

Prepublication version of Cheung, W.K., Mørch, A.I., Wong, K.C., Lee, C., Liu, J., Lam, M.H. (2007). Grounding Collaborative Learning in Semantics-based Critiquing. *Int'l J. Distance Education Technologies*, 5(2), pp. 40-55.

Grounding Collaborative Learning in Semantics-Based Critiquing

William K. Cheung¹, Anders I. Mørch², Kelvin C. Wong¹, Cynthia Lee³, Jiming Liu¹, and Mason H. Lam¹

¹ Department of Computer Science, Hong Kong Baptist University, Hong Kong
{william, kckwong, jiming, mason}@comp.hkbu.edu.hk

² InterMedia, University of Oslo, Norway
anders.morch@intermedia.uio.no

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 makes 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 three experiments. We end the paper by describing our plans for future work.

Keywords: Semantic Matching; Web-based Learning; Critiquing Systems; Knowledge Building; Essay Writing.

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 (Rice et al 1990; Weizman & Snow 2001). As a consequence, students' reading and writing skills do not keep pace with listening, viewing 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 im-

balance 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 results from three experiments, including comparing LSA with manual teacher feedback on a set of essays.

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 (Yamamoto et al 1998). These are key aspects of any design process. More specifically, 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.

The two basic activities of design are action and reflection ((Schön 1983), supporting composition and abstraction, respectively. Action means to create an artifact by selecting building blocks and combining them into functional arrangements and reflection means to evaluate the artifact from multiple viewpoints (McCall et al 1990). When this occurs without external disruption other than situation-specific feedback, it is referred to as reflection-in-action (Schön 1983). 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 means of an incomplete design (e.g. missing parts), inconsistency in arrangement of parts, or a need for restructuring the overall activity.

Design Critiquing

Computational support for reflection-in-action is provided with the critiquing approach (Fischer et al 1991; Qiu & Riesbeck 2004; Robbins & Redmiles 1998). Critiquing is defined as “presentation of a reasoned opinion about a product or action” created by a user with a computer (Fischer et al 1991). A Critiquing System integrates computational support for design-as-action and design-as-reflection and operational-

izes Schön's notion of "back talk" with computational critics (Fischer et al 1991). 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 (McCall et al 1990), ArgoUML (Robbins & Redmiles 1998), and the Java Critiquer (Qiu & Riesbeck 2004). These systems were developed for the domains of kitchen design, UML (Unified Modeling Language) and Java programming, respectively. For example Janus allows designers to create 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 (Robbins & Redmiles 1998). Similarly, the Java Critiquer identifies statements in a program that can be improved by readability and best practice (Qiu & Riesbeck 2004). 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 (Yamamoto et al 1998); and second, critic feedback can supplement teacher feedback on student essays in certain situations (after school hours, in distributed environments, distance education). In this context we propose to integrate knowledge building (a distributed collaborative learning activity) and LSA with critiquing in the following ways: 1) LSA to compute the critic feedback and 2) knowledge building to support joint reflection. This is different from past work on critiquing systems and educational applications of LSA. The previous work on LSA has focused almost exclusively on individual learning by integrating it with Intelligent Tutoring Systems (Steinhart 2001). A goal for us is to provide computer support for both action and reflection, and individual and collaborative learning.

Knowledge Building

Knowledge building (Scardamalia & Bereiter 1994) 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 from a research-like process by generating or responding to shared problems or questions, proposing tentative answers (personal explanations) and searching for deepening knowledge collaboratively.

Knowledge building and its subsequent refinement Progressive Inquiry (Hakkarainen et al 2002) 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 learn-

ing environments. Knowledge building environments are pedagogically designed discussion forums and include CSILE (Scardamalia & Bereiter 1994), Knowledge Forum, and Future Learning Environment (FLE) (Leinonen 2005). They are used in schools in Canada, Hong Kong and Scandinavia, as well as elsewhere in the world.

