Towards Logical Hypertext Structure
A Graph-Theoretic Perspective

Alexander Mehler[1], Matthias Dehmer[2], and Rüdiger Gleim[2]

[1] Universität Bielefeld, D-33501 Bielefeld, Germany
Alexander.Mehler@uni-bielefeld.de
[2] Technische Universität Darmstadt, D-64289 Darmstadt, Germany
{dehmer,gleim}@informatik.tu-darmstadt.de

Abstract. Facing the retrieval problem according to the overwhelming set of documents online the adaptation of text categorization to web units has recently been pushed. The aim is to utilize categories of web sites and pages as an additional retrieval criterion. In this context, the bag-of-words model has been utilized just as HTML tags and link structures. In spite of promising results this adaptation stays in the framework of IR specific models since it neglects the content-based structuring inherent to hypertext units. This paper approaches hypertext modelling from the perspective of graph-theory. It presents an XML-based format for representing websites as *hypergraphs*. These hypergraphs are used to shed light on the relation of hypertext structure types and their web-based instances. We place emphasis on two characteristics of this relation: In terms of *realizational ambiguity* we speak of functional equivalents to the manifestation of the same structure type. In terms of *polymorphism* we speak of a single web unit which manifests different structure types. It is shown that polymorphism is a prevalent characteristic of web-based units. This is done by means of a categorization experiment which analyses a corpus of hypergraphs representing the structure and content of pages of conference websites. On this background we plead for a revision of text representation models by means of hypergraphs which are sensitive to the manifold structuring of web documents.

1 Introduction

Text representation is a preliminary task of any approach to automatic text analysis. Among other things, this relates to the summarization, categorization, and mining of textual units. Analogously, *hypertext representation* is fundamental to *automatic hypertext analysis* [31]. This comprises, for example, the identification of patterns (e.g. compound documents [10], or small worlds in WWW graphs [1]), the categorization of links [3,27] or the retrieval of information from large hypertext bases [2]. In this context, the bag of words model of IR has been utilized as a starting point for hypertext representation just as HTML metadata, tags and link structures. That is, hypertexts are represented as vectors of features reflecting wording or markup as found in the hypertexts to be analyzed.

T. Böhme et al. (Eds.): IICS 2004, LNCS 3473, pp. 136–150, 2006.

In spite of promising results this approach stays in the framework of *text representation* as elaborated in IR. Particularly with regard to categorization the predominance of traditional representation models is evident: Categorization is designed as an assignment of predefined category labels to feature vectors without the preceding exploration of hypertext structures (e.g. compound nodes or paths).

In this paper we plead for an integrative view of graph-theoretical analysis and categorization. Our starting point is a system of hypertext structure types and their nondeterministic manifestation by web-based units. We propose a four-layer model of hypertext structure types and focus on the many-to-many relation to its instances as units of Internet-based communication. We place emphasis on two characteristics of this relation: In terms of *realizational ambiguity* we speak of functional equivalents to the manifestation of the same structure type. Conversely, in terms of *polymorphism* the phenomenon is addressed that the same hypertext unit may manifest different structure types. Polymorphism occurs when, for example, the same page provides information about different topics (e.g. a page as part of an academic's homepage lists courses beneath her biographical information) or serves different functions (e.g. a page offers the registration form of a research group beneath its brief description).

Our central hypothesis is that (comparable to natural language texts) realizational ambiguity and polymorphism are prevalent characteristics of web-based units. This has fundamental implications for hypertext categorization which normally presupposes to result in a non-overlapping separation of the object space, i.e. into an assignment of at most one category per object. If polymorphism is prevalent in this area, it does not make sense to view hypertext categorization as a process of disambiguating category assignments. As a consequence, two implications have to be balanced: Either the category system has to be revised, or – and this is our central thesis – the object space has to undergo a structural analysis as the result of which categorization *and* segmentation of the focal objects occurs. Since we view polymorphism to be a characteristic of web-based hypertexts, we expect *multiple*, *interdependent* categorizations to occur regularly. In other words: Proper hypertext categorization is bound to a preliminary structure analysis in which the regular realizational ambiguity and polymorphism of hypertext units is resolved. In order to support this line of argumentation we present a categorization of web pages of an area which is supposed to follow more stable authoring patterns and thus to be a profitable field of categorization: conference websites. This analysis operates on an XML-based representation format of hypertexts whose presentation is the second central focus of this paper. It is based on the idea to represent web-based units, their content and links as attributed typed directed nested *hypergraphs* [5].

The paper is organized as follows: After an outline of related work, our conceptual framework is presented in section (3): a four-layer model of hypertext structure types. This framework is used as the background of an XML-based format for representing web-based hypertexts as hypergraphs. The basic idea is to combine data-oriented representations of link structure, wording and markup

in a uniform model. In other words: The format integrates information relevant to structural analysis *and* categorization. It is proposed as a data-oriented alternative to the document-oriented representation of web pages as DOM trees based on their HTML tags [7]. Section (5) describes the automatic mapping of a corpus of conference websites onto a set of categories. The uniqueness of category assignment is measured in order to shed light on the range of polymorphism in the area under consideration. The practical relevance of our study is outlined in section (6). The paper closes with some conclusions.

