_{ij} := \begin{cases} w_{ij} & : \text{if } w_{ij} \text{ exists} \\ \mathcal{K} & : \text{else,} \end{cases}$$

denotes the converted distance matrix and  $w_{ij}$  denotes the shortest path from  $v_i$  to  $v_j$ .  $\mathcal{K}$  defines the *conversion constant*,  $|V|$  denotes the number of vertices of  $\mathcal{H}$  and BOTAFOGO et al. [2] set  $\mathcal{K} = |V|$ . From definition of  $C$  we conclude:  $C = 1$  iff  $\mathcal{H}$  is completely connected,  $C = 0$  iff  $\mathcal{H} = (V, \emptyset)$ . Now, assume for two hypertext graphs  $\mathcal{H}_1, \mathcal{H}_2$  holds  $C_1 \approx C_2$ . It is clear that the graph structures can be noticeable different. Therefore the index  $C$  is not suitable for determining intervals which contain similar hypertext structures. Hence, it is not possible to derive quality features, like "positive navigation behavior" from a certain value  $C^* \in [0, 1]$ .

In contrast to this, the unsupervised learning approach we suggest extends and improves the concept of graph theoretic indices mentioned significantly. Our new approach for analyzing web-based document structures consists of two steps:

- 1) Developing a method for measuring the similarity of generalized trees. Up to now there are no contributions in the area of the structural analysis of hypertext data that present results in terms of measuring the similarity of document structures by comparing the overall graph structure.
- 2) Application of multivariate data analysis methods, e.g., clustering methods.

There are already known approaches determining the similarity of web-based documents, e.g., [5], [11]. CRUZ et al. [5] presented results about the structural similarity of DOM-Trees [4] based on TDFA (*Tag Frequency Distribution Analysis*). JOSHI et al. [11] proposed an approach for measuring the structural similarity of web-based documents representing DOM-Trees on the basis of a *bag of path model*. However, our similarity measure is completely different from all these contributions, in the sense that it captures more structural information of a graph.

#### IV. SIMILARITY MEASURING OF GENERALIZED TREES

Now, we design a new similarity measure  $d$  for measuring the structural similarity of generalized trees. In the following we consider the unlabeled case, because the transition to the labeled case is possible by minor modifications (see Section (V)).

The main idea of this similarity measure is based on the derivation of property strings for each generalized tree and then to align the property strings representing our generalized trees by a *dynamic programming* technique [1] (see Fig. (3)). From the resulting alignment one obtains a value of the scoring function, which is minimized during the alignment process. The similarity of two generalized trees will be expressed by a cumulation of local similarity functions which weighs two types of alignments: *out-degree* and *in-degree* alignments on a generalized tree level. Since we are examining hierarchical graphs, we take a closer look at the out-degree and in-degree sequences (on a level  $i$ ), induced by the vertex sequences  $v_{i,1}, v_{i,2}, \dots, v_{i,\sigma_i}$  and their edge relations (see Fig. (3)). Now,

the more similar with respect to a *cost function*  $\alpha$  the out-degree and in-degree alignments on the levels  $i$  are, the more similar is the common structure of the graphs. Define  $r_k^{\hat{\mathcal{H}}^k} := v_{0,1}^{\hat{\mathcal{H}}^k}$ ,  $k \in \{1, 2\}$ , and let  $\hat{\mathcal{H}}^1$  be a given graph and  $v_{i,j}^{\hat{\mathcal{H}}^1}$ ,  $0 \leq i \leq h_1$ ,  $1 \leq j \leq \sigma_i$  denote the  $j$ -th vertex on the  $i$ -th level of  $\hat{\mathcal{H}}^1$ , analogous to  $v_{i,j}^{\hat{\mathcal{H}}^2}$  for  $\hat{\mathcal{H}}^2$ . Then the problem of determining the structural similarity between  $\hat{\mathcal{H}}^1$  and  $\hat{\mathcal{H}}^2$  is equivalent to computing the optimal alignment of

