

Grounding Collaborative Knowledge Building in Semantics-Based Critiquing

Anders I. Mørch¹, William K. Cheung², Kelvin C. Wong², Jiming Liu²,
Cynthia Lee³, Mason H. Lam², and Janti P. Tang²

¹ InterMedia, University of Oslo, Norway
anders.morch@intermedia.uio.no

² Department of Computer Science, Hong Kong Baptist University, Hong Kong
{willlliam, kcwong, jiming, mason, janti}@comp.hkbu.edu.hk

³ Language Centre, Hong Kong Baptist University, Hong Kong
cfklee@hkbu.edu.hk

Abstract. In this paper we investigate the use of Latent Semantic Analysis (LSA), Critiquing Systems, and Knowledge Building to support computer-based teaching of English composition. We have built and tested an English Composition Critiquing System that make use of LSA to analyze student essays and compute feedback by comparing their essays with teacher's model essays. LSA values are input to a critiquing component to provide a user interface for the students. A software agent can also use the critic feedback to coordinate a collaborative knowledge building session with multiple users (students and teachers). Shared feedback provides seed questions that can trigger discussion and extended reflection about the next phase of writing. We present the first version of a prototype we have built, and report the results from an informal experiment. We end the paper by describing our plans for future work.

1 Introduction

English is the preferred second language for many people and learning it occurs in many ways. For example, young people are quite apt in learning spoken English phrases when watching TV, browsing the Internet and communicating with peers on mobile phones (e.g. SMS). However, previous studies have shown these influences may have negative effect on vocabulary development [19, 27]. As a consequence, students' reading and writing skills do not keep pace with listening and speaking. Furthermore, English composition is primarily taught in the classroom and practiced in homework assignments, supported by qualified teachers and parents. These are important but scarce resources, creating an imbalance of textual and oral language exposure. We address this dilemma by augmenting classroom-based composition training integrated with computer support.

The paper is organized as follows. We start by characterizing English composition as a design activity and identify the components of a computer-based design environment to support it. Next, we explain how latent semantic analysis (LSA) can be

used to provide feedback on student compositions within this context, and how we have incorporated LSA as part of system architecture. We show a prototype of a critiquing system we have built, discuss our efforts in integrating it with a knowledge-building environment (FLE) and report the preliminary findings by comparing LSA with manual teacher feedback on a set of essays.

2 Related Work

Essay writing can be viewed as a design activity, producing a textual artifact - a document. A document consists of words and sentences. It has structuring (abstraction) and content production (composition) elements [28]. These are key aspects of any design process. Structuring defines the organization of the document in terms of sentences, paragraphs and sections (i.e. levels of abstraction); whereas content production is about finding words and phrases, and sequencing them into readable sentences, which again become part of paragraphs, and so on. A well-composed essay will communicate certain ideas, topics or themes about some area of shared concern. Intermediate level abstractions, such as paragraphs and sections, serve as placeholders for complex ideas extended over multiple paragraphs, so that the writers and readers can focus on one idea at a time while suppressing unimportant details.

Creative design is said to consist of two sub-activities, action and reflection, supporting composition and abstraction, respectively. Action means to create an artifact by selecting and combining building blocks into functional arrangements and reflection means to evaluate the artifact from multiple viewpoints [16]. When this occurs without external disruption other than situation-specific feedback, it is referred to as reflection-in-action [23]. In a good process of design, the designer will rapidly cycle between action and reflection until the design is completed. During this process, the “back talk” of the situation signals to the designer when there is a need to switch to the other mode. This is communicated by e.g. an incomplete drawing, inconsistency in arrangement of parts, a need for restructuring the task, etc.