2 Related Work

Structural analysis is a much considered topic in hypertext research. Beginning with the seminal article of Botafogo et al. [6], graph theory was utilized as a generic format of hypertext representation. This relates, for example, to the identification of spanning trees in hypertexts [6]. The idea is to markup hypertextual aggregates in order to enhance retrieval and browsing [22]. More recently, candidates of web-based hypertext types (e.g. web hierarchies, directories, corporate and web sites) have been identified by exploring the link structure of their pages [4]. Moreover, plenty of approaches deal with the structuring of single such types [24], their constitutive paths [20] and intentionally defined structural units as, for example, *compound documents* and their leaders [10], *logical domains* and their entry pages [19] or *logical documents* [28]. Whereas these approaches operate on the level of websites and their constituents, another group of approaches focuses on macrostructures. This relates to the distribution of links per node and web-topologies based on these distributions (e.g. the small-world problem [1]) as well as clusters of interrelated web pages and their hubs [9,16,21,22].

This paper also explores aggregates in web-based hypertexts which it conceives as informationally uncertain instances of latent authoring patterns. According to this view, structural analysis is always concerned with two aspects of markup: it includes (i) the exploration and annotation of patterns in concrete hypertexts whose frequent observation allows (ii) their abstraction as hypertext structure types which in turn serve as a precondition of pattern identification. In other words: We view structural analysis always to aim at a model of the underlying authoring patterns, and not only at a segmentation and annotation of their instances. As a preliminary step towards such a *grammar of authoring patterns*, the paper integrates three domains of research: graph theoretic modelling, structural analysis and explorative data analysis.

3 Logical Hypertext Structure

Our starting point is a four-level hypertext model which relies on the distinction of *abstract* structure types and their *concrete* instances as observable parts of the web (see figure 1). According to this view, compound hypertext document types are representations of stable authoring patterns on the level of homepages (e.g. academics' personal homepages). They are defined as systems of more elementary

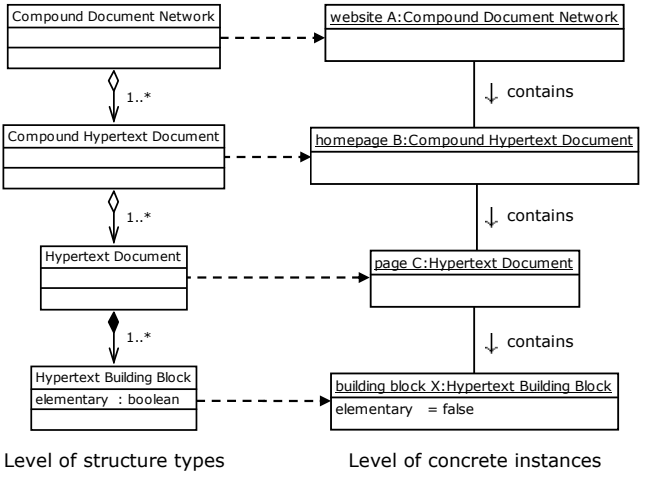

Fig. 1. A class and object diagram of hypertext structure types and their instances

hypertext document types and their regular dependency relations, whereby these document types are defined on their own as regular compositions of types of either complex or elementary building blocks. This multilevel type system is instantiated on four levels (see fig. 1):

1. On the lowest level *types of hypertext building blocks* are manifested by paragraph tags, horizontal rules, lists, tables, etc. Since these building blocks exist only as dependent parts of pages as elementary self-contained units of web-based communication, diagram (1) associates them with document types by *composition* which in UML is symbolized by a filled rhombus.
2. *Hypertext document types* and their dependency relations are instantiated by one or more web pages and their links, respectively. In the literature on web-based documents, these dependency relations have predominantly been classified with the help of rhetorical structure theory [17,26,29]. In terms of the Dexter Hypertext Reference Model [13] this means that the rhetorical relation (e.g. elaboration, exemplification, contrast) served by the target of a link l with respect to its source determines the type of l.
3. *Compound hypertext document types* are instantiated by *compound documents* in the sense of [10], i.e. by networks of pragmatically homogeneous web pages as instances of document types whose coherence is provided by the uniform intention of one or more hypertext authors. Document types are distinguished from *compound* document types by not only being pragmatically, but also functionally homogeneous in the sense of serving the same content function.
4. Finally, *compound document network types* are instantiated by *websites* and thus are characterized by pragmatic *and* content-based heterogeneity.