$$S_1 := r_1^{\hat{\mathcal{H}}^1} \circ v_{1,1}^{\hat{\mathcal{H}}^1} \circ v_{1,2}^{\hat{\mathcal{H}}^1} \circ \dots \circ v_{h_1,\sigma_{h_1}}^{\hat{\mathcal{H}}^1}, \quad (1)$$

$$S_2 := r_2^{\hat{\mathcal{H}}^2} \circ v_{1,1}^{\hat{\mathcal{H}}^2} \circ v_{1,2}^{\hat{\mathcal{H}}^2} \circ \dots \circ v_{h_2,\sigma_{h_2}}^{\hat{\mathcal{H}}^2}, \quad (2)$$

with respect to a *cost function*  $\alpha$  which evaluates the alignments. Here, we distinguish different types of alignments:  $(v, v)$  (vertex-vertex),  $(-, v)$  (gap-vertex), and  $(v, -)$  (vertex-gap). In order to determine the optimal alignment between two given graphs, we consider the Sequences (1), (2) where  $S_k[i]$  denotes the  $i$ -th position of the sequence  $S_k$  and it holds  $S_1[n] = v_{h_1,\sigma_{h_1}}^{\hat{\mathcal{H}}^1}$ ,  $S_2[m] = v_{h_2,\sigma_{h_2}}^{\hat{\mathcal{H}}^2}$ ,  $\mathbb{N} \ni n, m \geq 1$ ,  $S_k[1] = r_k^{\hat{\mathcal{H}}^k}$ ,  $k \in \{1, 2\}$ . The algorithm with the complexity  $O(|\hat{V}_1| \cdot |\hat{V}_2|)$  for finding the optimal alignment of  $S_1$  and  $S_2$  generates a matrix  $(\mathcal{M}(i, j))_{ij}$ ,  $0 \leq i \leq n$ ,  $0 \leq j \leq m$ . Now, we define the optimal alignment on the basis of the following dynamic programming algorithm [9]:

$$\mathcal{M}(0, 0) := 0,$$

$$\mathcal{M}(i, 0) := \mathcal{M}(i-1, 0) + \alpha(S_1[i], -) : 1 \leq i \leq n,$$

$$\mathcal{M}(0, j) := \mathcal{M}(0, j-1) + \alpha(-, S_2[j]) : 1 \leq j \leq m,$$

$$\mathcal{M}(i, j) := \min \begin{cases} \mathcal{M}(i-1, j) + \alpha(S_1[i], -) \\ \mathcal{M}(i, j-1) + \alpha(-, S_2[j]) \\ \mathcal{M}(i-1, j-1) + \alpha(S_1[i], S_2[j]), \end{cases}$$

for  $1 \leq i \leq n$ ,  $1 \leq j \leq m$ . In order to evaluate the alignments on each level, we defined [6] the functions

$$\gamma^{out} = \gamma^{out}(i, \sigma_1^{out}, \sigma_2^{out})$$

and

$$\gamma^{in} = \gamma^{in}(i, \sigma_1^{in}, \sigma_2^{in}),$$

$\sigma_k^{out}, \sigma_k^{in} \in \mathbb{R}$ ,  $k \in \{1, 2\}$  in an natural way and constructed a similarity measure  $d \in [0, 1]$  (on the basis of these functions).  $\gamma^{out}, \gamma^{in}$  are two-parametric functions, which detect the similarity of an outdegree and indegree alignment (on a level  $i$ ). Finally, if we assume a set of units  $U$  and a mapping  $\phi : U \times U \rightarrow [0, 1]$ , we called  $\phi$  a backward similarity measure if it satisfies the conditions

$$\phi(u, v) = \phi(v, u), \forall u, v \in U$$

and

$$\phi(u, u) \geq \phi(u, v), \forall u, v \in U.$$

Now we state the key result which has been proven in [6] for measuring the similarity for generalized trees.