The rationale for our wish to integrate 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' reflection needs, inspired by Schön's notion of reflective practice (Schön 1983). This is necessary but not sufficient in order to support the needs of a networked design community. Knowledge building, on the other hand, can add a multi-user dimension by supporting joint reflection, even though knowledge building was not originally conceived as such. Joint reflection occurs during "talk with peers" (e.g. Maybin et al 1992) in shared tasks and meaningful contexts, i.e. collaboratively addressing problems or questions shared by a community of learners in which shared understanding can emerge (Arnseth & Solheim 2002). 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 (Ludvigsen & Mørch 2003; Mørch et al 2004). 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, revolving around the respective schools' local situations, thus grounding the interaction.

Grounding is the process of making sure one's utterances are understood in communication with others, and the basis on which one builds further understanding. The concept of *common ground* has its roots in linguistics and arises from a model of conversation developed by Clark (Clark & Brennan 1991). It is suggested that that collaborating partners continually add and update information to the common ground and gradually improve understanding as the conversation proceeds. Similar notions like mutual belief, inter-subjectivity and shared knowledge have been applied to collaborative learning and problem solving (e.g. Arnseth & Solheim 2002; Baker et al 1999; Brennan 1998). However, grounding is not exclusively tied to communication and social interaction, and we propose four types of grounding that can impact the success and failure of computer-supported learning environments:

1. Communication and social interaction (linguistic grounding)
2. Practice situation (work-oriented grounding)
3. Artifacts and work-arounds (tacit grounding)
4. Knowledge base (semantic grounding)

Each of the four types of grounding may need attention when building computer support for collaborative learning and problem solving. Work-oriented grounding is the grounding that occurs at a workplace, e.g. when one is using tools and materials of a specific profession to create artifacts required for the business. Schön (1983) characterized this form of grounding as "reflective conversation with the materials of a situation." It connects professionals with the physical (material) world of their profession. On the other hand, tacit grounding is the form of grounding one resorts to when

selecting (sometimes without being aware of it) artifacts from the immediate environment to support an utterance (e.g. pointing to a watch as an excuse to leave a meeting early) or work-arounds (e.g. automatically selecting “back-up” technology when the primary technology fails). The latter is relevant when interacting with advanced learning environments. Finally, semantic grounding is the grounding that makes use of already established knowledge, i.e. the rules, facts and arguments defining a domain of interest.

The complexity the “grounding problem” (Brennan 1998) is in part related to the interdependencies among the four types of grounding just mentioned. It is outside the scope of this article to address them in detail. There readers are encouraged to consult the referenced sources. In the current project we focus on semantic grounding by integrating a knowledge-building environment with an LSA-based critiquing system.

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 pre-processing by the computer in various ways.

LSA computes first the semantic relationship between words using word co-occurrence statistics and then the similarity of two input texts (student and teacher) accordingly as follows. First, both input texts are segmented to form part of the corpus. Normally, the corpus should also be supplemented by additional related documents sourced from the Internet or student model essays. Then, the word-segment association matrix D is constructed. In the matrix D , each row typically stands for a unique word and each column stands for a text segment. Note that it is common to call each column the *feature vector* corresponding to a particular text segment. For the simplest case, each cell entry can be the frequency of a given word in a given text segment. As an example, consider the segment “International Conference on Web-based Learning focuses on research works that enhance teaching and learning experience”. If the j^{th} column corresponds to the aforementioned segment and the i^{th} row corresponds to the word “learning”, then the value in D_{ij} would be 2 as the word “learning” occurs two times in the segment. As weighting words based on their individual importance is known to be also effective in obtaining better matching results, we used entropy values instead for computing D_{ij} , given as

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

where N is the number of text segments in the stored corpus and f_{ij} is the frequency of the i^{th} word in the j^{th} text segment.