This multilevel type-instance model is illustrated by *conference websites* whose regular composition supports their expectation-driven processing: They

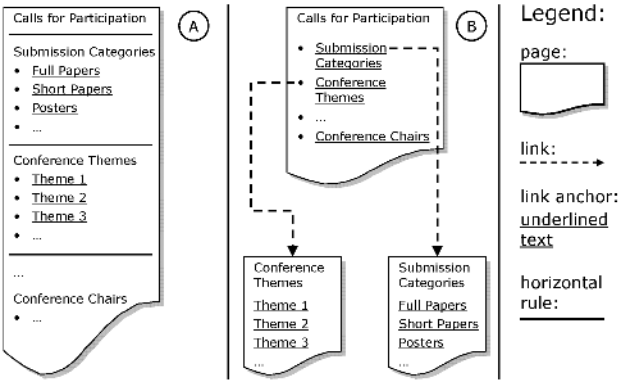

Fig. 2. Schematic drawings of functionally equivalent presentations of the same information as (A) a list and (B) a compound document of three pages

typically consist of pages on calls for participation, the conference program, additional workshops and tutorials etc. as instances of compound hypertext document types. The regular composition of conference websites can be observed on the level of their constitutive compound documents, too: A call for participation typically includes units about submission categories, topics, chairs and the submission procedure, whether they occur on the same page or not. Last but not least, the regular manifestation of building blocks is illustrated by HTML lists as the preferred means of enumerating conference topics.

The distinction of building blocks, document types and their recursive composition to networks of such types follows the linguistic differentiation of elementary text patterns and text (network) types. Evidently, this analogy does not stop with the alignment of both type systems. Rather, it is continued by what is called *realizational ambiguity*: Just as the same text pattern may be realized by different, but functionally equivalent text structures [23], the same kind of realizational ambiguity is observed in web-based communication. In other words: The realization of hypertext document types by websites, web pages, and (X)HTML building blocks is nondeterministic – their exist *functional equivalents* to the realization of the same hypertext structure type. This is once more illustrated by conference websites: In order to inform about a conference's calls for participation, the hypertext author may list all submission categories and conference themes as well as information about the submission procedure on one single page, possibly structured by horizontal rules and section headings (see figure 2.A). Functionally equivalent to this alternative is an instance of a document type leaded by a web page which enumerates the headers 'submission categories', 'conference topics', etc. as anchors of links to pages listing the corresponding subsets of categories (see figure 2.B). Obviously, these two alternatives do not exhaust the range of functional equivalents existing in this area. Moreover, realizational ambiguity is accompanied by what is called *polymorphism*. That is, the same web page may realize *different* hypertext document types. In

terms of conference websites polymorphism occurs when for example tutorials are listed on the same page as submission categories. The upper bound of this polymorphism is given by a conference website which manifests all constitutive document types (e.g. calls for participation, program, workshops, etc.) on a *single* page. Polymorphism and realizational ambiguity constitute the *n*:*m*-relation of hypertext structure types and their instances as distinguished in figure 1. Obviously, the range of this relation has not been explored so far.

In order to grasp this type-instance relation, we utilize the document structure model of [23] which has recently been proposed to account for the divergence of rhetorical, logical, and layout structure in written texts. Power et al. systematically report on examples where the same document structure is manifested by different but functionally equivalent texts. The added value of this approach is that it proposes a *document grammar* which supposes hierarchical text structures. This is in accordance with the linguistic tradition to represent structures as hierarchies as reflected by the OHCO-model, which describes texts as Ordered Hierarchies of Content Objects [25]. There is a long debate on the adequacy of the OHCO-model, which for the time being resulted in the proclamation of a poly-hierarchical text structure model [25]. It is well-known that even poly-hierarchies do not adequately model the network structures of hypertexts. This also holds for instances of (compound) hypertext document types: Although they prove to have a kernel hierarchical structure they also contain page internal links as well as *outside links* [4,10] which link pages of different sites and thus transcend the kernel hierarchy.

On this background, we view an extension of the model of Power et al., which also accounts for non-hierarchical links transcending the kernel (poly-)hierarchy, to be the favourite candidate for modelling the relation of hypertext structure types and their instances as shown in figure 1. This model is referred to by the term *LOGical hypertext dOcument Structure* (LOGOS). Its assumptions are:

- Observable web-structures are instances of usage-based and hence probabilistic hypertext structure types on at least four levels (see figure 1).
- The relation of these types to their instances is characterized by realizational ambiguity and polymorphism.
- Compound document types impose a kernel hierarchical structure on their instances on the basis of hierarchy constitutive links.
- An adequate model of hypertext structure types and their dependency relations is a *probabilistic grammar* which represents prototypical web structures on the level of compound document types.

A probabilistic grammar of compound hypertext documents is a formal representation of the yet unknown range of realizational ambiguity and polymorphism in this area. The goal of this paper is to shed light on this range as a preliminary study to such a grammar. It is approached by a quantitative analysis of the structure of conference websites as described in the subsequent sections.