**Theorem 4.1:** Let  $\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2$ ,  $0 \leq i \leq \rho$ ,  $\rho := \max(h_1, h_2)$ .

$$d(\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2) := \frac{\prod_{i=0}^{\rho} \gamma^{fin}(i, \sigma_1^{out}, \sigma_2^{out}, \sigma_1^{in}, \sigma_2^{in})}{\sum_{i=0}^{\rho} \gamma^{fin}(i, \sigma_1^{out}, \sigma_2^{out}, \sigma_1^{in}, \sigma_2^{in})}, \quad (3)$$

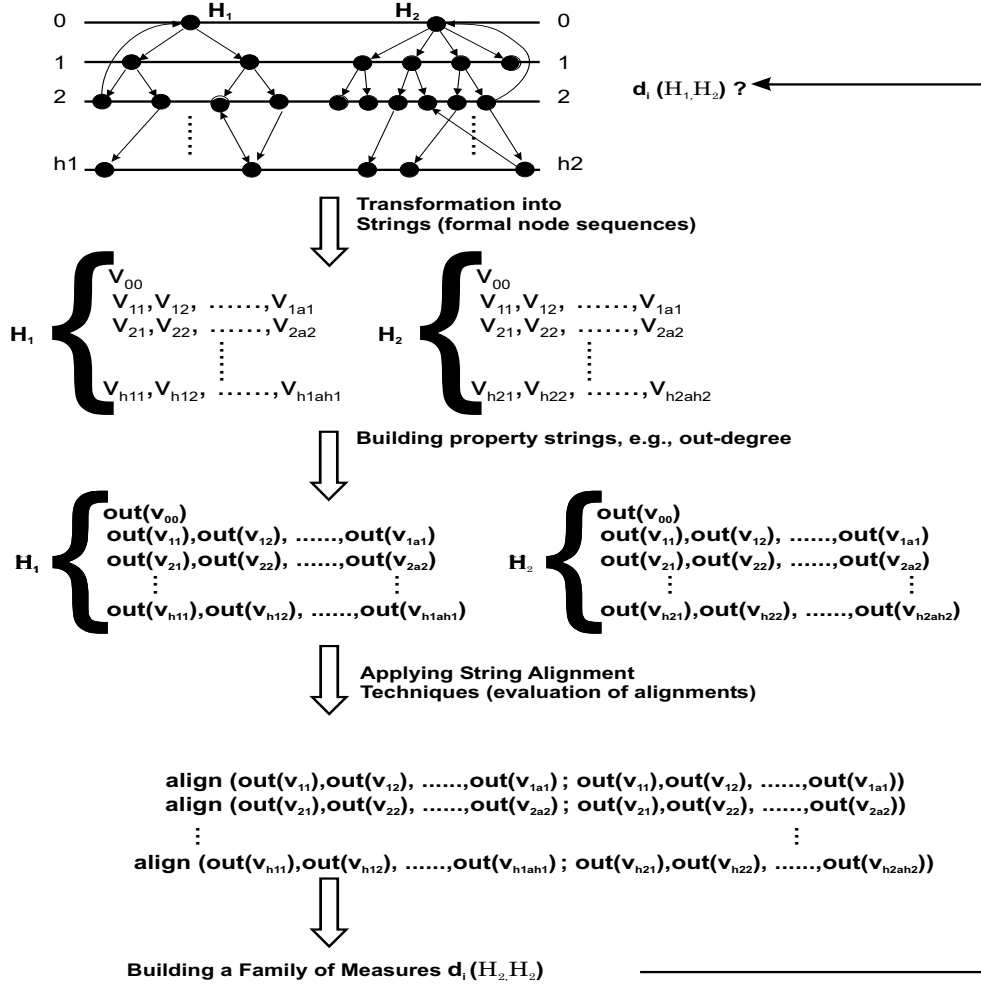


Fig. 4. Procedure for measuring the structural similarity of generalized trees. The big arrows denote the main transformation steps. The thin arrow shows the logical connection between the transformation steps and the computation of  $d_i(H_1, H_2)$ .

is a backward similarity measure, where  $\gamma^{fin}$  is defined as

$$\begin{aligned} \gamma^{fin} &= \gamma^{fin}(i, \sigma_1^{out}, \sigma_2^{out}, \sigma_1^{in}, \sigma_2^{in}) \\ &:= \zeta \cdot \gamma^{out} + (1 - \zeta) \cdot \gamma^{in}, \quad \zeta \in [0, 1]. \end{aligned}$$