Once the matrix D is computed, it will be first *decomposed* using Singular Value Decomposition (SVD) (Strang 1980), and then *trimmed* for some of the unimportant semantic dimensions (to be explained in the following paragraph), and finally *reconstructed* to a matrix with its dimension same as the original one. In particular, using

SVD, the matrix D can be 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 along its diagonal or otherwise zero. By comparing the diagonal elements of λ , we only keep those elements with large values and set the others to zero, with the effect that the dimension of λ is reduced. This is equivalent to removing the corresponding columns from P and rows from Q . The resulting “semantic space” is commonly considered the space that is spanned by the orthonormal columns of the matrix Q .

After the semantic space has been computed, the new D can be “reconstructed” from 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

$$\cos \theta = \frac{\langle x, y \rangle}{\|x\| * \|y\|}$$

where $\langle x, y \rangle$ is the inner product of vectors x and y , defined as $\langle x, y \rangle = x_1 y_1 + x_2 y_2 + \dots + x_n y_n$, and $\|x\|$ is the length of a vector x defined as $\|x\| = (\langle x, x \rangle)^{1/2} = (\sum_{i=1..n} x_i^2)^{1/2}$.

Using this similarity metric, words that have appeared in similar segments, and segments with similar semantic content, will be considered to be near one another (Steinhart 2001). Words that do not co-occur (e.g. bicycle and bike), but occur in similar contexts will also be grouped together.

Remark. Direct application of the above steps to our case is a costly computational process because it requires invocation of SVD each time the similarity value is computed. To alleviate the limitation, we, as suggested by Deerwester et al (1990), projected the text segments extracted from the latest submitted essay to the semantic space characterized by the orthonormal columns of the matrix Q computed based on the corpus *only*. The projection of the latest submission can then be accomplished using the corresponding $\lambda^{-1}P^T$ as the transformation matrix. Note that the projected vectors are sometimes called *pseudo* text segments. Similarly, the text segments extracted from the corpus can all be projected to the same semantic space so that they can be compared directly with the pseudo text segments of the latest submission. The major advantage of this is that we then only need to compute the SVD once (as far as the corpus is not changed) instead of per submission. Additional technical details on LSA can be found in Landauer et al (1998).

COMPONENTS OF A LEARNING ENVIRONMENT OF 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 es-

says, 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. 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.

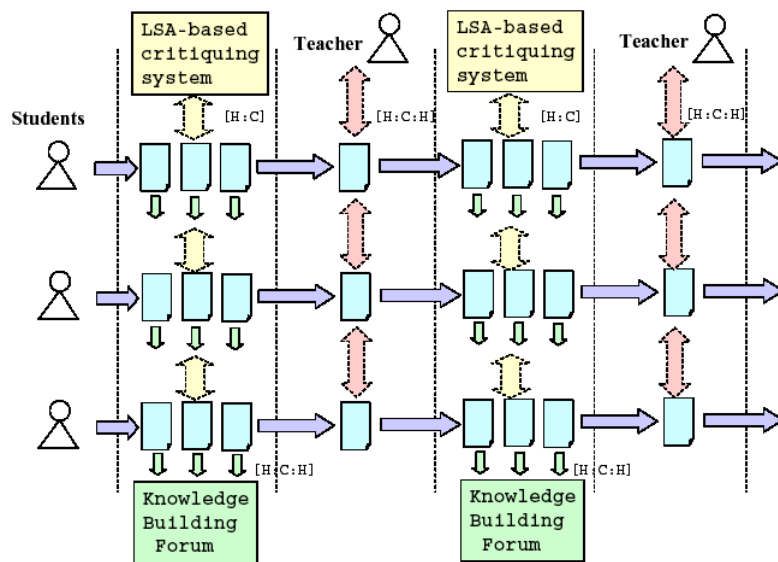


Fig. 1. English composition integrated learning environment system architecture.