4 Hypertext Representation

In order to approach a grammar of compound hypertext documents, a format is needed which allows to represent web-based hypertexts in a *uniform* and *standardized* as well as *flexible* and *extensible* way. That is, we do not only need a format expressive enough to represent the range and structural variety of functional equivalents of hypertext structure types. We also need a format which proves to be conceptually clear and computationally processable as regards the divergent tasks of automatic hypertext analysis. In order to approach this task of *adequate hypertext representation*, three requirements have to be met:

1. Page-internal and external, outside, inside, up, down and across links have to be mapped as well as the graph structures they induce (e.g. sequences, hierarchies and networks of interlinked pages).
2. User and system perspective (i.e what is seen on the screen vs. its underlying markup) have to be kept apart without ignoring their reciprocal mapping.
3. Link and node classification are two more use cases of machine learning. They demand a hypertext model which includes representations even of the wording of single pages – comparable to the bag-of-words model of IR, but with the difference that now graphs of such representations have to be managed since web pages are embedded into hypertext graphs.

The subsequent section presents an XML-based format for representing hypertexts as graphs. It is based on the Graph eXchange Language (GXL) [30], which has been developed as an XML-based format for data interchange between information systems. We utilize this format for hypertext representation:

4.1 XML-Based Hypertext Representation

Hypertext representation serves to map hypertexts onto instances of a format which supports the different tasks of hypertext analysis. The idea of our approach is that hypertexts are adequately modelled as graphs. This is in accordance with hypertext modelling [6,12]. We continue this tradition by using the GXL DTD against which automatically generated *hypertext graphs* are validated. The graph model of the GXL distinguishes six graph classes which we utilize as follows:

Graphs are ordered pairs (V, E) of a vertex set V and an edge set E. In the GXL, vertices are referred to as XML-elements named `node`. In the LOGOS framework, instances of this element are commonly used to represent single web pages identified by an `ID` and an GXL-attribute named `URI` (see table 1). Accordingly, instances of the elements `edge` and `rel`(ation) are commonly used to represent links of these nodes (see below).

Typed graphs are graphs with typed vertices and edges. We use typing to distinguish anchor nodes and page nodes as well as standard links and frame source links. Typing is manifested by the `type` element and its `xlink:href` attribute. Since we need several type systems to independently classify the same set of hypertext constituents, we also construct attributed graphs:

Table 1. A hypertext graph of a conference website (dots indicate omitted content)

```
<gxl xmlns:xlink="http://www.w3.org/1999/xlink">
  <graph hypergraph="true" edgemode="directed" id="HyperGraph0">
    <attr name="Title"><string>Hypertext 2004</string></attr>
    <node id="Page1">
      <attr name="URI"><locator xlink:href="http://www.ht04.org"/></attr>
      <attr name="Title">
        <string>Hypertext 2004 - Fifteenth Annual Conference on
        Hypertext and Hypermedia</string>
      </attr>
      <graph hypergraph="false" edgemode="directed" id="EmbeddedGraph2">
        <node id="Anchor5"> ... </node> ...
      </graph>
    </node> ...
    <node id="Page843">
       <attr name="URI">
         <locator xlink:href="http://www.ht04.org/cfpPapers.php"/>
       </attr>
       <attr name="Title">
         <string>Hypertext 2004 - Papers Call for Participation</string>
       </attr> ...
    </node> ...
    <rel id="HyperLink875">
      <attr name="LinkStructureType"><enum>kernellink</enum></attr>
      <relend direction="in" target="Page1" role="sourcepage"/>
      <relend direction="in" target="Anchor5" role="sourceanchor"/>
      <relend direction="out" target="Page843" role="targetpage"/>
    </rel> ...
  </graph>
</gxl>
```

Attributed graphs are graphs whose nodes and edges are assigned possibly nested bags, sets, tuples or sequences of boolean, integer, real or string valued attributes. In hypertext representation they are inter alia used to model the URL of a web page and its metatags as an attribute-value pair and a bag of such pairs enclosed by an GXL-attribute named `MetaTags`, respectively. Analogously, the content of a page (i.e. the wording and HTML tags enclosed by its `body` element) is modelled by a `TokenVector` element. This vector serves as a representation model in automatic hypertext categorization (see section 5). Furthermore, links are assigned a GXL-attribute named `LinkType` whose values distinguish between up, down, inside, outside and across links (see below). Finally, the GXL attribute model may also be used to represent HTML lists, tables and embedded objects as attributes of nodes.

