Bridging the Gap to Natural Language: A Review on Intelligent Tutoring Systems based on Latent Semantic Analysis

Wolfgang Lenhard, Department of Psychology, University of Würzburg, Röntgenring 10, 97070 Würzburg, Germany
wolfgang.lenhard@mail.uni-wuerzburg.de

Abstract. One of the major drawbacks in the implementation of intelligent tutoring systems is the limited capacity to process natural language and to automatically deal with unexpected or unknown vocabulary. Latent Semantic Analysis (LSA) is a statistical technique of automatic language processing, which can attenuate the “language barrier” between humans and tutoring systems. LSA-based intelligent tutoring systems address the goals of modelling human tutoring dialogues (AutoTutor), enhancing text comprehension and summarisation skills (State-The-Essence, Summary Street®, conText, Apex), training of comprehension strategies (iStart, a French system in development) and improving story and essay writing (Write To Learn, Select-a-Kibitzer, StoryStation). The systems are reviewed concerning their efficacy in modelling skilled human tutors and regarding their effects on the learner.

Keywords. intelligent tutoring systems, latent semantic analysis, writing-skills

INTELLIGENT TUTORING SYSTEMS

One-on-one tutoring is commonly considered to be the gold standard in education (Koschmann, 1996). Human tutors are able to guide the learning progress very efficiently and help students to considerably increase their knowledge and enhance their abilities (Bloom, 1984). In regular classrooms, of course it is extremely difficult to accomplish a situation, where students get the same amount of feedback and practice compared to direct instruction. In these situations, a computer program providing even half as much knowledge as a one-on-one human tutor would already mark a great success. In order to achieve this, a program would have to “engage the student in sustained reasoning activities and to interact with students based on a deep understanding of the student’s behaviour” (Corbett, Koedinger, & Anderson, 1997, p. 849). This is the main goal of intelligent tutoring systems (ITS).

These systems try to mimic human tutors by providing a *problem-solving environment*, a *domain expert module*, a *student model* and a *pedagogical module*. The problem solving environment, at minimum, consists of a plain text editor. It is the place where the student acts works on his or her task. The domain expert module contains the knowledge that the student should acquire. In classical ITS, it is an expert system that incorporates the behavioural repertoire to solve the tasks in the same way as students do it. The student model represents the student’s knowledge state, typically by including an overlay of the domain knowledge and a catalogue of bugs or knowledge gaps. Finally, the pedagogical module guides the learning process by incorporating a curriculum, structuring the knowledge transfer, using instructional strategies, as well as giving support and feedback.

Whiel ITS has many advantages, such as giving individualized feedback and enabling human tutors to focus on single students with more complex needs, the system is indeed far from perfect. One of the major drawbacks in the implementation of ITS, however, is the limited capacity to process natural language and to automatically deal with unexpected or unknown vocabulary (Puppe, 1992). While some authors of ITS make a virtue of necessity and underline the use of precise vocabulary as a key element for the intended learning progress, this circumstance poses a tight restriction to the knowledge transfer, because the learner not only has to grasp the meaning of the new information being given to him, but also has to use a restricted communication code. The narrow set of communication operations has been the central problem of ITS design (Koschmann, 1996).

LATENT SEMANTIC ANALYSIS

In order to model a real world tutoring process, as for example the dialogue between the teacher and a student, an ITS would have to possess a sufficient degree of verbal intelligence and semantic knowledge. Latent Semantic Analysis (LSA) is a statistical technique that meets these criteria insofar as it successfully represents those facets of word and text meanings that are reflected in their usage in written language. LSA is a statistical technique from the field of natural language processing (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990). It permits the extraction of the relations between words based on their common occurrences in texts. Its procedure is completely statistical in nature. Thus, the word meanings, reflected by the co-occurrences, are extracted completely without any specification of rules or dictionaries. This section gives a brief theoretical overview over LSA. With the "Handbook of Latent Semantic Analysis" (Landauer, McNamara, Dennis & W. Kintsch, 2007) an extensive and comprehensible textbook is at hand that covers both mathematical foundations and areas of application of LSA. It is a valuable source for all readers, who are interested in more specific information.

