Intelligent Information Systems 9999
ISBN 666-666-666, pages 1–14

Automatic Web Site Authoring with SiteGuide

V. de Boer, V. Hollink, and M.W. van Someren

Human-Computer Studies Laboratory, Informatics Institute, Faculty of Science, University of Amsterdam

Abstract

An important step in the design process for a web site is to determine which information is to be included and how the information should be organized on the web site's pages. In this paper we describe 'SiteGuide', a tool that automatically produces an information architecture for a web site that a user wants to create. SiteGuide takes as input URLs of web sites from the same domain as this target site and creates a model based on the common topics and structural features in these example sites. The tool presents this model to the user in various formats and it serves as a starting point for the information architecture of the new web site. Evaluation shows that SiteGuide is able to detect a large part of the common topics in example sites and to present these topics in an understandable form to its users. Web site set-ups made by amateur web designers that have used the SiteGuide's suggestions are of a higher quality than those made without the help of SiteGuide.

Keywords: web site authoring, web site clustering, information architecture

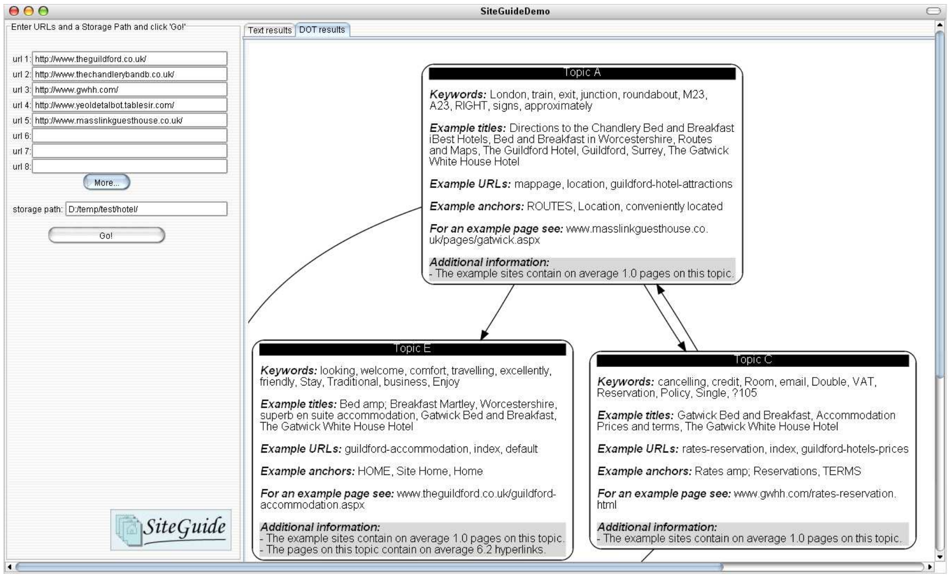
1 Introduction

Even the smallest companies, institutes and associations are expected to have their own web sites. However, designing a web site is a difficult and time-consuming task. Software tools that provide assistance for the web design process can help both amateur and professional web designers. In this paper we present a tool that supports designing the information architecture of a website.

Newman and Landay (2000) studied the current practices in web design and identified four main phases in the design process of a web site: discovery, design exploration, design refinement and production. A number of existing tools, such as Adobe Dreamweaver[1] and Microsoft Frontpage[2] provide help for the latter two phases, where an initial design is refined and implemented. For the discovery and design exploration phase, sketching tools such as DENIM (Lin *et al.*, 2000) have been developed to visually represent identified content and structure of the web site. These tools however, do not support collecting and structuring the content into an initial conceptual model (Falkovych and Nack, 2006).

[1] http://www.adobe.com/products/dreamweaver
[2] http://office.microsoft.com/frontpage

Figure 1: A screenshot of the SiteGuide system showing the graph representation of a number of topics for a web site of a hotel.



An important step in the discovery phase of web site design is reviewing web sites from the same domain as the target site (Newman and Landay, 2000). For instance, a person who wants to build a site for a small soccer club will often look at web sites of some other small soccer clubs. The information architectures of the examined sites are used as source of inspiration for the new site. Reviewing example sites can provide useful information, but comparing sites manually is very time-consuming and error-prone, especially when the sites consist of many pages.