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 the missing items in their text (compared with the corpus samples). 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 (Baeza-Yates et al 1999). The Analyzer then computes the word-segment association matrix. Singular Value Decomposition (Strang 1980) 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, as we described in the previous section.

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 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 insights into how individual students incorporate the missing themes, and participate as facilitators of student collaboration sessions to provide feedback when the students run out of ideas. 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. Maybin et al 1992) and pushes the learner's zone of proximal development (Vygotsky 1978). 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 (Littlewood 2003).

The LSA-based critiquing and knowledge building environment marks the contours of a "double-loop" learning process (see Figure 1). It alternates between inner (human-computer interaction) and outer (computer-supported 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 a well understood automated 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. In the remaining of this article we present our system development efforts (two prototypes) and evaluation results (three experiments).

System Prototypes

In order to support English essay writing as a design activity based on the models and techniques presented above, we had decided to reuse and integrate existing and freely available systems, making modifications if necessary. When selecting the critiquing component we considered both ArgoUML (Robbins & Redmiles 1998) and the Java Critiquer (Qiu & Riesbeck). The latter has the advantage of supporting the design of a textual artifact (program code), but ArgoUML has the advantage of being an open source system. We finally decided on ArgoUML due to its accessibility. However, we had to modify the system extensively (see Figure 2). In particular, we removed all the

features we did not need and adding the features that are unique to our domain. So, 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 perceives each essay as a sequence of sentences. We hid some of the Argo features such as the building block palette and the to-do list, which we anticipate to be useful in the future versions of our system. For instance, the building block palette could be useful for representing and manipulating more intermediate-level building blocks like paragraphs, sections and other higher level abstractions, which has shown to be useful for writing (Yamamoto et al 1998; Akin 1978) and can allow students to acquire skills in not only composition but also organization. It may simplify LSA preprocessing by reducing the need for sentence segmentation. Also, the to-do lists that can keep track of overlooked critic messages and suggest when they should be attended to can help students manage multiple missing sub-themes.

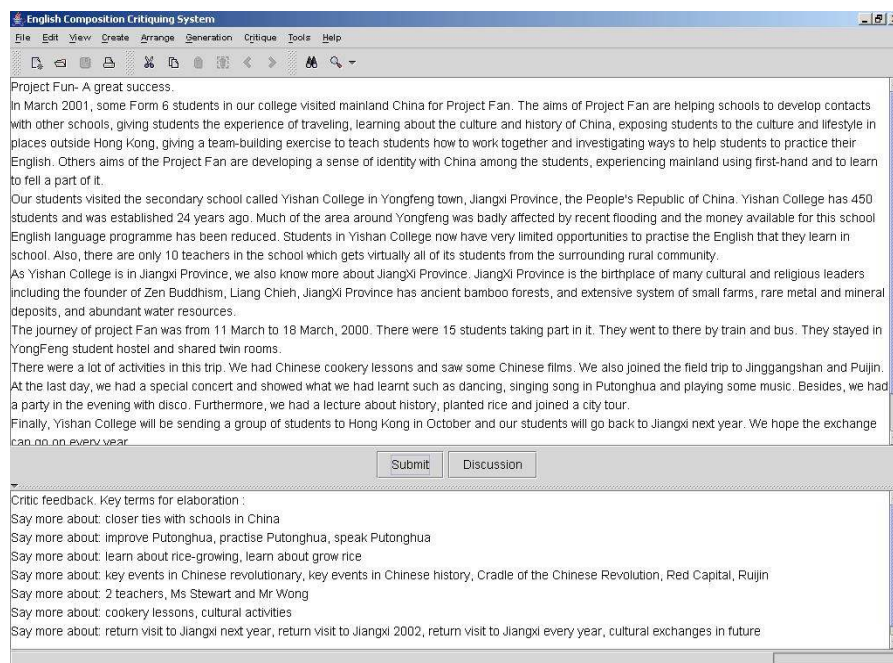


Fig. 2. The English composition critiquing system has a “Submit” button to generate LSA-based critique and a “Discussion” button to trigger a knowledge building session (Figure 3).