LSA is based on text corpora with each single text usually being split into documents, as for example paragraphs (Wiemer-Hastings, 1999). The information stored in the corpora can formally be represented by building a frequency matrix. The columns of this matrix contain the documents (n) and the rows contain the different words (m). Initially, each cell (a_{ij}) holds the frequency of a specific word in a specific document. Depending on the size of the corpora, the included text material, and the language, most cells contain zeros. Compared to the English language, this is especially true for languages with a high number of inflected words and rich compounds. This is the case with German and French, at least when no lemmatization (reduction of inflected words to their dictionary form) is applied to the raw text material. A medium sized corpus in the German language (e. g. 5 million words in 50 000 documents) usually results in a sparse matrix with a density below .05% (more than 99.95% of the cells hold a zero value).

This huge frequency matrix already comprises all the information that is subsequently utilised for text processing. Nevertheless, it is too large to reasonably apply it to similarity judgements because it contains a great deal of irrelevant information, called "noise". To reduce the noise, several steps are necessary. First, words with a very low frequency and stop words are usually filtered out. In the next step a weighting algorithm is applied to the matrix (Nakov, Popova, & Mateev, 2001) in order to emphasize words with a specific meaning. To every non-zero a_{ij}, both a local weighting function to increase or decrease importance of words with a text document and a global weighting function across the entire document collection is applied (Martin & Berry, 2007). As a consequence, a_{ij} = local(i, j)*global(i), where local(i, j) is the local function applied to the cell that holds the frequency of word i in text document j, and global(i) is the global function applied to word i. A very common weighting function is the so called log-entropy, which Dumais (1991) found to yield the best results.

Finally, the matrix is decomposed via Singular Value Decomposition (SVD) similar to the procession in a Principal Component Analysis (PCA). Contrary to the eigenvalue decomposition in PCA, where a decomposition of the square matrix of covariances is done, the SVD used in LSA is the decomposition of a rectangular matrix of weighted term frequencies (see. figure 1, mathematical description see Berry, Dumais, & O'Brien, 1995; Martin, & Berry, 2007). In fact any rectangular matrix, such as the n x m-matrix with the weighted word frequencies, can be decomposed into three orthogonal partial matrices, so that $X = T_o \times S_o \times D_o'$, where T_o and D_o' have orthonormal, and S_o has diagonal columns. T_o represents the term matrix (comparable to the factor values in PCA), D_o' the document matrix (comparable to the factor loadings in PCA) and S_o the matrix with the singular values (comparable to eigenvalues in PCA).

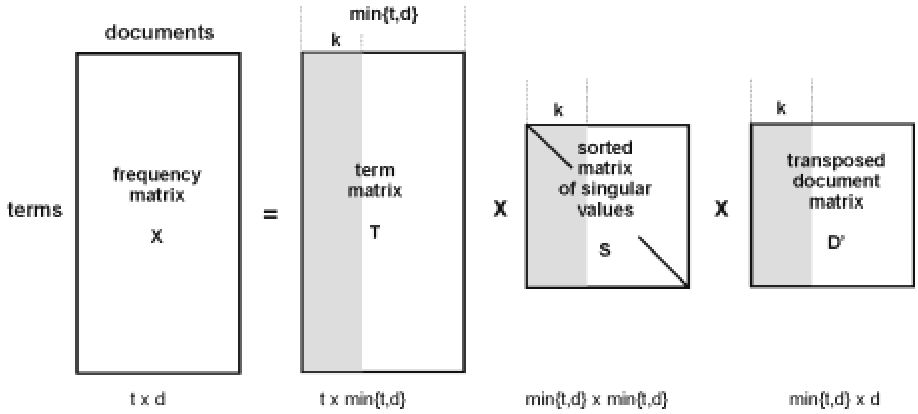

Fig 1. A Singular Value Decomposition results in three partial matrices. The original matrix is obtained again by multiplying the cells of the partial matrices. In the context of LSA, the matrices are reduced to approximately 100 to 500 dimensions, thus losing the biggest part of irrelevant information and eliminating noise from the original frequency matrix.