2.1 Design Critiquing

Computational support for reflection-in-action is provided with the critiquing approach [7, 18, 21]. Critiquing is defined as “presentation of a reasoned opinion about a product or action” created by a user with a computer [7]. A Critiquing System integrates computational support for design-as-action and design-as-reflection and operationalizes Schön’s notion of “back talk” with computational critics [7, 20]. Critics make the situation talk back so that non-expert designers can understand it, giving them task-specific feedback about the artifact-under-construction. Examples of critiquing systems are Janus [16], ArgoUML [20], and The Java Critiquer [18]. These systems were developed for the domains of kitchen design, UML (Unified Modeling Language) and Java programming, respectively. For example Janus allows designers to model kitchen designs at different levels of abstraction (from appliances to work centers), ArgoUML knows about the elements and relations of UML and can tell the designer when a software architecture diagram violates the rules of UML [21]. Simi-

larly, the Java Critiquer identifies statements in a program code that can be improved by readability and best practice [18]. These critics provide feedback on partially completed software artifacts, pointing out inconsistency and incompleteness in the design.

We believe the critiquing approach can be useful for computer-supported English composition for the following two reasons. First, writing can be modeled as a design activity [28], and second, critic feedback can supplement teacher feedback on student essays in certain situations (after school hours, in distributed environments). In this context we propose to integrate collaborative knowledge building and LSA with critiquing- Knowledge building to support collaborative reflection and LSA to compute the critic feedback. This is different from past work on critiquing systems and educational applications of LSA. The previous work on LSA has focused on individual learning by integrating it with Intelligent Tutoring Systems [24]. A goal for us is to provide computer support for both action and reflection, and individual and collaborative learning.

2.2 Collaborative Knowledge Building

Knowledge building [22] requires that new knowledge is not simply assimilated with the help of a more knowledgeable person or mediated by a computer system, but also jointly constructed through solving problems with peers by a process of building shared understanding. This type of teaching and learning takes its inspiration from pedagogical models such as problem-based learning and case-based instruction. These are models for teaching that require students to explore open-ended problems and generalize from exemplary cases. The basic idea of knowledge building is that students gain a deeper understanding of a knowledge domain through engaging in a research-like process by generating or responding to problems or questions, proposing tentative answers (personal explanations) and searching for deepening knowledge collaboratively.

Knowledge building and its subsequent refinement, Progressive Inquiry [8] are well suited to be supported by Internet technologies such as web-based discussion forums and have received considerable attention in the Computer Supported Collaborative Learning (CSCL) community. A reason for this is that the regularity of knowledge building, which is modeled after scientific discourse, provides students with a well-defined scaffolding structure built into the online learning environments. Knowledge building environments are pedagogically designed discussion forums and include CSILE [22], Knowledge Forum, and Future Learning Environment (FLE) [10]. They are used in schools in Canada, Hong Kong and Scandinavia, as well as elsewhere in the world.

The reason for our wish to integrate collaborative knowledge building with a critiquing system is twofold. First, critiquing systems do not provide full support of design-as-reflection because they address primarily individual designers' needs, inspired by Schön's notion of reflective practice [23]. Knowledge building, on the other hand, can add a multi-user dimension by supporting "collaborative reflection," even though knowledge building was not originally conceived as such. Collaborative reflection occurs during "talk with peers" [e.g. 15] in meaningful contexts, i.e. jointly

addressing problems or questions shared by a community of stakeholders in which shared understanding can emerge [2]. Knowledge building thus becomes an important part of the integrated collaborative learning and problem-solving environment.

Second, one of the authors has previously participated in a study to evaluate a knowledge-building environment (FLE) to support problem-based teaching of natural science in two high school classes in Norway [14, 17]. One of the results of this study was that students found knowledge building difficult. In particular they did not properly understand how to use the message categories to post messages in the forum. This was manifest in that interaction over time became less knowledge-building intense and more task-specific (localized), revolving around the respective schools' local situations, thus grounding the interaction. In the current project we address the grounding problem [2, 3, 4] computationally by integrating a knowledge-building environment with an LSA-based critiquing system.

2.3 Latent Semantic Analysis

Latent Semantic Analysis (LSA) is a mathematical technique for computing the semantic similarity between words and text segments with the help of a large corpus. The corpus can be a set of related documents. It can also be one document broken down into smaller text segments such as paragraphs or even sentences, as in our case. The input to LSA is the set of text segments, which may need processing by the computer in various ways.