For the knowledge-building component, we decided on another open source system, FLE (Future Learning Environment) (Leinonen 2005). FLE is a knowledge building environment developed in accordance with the progressive inquiry model (Hakkarainen et al 2002). 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.

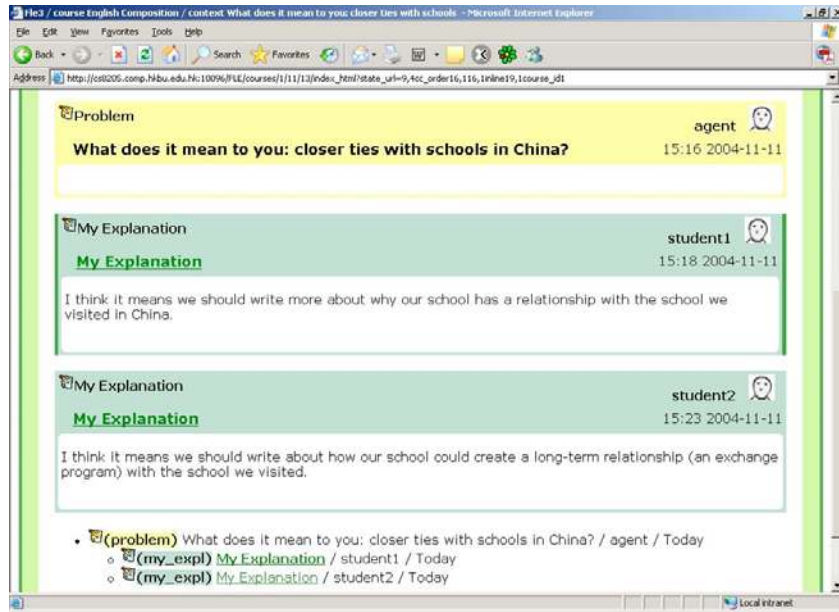


Fig. 3. The Fle3 Web-based knowledge-building forum shows a persistent storage of discussions related to students’ missing essay topics. An agent initiates a thread when it discovers an essay topic that is missing by more than one student. The KB forum has not yet been fully integrated into our environment and it has not yet been tested with students.

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 the two students receiving the same feedback have something in common so that they can resolve by information sharing and discussion. The reason why a discussion forum 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 the forum is built on our previous (empirical-based, system building) work on integrating agents with FLE (Dolonen et al 2003) and adaptive user interface agents (Liu et al 2003).

EVALUATION AND PRELIMINARY RESULTS

In order to assess the feasibility of our critiquing system regarding its ability to suggest missing/uncovered sub-themes, we conducted three experiments. It is through these experiments that we investigated the performance of LSA, studied the factors that can improve LSA performance and compared its performance with the conventional keyword matching technique.

Experiment 1: Performance of LSA

In this experiment, our objective was to investigate the feasibility of using LSA to suggest missing sub-themes. Seven high school students in Hong Kong were invited to write a 400 to 500-word essay on the topic “Write an essay about the experience of traveling to China”. At the same time, a teacher was asked to provide a number of sub-themes (25 in this study) of this topic, which the students were expected to include in their essays.

The teacher assessed the finished essays to identify the sub-themes that were missing, based on the set of predefined sub-themes. Then the essays were assessed by our system. Each text segment in the student essay was compared with each sample segment suggested by the teacher. If the semantic similarity (which was represented by the cosine value calculated by LSA) was below a preset threshold, we considered the sub-theme of the sample segment to be missing in the student essay. Finally, the missing sub-themes identified by the teacher and our system were compared to evaluate the performance of the system. The system identified 35 missing sub-themes in the 7 student essays, 22 of them were judged to be correct (i.e., also identified by the teacher as missing sub-themes), whereas the remaining 13 were considered inappropriate. On the basis of this, we get a tentative precision rate of 63%.