In this paper, we present 'SiteGuide', a system that helps web designers to create a setup for a new site. Its output is an initial information architecture for the target web site that shows the user what information should be included in the website and how the information should be structured. This information is presented to the user in both textual and visual representation of the identified information. It can be used as a standalone tool or its output can serve as a starting point for further design refinement. Usually, web designers are not experts on the content or domain of a new site and the domain experts are no designers. The goal of SiteGuide is to assist both groups by creating a first description of the content topics with a tentative structure for the site. Figure 1 shows a screenshot of the SiteGuide tool.

The SiteGuide system creates an initial information architecture for a new site by efficiently and systematically comparing a set of example sites identified by the user. SiteGuide automatically searches the sites for topics and structures that the sites have in common. For example, in the soccer club domain, it may find that most example sites contain information about youth teams or that pages about membership always link to pages about subscription fees. The common topics are brought together in a model of the example sites. The model is presented to the user and serves as an information architecture for the new web site.

SiteGuide can also be used in the design refinement phase of the web design process as a critic of a first draft of a site. The draft is compared with the model, so that missing topics or unusual information structures are revealed.

Section 2 describes the SiteGuide tool and how the commonalities are identified. In Section 3 we evaluate the tool. We compare our work to related work in Section 4 and end this paper with conclusions in Section 5.

2 The SiteGuide Tool

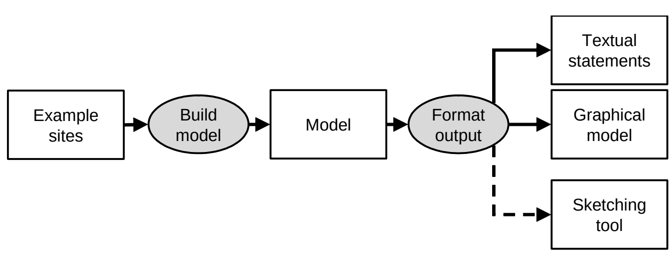
The main usage scenario for the Siteguide tool is the *modeling scenario* shown in Figure 2. In this scenario, the user employs SiteGuide to generate a first set-up for a web site based on a number of example sites. The user starts the interaction by inputting the URLs of the home pages of a small set of example web sites (typically 3 to 10). SiteGuide then scrapes and analyzes the sites and captures their commonalities in a web site model. This model forms the suggested information architecture for the new site. A model consists of a set of *topics* that appear in the example sites, which can be outputted in various formats, as described in Section 2.3 and 2.4. Section 2.1 describes the method used in the SiteGuide tool to extract the topics. A more detailed description of the method can be found in Hollink *et al.* (2009).

2.1 Clustering web pages

To construct the web site model, SiteGuide identifies common topics that occur on most example sites. For this, SiteGuide identifies pages of different example sites that handle on the same topic and forms clusters of these pages.

The clustering method must allow that pages appear in more than one cluster (a single page includes text about different topics). Also, a cluster may not contain a page from each site, because a topic may not be included in each site. Pages occur in more than one cluster, when they contain content about more than one topic.

FIGURE 2: The main usage scenario of the SiteGuide system.



As in most clustering tasks, the quality of an example site clustering is better when the pages in the clusters are more similar to each other. However, in most domains pages of one site are more similar to pages on the same site that handle on other topics than to pages of other sites on the same topic. As a result, a standard clustering method would mainly form groups of pages from one example site instead of identifying topics that span multiple sites. We solve this problem by focusing on similarities between pages from different sites. The quality of a clustering is determined by the quality of the individual clusters. We define the quality of a page cluster as the average similarity of the pages in the cluster to all other pages in the cluster *from other web sites*.

We use five different measures to determine the similarity between two web pages in a cluster:

- The **text similarity** is expressed as the cosine similarity between the terms on the pages (Salton and McGill, 1983). This measure enables SiteGuide to identify parts of the texts that pages have in common and ignore site-specific parts. Stop word removal, stemming and $tf \cdot idf$ weighting are applied to increase accuracy.
- **Anchor text similarity** is defined as the cosine similarity between the anchor texts of the links that point to the pages.
- **Page title similarity** is the inverse of the Levenshtein distance (Levenshtein, 1966) between the pages' titles.
- **URL similarity** is the inverse of the Levenshtein distance between the parts of the pages' URLs after the last slash ('/')
- The **link structure similarity** of a cluster is the proportion of the incoming and outgoing links of the pages that are mapped correctly. Two links in different link structures are mapped correctly onto each other if both their source pages and their destination pages are in the same cluster.