By construction we have  $d(\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2) \in [0, 1]$ . As a summary we note that our algorithm measures the similarity of two generalized trees by applying the technique of sequence alignments to outdegree and indegree sequences (on a level  $i$ ). These alignments have both global and local significance. On the one hand, the sequence alignments will be implemented in a global sense, to compute the optimal alignment between the sequences  $S_1$  and  $S_2$ . On the other hand, the alignments will be evaluated on the levels of the generalized trees by the function  $\gamma^{fin}$ . We note that the presented algorithm is suitable for the comparison of large generalized trees, because its complexity is enormously better than the complexity of methods which deal with isomorphic relations. As a summary Fig. (4) shows the overall procedure for measuring the structural similarity of web-based documents representing generalized trees.

## V. EXPERIMENTAL RESULTS

We applied our method to web-based document structures representing generalized trees and, hence, to a problem from web structure mining. To perform the evaluation, we used a corpus  $T_C$  which contains 500 conference websites from mathematics and computer science due to MEHLER et al. [13]. Starting from conference calendar websites, MEHLER et al. created the corpus with a java application that collects the conference links. Based on this set of links, we extracted the websites from the web by HyGraph [8]. Our evaluation is based on the two following steps:

- 1) Examining the cumulative similarity distribution of  $T_C$  on the basis of website structures, that is we represent a conference website  $w \in T_C$  as a unlabeled generalized tree. This examination is impossible without a meaningful similarity measure which covers the complex graph structure of two graph-based documents. Thereby, the interpretation of the cumulative similarity distribution leads us to various applications.
- 2) Application of agglomerative clustering to the obtained

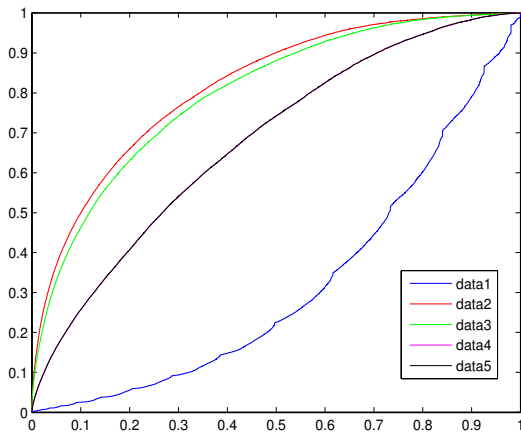


Fig. 5. The  $x$ -axes corresponds to the values of  $d \in [0,1]$  and the  $y$ -axes represents the cumulative similarity distributions for  $D_1$ - $D_5$ .

similarity matrix where the web-based documents are represented by their DOM-Trees [4]. Here, a DOM-Tree is represented by a labeled generalized tree.

**Definition 5.1:** In terms of  $T_C$  we define data classes  $D_1 - D_5$  which are manifest by the following parameter spectra:

- $D_1$ :  $\zeta = 1.0$  (solely alignments of Kernel-edges); parameter settings:

$$\sigma_{out}^1 = 1.0, \sigma_{out}^2 = 2.0, \sigma_{in}^1 = 1.0, \sigma_{in}^2 = 2.0.$$

- $D_2$ :  $\zeta = 0.3$ ; parameter settings:

$$\sigma_{out}^1 = 1.0, \sigma_{out}^2 = 1.0, \sigma_{in}^1 = 1.0, \sigma_{in}^2 = 1.0.$$

- $D_3$ :  $\zeta = 0.5$ ; parameter settings:

$$\sigma_{out}^1 = 1.0, \sigma_{out}^2 = 1.0, \sigma_{in}^1 = 1.0, \sigma_{in}^2 = 1.0.$$

- $D_4$ :  $\zeta = 0.5$ ; parameter settings:

$$\sigma_{out}^1 = 3.0, \sigma_{out}^2 = 3.0, \sigma_{in}^1 = 3.0, \sigma_{in}^2 = 3.0.$$

- $D_5$ :  $\zeta = 0.5$ ; parameter settings:

$$\sigma_{out}^1 = 2.0, \sigma_{out}^2 = 2.0, \sigma_{in}^1 = 2.0, \sigma_{in}^2 = 2.0.$$

