A NEW DATA-DRIVEN APPROACH FOR MULTIMEDIA PRESENTATION PLANNING

Amanda Stent
Department of Computer Science
State University of New York at Stony Brook
email: stent@cs.sunysb.edu

Hui Guo
Department of Computer Science
State University of New York at Stony Brook
email: huguo@cs.sunysb.edu

ABSTRACT

A number of Multimedia Presentation Systems have been built for technical documentation, traffic management systems, educational software and other applications. However, these systems use handwritten presentation planning operators, which limits their flexibility. In this paper we describe a new data-driven approach for multimedia presentation planning. We present algorithms for acquiring presentation plan operators from Web pages, and for using these operators to generate multimedia presentations. We also show how to use machine learning to adapt multimedia presentation planning to human presentation preferences.

KEY WORDS

Human Computer Interfaces, Multimedia Communication Systems, Collaborative Systems and Applications

1 Introduction

Multimedia has proven to be an efficient and effective form of presenting information from complex data sources. In our information-rich world, multimedia presentations are all around us: on our computers, PDAs and cellphones, in our books, magazines and newspapers, and in our workplaces, schools and homes. As a result, the need for Multimedia Presentation Systems (MMPSs), which generate multimedia presentations automatically, is growing rapidly.

Multimedia presentation planning is an inherently complex process. This complexity has two basic causes: the *size of the choice space* for multimedia presentations, and the *need for good heuristics* for pruning the choice space and finding optimal paths through it [3].

Consider first the size of the choice space for multimedia presentations. The basic elements are pieces of content and presentation elements (e.g. tables, pictures, charts, text, speech). The planning process itself involves four tasks. Relevant content must be selected and organized in accordance with some communicative goal (*content selection*). These facts must be mapped to the presentation elements that will be used to present them (*presentation element selection*). These presentation elements must be arranged temporally and spatially (*presentation layout*), and global and local presentation features (e.g. color, size, grouping) must be set (*presentation style*). To get an idea of the complexity of these tasks, consider only the task of assigning content to presentation elements. If each piece of content must be assigned to exactly one presentation element, and there are c pieces of content and p presentation elements, then the number of assignments of content to presentation elements is p^c. If we permit redundancy (i.e. each piece of content can be assigned to one or more presentation elements) then the number of assignments becomes $(2^p - 1) * c$. Many of these assignments, though theoretically possible, are infeasible or have poor presentation style; however, it is difficult to automatically define a subset of feasible assignments for a particular domain and goal that is large enough to permit interesting variation.

Now consider the problem of searching through this choice space for 'good' presentations. Most people do not focus on the presentation aspects of a good presentation, because it permits them to focus on the content and on their current task. However, they can identify faults of a bad presentation; they may say that it doesn't contain the information they need or contains too much information, is unclear, is badly organized, or is unattractive. These terms are helpful for human presentation designers, but are too vague for MMPSs, and very few people (usually graphic designers or statisticians) can explain their intuitions about presentation design clearly (c.f. [8, 9]). A presentation planning system should therefore contain methods for transforming users' presentation preferences into quantitative metrics that eliminate poor presentation choices and/or select good ones.

This paper describes a new data-driven approach to multimedia presentation planning. Section 3 describes our algorithm for acquiring presentation operators automatically. Section 4 describes how we generate multimedia presentations. In Section5, we show how we can use supervised learning to adapt multimedia presentation planning for human preferences. We sum up in Section 6.

2 Related Work

Multimedia generation is usually performed in a pipeline process: content selection, presentation planning, tactical or media-specific realization, and production [1]. Existing multimedia generation systems usually assign content to media during presentation planning.

There are two ways of representing presentation patterns for multimedia presentation planners. Schemas are templates for an entire presentation [14], while presentation operators are rules that specify how to break down a presentation goal into subgoals and eventually into parts of a presentation [18]. Multimedia presentation planners typically use an AI-style planner (e.g. [6]). The input is a communicative goal and a set of facts organized using a model of discourse structure (e.g. RST [17]). Operators are selected from the list of available operators to decompose goals into subgoals and actions; the actions may be layout actions or calls to media-specific generators. Planning continues until there are no more goals (success condition), or no more operators (failure condition).