The quality of a page cluster C is a combination of the five similarity measures:

$$quality(C) = \sum_{sim_i \in Sims} (w_i \cdot sim_i(C)) - \alpha \cdot S_C$$

Here $Sims$ are the five similarity measures, which are weighted with weighting parameters w_i. S_C is the number of example sites that have pages in cluster C. The term $-\alpha \cdot S_C$ subtracts a fixed amount (α) for each of the S_C sites in the cluster. Through the parameter α, the size of the clusters is automatically geared to the number of sites that address the same topic, so that we do not need to specify the number of clusters beforehand. Hollink *et al.* (2008) describes how good values for the parameters w_i and α can be determined.

We use a form of hill climbing search the space of all possible clusterings heuristically for a clustering with a high quality score. For this, we start with an initial clustering and modify the clustering at each iteration. The modification resulting in the highest overall page clustering similarity is kept as the new clustering. This is done until no significant improvements can be made. For a more detailed description, we direct the reader to Hollink *et al.* (2009).

2.2 From Clustering to Model

The next step is to transform the clustering into a model consisting of descriptions of topics that occur on most of the example sites. Each cluster becomes a topic in the model. Topics are characterized by five key phrases, the *characterizing features*. Theseare extracted from the contents of the pages in the cluster, the titles of these pages, anchor texts of links pointing to the pages and terms from the page URLs. SiteGuide lists all terms from the contents of the pages and all URLs, titles and anchor texts. For each type of term we designed a measure that indicates how descriptive the term or phrase is for the topic and the most descriptive terms are used as the characterizing features. In addition, the most central page in the cluster (the page with the highest text similarity to the other pages in the cluster) becomes an example page for that topic.

To show how topics are positioned/embedded in a site SiteGuide uses *structural features* of topics: the average number of papers in a site on the topic, the average number of ingoing and outgoing links of a page that belongs to the topic and the number of links between all pages of the topic and pages of other topics (e.g., pages on topic A frequently link to pages on topic B). To find the structural features of the topics, SiteGuide analyzes the pages and links of the corresponding page clusters. It determines for each site over how many pages the information on a topic is spread and counts the number of incoming and outgoing links. Furthermore, it counts how often the topic links to each other topic.

2.3 Presenting the output to the user

The SiteGuide user can select to either have the model presented to him in a textual or graphical manner. In the first case, the user is presented with a list of all topics. If the user selects one of the generated topics, SiteGuide then displays the characterizing and structural features in natural language statements.

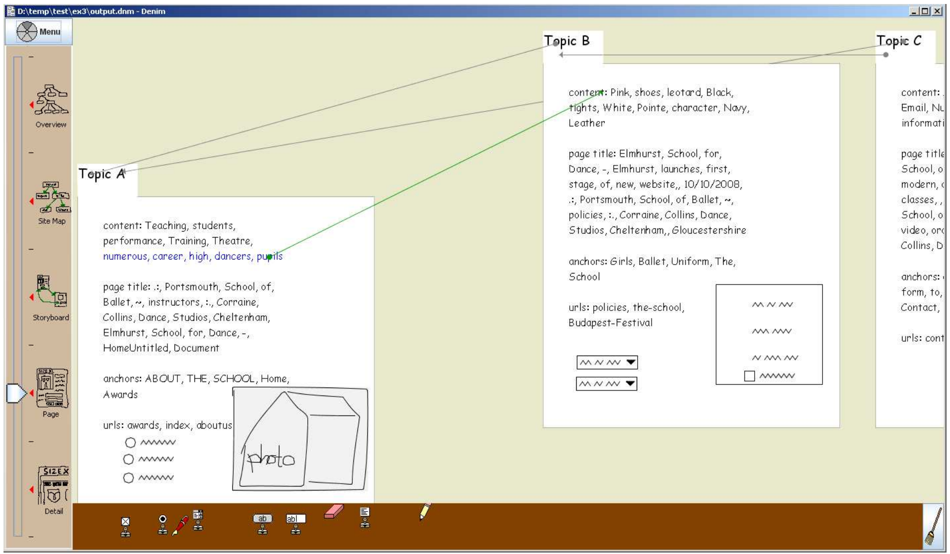
If the graphical representation is selected, the resulting model is presented as a graph which is generated using the Graphviz graph visualization package[3]. In this graph the topics are presented as boxes, listing both the characterizing and structural features. The topics are connected by arrows that represent the frequently occurring links between these topics. SiteGuide presents the graph in a result window. An example of such a graph can be seen in the screenshot in Figure 1. The experiments described in Section 3 evaluate both output types.