LSA computes the similarity of two input texts (student and teacher) as follows. First, both input texts are segmented to form the corpus. Then, the word-segment association matrix D is constructed. In the matrix D , each row stands for a unique word and each column stands for a text segment. Each cell entry can be the frequency of a given word in a given text segment. As an example, consider the segment "The 2004 IEEE International Conference on Electronic Technology, Electronic Commerce and Electronic Service." If the j^{th} column corresponds to the aforementioned segment and the i^{th} row corresponds to the word "Electronic", then the value in D_{ij} would be 3 as "Electronic" occurs three times in the segment. As weighting the words based on their individual importance is known to be effective in obtaining better matching results, we use entropy values instead of computing D_{ij} given as

$$D_{ij} = \log(f_{ij} + 1.0)^* \left\{ 1 - \frac{1}{\log(N)} \sum_{j=1}^N \left[\frac{f_{ij}}{gf_i} \log \frac{f_{ij}}{gf_i} \right] \right\}$$

$$gf_i = \sum_{j=1}^N f_{ij}$$

where N : number of text segments in the stored corpus

f_{ij} : frequency of the i^{th} word in the j^{th} text segment

Once the matrix D is computed, it is decomposed using Singular Value Decomposition (SVD) [25]. The matrix D is expressed as a unique product of three matrices: $D = P\lambda Q'$ such that P and Q have orthonormal columns and λ contains the singular

values on the diagonal. By comparing the diagonal elements of λ , we only need to keep those elements with large values and can set the others to zero, with the effect that the dimension of λ will be reduced. This is equivalent to removing the corresponding columns from P and rows from Q . The new P and Q then define the revised “semantic space.” Words that have appeared in similar segments, and segments with similar semantic content, will be positioned near one another [24]. Words that do not co-occur (e.g. bicycle and bike), but occur in similar contexts will also be grouped together.

After the semantic space has been computed the new D can be “reconstructed” based on the new P and Q . The similarity between two text segments can then be computed by calculating the geometric cosine between their corresponding vectors in D , given as

$$\cos \theta = \frac{X \bullet Y}{\|X\| * \|Y\|}$$

where $X \bullet Y$ is the inner product of vectors x and y and defined as follows

$$X \bullet Y = x_1 y_1 + x_2 y_2 + \dots + x_n y_n, \text{ and } \|X\| \text{ is the length of a vector } X,$$

$$\text{defined as } \|X\| = (X \bullet X)^{1/2} = (\sum_{i=1..n} x_i^2)^{1/2}$$

Remark. In order to allow the latest submitted essays to be compared with the corpus in the derived semantic space; one can project them by the SVD results. These are referred to as pseudo documents [5]. This has the advantage that we only need to compute the SVD once instead of per submission. Additional technical details on LSA can be found in [9].

3 Components of a Learning Environment for Essay Writing

We have incorporated *LSA* together with *critiquing* and *knowledge building* to form an integrated learning environment for English Essay Writing. The LSA-based critiquing component of this environment allows us to compare student and model essays and provide critic feedback to the students when they submit their work in progress, whereas the knowledge building component provides support for collaboratively resolving critic feedback that is not well understood by the students on their own, grounding it in a relevant context. The overview of this environment is shown in Figure 1 and the workings of its components are explained below.

The teacher first decides on the topic to be taught and writes and/or collects a set of samples articles and essays that represent the domain in some detail. These samples are then input into the system so that the LSA analyzer can build a semantic space for the domain. Student model essays, suggested answers by teachers, as well as articles from external sources (which could be anything from on-line newspapers to scanned essays of textbooks) constitute this set.