A reason for this relatively low number is the small size of the corpus. We used a corpus of about 3,000 words to build the semantic space. This is a smaller corpus than what has been used in related studies, such as TASA-all (a large knowledge space consisting of text samples from the K12 (grade 1-12) curriculum in the United States) (Steinhart 2001). 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. Therefore, another experiment is conducted.

Experiment 2: Enhancements to LSA

In this experiment, our objective was to improve the performance of LSA. We proposed some enhancements to the generic LSA.

In addition to the three pre-processing steps: 1) removal of stop words, 2) change of plural nouns to singular ones and 3) change of verbs in different tenses to their present tense form (using WordNet) as performed in Experiment 1, the system was

further enhanced by converting the adverbs to their equivalent words in the adjective form to further unify the words with the same meaning. In addition, the top 20 words with the highest entropy value were removed as these words were more evenly distributed in the sub-theme paragraphs and corpus, which could have a negative effect on the system's ability to discriminate between texts. It was believed that these words did not provide any value-added semantic information for identifying sub-themes. For example, in the context of "Mobile Phone Impact", the words "mobile" and "phone" do not add new semantic information for discriminating among the different sub-themes associated with mobile phones. Furthermore, after the removal of the aforementioned top 20 words, the remaining top 25 words in all the sub-theme vectors after performing SVD were further checked. If these words appeared in more than or equal to half number of the sub-themes, they were removed. The objective was to make the sub-themes more distinguishable.

Apart from enhancing the preprocessing steps, we also attempted to further enrich the text segment representation. The enhancement was based on the observation that the semantic context of a sentence, sometimes, can only be captured by considering its neighboring sentences. To incorporate this enhancement, we modified each column of the LSA matrix by adding to it a weighted version of the sum of the columns corresponding to its neighbors. One can consider this idea to be similar to the moving window concept commonly used in image processing¹.

In this experiment, twelve students were invited to write a composition on the topic of "Mobile Phone Impact". A teacher was asked to identify the covered sub-themes in the composition and a total of 26 sub-themes were marked. The performance of generic LSA and the enhanced LSA were compared. The threshold on the cosine value was set to be 0.15 for determining whether a sub-theme was found. Without the enhancements, our system identified 23 sub-themes, out of which 11 were members of the "Mobile Phone Impact" dataset, giving an overall precision of 0.32 and a recall of 0.42. With the proposed enhancements, 20 out of the 27 sub-themes identified by our system were judged to be correct. This gives a precision of 0.74 and a recall of 0.77. We consider the improvement to be significant.

Experiment 3: Performance Comparison with Simple Keyword Matching

The objective of Experiment 3 was to verify the performance gain brought by LSA using simple keyword matching as the baseline. Keyword matching was implemented using the "Mobile Phone Impact" dataset as adopted in Experiment 2. We completed all the steps described in the previous section, excluding all the LSA related steps (i.e. the use of corpus, SVD and sub-theme keyword removal). Not surprisingly it was found that keyword matching was inferior to LSA. In particular, the keyword matching method identified 34 sub-themes, out of which only 11 were found to be correct, giving a precision of 0.32 and a recall of 0.42. Figure 4 shows the recall-precision curves of the enhanced LSA (the dotted curve) and the keyword matching (the solid curve). The corresponding break-even points were 0.71 and 0.32 respectively. This result indicates that the enhanced LSA is a significant factor contributing to the accuracy of the sub-theme identification process.