A number of MMPSs have been built for technical documentation, traffic management systems, educational software and other applications (e.g. WIP [7], SAGE [16], AutoBrief [15]). However, these systems use handwritten rules, which lead to a lack of adaptability in presentation planning and an inability to model human presentation preferences other than those of the system designer.

Research on adaptation in multimedia information presentation has typically been done in the context of automatic Web page generation. A good summary of work on adaptive hypermedia is [2]. Researchers have studied adaptation of Web page text, layout and presentation style. For example, graphical elements of a presentation/interface can be adapted to the user's perceived needs, interests and cognitive load [10, 11]. Web page layout can be adapted to user expertise [4]. Finally, presentation style can be adapted to user age [5] and to specific disabilities [12].

Some recent work has been done on mining of Web pages for semantic structure. [19] presents an algorithm for analyzing HTML pages based on visual clues to extract semantic information. We use similar techniques to extract presentation operators.

3 A Data-Driven Approach for Acquiring Presentation Planning Operators

As the need for automatic multimedia presentation planning grows, designing schemas or operators by hand becomes less feasible. Here we present a data-driven approach for multimedia presentation planning, in which we acquire presentation planning operators from human-created presentations in structured formats, particularly HTML (but also SMIL, XML). The presentation planning operators acquired in this fashion are domain-dependent, but a different set can be easily acquired for each new domain given a fairly small amount of training data in the form of human-created presentations (many of which can be found on the World Wide Web). We have applied this approach to learn presentation planning operators for recommendations of single computers and houses.

Our system architecture is shown in Figure 1. Given a communication goal, example presentations represented in a markup language, a database of content and an ontology (if necessary), the system can learn presentation operators automatically.