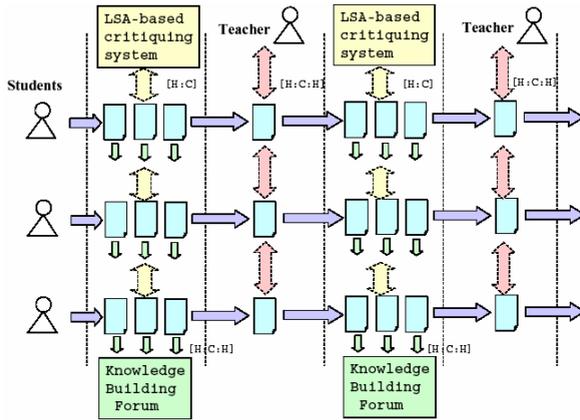


Fig. 1. English composition integrated learning environment system architecture

The students write their essays using the English Composition Critiquing System (see below). When they require assistance they can request automated feedback (critique), which points out missing items in their text (compared with the corpus essays). Before the text can be input into LSA all the articles are broken down into sentences and preprocessed by techniques such as stop-word removal and stemming as described above. The Analyzer then computes the word-segment association matrix. Singular Value Decomposition [25] is performed on the matrix and the semantic similarity between all possible sentence pairs, one from the student and the other from the model samples, is computed. This allows the system to identify the sentences in the model essays that contain themes that are missing in the students' submissions.

The final steps are semantic matching and summarization. The identified sentences containing the missing themes can be summarized as a trunk of keywords or short phrases preset by the teacher or automatically by the system, using computational text summarization techniques. This will result in a summary that is reported as critic feedback in the user interface. In the prototype we describe below, we have modeled our critics' feedback based on the phrasing and organization of Hong Kong English teachers' marking schemes. When the critique is presented as feedback immediately after the students have completed a part of their essay, it will allow them to revise their essays in a meaningful context.

The roles of teachers and students could be much more active than merely providing model samples and improving essays based on the predefined critic feedback. Teachers can monitor how well the different themes are handled by the students. They may provide more insight into how individual students incorporate missing themes, and participate as facilitators of student collaboration sessions to provide feedback when the students run out of ideas [11]. Their participation serves the purpose of supportive interaction through which an expert assists a group of learner to develop a higher level of understanding [e.g. 15] and pushes the learner's zone of proximal development [26]. A recent large-scale language learning survey has confirmed the observation that most

students in East Asian and European countries have a positive attitude towards cooperating in groups in order to achieve common goals, and they would like to see themselves as active participants in the classroom learning process [12].

The LSA-based critiquing and knowledge building environment marks the contours of a “double-loop” learning process (see Figure 1), alternating between inner (individual) and outer (collaboration) phases. The process can be repeated several times before the students submit their final essay for grading or commenting on by the teacher. In a good process of writing, we anticipate this learning environment will support reflection-in-action at two levels: 1) individual (inner loop) activity when students switch between essay composition and modification by responding to well understood critique and 2) collaborative (inner + outer loop) activity by entering a collaborative mode of interaction through responding to critique that is not well understood or where the understanding can be broadened or made more interesting for the students by sharing their ideas with others. Whether or not our computational environment can provide adequate scaffolding for reflection-in-action in English essay writing at these two levels is currently a hypothesis. Its conceptual basis and technological platform are provided in this paper.

3.1 System Prototypes

In order to support English essay writing as a design activity based on the models and techniques presented above we decided to reuse and integrate existing systems, making modifications when necessary. When selecting the critiquing component we considered both ArgoUML [21] and the Java Critiquer [18]. The latter has the advantage that it supports the design of a textual artifact (program code), but ArgoUML has the advantage it is a freely downloadable, open source system. We decided on ArgoUML due to its accessibility. However, we had to modify the system extensively, with the effect its UML features are no longer visible (see Figure 2).

