Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2001 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2001

An Intelligent Learning Assistant in Multimedia-based Interactive E-Learning

Jinwei Cao
University of Arizona

Dongsong Zhang
University of Arizona

Follow this and additional works at: http://aisel.aisnet.org/amcis2001

Recommended Citation

Cao, Jinwei and Zhang, Dongsong, "An Intelligent Learning Assistant in Multimedia-based Interactive E-Learning" (2001). *AMCIS 2001 Proceedings*. 27.
http://aisel.aisnet.org/amcis2001/27

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2001 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

AN INTELLIGENT LEARNING ASSISTANT IN MULTIMEDIA-BASED INTERACTIVE E-LEARNING

Jinwei Cao
Center for the Management of Information
University of Arizona
jcao@cmi.arizona.edu

Dongsong Zhang
Center for the Management of Information
University of Arizona
dzhang@bpa.arizona.edu

Abstract

Web-based educational systems provide students learning opportunities at anytime and anywhere. But most of them lack capability of tailoring the learning materials to meet individual needs. Intelligent Tutoring Systems (ITS) and Adaptive Hypermedia Systems (AHS) have explored solutions to this problem, but they have difficulty to interact with users efficiently. In this paper we introduce a new approach to creating an intelligent learning assistant in our Learning by Asking (LBA) project, a Multimedia-based Interactive e-Learning system, in which we combine ITS and AHS technologies together. Additionally, we use natural language processing to allow users to input questions in conversational English, multimedia materials (video clips) to represent domain knowledge, and case-based reasoning to produce personalized guidance to individual users based on their different questions and learning history. Integrating those techniques together, the intelligent learning assistant in the LBA system features an adaptive two-way communication between users and the system, thus provides a natural learning environment.

Keywords: Learning assistant, interactive e-learning, intelligent tutoring systems, adaptive hypermedia systems, case based reasoning

Introduction

With the development of telecommunication and the Internet technologies, Web-based distance education gives remote students the same chance of studying as local students at any time and anywhere. But students who take part in distance education differ greatly in their goals, professional background, interests and knowledge of the subject matter. It has been well recognized, therefore, that one of the primary disadvantages of traditional distance learning via the Internet is the lack of capability of tailoring the learning materials to meet individual needs (Brusilovsky 1998). Thus, seeking solutions that can dynamically adjust learning guidance and strategies is highly desired.

Artificial intelligence technologies provide some useful methods to achieve such a goal of adaptive education without real instructors. Intelligent Tutoring Systems (ITS) and Adaptive Hypermedia Systems (AHS) are two of such methods used during the past decades (Burns 1988; Brusilovsky 1994). ITS brings three types of knowledge - expert knowledge, student diagnostic knowledge and instructional knowledge - together to build an "intelligent tutor" (Burns 1988). The other method, AHS, enhances classic hypermedia by adapting the content of a hypermedia page to the users' knowledge and goals, as well as by suggesting the most relevant links to follow (Brusilovsky 1994). Although very few earlier stand-alone ITS used adaptive hypermedia, today some Web-based adaptive educational systems have shown the tendency to adopt a hybrid of both ITS and AHS.

However, a system cannot get real adaptive interaction with users by merely merging those two methods together. AHS generally only provides techniques for adaptive presentation and adaptive navigation support without user initiated input (Brusilovsky 1996). The problem of interacting with users also has perplexed the researchers of ITS for a long time because of the difficulty of using natural language (Anderson 1988). To solve this problem and to provide a real user-centered learning environment on the Web, we present a new approach to creating an intelligent learning assistant in our Learning by Asking (LBA) project that is developed for multimedia-based interactive e-Learning.

The LBA system allows users to learn via the continuous interaction with 'virtual mentors'. For example, users ask questions in natural language and learn by retrieving and viewing appropriate video clips that contain the expert's answers (Zhang et al. 2000). It is motivated by constructivism learning theory that emphasizes the active participation and reflection by the learners who are encouraged to take the initiative in interacting with instructors (Simons, 1992). Considering that a novice in a certain domain may know little about the domain at all and may not know any question to ask, we have to provide a learning assistant in the LBA system. This assistant should give users the navigation guidance that tell them which questions they may be interested in when they begin to learn or after they have asked a couple of questions. Also, in order to help users select from the automatic retrieved video clips, the learning assistant should narrow the scope by providing suggested clips from domain experts that can answer the question. Our approach of building such an intelligent learning assistant integrates ideas from both ITS and AHS with some other AI technologies, namely natural language processing and case-based reasoning. Currently we are developing a prototype LBA system that teaches undergraduate students about the Internet and its basic applications. The prototype intelligent learning assistant provides dialogue-like two-way communications between students and virtual mentors, and such a procedure is very close to natural learning. In this paper we introduce the design and the implementation of the intelligent learning assistant in LBA. First of all we describe the related research. Then we describe the LBA system architecture and the approach we use to get adaptive instructional guidance. Finally, we demonstrate the preliminary results and the future direction.