In general the computation of the cumulative similarity distribution of a corpus of graph-based hypertext structures opens new perspectives, e.g.:

- The distribution in terms of structural similarity of graph-based hypertexts is unknown, because there is no efficient graph similarity measure available, which capture enough structural information for measuring the structural similarity. Our graph similarity measure  $d$  together with the cumulative similarity distribution provides a better understanding of web-based hypertexts and their interactions.
- Suppose we have a test corpus  $T$  of graph-based hypertexts from a specific web-genre, e.g., conference websites. MEHLER et al. described in [14] the interesting problem of computing a structural prototype of  $T$  by a graph median. As a preprocessing step for the *graph prototyping* [14] we can decide, on the basis of the cumulative

TABLE I

RESULTS OF PERFORMANCE EVALUATION OF THE CLUSTERING PROCEDURE CONCERNING  $W_1$ .

$C_i$	cluster	precision	recall
$C_1$	staff web pages	84%	99%
$C_2$	lecture announcements	76%	91%
$C_3$	summary web pages	65%	92%
$C_4$	lecture materials	85%	60%
$C_5$	Download pages	72%	84%

similarity distribution, how structurally different the web-based document structures are.

Here, we compute the cumulative similarity distribution of  $T_C$  based on Definition (5.1) in order to examine the navigation strategies in terms of a web-genre. We assume that a generalized tree reflects all possible navigation paths of a graph-based conference website. The computation and interpretation of the cumulative similarity distribution of  $T_C$  leads us to the question how different the navigation strategies within the specific web-genre are. As example, we choose the web-genre of conference websites. In order to discuss the cumulative similarity distribution of  $T_C$  (see Fig. (5)) we note that the data classes  $D_1$ - $D_5$  are manifest by the same corpus  $T_C$ . We obtain a certain data class by only varying the parameters mentioned in Definition (5.1). Now, by varying the parameters we find the parameter tuple  $(\zeta, \sigma_{out}^1, \sigma_{out}^2, \sigma_{in}^1, \sigma_{in}^2)$  which captures enough structural information during the similarity measuring. In the following we notice that the plot of class  $D_1$  differs in principle from the plots of data classes  $D_2$ - $D_5$ . We recognize that, e.g., 20% of the conference websites have already the similarity value  $d \leq 0.5$ . Unlike 90% of the conference websites in  $D_2$  have the similarity value  $d \leq 0.5$ . In summary, we conclude from Fig. (5), that the similarity values of the conference websites in  $D_1$  were significantly higher compared to the conference websites of data classes  $D_2$ - $D_5$ . This is plausible, because the conference websites in  $D_1$  are treated solely as rooted trees without Cross-edges, Up-edges and Down-edges. Hence, the main part of the conference websites of  $D_1$  is significantly less structurally different than the websites of the remaining data classes. In terms of  $D_2$ - $D_5$  the situation is inverted: In consideration of all types of conference websites the main part of the graph-based hypertext structures are structurally dissimilar on the basis of  $d$ . The plot of  $D_4$  equals the plot of  $D_5$ . Finally we notice that for the data classes  $D_2$ - $D_5$  the main part of all possible navigation strategies are very different within our web-genre. This is reflected by psychological features of hypertext navigation, e.g.:

- Certain strategies of treatment
- Existence of previous knowledge
- Specific user preferences

The next application consists of an evaluation of a set of websites  $\{W_1, W_2\}$ <sup>1</sup> which are treated as sets of their DOM-Trees [4]. Now we have to modify our similarity measure  $d$

<sup>1</sup> <http://www.algo.informatik.tu-darmstadt.de>, <http://www.sec.informatik.tu-darmstadt.de>.