The system building approach we took was to start with the existing system, removing all the features we did not need and adding the features that are unique to our domain. In retrospect we could have built the equivalent functionality from scratch in about the same time. However, some of the Argo features we have hidden, such as the building block palette and to-do list we anticipate to be useful in future versions of our system. For example, the current version requires students to input their essays in terms of characters and words (i.e. the composition area is a text processing window), whereas LSA Analyzer requires sentences as input. Furthermore, we know from previous studies that designers work at multiple levels of abstraction [1], which in the context of essay composition means words, sentences, paragraphs, sections, and other higher level structures [28]. Therefore, providing support for direct manipulation of intermediate-level building blocks is another way to extend the current system. It will allow the students to acquire skills in organization as well as composition, and it may simplify LSA preprocessing by reducing the need for sentence segmentation. Furthermore, to-do lists that can keep track of overlooked critic messages and suggest when they should be attended to, can help students to manage multiple missing sub-themes. This is part of future work.

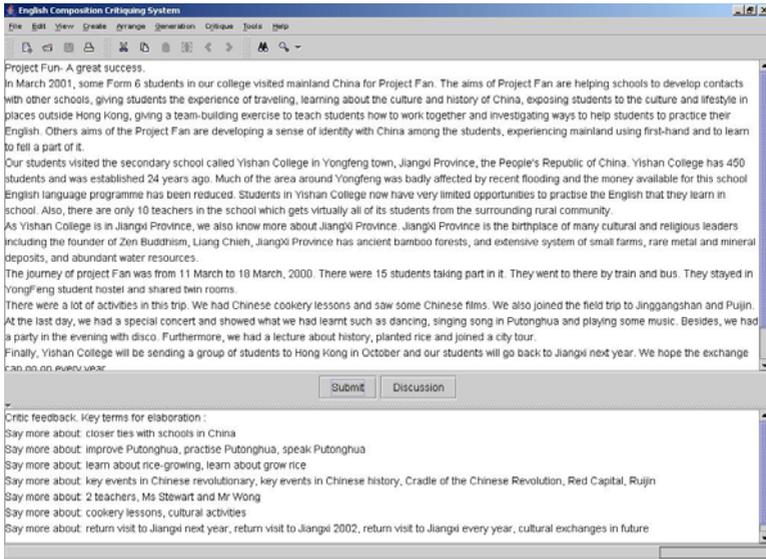


Fig. 2. The English composition critiquing system has a “Submit” button to generate LSA-based critique and a “Discussion” button to trigger a knowledge building session (Figure 3)

For the knowledge-building component we decided on another open source system, FLE (Future Learning Environment) [10]. FLE is a knowledge building environment developed in accordance with the progressive inquiry model [8]. It is an asynchronous, web-based groupware for computer-supported collaborative learning. It is designed to support collaborative learning in the form of a discussion forum with message categories (knowledge types) named after the stages of the progressive inquiry model. These stages and corresponding categories can help students improve their collaboration and ability to solve open-ended problems. The categories that are provided with the system (Fle3) are “*problem*,” “*my explanation*,” “*scientific explanation*,” “*summary*” and “*comment*.” Two of these categories are displayed in Figure 3.

Figure 3 shows the reader’s interface of the knowledge-building forum of Fle3 from a simulated session involving two students who have been invited to join the forum by a coordinator agent to resolve a missing essay topic. The missing essay topic is picked up by the agent and serves as a seed question. In knowledge building these initial questions are often formulated by teachers, based on their knowledge of the subject to be taught. In this case it is handled by a software agent based on its ability to identify students who receive the same feedback, and a belief that two students receiving the same feedback have something in common that they can resolve by information sharing and discussion. The reason why discussion may be the appropriate form of resolving the feedback is based on the fact that missing sub-themes define open-ended questions, i.e. they can be addressed in many different ways. We have not yet tested these claims, but it builds on our previous (empirical-based, system building) work on integrating agents with FLE [6] and adaptive user interface agents [13].

States) [24]. The TASA-all corpus comprises approximately eleven million words. We believe that a larger corpus for constructing our semantic space will further improve the accuracy of our system in identifying missing sub-themes. We will explore this hypothesis by varying corpus size for different domains in our further work

5 Conclusions and Directions for Further Work

Many students find essay writing stressful because they do not have sufficient ideas to fully cover the topic they are asked to write about. They usually run out of ideas before they have completed their essays. When the class size is large and when running in-class writing exercises, it is difficult for teachers to give proper feedback to individual students on missing topics to address in their rewrite, because it requires considerable amount of teachers' time.