Directed graphs are graphs whose edges are ordered pairs of nodes, *adjacent from* their source and *adjacent to* their target node. They are the default means of representing HTML links whose source and target anchors belong to the same web page, i.e. page internal links (see link C in figure 3). This is done

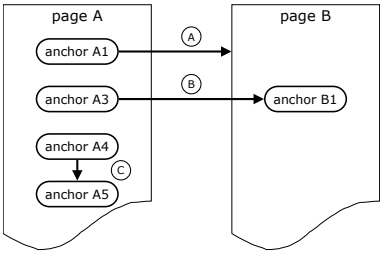

Fig. 3. Internal (C) and external links with (B) and without (A) target anchor

with the help of two attributes assigned to the `edge` element (see table 1): `from` and `to` take as values the `ID` of the corresponding source and target node anchor, respectively. Thus, they behave as `IDREF` attributes in XML. In spite of this preferred usage, `edge` elements, their attributes and content model are not restricted to map only HTML links. As the GXL model of hypergraphs (see below) shows, even sophisticated links following the XLink standard can be modelled in GXL – it is this expressive power of the GXL which underlies our decision to use it as the primary format for representing web-structures.

Ordered graphs are directed graphs whose arcs are assigned ordinal numbers reflecting any order dependent on their respective source node. In linguistics, such assignments can be used to model the syntagmatic order of immediate constituents of superordinate nodes. In hypertext representation they are analogously used to model the order of links (dependent on the textual order of their anchors) which are adjacent *from* the same node. This order is manifested by a GXL-attribute named `order` and assigned to the respective `rel`(ation) element.

Stratified graphs are graphs whose nodes (may) embed graphs on their own. In hypertext representation they are used to model page-internal structures composed of links whose source and target anchors belong to the same page (see link C in figure 3). In order to map this membership, the graph spanned by the links internal to a page is included by the content model of the node modelling this page. Following this approach, the GXL realizes a kind of document-oriented modelling – *complementing its predominant data-oriented character*. Since page-internal links simply consist of a (possibly attributed) association of two anchors belonging to the same page, the `edge` element suffices as the GXL analogon of graph theoretic edges to model this kind of links. In case of all other links, so called hyperedges of hypergraphs are used:

Hypergraphs are graphs whose *hyper*edges are subsets of the vertex set V. Hyperedges may also be ordered and directed. This qualifies them for modelling HTML links whose anchors belong to different web pages (see link B in figure 3). Table (1) illustrates an instance of the element `rel`(ation) which models a link of two pages (identified by `Page1` and `Page843`). The content model of the hyperedge in question comprises a `relend` element targeting at `Page1` as its `sourcepage`, a `rel`(ation)`end` targeting at `Page843` as its `targetpage`, and a `relend` element targeting at the link's source page anchor.

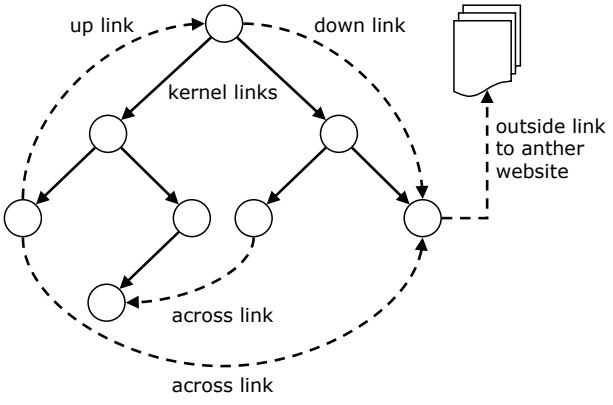

Fig. 4. The kernel hierarchy with additional up, down, across and outside links

Links with a target anchor specification in the URL value of their `href` attribute (see link B in figure 3) are modelled as `rel` elements with a `relend` element of role `targetanchor`. Since relation ends can be extended by any GXL-attribute and since hyperedges of this kind are not restricted regarding the number of their targets, they allow to model any relation of any valency. In other words, hyperedges are the preferred means to represent complex links of the XLink standard *using the same representation format as in case of* HTML *links*.

With the help of the GXL, web-based hypertexts on the level of compound documents, their constitutive pages and the internal link structure of these pages are mapped onto XML documents – henceforth called *hypertext graphs* – as instances of the GXL DTD. As claimed by the LOGOS model in section (3), these hypertext graphs have a kernel hierarchical structure whose constitutive links are tagged as `rel`(ation) elements of `LinkType kernellink` (see for example `HyperLink875` in table 1). This kernel hierarchy is illustrated by a conference website leaded by a title page and continued by a page on the corresponding call for papers which in turn is continued by a page on style sheets, etc. The root of the kernel hierarchy of a hypertext graph is identified by the GXL-attribute `RootID` of the global `graph` element. The web page corresponding to that root has to be explored in advance as the leader (or home page) in the sense of [10]. The compound document leaded by that page is for the time being identified by a breadth first search starting with the leader and identifying the spanning tree rooted by it.

Kernel links are distinguished from up, down, inside, outside and across links [4,10,26], which in the following are defined on the basis of the kernel hierarchy spanned by the kernel links of the respective hypertext graph (see figure 4):

Kernel links associate dominating nodes with their immediately dominated successor nodes.