2.4 Exporting the output to the DENIM tool

To allow the user to easily use the generated model as an initial interaction design within a web design process, SiteGuide also offers the option to export the model to the DENIM tool (Lin *et al.*, 2000). The DENIM tool is a web design sketching tool that offers the users the option to design at different levels of refinement. The semantical zooming feature allows the user to 'zoom out' and sketch a site map for the web site on the one hand and 'zoom in' on specific web pages to enter content and interaction options on the other hand. Web designs sketch in DENIM can be

[3]http://www.graphviz.org

FIGURE 3: A screenshot of part of a DENIM web site sketch expanded from an imported SiteGuide web site setup for a dance school.



exported to HTML for quick testing. These functions make DENIM a very usable and rapid web design and prototyping tool.

DENIM uses an XML representation to store its sketches. SiteGuide can export its topics and structural features in this XML format and this can be imported in DENIM. DENIM uses an XML representation to store its sketches. The SiteGuide generated web site model can be exported to this format. SiteGuide topics become DENIM pages with the page content consisting of the characterizing features. The links between topics are used to generate site map links in DENIM. The resulting file can be loaded in DENIM where it can be further refined by sketching. Through the DENIM tool, the model can be easily converted to a working HTML web site for testing. Figure 3 shows an example of a SiteGuide model loaded into DENIM. Here a setup for a small dance school is shown, including a home page and a page describing the required clothing. The figure shows that in DENIM links, images and user interaction elements can be added by sketching.

2.5 Critiquing scenario

The SiteGuide model can also be used for a related usage scenario, the *critiquing scenario*. Here the user has already created a first draft version of his new site, which serves as additional input for SiteGuide. For this scenario we developed a variant of the web site comparison method which enables SiteGuide to compare the example site model to the draft. SiteGuide first constructs a web site model based on the example sites and then maps the found topics to the draft site's pages. SiteGuide searches for differences between the draft and the example site model. It determines which topics in the model do not have corresponding pages in the draft and reports that these topics are missing on the new site. Finally, it compares the structural features of the topics in the new site to the common structural features in the example site model and reports the differences.

3 Evaluation

To evaluate whether SiteGuide provides useful assistance to users who are building a web site, we set up an evaluation study that answers two questions: 1) Do the discovered clusters and topics represent the subjects that are really addressed at the example sites and are the textual topic descriptions understandable for humans? and 2) Do people actually build better web sites when using the SiteGuide tool? To answer the first question we compared the model created by SiteGuide to manually created gold standard model. For the second question, we evaluated the web designs of test subjects who used SiteGuide with designs of subjects who had not used SiteGuide.

3.1 Evaluation of the Model

In a previous evaluation study (Hollink *et al.*, 2009), we evaluated SiteGuide's clustering algorithm. For this, we automatically compared SiteGuide's page clustering to this gold standard clustering. Results showed that SiteGuide is indeed able to find these clusters to some degree and they suggested that the clusters could provide useful recommendations to the user. In the same study, we also evaluated SiteGuide in the critiquing scenario and concluded that SiteGuide is able to detect a substantial amount of the differences between a draft site and example sites.

In this paper, we test whether the generated model topics correspond to a gold standard. For this, we used web sites from three domains: windsurf clubs, primary schools and small hotels. For each domain 5 sites were selected as example sites. Table 1 shows the main properties of these three domains.

From these example sites we manually created gold standards. The gold standard models consisted of a clustering of the example site. For each cluster with pages from three or more different web sites (frequent topics) we manually constructed textual descriptions. The textual descriptions were 1 or 2 sentences in length and contained around 20 words. For example, in the school domain one topic was described as '*These pages contain a list of staff members of the schools. The lists consist of the names and roles of the staff members.*'. Table 1 shows the number of pages in the domains and the amount of frequent topics found.

To validate the objectivity of the gold standards, for one domain (hotels) we asked another person to create a second gold standard. We compared the topics in the two gold standards and found that 82% of the topics were found in both gold standards. From this we concluded that the gold standards were quite objective and are an adequate means to evaluate the output of SiteGuide.