Related Research

The design of the intelligent learning assistant in the LBA system is primarily inspired by ITS and AHS concepts.

Much prior ITS has been developed since 1970s'. In an ITS, the role of a human teacher is replaced by a computer tutor, which basically consists of four main interconnected modules, as shown in Figure 1 (Burns 1988).

1. *Domain Knowledge Module,* which is represented as the domain knowledge base;
2. *Teacher Module,* which contains teaching strategies for guiding the learning process of students;
3. *Student Module,* which contains information that is specific to each individual student;
4. *User Interface Module,* which enables interaction among student, teacher and domain knowledge.

Most of the earlier ITS were stand-alone programs and focused on interactive problem solving support. Such an ITS is limited to the domains that have explicit procedural knowledge such as computer programming or calculating. In many other walks of life, however, students will need to understand the basic principles and facts in a domain. Referring to the ACT-R theory of skill knowledge (Anderson, 1993), such factual or experiential knowledge is called declarative knowledge. Unfortunately, those earlier stand-alone ITS had difficulty in engaging in such declarative tutoring (Anderson 1988). But since the adaptive educational systems developed from stand-alone to Web-based, AHS shows the potential of coming up with such declarative knowledge.

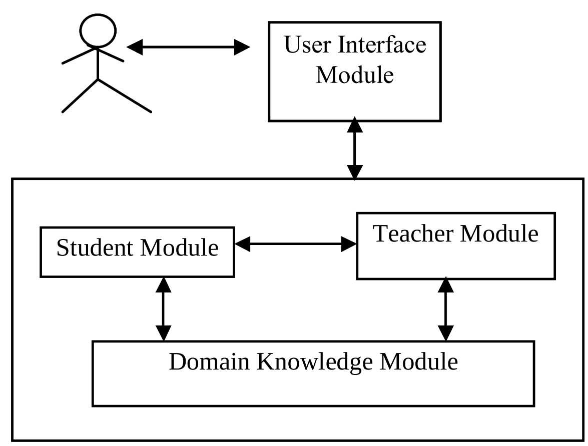

Figure 1. General ITS Architecture

Brusilovsky defined in 1996 that "adaptive hypermedia systems are all hypertext and hypermedia systems that reflect some features of the user in the user model and that apply this model to adapt various visible aspects of the system to the user". Hypermedia, which is a combination of multimedia and hypertext, has a node-and-link structure inherent to the organization of information and is based on integrated media (Heller 1992). Since a semantic network can easily represent declarative knowledge (Carbonell 1970), hypermedia has provided a good method to do declarative tutoring through reorganizing the nodes and links in the knowledge network. Also some research results have shown that multimedia, such as video or audio, in hypermedia systems can represent the declarative knowledge more vividly than plain text, in order to make students learn more efficiently (Zhang 1995).

Similar to ITS, an AHS also uses a *domain model* to contain a set of domain concepts. The concepts in the domain model are related to each other, thus forming a kind of semantic network. For each concept in the domain model, an individual user model called an *overlay model* stores some value as an estimation of the user knowledge level on this concept. But AHS in education ignores instructional strategies, which are represented by the teacher module in ITS, so it lacks control of the whole learning process.

Currently, some Web-based educational systems have integrated both ITS and AHS technologies. For example, **ELM-ART** is a Web-based ITS presented as an on-line intelligent textbook to teach an introductory LISP course at the University of Tier (Brusilovsky et al. 1996). ELM-ART makes a good user model and adaptive presentation and navigation, but it only has text material and is not as efficient as multimedia materials when teaching some descriptive concepts (Zhang 1995). **MANIC** is a project that use multimedia technology to deliver stored course materials such as class notes/overheads and the audio/video of classroom lectures to students over the Internet (Stern et al. 1997). MANIC adopts multimedia materials but it did not have efficient user model and navigation control. Also, both of these systems show another limitation of many ITS and AHS; that is, they did not provide the function to let users input their questions and only allowed them to go along the existing links. So the interaction between users and system is limited to some extent. Maybe this is due to the difficulty of natural language processing, just as John Anderson said in 1988: "…natural language is the Achilles' heel of any effort to do such declarative tutoring." And until now this is still the bottleneck that perplexes most of the adaptive educational systems.