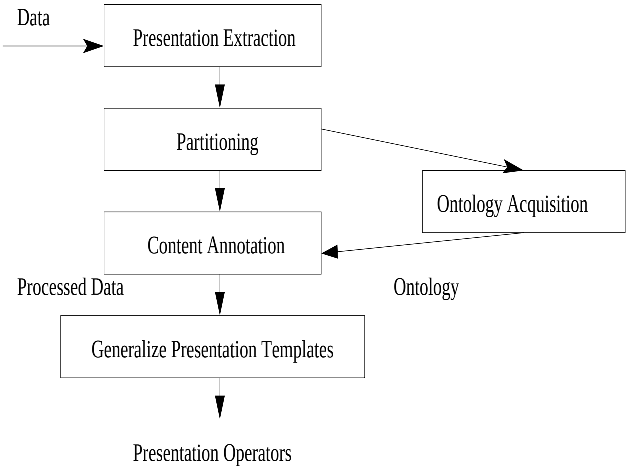


Figure 1. MMPS architecture

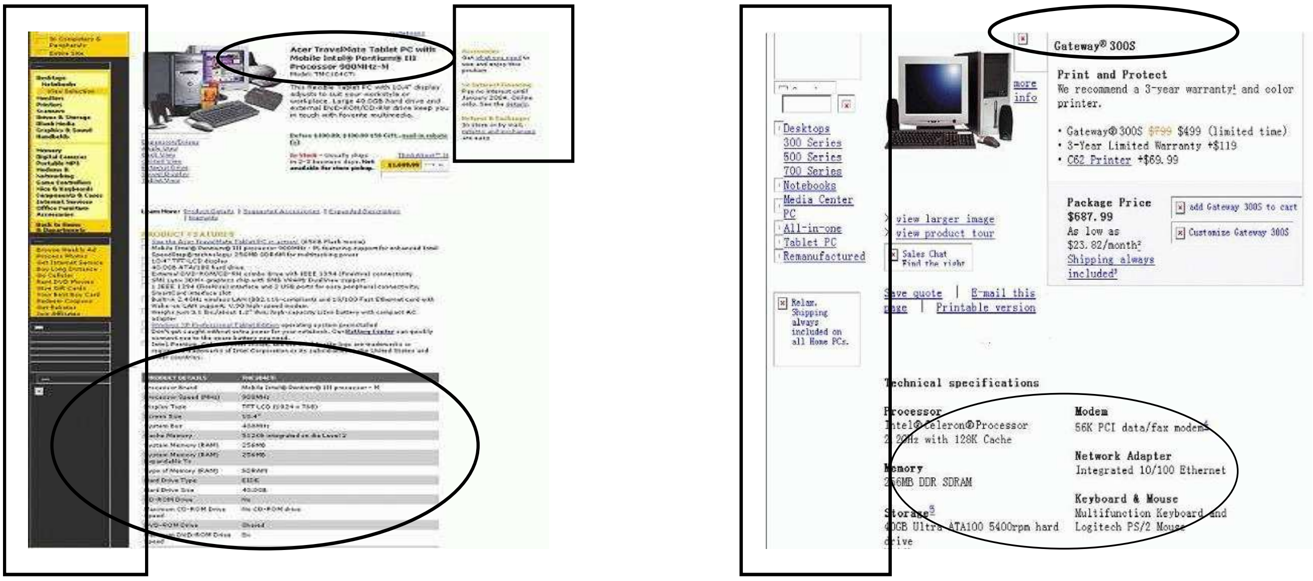
3.1 The Operator Acquisition Process

Our automatic operator acquisition process has four stages: *presentation extraction, presentation partitioning, content annotation* and *operator construction and generalization*. In this section, we describe the stages in detail.

Presentation Extraction Presentations contained in Web pages may be surrounded by extraneous information, e.g. navigation links and advertisements. To extract the presentation itself, we first group pages from the same website together. By locating similar subtrees in the DOM trees for pages from the same site, we can identify menus, navigation elements, advertisements and other extraneous information. This information is removed. Figure 2 shows two presentations in our training data; the extraneous information is marked with squares.

Presentation Partitioning To identify the layout of a presentation we must partition it into its presentation elements. Each element corresponds to a visually distinguishable part of the presentation (e.g. a table, a list, a picture, text, a title). For example, a presentation of a computer may comprise an element describing the computer and another element containing a table of features of the computer. We partition each presentation automatically by labeling subtrees rooted at presentation element tags (e.g. <TABLE>, <UL>). Some of the presentation elements in the two presentations in Figure 2 are circled. We use heuristics to label presentation elements that are not clearly marked using the markup language (such as the table in the second presentation in Figure 2).

Content Annotation After partitioning, we automatically annotate each presentation element for the content type(s) it contains. In order to do this, we need a taxonomy or lexicon for the domain. If we do not have an



Acer TravelMate Tablet PC with Mobile Intel® Pentium® III Processor 900MHz-M

Gateway® 300S

Print and Protect
We recommend a 3-year warranty¹ and color printer.

- Gateway® 300S ~~$799~~ $499 (limited time)
- 3-Year Limited Warranty +$119
- C62 Printer +$69.99

Package Price
$687.99
As low as
$23.82/month²
Shipping always included³

Technical specifications

Processor
Intel® Celeron® Processor 2.2GHz with 128K Cache

Memory
256MB DDR SDRAM

Storage⁵
40GB Ultra ATA100 5400rpm hard drive

Modem
56K PCI data/fax modem⁴

Network Adapter
Integrated 10/100 Ethernet

Keyboard & Mouse
Multifunction Keyboard and Logitech PS/2 Mouse

Figure 2. Pages from Bestbuy.com and Gateway.com

existing taxonomy or lexicon for the current domain, we can sometimes acquire one automatically from our data. Many presentations contain an attribute-value table (or tables). We can compare the tables in different presentations to find the frequencies of occurrence of the values for each attribute. These values can be used to train a probabilistic lexical database. Table 1 shows some stem entries in our automatically-acquired lexical database for computer recommendations.

Stem Entries	Concepts	Counts
GB	Memory—Size Hard Drive—Size	193 822
DDR	Memory—Type	819
…	…	…

Table 1. Stem entries in the lexical database

Given a taxonomy or lexicon, we can label textual elements of a presentation (e.g. table rows, text, titles, list items). We compute the probability of each text being in each category in the taxonomy, and assign the text the category label with the highest probability. For example, for the text ‘1GB DDR’, we remove the number and split it into two words: ‘GB’ and ‘DDR’. From the lexical database, we know that the probability of ‘GB’ being in ‘Hard Drive — Size’ is 822/(822+193) = 81%, while the probability of it being in ‘Memory — Size’ is 193/(822+193) = 19%. For the word ‘DDR’, the probability of being in ‘Memory — Type’ is 819/819 = 100%. However, ‘DDR’ has never appeared in ‘Hard Drive’. So we assign a small probability, e.g. 1%, to it. Now we compute the probabilities of ‘1GB DDR’ being in these two categories:

```
Hard Drive: 81%*1% = 0.0081
```

```
Memory: 19%*99% = 0.1881
```