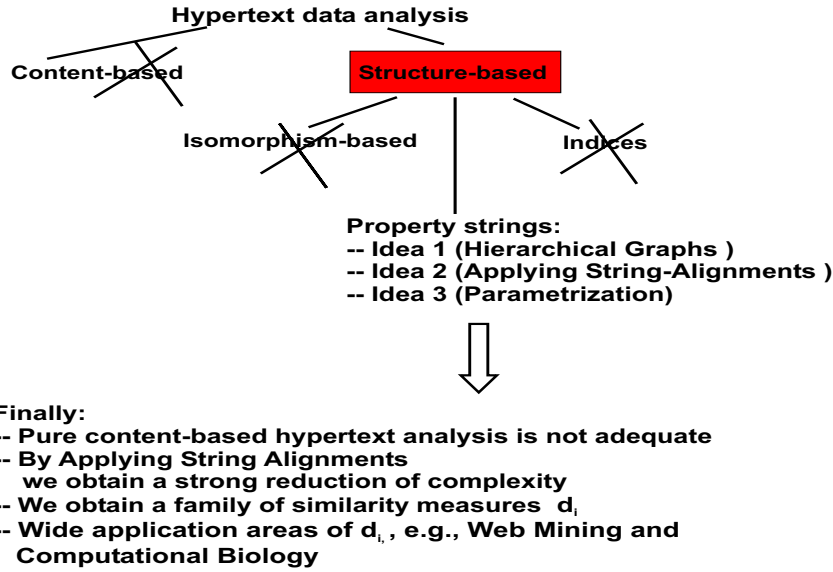


Fig. 6. Overall Approach for measuring the structural similarity of web-based document structures.

TABLE II  
RESULTS OF PERFORMANCE EVALUATION OF THE CLUSTERING  
PROCEDURE CONCERNING  $W_2$ .

$C_i$	Cluster	precision	recall
$C_1$	Staff web pages	99%	99%
$C_2$	summary web pages	99%	99%
$C_3$	seminar announcements	75%	99%
$C_4$	lecture announcements	93%	93%
$C_5$	techn. doc. pages	50%	99%

because our generalized trees are vertex labeled by HTML-markups. We define in a similar way

$$\gamma_m^{fin}(i, \hat{\sigma}_{out}^1, \hat{\sigma}_{out}^2, \hat{\sigma}_{in}^1, \hat{\sigma}_{in}^2) := (1 - \zeta_m) \cdot \gamma_m^{fin}(i, \hat{\sigma}_{out}^1, \hat{\sigma}_{out}^2, \hat{\sigma}_{in}^1, \hat{\sigma}_{in}^2) + \zeta_m \cdot \gamma_m(i, \sigma_m),$$

where  $\zeta_m \in [0, 1]$ ,  $\sigma_m \in \mathbb{R}$  and  $\gamma_m(i, \sigma_m)$  state a cumulative function which detects the similarity of vertex markup alignment on a level  $i$ . With  $\zeta_m = 0.5$  ( $\sigma_{out}^1 = 1.0$ ,  $\sigma_{out}^2 = 2.0$ ,  $\sigma_{in}^1 = 1.0$ ,  $\sigma_{in}^2 = 2.0$ ) and on the basis of a simple scheme which expresses the similarity relations between vertex labels, we apply an *agglomerative clustering method*<sup>2</sup> [3] to the obtained similarity matrices  $(d_{ij})_{ij}$ ,  $1 \leq i \leq |W_1|$ ,  $1 \leq j \leq |W_1|$  and  $(d_{ij})_{ij}$ ,  $1 \leq i \leq |W_2|$ ,  $1 \leq j \leq |W_2|$ ,  $d_{ij} \in [0, 1]$ .  $|W_1|$  and  $|W_2|$  denote the number of extracted DOM-Trees from  $W_1$ ,  $W_2$ , respectively.

As a result we found cluster  $C_i$  which contain graph-based document structures where the  $C_i$  manifest structure types, e.g., staff web pages, and special structured template web pages. We evaluated the performance of clustering with the well know measures  $recall = \frac{|M_r \cap M_g|}{|M_r|}$  and  $precision$

$= \frac{|M_r \cap M_g|}{|M_g|}$  [3]. Here,  $M_r$  denotes the number of all relevant documents and  $M_g$  the number of documents found, based on a Cluster  $C$ . The results in the Tables (I), (II) show clearly that the agglomerative clustering method created type classes, which contain structurally meaningful web pages. The high precision values express that the web pages found in a cluster  $C_i$  are actually relevant and the high recall values state that the web pages, which are relevant for Cluster  $C_i$ , were actually found. In comparison with the remaining precision values the cluster  $C_5$  in Table (II) achieves a lower precision value. However,  $C_5$  has a high recall value. That means, (i) some web pages found related for  $C_5$  are not relevant and (ii) all relevant web pages for  $C_5$  were found. Regarding to recall the situation for  $C_4$  in Table (I) is reversed. Hence, we have shown that  $d$  is successfully applicable for the structural filtering of web pages.