LBA System Architecture and Design

Comparing with the above Web-based educational systems, the LBA system has some new features. For example, it allows students to ask questions in natural language, and it uses video clips that contain the experts' answers to represent the domain knowledge. The intelligent learning assistant in the LBA system helps to guide such a "learning by asking" process.

LBA is implemented as a client-server architecture that is depicted in Figure 2. It consists of three parts: a Web client, a Web server and a video repository. To make these parts work, we first videotape some experts in a particular domain according to different topics during the lectures or interviews. Then, videos are logically segmented into individual clips in a way in which each clip focuses on a specific subject or can answer a question. In our system, these video clips called "Virtual Mentors" represent domain expert knowledge. They are stored in a video repository (Video Streaming Server). A user can ask a question in plain English through the user interface. The question is then sent to the Web server, and finally a list of pertinent video clips from video searching and the guidance from the intelligent learning assistant will be returned to the user simultaneously.

Web Client

Individual students get into the learning environment through a Web user interface. They can ask questions through typing natural language queries. The guidance from the intelligent learning assistant and the links to pertinent video clips from video searching will be displayed on the interface at the same time. In the LBA system, the client side is platform-independent. Users only need a Web browser, a video player and a sound card to watch the video presentation.

Web Server

The Web server contains three modules explained as following:

1) Question understanding module, which is used to parse questions based on syntactic and semantics analysis;
2) Learning Assistant module, which works with knowledge base and inference engine to provide the intelligent guidance for students. It will be discussed in the following section;
3) Video clip retrieval module, which determines how to search in the metadata database and identify candidate video clips whose contents might answer the questions.

Video Repository

All the video files are stored in this video-streaming server. They are logically segmented into short clips that can answer specific questions