Since the category ‘Memory’ has higher probability, we label the text with ‘Memory’. Non-text elements (e.g. images) are labeled with the label of the textual element(s) surrounding them (e.g. the image caption). At the end of this process, any presentation element with no content label is discarded.

Generalizing presentation operators After partitioning and content annotation, we have a marked-up, segmented version of each presentation called the *presentation template* for that presentation. Figure 3 is an example of a presentation template.

We automatically split the presentation template into presentation operators by splitting at subtrees in the DOM tree that correspond to presentation elements or presentation element groupings. With each presentation operator, we store its content types (assigned during content annotation). We also store presentation style information from the training data (e.g. the size of an image, the font style of a piece of text, the background color of a table row).

Finally, we store presentation layout information with each presentation operator. For example, images are to the left of text descriptions in some presentations, and on the right in others. On the other hand, text descriptions are above feature tables in most presentations. To learn presentation layout information, we traverse the DOM tree of each presentation template and record the relative positions of presentation elements. Then we compute the probability of specific presentation elements given context. For example, we may get the following result:

$$P(FeatureTable|Image\&\&Description) = 6.176e-4$$

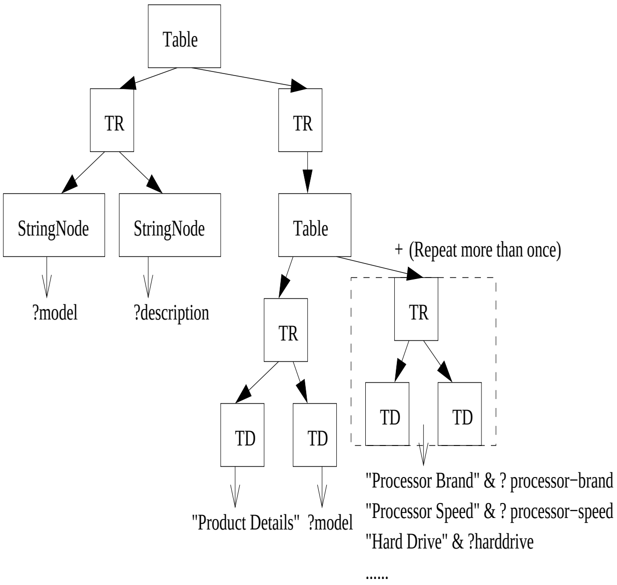


Table
TR
TR
StringNode
StringNode
Table
+ (Repeat more than once)
?model
?description
TR
TR
TD
TD
TD
TD
"Processor Brand" & ? processor−brand
"Processor Speed" & ? processor−speed
"Hard Drive" & ?harddrive
......
"Product Details"
?model

Figure 3. A presentation template

This means the probability of a feature table occurring between an image and a text description is $6.176e - 4$.

By storing presentation operators rather than presentation templates, we can generate previously unseen presentations. However, we can also use the presentation layout and content mapping information to select combinations of presentation operators that have high probability given the training data, and therefore probably have good presentation style.

4 Multimedia Presentation Planning

The multimedia presentation planning process starts with a content plan (e.g. to describe a computer, to compare two computers). We use a generate-and-rank approach to multimedia presentation generation. Our presentation planner instantiates all operators matching the communicative goal and content in the content plan. Some operators have presentation subgoals, which are filled in recursively. We combine the instantiated operators in all possible ways, and rank the resulting presentations using the presentation layout probabilities. The top-ranked presentation(s) that cover all or most of the content plan are candidates for output.

For example, Figure 4 shows two presentations generated by our system using training data like that shown in Figure 2. The left-hand one got an average human rating of 4 for both clarity of organization and attractiveness. The right-hand one got an average human rating of 1 for clarity of organization and of 1.7 for attractiveness. The database of presentation operators used in this version of the system was trained on 100 presentations from 4 different websites. We have also trained a presentation planner for house recommendations using a database of 400 presentations from 4 different websites.