## VI. CONCLUSIONS

The main contributions of this paper are shown in Fig (6). Fig (6) shows the overall idea of our new approach for measuring the structural similarity of web-based documents represented by generalized trees. The first main idea from Fig (6) characterizes the transformation of web-based documents into hierarchical and directed graphs we call generalized trees. Then, by transforming the generalized trees into property strings we applied string alignment techniques for measuring the structural similarity of the generalized trees. Finally, by parameterization, that is the functions which evaluate the alignments have parameters, we obtain a high flexibility of our new graph similarity method. From this it is clear that it is possible to define new graph similarity measures  $d_i$  which can be optimized by parameterization for special graph classes. In the present paper we examined only one similarity measure  $d$  presented in Section (IV). In order to motivate our method we stated in Section (II) some results from hypertext category

<sup>2</sup>We used Average Linkage [3].

rization. Following these results we did not concentrate on the content-based hypertext analysis but on the structural analysis of our web-documents. We repeated some problems that graph theoretic indices have in capturing important structural information of graphs. Then, we designed a new method to measure the structural similarity of generalized trees. The new similarity measure is based on the representation of generalized trees as linear strings. We call these strings property strings, because their components represent structural properties of the generalized trees. The similarity of two generalized trees is then defined as the optimal alignment of the corresponding property strings of the generalized trees. From this definition it is clear, that our similarity measure is also different from measures, which are based on isomorphic relations [17], [23]. We demonstrated in Section (V) that the similarity measure  $d$  has important applications in web structure mining [3]. We have shown that the cumulative similarity distribution provides useful information about the test corpus  $T_C$ . On the basis of Definition (5.1) we answered the question how structurally different the conference websites of  $T_C$  are. In our experiment in Section (V) the corpus  $T_C$  comprised 500 generalized trees and the average number of nodes per tree was 23. Because our similarity measure is a parametric measure depending on  $(\zeta, \zeta_m, \sigma_{out}^1, \sigma_{out}^2, \sigma_{in}^1, \sigma_{in}^2, \hat{\sigma}_{out}^1, \hat{\sigma}_{out}^2, \hat{\sigma}_{in}^1, \hat{\sigma}_{in}^2)$  we are able to emphasize different structure types of generalized trees during the evaluation of the alignment. For example, by setting  $\zeta = 1$  we consider a hypertext structure as a directed rooted tree. More precisely, we align only the out-degree property strings induced by edges from the underlying directed rooted tree. If we set  $\zeta = 0$ , we align the property strings induced by in-degree sequences only. In most of the cases we used  $\zeta = \frac{1}{2}$ , which weighs in- and out-degree sequences equally,  $\gamma^{fin} = \frac{\gamma_{out}}{2} + \frac{\gamma_{in}}{2}$ . In the case of labeled generalized trees we considered in Section (V) websites as sets of their DOM-Trees ( $|W_1| + |W_2| = 270$ ). We applied agglomerative clustering to the obtained similarity matrix and found type classes, which contain structurally meaningful web pages. We evaluated the clusters found by comparing them with labels manually assigning to the web pages and found high recall and high precision values. That means web pages found in a cluster  $C_i$  are actually relevant and web pages which are relevant for cluster  $C_i$  were actually found. Hence, our similarity measure  $d$  is applicable to filter web pages according to their structural organization. The interpretation of our results from the filtering process is that similar DOM-Trees tend to have a similar content and layout elements. Furthermore, clusters tell us something about the meaning of web pages. Altogether we found document groups which are comparable on the basis of their structure types.