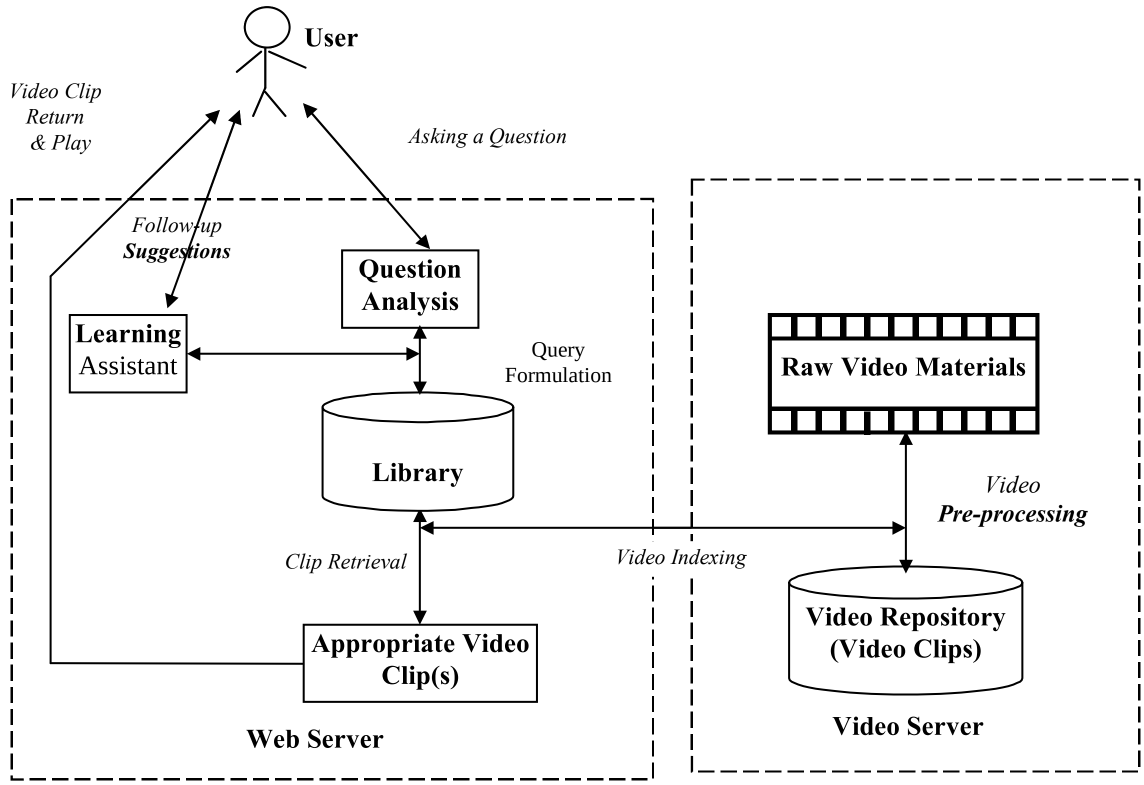

Figure 2. Overall Architecture of LBA System

Intelligent Learning Assistant in LBA

The intelligent learning assistant module in LBA is a key component that enables the intelligent features of the LBA system. We integrate both ITS and AHS technologies together with case-based reasoning into the design of this intelligent learning assistant. Also, this assistant works with the question understanding module in the LBA system to get the users' questions in natural language.

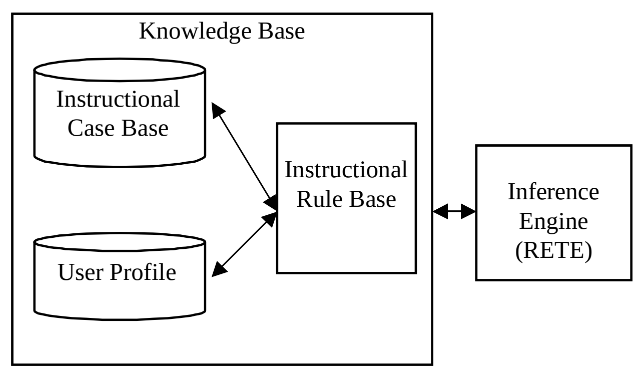

Figure 3. Architecture of Intelligent Learning System

Depicted in Figure 3, the learning assistant module includes a knowledge base and an inference engine. Here the knowledge base contains three sub-components and maps into the structure of a traditional ITS's knowledge base, namely, an instructional case base containing domain knowledge, an instructional rule base representing the teacher module, and a user profile representing the student module. They are discussed in details in the rest of this section.

Instructional Case Base

The objective of this case base is to represent domain knowledge. In LBA, each video clip focuses on a specific subject and can be viewed as a knowledge node in the domain knowledge network. Each clip can answer certain questions. So the association between two different questions or between a question and a clip can be seen as the link in the knowledge network. The knowledge represented in video clips is declarative and descriptive, and is difficult to be expressed as explicit rules in rule-based reasoning (Watson 1997).

However, this problem of knowledge representation can be solved by case-based reasoning techniques (CBR) (Watson 1997). Simply speaking, a case-based reasoner solves new problems by adapting solutions that were used to solve old problems (Riesbeck et al. 1989). So typically CBR is described as a cyclical process comprising four steps:

1) Retrieve the most similar case(s).
2) Reuse the case(s) to attempt to solve the problem.
3) Revise the proposed solution if necessary.
4) Retain the new solution as a part of a new case.

According to CBR, each individual case contains a problem description and a stored solution to build an association between them. Similarly in this learning assistant's instructional case base, each case is composed of a question that users are likely to ask and the related suggestions from an expert who clearly understands the domain knowledge network. Such a suggestion can be divided into two parts. One is a list of video clips that can most likely answer the question in the experts' mind. The other is a list of follow-up questions that the users might be interested in after understanding the current question. So the cases can set up associations both between different questions and between questions and clips. When students ask particular questions, the learning assistant will at first search the case base and retrieve the most similar cases. After that, it reuses the cases to attempt to give a suggestion. But in order to make the case specific to an individual user, the proposed suggestions need to be revised by applying instructional rules and considering the current learning status of the user. We will describe the instructional rules and user profile in detail in the next subsection.