We believe the use of our semantic-based critiquing system can support students by autonomously suggesting what missing sub-themes they should pay attention to when revising their essays. Students can submit their draft essays to the system for feedback whenever they are running out of ideas. If the feedback is incomplete or poorly understood (e.g. due to LSA truncation steps) the students can enter a system-initiated, contextualized discussion forum that provides support for collaborative knowledge building using the pedagogically informed progressive inquiry model. We believe that this can help students to enrich their essay content with more vocabulary in a context that is meaningful to them, thus grounding their learning activity. We are also interested in how the students view the critiquing system and to what extent the knowledge-building forum will be used. On the technical (algorithmic) side, we want to investigate the factors that will affect the performance of LSA in the essay-writing domain. For instance, how to determine both the optimal number of dimensions of the semantic space and the optimal threshold value for similarity matching are questions that require further research to answer.

Acknowledgments

This work was partly supported by Faculty Research Grant of Hong Kong Baptist University-FRG/02-03/II-38 and Group Research Grant of Research Grant Council (RGC) - HKBU 2/03/C. The first author was as a Visiting Research Scholar at CS department, Hong Kong Baptist University while this paper was written, and he thanks InterMedia and Faculty of Education, University of Oslo for a sabbatical leave.

References

1. Akin, O: How Do Architects Design? In: Latombe, J. (ed.): *Artificial Intelligence and Pattern Recognition in Computer Aided Design*. North-Holland, New York, NY (1978) 65-104
2. Arnseth, H. C., Solheim, I: Making Sense of Shared Knowledge. In G. Stahl (ed.): *Proceedings of CSCL 2002 – Conference for Computer Support for Collaborative Learning (Boulder, 2002)* 102-110