5 Using Boosting to Select Presentations

Because we typically have to train on a very small amount of training data, many of our presentations are not ones a human would find attractive or well-organized. To model human presentation preferences, we use the AdaBoost algorithm [13].

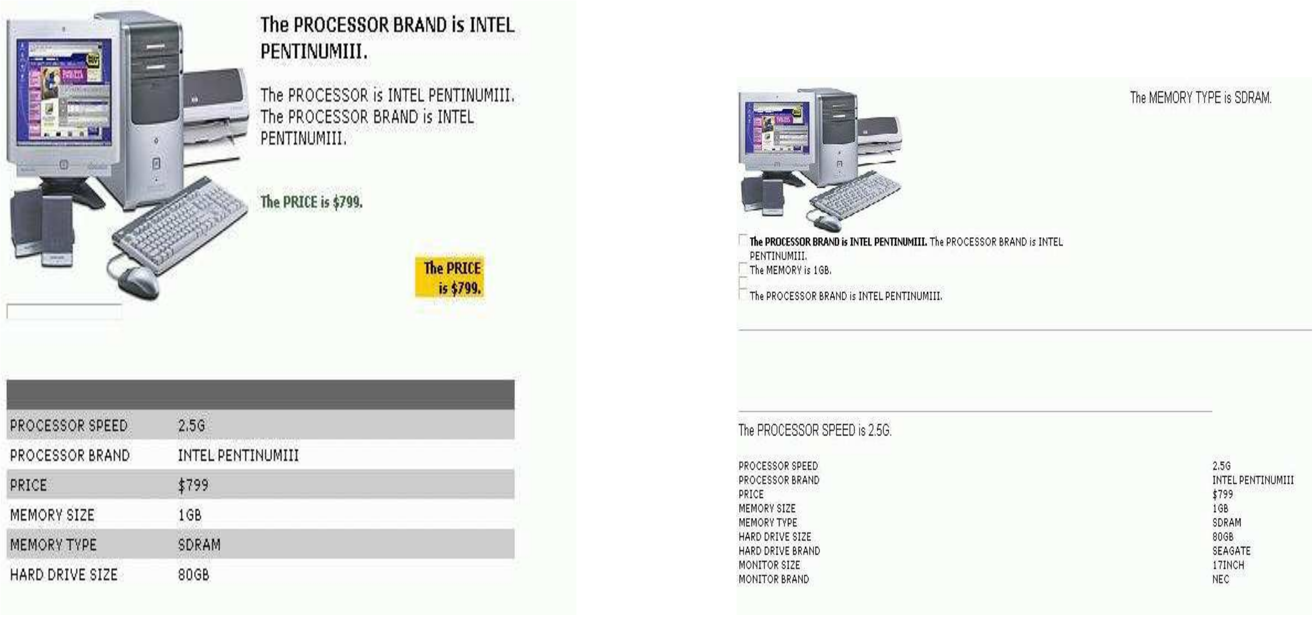
Examples and Feedback We trained our generation system on 100 computer presentations from 4 different websites (a very small amount of training data). We then automatically generated presentations for each of six input content plans. Two of these content plans included three facts about one computer; two included six facts about one computer; and two included all the facts from the database about that computer. We selected 15 presentation candidates for each input: 12 from the most highly ranked and 3 random other candidates. We also selected one presentation from our training data for each input, giving 96 presentations total. We asked three human judges to rate each of these 96 presentations by indicating their degree of agreement on a scale from 0 (“Strongly Disagree”) to 4 (“Strongly Agree”) with the following two statements:

Statement 1. This presentation is clear and well organized.
Statement 2. This presentation is attractive.

The result was two scores for each presentation for each judge. For boosting, we used the average of the judges’ ratings for each question for each presentation.

Features We encoded each presentation as a set of features. We used five types of feature for each presentation: absolutePosition, relativePostion, size, style and contains. **AbsolutePosition** features indicate the position of presentation elements (e.g. a table, a line of text) in the presentation. **Size** features indicate the width and length of units. These are integer-valued features. **RelativePosition** features indicate whether an element is above or next to another element. **Style** and **contains** features indicate the appearance and content of units. These are binary-valued features. Example features are shown in Table 2. We checked that given these features about a presentation, a human who has not previously seen the presentation can draw it. In other words, a presentation is completely described using the features.