Natural language processing in the question understanding module plays an important role in the case representation and retrieval. We used Conexor iSkim (Voutilainen 2000), a natural text processing component for IT applications, to help parse the questions and obtain the indexing terms. iSkim is able to analyze English texts and produces five types of information: part of speech (POS), lemma, morphology, light syntax, and named entity recognition. We encode the output of iSkim to a question template shown in the example below. Each of the possible questions is processed to such a question template and is stored as part of a case. The suggestion part of the case has pointers to the metadata of video clips and pointers to other questions. All the cases are stored in the database. The incoming questions from users are parsed to fill out the question template and the system intends to search for closest matching questions in the cases. Finally by asking several questions to confirm the matching questions, the known suggestions to the best-matched question will be offered. A concrete example is as follows:

Suppose a user asks a question: *What was the year that the Internet was first used?*

The question understanding module will parse this question and generate a question template as:

Question Category: (11) What + BE + <time>
Answer Type: <time>
Person: NULL
Organization: NULL
Verb Phrase: first use
Object: Internet
Time: NULL
Location: NULL

Since the Object slot usually contains the keywords of the question, the question understanding module will give the synonyms of the words in this slot. In this particular example, they include “network”, “web”, “www”, etc. Also we have a collection of rules that can find the similar question categories, for example, question category *(11) What + BE + <time>* is identical to question category *(42) When*. All the case questions stored in the database have the same structure as the above question template. So if there is a record in the database as:

Question Category: (42) When
Answer Type: <time>
Person: people
Organization: NULL
Verb Phrase: begin, use
Object: Internet
Time: NULL
Location: NULL
Question: When did people begin to use the Internet?
Suggestion ID: 16

After searching by all the attributes and the synonyms, as well as applying the question category rules, we find this is a matched record and will get the suggestion ID from this record (suggestion 16). Finally the content of the suggestion 16 will be delivered to the user. The contents may suggest that the user see clip 4 and ask other related questions as "What is the Internet?" "How to use the Internet?" etc.

Instructional Rule Base

In the instructional rule base, the rules are the representations of teaching strategies for individual students. The retrieved suggestions from instructional case base are generic suggestions to every student. To make the learning assistant personalized and intelligent, we build these instructional rules to work on the facts coming from the case base and the user profile, and then revise the general instruction case of a certain question based on the background and learning history of students. The instructional rules, obtained from the human instructors who teach in the domain, are represented by the production rules. For example:

R1
IF *student A asks question B that exists in case C* AND *A is at level 1 (novice level) in this domain*
THEN *fetch the suggestion1 from case C*

R2
IF *A has watched clip D before* AND *D is in suggest 1*
THEN *Remind A that he/she has viewed D before*

In the LBA system we implemented these rules by JESS (Java Expert System Shell, http://herzberg.ca.sandia.gov/jess/) script language and embedded the JESS script in a Java program. JESS is a rule engine and scripting environment inspired by the CLIPS expert system shell and written entirely in Sun's Java language by Sandia National Laboratories. Java's portability makes sure that the rule base is cross-platform and is fit for the Web application.

User Profile

This is a dynamic database that contains general information and the learning status of each user. The basic information of individual users such as username and password, the background knowledge and personal interest will be stored in the database when users register for the system. The user records of their previous learning behaviors, such as the questions they have asked, the video clips they have seen, and other relevant information, will be captured on the fly and stored in the database. Each user has a score representing his or her domain knowledge level. This score is a weighted sum of all the above information and is dynamically changed with the student's learning process. The instructional rules will give personalized guidance to each individual user based on the score and his/her learning history.

Inference Engine

The Inference Engine (IE) is used to interpret the knowledge. In the LBA system, it is provided by JESS, using a very efficient method known as Rete algorithm (Forgy 1982). Therefore, we do not need to concern ourselves about the design of the IE, and this reduces the difficulty and complexity of the implementation.

Preliminary Implementation

So far, we have built a prototype of the LBA system in the domain of introduction to the Internet. As to the intelligent learning assistant, we created a relational database to hold information on the instructional case base, the user profile and metadata of all video clips. Currently we have about 150 cases, and those cases point to almost 100 real-format video clips in the video repository. The size of the case base, the rule base and the video depository is not very big but it is enough for testing purpose. We use JESS as a core tool to set up the rule base and let its RETE engine get the adaptive instructional guidance. A user-friendly interface has been built using HTML. Java Servlet and JDBC (Java DataBase Connectivity) technologies are used to connect the front end client and back end knowledge base, to retrieve the case(s), as well as to finally return the video search results and personalized guidance to the user computer. The learning assistant has integrated with other main components in the LBA system and is currently under evaluation.