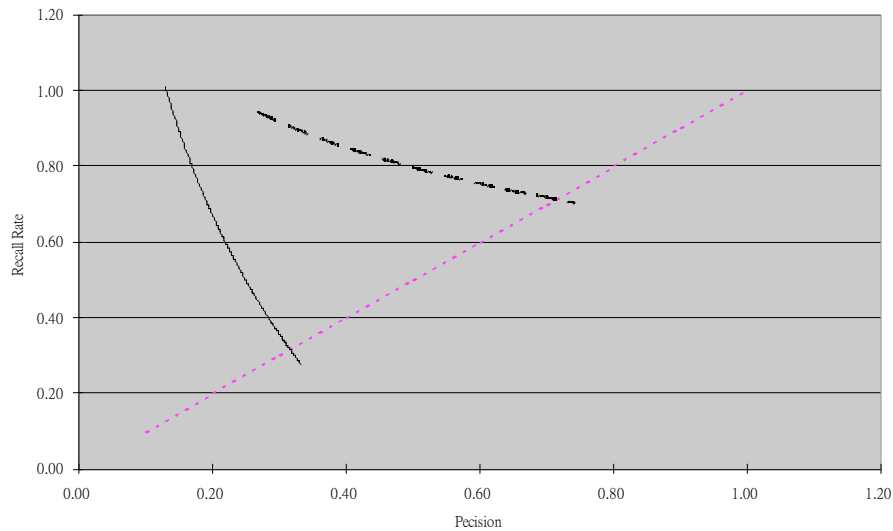


Fig. 4. Performance comparison between the enhanced LSA and keyword matching applied to identifying sub-themes in student essays.

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 the missing sub-themes because it requires considerable amount of teachers' time.

We believe that 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 run 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 knowledge building according to the progressive inquiry pedagogical model. We believe that this combination of theory foundation and computer support for individual and collaborative learning can help students enrich their essay content with a richer vocabulary in contexts that are meaningful to them. We are also interested in ascertaining the way in which students view the critiquing system and the extent to which the knowledge-building forum will be used. On the technical (algorithmic) side, it is worth investigating the factors that will affect the performance of LSA in the essay-writing domain. Knowing how to determine both the optimal number of dimensions of the semantic

space and the optimal threshold value for similarity matching are important and these questions 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 second author was as a Visiting Research Scholar at Department of Computer Science, Hong Kong Baptist University while this paper was written, and he thanks InterMedia and Faculty of Education, University of Oslo for a sabbatical leave.

NOTES

¹In our experiment, the weighting factor for the neighboring sentence is 0.2 where the neighbor of a sentence is defined as the sentences following it up to the end of the current paragraph.

REFERENCES

- Akin, O. (1978). How Do Architects Design? Latombe, J. (ed.): *Artificial Intelligence and Pattern Recognition in Computer Aided Design*. North-Holland, New York, NY, 65-104
- Arnseth, H. C., Solheim, I. (2002). Making Sense of Shared Knowledge. G. Stahl (ed.): *Proceedings of CSCL 2002 – Conference for Computer Supported Collaborative Learning (Boulder, 2002)*, 102-110
- Baeza-Yates, R., Ribeiro-Neto, B., (1999). *Modern Information Retrieval*, Addison Wesley.
- Baker, M.J., Hansen, T., Joiner, R., Traum, D (1999). The Role of Grounding in Collaborative Learning Tasks. Dillenbourg, P. (ed.): *Collaborative Learning: Cognitive and Computational Approaches*. Pergamon/Elsevier, Amsterdam, The Netherlands, 31-63
- Brennan, S.E. (1998). The Grounding Problem in Conversations with and through Computers. Fussell, S.R., Kreuz, R.J. (eds.): *Social and Cognitive Approaches to Interpersonal Communication*. Lawrence Erlbaum, Mahwah, NJ, 201-225
- Clark, H., Brennan, S. (1991). Grounding in Communication. Resnick, L.B., Levine, M, Teasley (eds.): *Perspectives on Socially Shared Cognition*. American Psychological Assoc., Washington, DC, 127-149
- Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R. (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science* 41, 6, 391-407