For each of these domains, SiteGuide generated a web site model based on the example sites. Since we only wanted to evaluate how well the topics could be interpreted by a user, we did not output the structural features. We restricted

TABLE 1: Properties of the evaluation domains and the gold standards (g.s.): the total, minimum and maximum number of pages in the example sites and the number of topics that were found in at least 50% of the sites (frequent topics).

domain	total pages	min-max pages	frequent topics in g.s.
hotel	59	9-16	7
surfing	120	8-61	12
school	154	20-37	17

the output for a topic to 10 content keywords and to 3 phrases for page titles, URLs and anchor texts. We also displayed for each topic a link to the example page. SiteGuide's generated web site models contained 34 topics across the three domains. We asked 5 evaluators to interpret these 34 topics and to write a short description of what they thought each topic was about. We required the descriptions to be of the same length as the gold standard descriptions (10-30 words). None of the evaluators were domain experts or expert web site builders.

It took the evaluators on average one minute to describe a topic, including typing the description. By comparison, finding the topics in the example sites by hand (for the creation of the gold standard) took about 15-30 minutes per topic.

An expert coder determined whether the interpretations of the evaluators described the same topics as the gold standard descriptions. Since both the evaluators' topic descriptions and the gold standard topic descriptions were natural language sentences, it was often difficult to determine whether two descriptions described the exact same topic. We therefore had the coder classify each description in one of three classes: a description could either have a partial match or an exact match with one of the gold standard topics or have no matching gold standard topic at all. An exact match means that the description describes the exact same topic as the gold standard topic. A partial match occurs when a description describes for instance a broader or narrower topic. To determine precision the coder matched all topic descriptions of the evaluators to the gold standard descriptions. To determine recall the coder matched all gold standard descriptions to the evaluators' descriptions in the same manner. To determine the objectivity of the coding task, we had a second expert coder perform the same task. The two coders agreed on 69% of the matches, considering the possible variety in topic descriptions, we consider this an acceptable level of inter-coder agreement.

Averaged over the domains and evaluators, 94% of all evaluators' topic descriptions were matched to one of the gold standard topics when both partial and exact matches were considered. In other words, 94% of the topics that were found by SiteGuide corresponded (at least partially) with a topic from the gold standard and were also interpreted as such. When only exact matches were counted this figure was still 73%. This indicates that for most topics SiteGuide is capable of generating an understandable description and that the found topics are the important topics that should be found according to the gold standard.

We calculate precision and recall as the relaxed precision and recall (Ehrig and Euzenat, 2005, see):

TABLE 2: The results of the manual evaluation for the three domains expressed in relaxed precision, recall and f-measure.

domain	precision	recall	f-measure
hotel	0.90	0.54	0.68
school	0.78	0.54	0.64
surf	0.81	0.63	0.71

$$\text{precision} = \frac{|T_{exact}| + 0.5 \cdot |T_{part}|}{|T_{SiteGuide}|}$$

$$\text{recall} = \frac{|T_{exact}| + 0.5 \cdot |T_{part}|}{|T_{g.s.}|}$$

where T_{exact} and T_{part} are the exact and partially matching topics and $T_{SiteGuide}$ and $T_{g.s.}$ are the topics generated by SiteGuide and the topics in the gold standard respectively. We also report the f-measure, based on relaxed precision and recall.

In Table 2 we display the relaxed precision, recall and f-measure values for the three domains. The average precision over all domains is 0.83, showing that most of the topics that SiteGuide found can indeed be interpreted and that these interpretations correspond to topics of our gold standard. The average recall is 0.57, which means that more than half of the topics from the gold standard were correctly identified and outputted by SiteGuide. Both precision and recall do not vary much across domains, indicating that SiteGuide is capable of identifying and displaying topics in a wide range of domains.

3.2 Evaluation of web site design support

The evaluation experiment described in the previous sections shows that the identified topics are of a certain quality. In this section we show that the use of SiteGuide's output actually improves the early phases of the web design process.