## REFERENCES

- [1] R. Bellman, *Dynamic Programming*. Princeton University Press, 1957
- [2] R. A. Botafogo, B. Shneiderman: *Structural analysis of hypertexts: Identifying hierarchies and useful metrics*, ACM Trans. Inf. Syst. 10 (2), 1992, 142-180
- [3] S. Chakrabarti: *Mining the Web. Discovering Knowledge from Hypertext Data*, Morgan and Kaufmann Publishers, 2003
- [4] S. Chakrabarti: *Integrating the document object model with hyperlinks for enhanced topic distillation and information extraction*, Proc. of the 10th International World Wide Web Conference, Hong Kong, 2001, 211-220
- [5] I. F. Cruz, S. Borisov, M. A. Marks, T. R. Webb: *Measuring Structural Similarity Among Web Documents: Preliminary Results*, Lecture Notes In Computer Science, Vol. 1375, 1998
- [6] M. Dehmer, *Strukturelle Analyse web-basierter Dokumente*, Ph.D Thesis, Department of Computer Science, Technische Universität Darmstadt, 2005, unpublished
- [7] M. Dehmer, R. Gleim, A. Mehler: *Aspekte der Kategorisierung von Webseiten*, GI-Edition - Lecture Notes in Informatics (LNI) - Proceedings, Jahrestagung der Gesellschaft für Informatik, Informatik 2004, Ulm/Germany, 2004, 39-43
- [8] R. Gleim: *HyGraph – Ein Framework zur Extraktion, Repräsentation und Analyse webbasierter Hypertextstrukturen*, Beiträge zur GLDV-Tagung 2005, Bonn/Germany, 2005
- [9] D. Gusfield: *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*, Cambridge University Press, 1997
- [10] T. Jiang, L. Wang, K. Zhang: *Alignment of trees - An alternative to tree edit*, Theoretical Computer Science, Elsevier, Vol. 143, 1995, 137-148
- [11] S. Joshi, N. Agrawal, R. Krishnapuram, S. Negi: *Bag of Paths Model for Measuring Structural Similarity in Web Documents*, Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), 2003, 577-582.
- [12] Mehler A.: *Textbedeutung. Zur prozeduralen Analyse und Repräsentation struktureller Ähnlichkeiten von Texten*, Peter Lang, Europäischer Verlag der Wissenschaften, 2001
- [13] A. Mehler, M. Dehmer, R. Gleim: *Towards logical hypertext structure. A graph-theoretic perspective*, Proc. of I2CS'04, Guadalajara/Mexico, Lecture Notes in Computer Science, Berlin-New York: Springer, 2004
- [14] A. Mehler, R. Gleim, M. Dehmer: *Towards structure-sensitive hypertext categorization*, to appear in: Proceedings of the 29-th Annual Conference of the German Classification Society, 2005
- [15] S. M. Selkow: *The tree-to-tree editing problem*, Information Processing Letters, Vol. 6 (6), 1977, 184-186
- [16] T. F. Smith, M. S. Waterman: *Identification of common molecular subsequences*, Journal of Molecular Biology, Vol. 147 (1), 1981, 195-197
- [17] F. Sobik, *Graphmetriken und Klassifikation strukturierter Objekte*, ZKI-Informationen, Akad. Wiss. DDR, Vol. 2 (82), 1982, 63-122
- [18] J. R. Ullman, *An algorithm for subgraph isomorphism*, J. ACM, Vol. 23 (1), 1976, 31-42
- [19] P. H. Winne., L. Gupta, J. C. Nesbit: *Exploring individual differences in studying strategies using graph theoretic statistics*, The Alberta Journal of Educational Research, Vol. 40, 1994, 177-193
- [20] A. Winter: *Exchanching Graphs with GXL*, <http://www.gupro.de/GXL>
- [21] Y. Yang, S. Slattery, R. Ghani: *A study of approaches to hypertext categorization*, Journal of Intelligent Information Systems, Vol. 18 (2-3), 2002, 219-241
- [22] K. Zhang, D. Shasha: *Simple fast algorithms for the editing distance between trees and related problems*, SIAM Journal of Computing, Vol. 18 (6), 1989, 1245-1262
- [23] B. Zelinka, *On a certain distance between isomorphism classes of graphs*, Časopis pro řest. Matematiky, Vol. 100, 1975, 371-373