Results Using 6-fold cross validation, we repeatedly trained our presentation planner using the presentations from five of the six inputs, and tested on the presentations from the sixth. Our evaluation metric is the average of the human judges’ ratings for each question for each presentation. We consider presentations rated 3 or higher by the human judges to be ’good’ presentations; those rated lower than 2 are considered to be ’bad’ presentations. For each input, we compare:



The PROCESSOR BRAND is INTEL PENTINUMIII.

The PROCESSOR is INTEL PENTINUMIII. The PROCESSOR BRAND is INTEL PENTINUMIII.

The PRICE is $799.

The PRICE is $799.

PROCESSOR SPEED	2.5G
PROCESSOR BRAND	INTEL PENTINUMIII
PRICE	$799
MEMORY SIZE	1GB
MEMORY TYPE	SDRAM
HARD DRIVE SIZE	80GB

The MEMORY TYPE is SDRAM.

□ **The PROCESSOR BRAND is INTEL PENTINUMIII.** The PROCESSOR BRAND is INTEL PENTINUMIII.
□ The MEMORY is 1GB.
□
□ The PROCESSOR BRAND is INTEL PENTINUMIII.

The PROCESSOR SPEED is 2.5G.

PROCESSOR SPEED	2.5G
PROCESSOR BRAND	INTEL PENTINUMIII
PRICE	$799
MEMORY SIZE	1GB
MEMORY TYPE	SDRAM
HARD DRIVE SIZE	80GB
HARD DRIVE BRAND	SEAGATE
MONITOR SIZE	17INCH
MONITOR BRAND	NEC

Figure 4. A highly-ranked generated presentation (left) and a low-ranked generated presentation (right)

Type of Feature	Example
AbsolutePosition	leftCorner_image_x : 0
RelativePosition	above_image_featureTable : 1
Size	size_image_width : 3
Style	style_title_bold : 1
Contains	contains_featureTable _entry(processor) : 1

Table 2. Examples of Presentation Features

- HUMAN: the human ratings of those presentations rated 3 or higher by the human judges;
- BOOST: the human ratings of those presentations rated 3 or higher by the presentation planner adapted using boosting; and
- BASE: the human ratings of those presentations that got the highest probability in our baseline presentation planner.

Table 3 summarizes the difference between HUMAN, BOOST and BASE for statement 1, statement 2 and the average score for these two statements. We did t-tests to compare HUMAN to BOOST to BASE. For statement 1, BOOST is significantly better than BASE (df = 29.6, $p < 0.05$) and significantly worse than HUMAN (df = 27.3, $p < 0.01$). For statement 2, BOOST is significantly better than BASE (df = 28, $p < 0.01$) and significantly worse than HUMAN (df = 27, $p < 0.05$). For the average of statements 1 and 2, BOOST is significantly worse than HUMAN (df = 27, $p < 0.001$) but not significantly better than BASE. This shows that with boosting we can model specific human preferences (e.g. about attractiveness, clarity of organization) but that these preferences may conflict with each other when combined.

Statement	Set	Min	Max	Mean
Statement1	HUMAN	3.0	4.0	3.37
	BOOST	1.3	4.0	2.74
	BASE	1.0	4.0	2.32
Statement2	HUMAN	3.0	4.0	3.34
	BOOST	1.0	4.0	2.81
	BASE	1.0	4.0	2.16
Average	HUMAN	3.0	4.0	3.36
	BOOST	1.3	4.0	2.32
	BASE	1.0	4.0	2.23

Table 3. Summary of the human-assigned scores

We also looked at how many of the presentations rated 3 or higher by the trained presentation planner were actually not 'bad' (human rating 2 or lower). For statement 1, the trained presentation planner rated 21 presentations 3 or higher, of which 16 had a human rating of higher than 2. For statement 2, the trained presentation planner rated 20 presentations 3 or higher, of which 16 had a human rating of higher than 2. When the ratings of the presentation planners trained separately on statement 1 and statement 2 were averaged, of 17 resulting presentations that were rated 3 or higher, 13 had a human rating of higher than 2. This shows that although our trained presentation planner can identify 'not bad' presentations, it also spuriously identifies roughly

25% of 'bad' presentations as 'good'. We think that with more training data the number of spurious 'good' presentations would decline.