Dolonen, J., Chen, W., Mørch, A. (2003). Integrating Software Agents with FLE. Wasson, B., Ludvigsen, S., Hoppe, U. (eds.): *Proceedings of Conference of Computer Supported Collaborative Learning (CSCL 2003)*. Kluwer Academic, Dordrecht, The Netherlands, 157-161

Fischer, G., Lemke, A.C., Mastaglio, T., Mørch, A. I. (1991). The Role of Critiquing in Cooperative Problem Solving. *ACM Transactions on Information Systems* 9, 3, 123-151

Hakkarainen, K., Lipponen, L., Järvelä, S. (2002). Epistemology of Inquiry and Computer-Supported Collaborative Learning. T. Koschmann, Hall, R. and Miyake, N. (eds.): *CSCL 2: Carrying Forward the Conversation*. Lawrence Erlbaum, Mahwah, NJ, 129-156

Landauer, T.K., Foltz, P.W., Laham, D. (1998). An Introduction to Latent Semantic Analysis. *Discourse Processes* 25, 259-284

Leinonen, T. (2005). 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)

Littlewood, W. (2003). 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*

Liu, J., Wong, K.C., Hui, K.K. (2003). An Adaptive User Interface Based on Personalized Learning. *IEEE Intelligent Systems* 18, 2, 52-57

Ludvigsen, S., Mørch, A. (2003). Categorization in Knowledge Building: Task-specific Argumentation in a Co-located CSCL Environment. Wasson, B., Ludvigsen, S., Hoppe, U. (eds.): *Proceedings of Conference of Computer Supported Collaborative Learning (CSCL 2003)*. Kluwer Academic, Dordrecht, The Netherlands, 67-76

Maybin, J., Mercer, N., Stierer, B. (1992). Scaffolding Learning in the Classroom. Norman, K. (ed.): *Thinking Voice*. Hodder & Stoughton, London, UK, 186-195

McCall, R., Fischer, G., Mørch, A. (1990). Supporting Reflection-in-Action in the Janus Design Environment. McCullough, M., Mitchell, W.J., Purcell, P. (eds.): *The Electronic Design Studio*, MIT Press, Cambridge, MA, 247-259

Mørch, A., Omdahl, K., Ludvigsen, S. (2004). 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, 31-33

Qiu, L., Riesbeck, C.K. (2004). Making Critiquing Practical: Incremental Development of Educational Critiquing Systems. *Proceedings of the 2004 International Conference on Intelligent User Interfaces*. ACM Press, New York, 304-306

Rice, M.L., Huston, A.C., Truglio, R.T., Wright, J.C. (1990). Words from Sesame Street: Learning Vocabulary while Viewing. *Developmental Psych* 26, 3, 421-428

Robbins, J.E., Redmiles, D.F. (1998). Software Architecture Critics in the Argo Design Environment. *Knowledge-Based Systems* 5, 1, 47-60

Scardamalia, M., Bereiter, C. (1994). Computer Support for Knowledge-Building Communities. *The Journal of the Learning Sciences* 3, 3, 265-283

Schön, D.A. (1983). The Reflective Practitioner: How Professionals Think in Action. *Basic Books*, New York, NY

Steinhart, D.J. (2001). 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*

Strang, G. (1980). *Linear Algebra and Its Application*, 2nd edition. Academic Press, London, UK

Vygotsky, L. (1978). Mind in Society – *The Development of Higher Psychological Processes*. Harvard University Press, Cambridge, MA

Weizman, Z.O., Snow, C. (2001). Lexical Input as Related to Children's Vocabulary Acquisition: Effects of Sophisticated Exposure and Support for Meaning. *Developmental Psychology* 37, 2, 265-279

Yamamoto, Y., Takada, S., Gross, M., Nakakoji, K. (1998). Representational Talkback – An Approach to Support Writing as Design. *Proceedings of the Asia Pacific Conference on Computer Human Interaction (Kanagawa, Japan)*, IEEE Computer Society, 125-131