For this experiment, we asked 12 participants to make a web site setup for three different web sites: a primary school, a small hotel and a scientific conference. We randomly divided the participants into a test group who used SiteGuide and a control group who did not use the tool. The order in which the different domains were presented were varied in such a way that for both the test group and the control group, all six possible orders occurred once. The participants were all computer science Ph.D. students with some familiarity with web sites in general. None of them had any significant web design experience. The goal of this experiment is to show that SiteGuide supports the early web site design process in terms of speed, ease and resulting quality of the design. For each of the three web sites, the participants were asked to sketch a web site on a piece of paper with pages being represented by boxes and links represented by arrows. Pages must contain a title and a short description of its contents. Additional comments could also be added to the sketch.

Participants in both groups were presented with URLs of 5 example sites but they were instructed that consulting these sites was optional and that other web sites could also be visited for inspiration. The participants in the test group also received a SiteGuide graphical model output, constructed from the 5 example sites. They were instructed that this could also be used for inspiration, but that it was not compulsory. For the creation of a single setup sketch, the participants had 10 minutes. We measured the perceived performance by using a user questionnaire. For a more objective performance measure, a web design expert analyzed result sketches.

3.2.1 Perceived Performance

After completing a web site design, the participants filled out an evaluation form that was used to measure the perceived performance. The participants were asked to give a score on a five point scale for the difficulty of the creation task, the quality of the resulting setup and the completeness of the setup and the usefulness of the provided information. The participants were also asked whether or not they felt that they had enough time for sketching the web site setup.

The test group was asked to what degree they felt their design was based on the SiteGuide setup. They were also asked whether the SiteGuide setup was helpful and easily understandable and to what degree SiteGuide's output made it easier and less time-consuming to construct the sketch. Finally they were asked whether the SiteGuide setup helped to create a better, more complete setup. Both the test and the control groups were asked these same questions for the example sites as well. All these questions were answered using a five-point scale. Additional comments about the task, the example sites and the SiteGuide output were also collected.

We performed a multivariate ANOVA analysis measuring the influence of group, web site type and the order on perceived performance. One result that was significant was the effect of the task type (the type of target web site) on the perceived ease the setup construction ($F = 6.53, p = 0.005$): The average score for both the hotel site and the conference site was 2.33 (between *easy* and *neutral*), whereas the construction of the school site was perceived to be much harder with an average of 3.5 (between *neutral* and *difficult*). The task type was also a significant factor for the scores on the perceived completeness of the setup ($F = 5.084, p = 0.014$) and the satisfaction with the end result ($F = 4.91, p = 0.0163$).

The usefulness of the provided information was rated higher by the control group than by the test group with an average score of 4.17 and 3.17 respectively ($F = 4.49, p = 0.0445$). From control and test group comments, we gathered that both groups perceived the example web sites to be easily understandable, while the SiteGuide output was perceived to be relatively difficult to understand in some cases. The model evaluation indicated that users are able to understand SiteGuide's topics. Combining the two findings, we conclude that although it takes more effort, the topics are well understandable in the end.

For the perceived time shortage, we did not find any significant effects. On average, the test group participants perceived the creation of all websites to be slightly easier although this is also not a significant ($F = 2.13, p = 0.157$) difference.

The score for the perceived use of the example sites in the control group was 3.33. For the test group this value is 2.78. This indicates that the test group perceives that their final sketch is to a lesser degree based on the example sites compared to the control group. Moreover, the test group perceived that their final sketch was just as heavily based on the SiteGuide setup (average score 2.72). This indicates that the test group participants perceive to have used the SiteGuide setup to some extent.

A number of the additional comments provided by the test group participants state that the SiteGuide example setup was used to check the completeness of sketched setups created by that group. This is an interesting usage scenario that corresponds somewhat to the critiquing scenario as described in Section 2.5.

3.2.2 Expert Evaluation

The results from the user questionnaire analysis tell us how users perceived working with SiteGuide, but do not give information about the quality of the resulting web designs. To measure the objective quality of the web site setup sketches, we used an expert evaluation of the primary school results. We chose the school domain since this was perceived to be the hardest task.

A double-blind evaluation was performed by a web design professional. He was provided with the 12 sketches of the primary school web sites made by the participants in both groups. The expert was asked to first make an ordering of the 12 sketches based on their general quality. The expert then assigned a rating to each sketch for five quality criteria:

- **completeness**: whether the web site setup contains all information that is needed.
- **relevancy**: to what amount the information in the web site set up is relevant and useful
- **detailedness**: whether the web site setup is specified in appropriate detail
- **content structure**: whether the information in the setup is appropriately structured over the pages
- **link structure**: whether the links are appropriate

For each of these criteria, the expert was asked to assign an evaluation using a five-point scale (*very bad, bad, fair, good* or *very good* corresponding to an evaluation score of 1-5). For each setup, the expert also provided a short description explaining the ranking and scores.