3. Baker, M.J., Hansen, T., Joiner, R., Traum, D.: The Role of Grounding in Collaborative Learning Tasks. In: Dillenbourg, P. (ed.): *Collaborative Learning: Cognitive and Computational Approaches*. Pergamon/Elsevier Science, Amsterdam, The Netherlands (1999) 31-63
4. Brennan, S.E.: The Grounding Problem in Conversations with and through Computers. In: Fussell, S.R., Kreuz, R.J. (eds.): *Social and Cognitive Approaches to Interpersonal Communication*. Lawrence Erlbaum, Mahwah, NJ (1998) 201-225
5. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science* 41, 6 (1999) 391-407
6. Dolonen, J., Chen, W., Mørch, A.: Integrating Software Agents with FLE. In: Wasson, B., Ludvigsen, S., Hoppe, U. (eds.): *Proceedings of Conference of Computer Supported Collaborative Learning (CSCL 2003)*. Kluwer Academic, Dordrecht, The Netherlands (2003) 157-161
7. Fischer, G., Lemke, A.C., Mastaglio, T., Mørch, A. I.: The Role of Critiquing in Cooperative Problem Solving. *ACM Transactions on Information Systems* 9, 3 (1991) 123-151
8. Hakkarainen, K., Lipponen, L., Järvelä, S.: Epistemology of Inquiry and Computer-Supported Collaborative Learning. In: T. Koschmann, Hall, R. and Miyake, N. (eds.): *CSCL 2: Carrying Forward the Conversation*. Lawrence Erlbaum, Mahwah, NJ (2002) 129-156
9. Landauer, T.K., Foltz, P.W., Laham, D.: An Introduction to Latent Semantic Analysis. *Discourse Processes* 25 (1998) 259-284
10. Leinonen, T.: Fle3 > Future Learning Environment. Website hosted by UIAH Media Lab, University of Art and Design Helsinki, Finland, <http://fle3.uiah.fi/> (retrieved May, 2nd, 2005).
11. Littlewood, W.: The Use of Collaborative Interaction Techniques in Teaching EAP. *HKBU Papers in Applied Language Studies*, Language Centre, Hong Kong Baptist University, Kowloon Tong, HK (2001)
12. Littlewood, W.: Students' Perception of Classroom Learning in East Asia and Europe. *HKBU Papers in Applied Language Studies*, Language Centre, Hong Kong Baptist University, Kowloon Tong, HK (2003)
13. Liu, J., Wong, K.C., Hui, K.K.: An Adaptive User Interface Based on Personalized Learning. *IEEE Intelligent Systems* 18, 2 (2003) 52-57
14. Ludvigsen, S., Mørch, A.: Categorization in Knowledge Building: Task-specific Argumentation in a Co-located CSCL Environment. In: Wasson, B., Ludvigsen, S., Hoppe, U. (eds.): *Proceedings of Conference of Computer Supported Collaborative Learning (CSCL 2003)*. Kluwer Academic, Dordrecht, The Netherlands (2003) 67-76
15. Maybin, J., Mercer, N., Stierer, B.: Scaffolding Learning in the Classroom. In: Norman, K. (ed.): *Thinking Voice*. Hodder & Stoughton, London, UK (1992) 186-195
16. McCall, R., Fischer, G., Mørch, A.: Supporting Reflection-in-Action in the Janus Design Environment. In: McCullough, M., Mitchell, W.J., Purcell, P. (eds.): *The Electronic Design Studio*, MIT Press, Cambridge, MA (1990) 247-259
17. Mørch, A., Omdahl, K., Ludvigsen, S.: Knowledge Building in Distributed Collaborative Learning: Organizing Information in Multiple Worlds. *Proceedings of the Human Factors in Computing Systems (CHI 2004) Workshop on Designing for Reflective Practitioners*, also Technical Report UCI-ISR-04-2, Institute for Software Research, University of California, Irvine, CA (2004) 31-33
18. Qiu, L., Riesbeck, C.K.: Making Critiquing Practical: Incremental Development of Educational Critiquing Systems. *Proceedings of the 2004 International Conference on Intelligent User Interfaces*. ACM Press, New York (2004) 304-306
19. Rice, M.L., Huston, A.C., Truglio, R.T., Wright, J.C. Words from Sesame Street: Learning Vocabulary while Viewing. *Developmental Psychology* 26, 3 (1990) 421-428

20. Robbins, J.E.: Design Critiquing Systems, Technical Report UCI-98-41, Department of Informatics, University of California, Irvine, CA (1998)
21. Robbins, J.E., Redmiles, D.F.: Software Architecture Critics in the Argo Design Environment. *Knowledge-Based Systems* 5, 1 (1998) 47-60
22. Scardamalia, M., Bereiter, C.: Computer Support for Knowledge-Building Communities. *The Journal of the Learning Sciences* 3, 3 (1994) 265-283
23. Schön, D.A.: *The Reflective Practitioner: How Professionals Think in Action*. Basic Books, New York, NY (1983)
24. Steinhart, D.J.: *Summary Street: An Intelligent Tutoring System for Improving Student Writing Through the use of Latent Semantic Analysis*. Ph.D. thesis, Dept. of Psychology, University of Colorado, Boulder, CO (2001)
25. Strang, G.: *Linear Algebra and Its Application*, 2nd edition. Academic Press, London, UK (1980)
26. Vygotsky, L.: *Mind in Society – The Development of Higher Psychological Processes*. Harvard University Press, Cambridge, MA (1978)
27. Weizman, Z.O., Snow, C.: Lexical Input as Related to Children's Vocabulary Acquisition: Effects of Sophisticated Exposure and Support for Meaning. *Developmental Psychology* 37, 2 (2001) 265-279
28. Yamamoto, Y., Takada, S., Gross, M., Nakakoji, K.: Representational Talkback – An Approach to Support Writing as Design. *Proceedings of the Asia Pacific Conference on Computer Human Interaction (Kanagawa, Japan)*, IEEE Computer Society (1998) 125-131