Down links associate nodes of the kernel hierarchy with one of their (dominated) successor nodes in terms of that kernel hierarchy.

Up links associate analogously nodes of the kernel hierarchy with one of their (dominating) predecessor nodes.

Across links associate nodes of the kernel hierarchy none of which is an (im-)mediate predecessor of the other in terms of the kernel hierarchy.

Inside links are node (i.e. page) internal links.

Outside links associate nodes of the kernel hierarchy with nodes of other websites.

These types of links are tagged by means of the GXL-attribute named `LinkType` whose string-value belongs to the set {`uplink`, `downlink`, `insidelink`, `outsidelink`, `acrosslink`}. Table (3) lists the frequencies of these types as found in our test corpus of 13,481 pages of 1,000 conference/workshop websites from computer science and mathematics.

According to our LOGOS model, web-based hypertexts are represented as typed attributed directed ordered hypergraphs supplemented by graph stratification and markup of a kernel hierarchy. This is done with the help of the GXL as a uniform format for representing hypertexts, their nodes, links and other building blocks. These hypertext representations serve as a *uniform* input/output format of any subsequent qualitative/quantitative hypertext analysis. It is this general requirement hypertext graphs are generated to meet.

5 Hypertext Categorization

Hypertext categorization is the task of automatically assigning category labels to hypertext units [8]. It utilizes HTML markup, metatags and link structure beyond representations of the units' wording as input of feature selection [11,31]. In our categorization experiment pages of conference websites are categorized (see table 2). The aim is to shed light on the range of polymorphism which we expect to be accompanied by a regular multiple-categorization. If polymorphism is a characteristic of web-based units, their pages cannot serve as the elementary unit of hypertext categorization, since polymorphic pages simultaneously instantiate several categories. In order to verify this we use Support Vector Machine (SVM) classification which proves to be successful in text categorization especially in cases of high dimensional, sparse and noisy feature vectors [15]. Handling noisy data is of outmost importance in hypertext categorization since we cannot expect to observe instances of well established authoring practices comparable to well established text types.

To implement this we use the LibSVM implementation of SVMs [14]. We use C-SVM and an RBF-*kernel function* of type $K(u,v) := e^{-\gamma\|u-v\|^2}$. In order to derive optimal parameter vectors (C,γ) for our training sets we perform a search in the parameter space $P := \{(C,\gamma)|C = 2^g, \gamma = 2^s, g \in \{-4,0,4,\ldots,20\}, s \in \{-16,-12,-8,\ldots,8\}\}$ in combination with a 5-fold cross validation. For each category we choose those parameter vector which minimizes the error of cross validation with respect to the training set. As features all tokens and HTML tags enclosed by the `body` and `head` element of the respective page are used. The training set of category C_i was designed as follows: starting from the overall

Table 2. The set of categories and their uniqueness coefficient U_i

category	label	prec.	recall	acc.	#matchings	U_i
submission and author instructions	C_1	29,1%	99,0%	70,8%	2107	0,10
call for papers	C_2	41,6%	99,0%	82,5%	2661	0,05
important dates	C_3	41,2%	99,0%	90,4%	1992	0,05
committees	C_4	50,0%	99,2%	88,2%	1546	0,24
accepted papers	C_5	66,6%	99,0%	72,1%	3846	0,02
topics and general information	C_6	35,0%	99,1%	90,4%	3616	0,02
program	C_7	25,5%	66,0%	68,4%	2716	0,14
travel and accommodation	C_8	50,0%	99,2%	80,3%	2245	0,03
venue	C_9	32,0%	99,0%	66,3%	3045	0,02
invited speakers	C_{10}	25,0%	99,0%	80,1%	2206	0,01
registration	C_{11}	46,1%	99,0%	71,3%	3339	0,03
sponsors	C_{12}	41,6%	99,0%	82,9%	4627	0,03
workshops	C_{13}	52,1%	99,2%	94,1%	1141	0,02

training set (of about 800 pages) all positive examples of C_i were selected and a random sample of negative examples was chosen whose cardinality equaled the set of positive examples of C_i in the overall set, whereby the negative examples were uniformly distributed over the set of remaining categories. The evaluation of performance on the basis of our corpus (see table 3) is shown in table 2. It demonstrates very high recall, but low precision values; with the exception of category C_7, the categories are in many cases wrongly applied. In other words: the categorization is highly error-prone. This result is confirmed by the uniqueness coefficient $U_i \in [0, 1]$ which relates the number of test cases assigned solely to category C_i to the total number of assignments to this category, where $[\![C_i(u)]\!] = 1$ iff the page u belongs to category C_i; $|C| = 13$ is the cardinality of the set of categories:

$$U_i := \frac{|\{u \in U \mid C_i(u) \wedge \neg(C_1(u) \vee \ldots \vee C_{i-1}(u) \vee C_{i+1}(u) \vee \ldots \vee C_{|C|}(u))\}|}{|\{u \in U \mid C_i(u)\}|}.$$