Viewed over all 12 setups, the expert assigned mostly average scores. None of the criteria for none of the setups was evaluated as being *very bad* while only 4 *very good* evaluations were provided. The average evaluation scores show a strong correlation with the assigned ranking (a Pearson correlation coefficient value of -0.89). This is an indication that the chosen evaluation criteria matches the expert's notion of general web site quality. The average evaluation score, averaged over all 12 setups was 3.23, slightly above *fair*, with a standard deviation of 0.60.

TABLE 3: Average rank and evaluation score for the test and control group. Significance is tested with an unpaired one-tailed Mann-Whitney U test for the ranks and an unpaired one-tailed T-test for the 5 criterion scores and their average.

	Without SiteGuide (control group)	With SiteGuide (test group)	Difference	Significance
Rank	8.33	4.67	-3.67	$p = 0.047$
Completeness	3.00	3.33	0.33	$p = 0.282$
Relevancy	3.00	3.17	0.17	$p = 0.367$
Detailedness	2.67	3.33	0.67	$p = 0.064$
Content structure	3.00	4.17	1.17	$p = 0.006$
Link structure	2.83	3.83	1.00	$p = 0.022$
Average score	2.90	3.57	0.67	$p = 0.026$

Table 3 shows the average rank and evaluation scores for both the test group and the control group. It also shows the average over all scores. The average rank of the setups created by the test group is significantly higher than that of the control group according to the Mann-Whitney U-test. From this we can conclude that the setups made by participants presented with SiteGuide are considered to be better than those made without SiteGuide.

For all five criterion scores, the test group receives a higher average score. For a significance level of $\alpha = 0.05$, this difference is significant for content structure and link structure and the average over the 5 criterion scores. It is marginally significant for detailedness. These results show that for this domain, users design web sites that have a more appropriate level of detail and are that better structured both in their page content and the links.

To analyze the objective acceptance of the SiteGuide setup for the school task, we compared the topics in the presented SiteGuide setup to the resulting sketches from the test group. For each page in a participant's sketch, we checked if a corresponding topic occurred in the SiteGuide setup. In the same manner as described in Section 3.1, we assigned a score of 1, 0.5 and 0 to full, partial and no matches to this 'gold standard' respectively. The acceptance score for a participant is the average score for all that participant's sketched pages, corresponding to the relaxed precision of Section 3.1. This average acceptance score for the test group is significantly higher than that of the control group (respectively 0.55 and 0.13, significant at $p = 0.047$). This indicates that, as was also perceived by the participants, the SiteGuide information is actually used in the final setup.

4 Related work

Existing tools for assisting web site development help users with the technical construction of a site. Tools such as Dreamweaver[1] or Frontpage[2] allow users to create web sites without typing HTML. Other tools evaluate the design and layout of a site on usability and accessibility, checking for example for dead links and buttons and missing captions of figures (see for an overview Web Accessibility

Initiative 2008; Brajnik 2004). However, none of these tools help users to choose appropriate contents or structures for their sites.

Our approach is in spirit related to the idea of ontologies. The goal of an ontology is to capture the conceptual content of a domain. It consists of structured, formalized information on the domain. The web site models presented in this paper can be viewed as informal ontologies as they comprise structures of topics that occur in sites from some domain. However, our models are not constructed by human experts, but automatically extracted from example sites.

Another related set of tools are tools that improve link structures of web sites, such as PageGather (Perkowitz and Etzioni, 2000) and the menu optimization system described in Hollink *et al.* (2007). These tools do not provide support on the contents of a site. Moreover, they need usage data, which means that they can only give advice about sites that have been online for some time.

The algorithm that underlies the SiteGuide system is related to methods for high-dimensional clustering, which are, for instance, used for document clustering. However, there are several important differences. The task of SiteGuide is to find topics in a set of web sites instead of an unstructured set of documents. It is more important to find topics that appear in many sites than to group pages within sites. This is reflected in the similarity measure. Another difference is that in SiteGuide the relations (links) between pages are taken into account. The extent to which relations between pages within one site match the relations in other sites contributes to the similarity between pages. Also, the final model of the sites includes relations between topics.