6 Conclusions and Future Work

In this paper, we have described an algorithm for data-driven multimedia presentation planning. We have shown that it is possible to acquire presentation planning operators from human-created presentations represented using a markup language, and that these operators can be adapted to model specific human preferences about multimedia presentation style. Our results indicate that this line of research shows promise, but considerable work remains to be done. We plan to test our operator acquisition process on more complex presentations, such as comparisons, schedules and animations. We also plan to explore presentation planning as an application for user modeling. For example, we plan to ask more judges to rate presentations, use their ratings to group them into user categories, and then adapt our system for each user category.

References

[1] M. Bordegoni, G. Faconti, S. Feiner, M. Maybury, T Rist, S. Ruggieri, P. Trahanias, and M. Wilson. A standard reference model for intelligent multimedia presentation systems. *Computer Standards and Interfaces*, 18(6, 7):477–496, December 1997.

[2] P. Brusilovsky and M. Maybury. From adaptive hypermedia to the adaptive web. *Communications of the ACM special issue on the Adaptive Web*, 45(5), 2002.

[3] C.Elting, S.Rapp, G.Mohler, and M.Strube. Architecture and implementation of multimodal plug and play. In *Proceedings of ICMI 2003*, 2003.

[4] I. Cruz, K. James, and D. Brown. Integrating layout into multimedia data retrieval. In *Proceedings of the AAAI Fall Symposium on 'Using Layout'*, 1999.

[5] B. DeCarolis and S. Pizzutilo. From discourse plans to user-adapted hypermedia. In *User Modeling: Proceedings of the Sixth International Conference*, 1997.

[6] E.Andre and T.Rist. Coping with temporal constraints in multimedia presentation planning. In *Proceeding of AAAI-96*, volume 1, pages 142–147, 1996.

[7] E.Andre, W.Finkler, W.Graf, and T.Rist. *Intelligent Multimedia Interfaces*, chapter WIP: The Automatic Synthesis of Multimodal Presentations, pages 75–93. AAAI Press, 1993.

[8] E.Tufte. *Evisioning Information*. Graphics Press, 1990.

[9] E.Tufte. *The Visual Display of Quantitative Information*. Graphics Press, 2001.

[10] M. Light and M. Maybury. Personalized multimedia information access: Ask questions, get personalized answers. *Communications of the ACM special issue on the Adaptive Web*, 45(5), 2002.

[11] D. Reitter, E. Panttaja, and F. Cummins. UI on the fly: Generating a multimodal user interface. In *Proceedings of HLT/NAACL 2004*, 2004.

[12] J. Richards and V. Hanson. Web accessibility: A broader view. In *Proceedings of WWW 2004*, 2004.

[13] R.Schapire. A brief introduction to boosting. In *Proceedings of the Sixteenth International Joint Conference on AI*, 1999.

[14] S.Feiner and K.McKeown. *Intelligent Multimedia Interfaces*, chapter Automating the Generation of Coordinated Multimedia Explanations, pages 117–138. AAAI Press, 1993.

[15] S.Kerpedjiev, G.Carenini, S.Roth, and J.Moore. AutoBrief: a multimedia presentation system for assisting data analysis. *Computer Standards & Interfaces*, 18(6-7):583–593, 1997.

[16] S.Kerpedjiev, G.Carenini, S.Roth, and J.Moore. Integrating planning and task-based design for multimedia presentation. In *Proceedings of the 1997 International Conference on Intelligent User Interfaces,*, pages 145–152, 1997.

[17] W.Mann and S.Thompson. *The Structure of Discourse*, chapter Rhetorical Structure Theory: A Theory of Text Organization. Ablex Publishing Corporation, 1987.

[18] W.Wahlster, E.Andre, W.Finkler, H.Profitlich, and T.Rist. Plan-based integration of natural language and graphics generation. *Artificial Intelligence*, 63:387–427, 1993.

[19] Y.Yang and H.Zhang. HTML page analysis based on visual cues. In *Sixth International Conference on Document Analysis and Recognition (ICDAR '01)*, 2001.