Table (2) demonstrates the extremely low discriminatory power of our category set. There are at least four possible reasons for this deficiency: Either, the category set is erroneous in the sense of not being fine-grained enough, for example, or the training corpora have to be redesigned, or SVM categorization has to be replaced by another method, or – and this is our preferred reading – the pages in question are systematically polymorphic: they simultaneously manifest – as a sample of the websites has shown – more than one function in conference announcement. That is we do not directly cast doubt on the category set, but rather argue for a preceding exploration of patterns of page internal structures in order to disentangle functional equivalents and polymorphic units as a preliminary step to any categorization. In this sense, hypertext categorization is bound to an integration of vector space and structure oriented models.

Table 3. The corpus of conference web sites used in the categorization experiment

Attribute	Value	Attribute	Value
Number of web sites	1,000	Number of up links	10,535
Number of web pages	13,481	Number of down links	13,012
Number of frame set links	1,236	Number of across links	43,145
Number of kernel links	12,382	Number of internal links	6,323

6 Applications

In hypertext analysis various graph theoretic measures are investigated [6]. These measures generally ignore syntactic, semantic and pragmatic types of nodes and links as well as the sub-structures they induce. We plan to develop measures which focus on that deficit. We especially focus on measures that are based on node sequences and sequence alignments. Measures and *spectral algorithms* to describe maximal subgraphs in the sense of semantic similar regions will be defined on the Matrix $(s_{ij})_{ij}, 1 \leq i \leq n, 1 \leq j \leq n, i, j \in \mathbb{N}$, where s_{ij} is the similarity between the text sequences representing the nodes v_i and v_j computed by the metric explained in [18]. The application area of these measures is the exploration of WWW patterns, i.e. the classification of web-units based on fine-grained representations of their structuring.

A further application area of structural hypertext analysis is the enhancement of browsing and information retrieval from large hypertext bases. The idea is to gain an additional guideline for browsing and retrieval from the markup of hypertext aggregates above the level of elementary nodes and links. We also follow this line of argumentation, but propose a further field of application, namely *large-scale hypertext authoring and maintenance*. This proposal follows Power et al. [23] who describe a text authoring tool which allows a text designer to chose among a set of parameters which control the manifestation of the same logical document structure by different texts. Analogously, we plan an authoring tool which automatically produces and maintains web pages on the basis of their GXL hypertext graphs and the designers choice of functional equivalents to the manifestation of hypertext structure types. The application scenario of this tool is a large scale intranet or website with a tremendous number of heterogeneously designed web pages. The first task is to map these pages onto a set of GXL hypertext graphs. The next step would be a redesign by standardizing, for example, the functional equivalents to the manifestation of the same structure type (e.g. presentation of project descriptions). That is, we plan to use hypertext representation not only for the *analysis* of existing, but also for the *synthesis* of new web documents.

7 Conclusions

Starting from a four-level hypertext structure model we introduced the concepts of realizational ambiguity and polymorphism. That is, we view web-based units

to be informationally uncertain manifestations of latent authoring patterns. We argued that in order to derive an adequate hypertext model this informational uncertainty has to be explored first. In order to show this we performed a categorization experiment according to which even higher performing categories do not prove to be stable predictors of hypertext units – due to multiple and fuzzy categorizations. Our conclusions are twofold: First, we argue for structure analysis which tries to identify functional equivalents to the manifestation of hypertext structure types and to resolve the pages' inherent polymorphism as a preliminary step of any hypertext categorization. We are convinced that without such an analysis it does not make sense to compare different websites since there may exist hypertext graphs which manifest the same structure types by means of radically divergent surface structures and at the same time there may exist graphs which manifest divergent structure types by means of similar surface structures. Second, we plead for a reconstruction of (hyper-)text representation models used in content analysis, which depart form the vector space model in the sense that they map both, the structure and content of units to be represented.