Conclusions and Future Work

In this paper, we present the design and implementation of an intelligent learning assistant module in the ongoing LBA project, a multimedia-based interactive e-Learning system. This module provides users an environment similar to their natural learning processes. When the cases and rules in the knowledge base increase gradually, the learning assistant will work as a virtual mentor on-line.

The future research will focus on evaluation and improvement of the intelligent learning assistant. We intend to use the undergraduate students who register for the course " Introduction to the Internet" at our University as our subjects to test the first prototype. Automatic knowledge acquisition is another topic that attracts us. It has been recognized as a bottleneck for knowledge based systems for a long time. Since the case base we are building cannot hold all the possible questions from the students, and sometimes the suggestions from the domain experts are not complete and precise, it is appealing to develop an automatic knowledge acquisition method for the learning assistant. Ideally such a method can automatically capture new cases and insert them into the case base, and it can modify some existing cases and rules based on statistical results. With such an automatic knowledge acquisition method, the learning assistant can keep learning by itself, and the knowledge base will evolve incrementally.

References

Anderson, J. R. "The expert Module," in *Foundations of Intelligent Tutoring Systems*, Martha C. Polson and J.Jeffrey Richardson (eds.), Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, pp. 21-53, 1988.

Anderson, J. R. *Rules of the Mind*, Hillsdale, NJ: Lawrence Erlbaum, 1993.

Brusilovsky, P. "Adaptive Hypermedia: An Attempt to Analyze and Generalize," in *Workshop on Adaptive hypertext and hypermedia* at UM'94, Hyannis, Cape Cod, MA, 1994.

Brusilovsky, P. "Methods and Techniques of Adaptive Hypermedia," *User Modeling and User-Adapted Interaction*, 6, 1996, pp. 87-129.

Brusilovsky, P., Schwartz, E., and Weber, G. "ELM-ART: an Intelligent Tutoring System on World Wide Web," *ITS96- The Workshop: Architecture and Methods for Designing Cost-Effective and Reusable ITS*, Montreal, June 10-14, 1996.

Brusilovsky, P. "Adaptive Educational Systems on the World-Wide-Web: A Review of Available Technologies," *"Web-based ITS" at ITS'98, 4th International Conference in Intelligent Tutoring Systems*, San Antonio, Texas, 16-19 August, 1998.

Burns, H. L., and Capps, C. G. "Foundations of Intelligent Tutoring Systems: An Introduction," in *Foundations of Intelligent Tutoring Systems*, Martha C. Polson and J.Jeffrey Richardson (eds.), Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, 1988, pp. 1-19.

Carbonell, J. R. "AI in CAI: An artificial intelligence approach to computer-aided instruction," *IEEE Transactions on Man-Machine Systems*, 11, 1970, pp.190-202.

Forgy, C. L. "Rete: A Fast Algorithm for the Many Pattern/Many object Pattern Match Problem," *Artificial Intelligence,* 19, 1982, pp. 17-37.

Heller, L. "Hypermedia" in *Computer-based instruction: Methods and Development*. Prentice Hall, Englewood Cliffs, NJ, 1992.

Riesbeck, C. K., and Schank, R. S. *Inside Case-Based Reasoning,* Northvale, NJ: Lawrence Erlbaum Associates, 1989.

Simons, P. R. J. "Constructive learning: the role of the learner," in *Designing for Constructive Learning*, T. M. Duffy, J. Lowyck, and D. H. Jonassen (eds.), Berlin: Springer-Verlag, 1992.

Stern, M., Steinberg, J., Lee, H.I., Padhye, J., and Kurose, J. "MANIC: Multimedia Asynchronous Networked Individualized Courseware," in *Educational Media and Hypermedia*, 1997.

Voutilainen, A. "Helsinki taggers and parsers for English," in *Corpora Galore: Analysis and techniques in describing English,* John M. Kirk (ed.), Rodopi: Amsterdam & Atlanta. 2000.

Watson, I. D. *Applying Case-Based Reasoning - Techniques for Enterprise Systems*, Morgan Kayfmann Publishers, Inc., San Francisco, CA, 1997.

Zhang, D. S., and Nunamaker, J. F. "A Multimedia-Enabled System for Interactive Self-paced Learning," *Proceedings of the IASTED International Conference*, November 19-23, Las Vegas, 2000, pp. 351-355.

Zhang, W. "Multimedia, Technology, Education and Learning," in *Technological Innovations in Literacy and Social Studies Education*, 1995. Available: http://tiger.coe.missouri.edu/~sslit95/