5 Conclusions

The SiteGuide system provides assistance to web designers who want to build a web site but do not know exactly which content must be included in the site. It automatically compares a number of example web sites and constructs a model that describes the features that the sites have in common. SiteGuide can be used as a standalone tool to generate an information architecture that can be used in the early stages of web desing. Alternatively, this set up can be imported into the DENIM web design tool. In addition, SiteGuide can show differences between example sites and a first version of a new site in a critiquing scenario.

SiteGuide was applied to example web sites from multiple domains. In these experiments, SiteGuide proved able to find many topics that the sites had in common. Moreover, the topics were presented in such a way that humans could understand what the topics were about, although the results from the questionnaire indicated that this required some extra effort.

A user experiment showed that SiteGuide helps users to create better web site setups. The participants incorporated more than half of the SiteGuide topics suggested by SiteGuide into their design sketch. Although the task was not perceived to be easier by the participants, the resulting web site setups were evaluated to be of a higher quality by a web design expert.

Comments by the participants suggest that the critiquing scenario is also an effective way to use SiteGuide. Support for this scenario will be implemented in the next version of the tool and a user evaluation will be conducted to test its performance on this task. Further user evaluation studies can also determine the value of SiteGuide in providing input for the DENIM web design tool and the quality of the final product: the web sites themselves.

We also aim to extend SiteGuide with a number of new features. A web search component will be implemented that uses the metrics developed within the SiteGuide project to retrieve pages similar to those in the input pages from the web. This could be used to find example sites automatically. Semantics will be added to the similarity measure in the form of WordNet relations. Finally, SiteGuide could output additional features such as style features (e.g., colors and use of images) or the amount of tables, lists, forms, etc. More research is needed to determine how these features can be compared automatically.

References

G. Brajnik (2004), Using automatic tools in accessibility and usability assurance processes, in *Proceedings of the Eighth ERCIM Workshop on User Interfaces for All, Vienna, Austria*, pp. 219–234.

M. Ehrig and J. Euzenat (2005), Relaxed precision and recall for ontology matching, in *Proceedings of the K-Cap 2005 Workshop on Integrating Ontologies, Banff, Canada*, pp. 25–32.

K. Falkovych and F. Nack (2006), Context aware guidance for multimedia authoring: Harmonizing domain and discourse knowledge, *Multimedia Systems*, 11(3):226–235.

V. Hollink, V. De Boer, and M. Van Someren (2009), SiteGuide: An example-based approach to web site development assitance, in *Proc. of the Fifth International Conference on Web Information Systems and Technologies (to appear), Lisboa, Portugal*.

V. Hollink, M. Van Someren, and V. De Boer (2008), Capturing the needs of amateur web designers by means of examples, in *Proceedings of the 16th Workshop on Adaptivity and User Modeling in Interactive Systems, Würzburg, Germany*, pp. 26–31.

V. Hollink, M. Van Someren, and B. Wielinga (2007), Navigation behavior models for link structure optimization, *User Modeling and User-Adapted Interaction*, 17(4):339–377.

V. I. Levenshtein (1966), Binary codes capable of correcting deletions, insertions and reversals, *Soviet Physics Doklady*, 10(8):707–710.

J. Lin, M. W. Newman, J. I. Hong, and J. A. Landay (2000), DENIM: finding a tighter fit between tools and practice for Web site design, in *CHI '00: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 510–517, ACM, New York, NY, USA, ISBN 1-58113-216-6, doi:http://doi.acm.org/10.1145/332040.332486.

M. W. Newman and J. A. Landay (2000), Sitemaps, storyboards, and specifications: A sketch of web site design practice, in *Proceedings of the Third Conference on Designing Interactive Systems, New York, NY*, pp. 263–274.

M. Perkowitz and O. Etzioni (2000), Towards adaptive web sites: Conceptual framework and case study, *Artificial Intelligence*, 118(1-2):245–275.

G. Salton and M. J. McGill (1983), *Introduction to Modern Information Retrieval*, McGraw-Hill, New York, NY.

Web Accessibility Initiative (2008), Web accessibility evaluation tools: Overview, *http://www.w3.org/WAI/ER/tools/*, last accessed: July 4, 2008.