References

1. L. A. Adamic. The small world of web. In S. Abiteboul and A.-M. Vercoustre, editors, *Research and Advanced Technology for Digital Libraries*, pages 443–452. Springer, Berlin/Heidelberg/New York, 1999.
2. M. Agosti and A. F. Smeaton. *Information Retrieval and Hypertext*. Kluwer, Boston, 1996.
3. J. Allan. Automatic hypertext link typing. In *Proceedings of the 7th ACM Conference on Hypertext*, pages 42–52. ACM, 1996.
4. E. Amitay, D. Carmel, A. Darlow, R. Lempel, and A. Soffer. The connectivity sonar: detecting site functionality by structural patterns. In *Proc. of the 14th ACM conference on Hypertext and Hypermedia*, pages 38–47, 2003.
5. C. Berge. *Hypergraphs: Combinatorics of Finite Sets*. North Holland, Amsterdam, 1989.
6. R. A. Botafogo, E. Rivlin, and B. Shneiderman. Structural analysis of hypertexts: Identifying hierarchies and useful metrics. *ACM Transactions on Information Systems*, 10(2):142–180, 1992.
7. S. Chakrabarti. Integrating the document object model with hyperlinks for enhanced topic distillation and information extraction. In *Proc. of the 10th International World Wide Web Conference, Hong Kong, May 1-5*, pages 211–220, 2001.
8. S. Chakrabarti, B. Dom, and P. Indyk. Enhanced hypertext categorization using hyperlinks. In L. Haas and A. Tiwary, editors, *Proceedings of ACM SIGMOD International Conference on Management of Data*, pages 307–318. ACM, 1998.
9. S. Chakrabarti, M. Joshi, K. Punera, and D. M. Pennock. The structure of broad topics on the web. In *Proc. of the 11th Internat. World Wide Web Conference*, pages 251–262. ACM Press, 2002.
10. N. Eiron and K. S. McCurley. Untangling compound documents on the web. In *Proceedings of the 14th ACM conference on Hypertext and hypermedia, Nottingham, UK*, pages 85–94, 2003.
11. J. Fürnkranz. Using links for classifying web-pages. Technical report, TR-OEFAI-98-29, 1998.

12. J. Furner, D. Ellis, and P. Willett. The representation and comparison of hypertext structures using graphs. In M. Agosti and A. F. Smeaton, editors, *Information Retrieval and Hypertext*, pages 75–96. Kluwer, Boston, 1996.
13. F. Halasz and M. Schwartz. The Dexter hypertext reference model. *Communications of the ACM*, 37(2):30–39, 1994.
14. C.-W. Hsu, C.-C. Chang, and C.-J. Lin. A practical guide to SVM classification. Technical report, Department of Computer Science and Information Technology, National Taiwan University, 2003.
15. T. Joachims. *Learning to classify text using support vector machines*. Kluwer, Boston, 2002.
16. J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.
17. R. Kuhlen. *Hypertext: ein nichtlineares Medium zwischen Buch und Wissensbank*. Springer, Berlin/Heidelberg/New York, 1991.
18. M. Li, X. Chen, L. Xin, B. Ma, and P. M. Vitányi. The similarity metric. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 863–872. ACM Press, 2003.
19. W.-S. Li, O. Kolak, Q. Vu, and H. Takano. Defining logical domains in a web site. In *Proc. of the 11th ACM on Hypertext and Hypermedia*, pages 123–132, 2000.
20. Y. Mizuuchi and K. Tajima. Finding context paths for web pages. In *Proceedings of the 10th ACM Conference on Hypertext and Hypermedia*, pages 13–22, 1999.
21. S. Mukherjea and Y. Hara. Focus+context views of world-wide web nodes. In *Proceedings of the eighth ACM conference on Hypertext*, pages 187–196, 1997.
22. P. Pirolli, J. Pitkow, and R. Rao. Silk from a sow's ear: Extracting usable structures from the web. In *Proc. of the ACM SIGCHI Conference on Human Factors in Computing*, pages 118–125. ACM Press, 1996.
23. R. Power, D. Scott, and N. Bouayad-Agha. Document structure. *Computational Linguistics*, 29(2):211–260, 2003.
24. G. Rehm. Towards automatic web genre identification – a corpus-based approach in the domain of academia by example of the academic's personal homepage. In *Proc. of the Hawai'i Internat. Conf. on System Sciences*, January 7-10 2002.
25. A. Renear. Out of praxis: Three (meta)theories of textuality. In K. Sutherland, editor, *Electronic Text. Investigations in Method and Theory*, pages 107–126. Clarendon Press, Oxford, 1997.
26. L. Routledge, B. Bailey, J. van Ossenbruggen, L. Hardman, and J. Geurts. Generating presentation constraints from rhetorical structure. In *Proceedings of the 11th ACM Conference on Hypertext and Hypermedia*, pages 19–28. ACM, 2000.
27. E. Spertus. ParaSite: mining structural information on the web. In *Selected papers from the sixth international conference on World Wide Web*, pages 1205–1215. Elsevier, 1997.
28. K. Tajima and K. Tanaka. New techniques for the discovery of logical documents in web. In *Internat. Symposium on Database Applications in Non-Traditional Environments*, pages 125–132. IEEE, 1999.
29. M. Thüring, J. Hannemann, and J. M. Haake. Hypermedia and cognition: Designing for comprehension. *Communications of the ACM*, 38(8):57–66, 1995.
30. A. Winter, B. Kullbach, and V. Riedinger. An overview of the GXL graph exchange language. In S. Diehl, editor, *Software Visualization*, pages 324–336. Springer, Berlin/Heidelberg, 2002.
31. Y. Yang, S. Slattery, and R. Ghani. A study of approaches to hypertext categorization. *Journal of Intelligent Information Systems*, 18(2-3):219–241, 2002.