By ordering the columns of all three matrices according to the size of the singular values of S_o, all but the first (and thus most important) k dimensions can be deleted in order to get the least-squares best fit approximation to X with k dimensions. By the reduction of the number of extracted dimensions to a minimum, noise is excluded and the amount of data and memory consumption is downsized. What is even more important, the dimension reduction leads to the generalisation of word and text meanings and the compression of knowledge. The retrieved vectorial representation of the semantic content is not simply some kind of word occurrence statistic any more, but rather an abstraction of the *latent semantic content* reflected by the common usage of words (thus *Latent* Semantic Analysis).

Contrary to PCA, there are no criteria as to how many dimensions should be extracted. Numbers of 300 to 500 dimensions turned out to work best (Dumais, 1991; Graesser et al., 1999; Nakov, 2000; Wild, Stahl, Stermsek, & Neumann, 2005). In order to abridge the extremely resource intense process of a complete SVD with subsequent dimension reduction, the Lanczos algorithm (Lanczos, 1950) is commonly used. It is an iterative method, which proved to be accurate and efficient for the extraction of a limited number of dimensions from large and sparse matrices (Martin & Berry, 2007).

Eventually, the results of the SVD establish an n-dimensional orthogonal space, or "semantic space", where the terms and documents are distributed according to their common usage. Thus, the vector of a term partly represents its semantic content. The position within the semantic space reflects the relation of the meaning to other words or documents. As a result, words occur near to each other in the semantic space if they are often used in the same contexts, regardless of whether they are actually used in the same documents or not (higher order cooccurrences; Lemaire, & Denhière, 2004; Kontostathis, & Pottenger, 2002). Thus, LSA is not simply the computation of co-occurrence statistics. It is not a keyword matching algorithm, but "rather reflects … semantic relatedness, regardless of the actual words used" (Franzke, Kintsch, Caccamise, Johnson, & Dooley, 2005).

Even using the Lanczos algorithm, the SVD is a relatively resource intense computation process. Once it is finished, it enables high-performance similarity judgements between words or texts. There are several possible similarity measures like the Euclidian distance or the cosine of the angle between two vectors. The cosine turned out to be a robust similarity measure (Landauer, Laham, Rehder, & Schreiner, 1997; Rehder, Schreiner, Wolfe, Laham, Landauer, & Kintsch, 1998). Moreover, it yields a simple interpretation because it can be used just like a linear correlation.

New text material can be compared by projecting it into the semantic space, a process called "folding in". Simply speaking, the words of the new text are filtered and weighted in the same way as the original text corpus. Subsequently the vectors of the words are summed up to a new vector whose length and direction represent the meaning of the new text.

In contrast to other methods of automated text analysis, LSA is able to categorize semantically related texts as similar, even when they do not share a single word. For example, the two sentences "A penguin is a bird that lives on fish and krill" and "Penguins are birds, which eat crabs and fishes" have a cosine of .763, despite the fact, that they do only share the word "and" (computation done with word material in German language, plural forms treated as distinct words). The first sentence has only a cosine of .563 with the sentence "A whale is a marine mammal that lives on fish and krill" although there is a large word overlap between the two sentences (demonstration available under Lenhard, Baier, Schneider, & Hoffmann, 2006)[1].

Despite its elegance there are serious limitations to LSA. One of the main points is the lack of syntax, word order and negation. For an LSA-system "heaven" and "hell" are more or less the same, because the two concepts are highly interconnected to each other (Steinhart, 2001, p. 46). As a result, LSA-systems in general do not work well on topics and tasks that rely highly on argumentation structure and logic (Landauer, Foltz & Laham, 1998). As a consequence, the usage of expository text when computing semantic spaces as well as comparing texts usually yields superior results, because these texts do less rely on stylistic devices. Moreover LSA usually performs better on texts containing

---

[1] The space underlying the example consists of texts from the domains of biology, geography and geology from school books, encyclopaedias and internet pages. The texts were extracted and split into paragraphs automatically. I converted all words in the texts to lower case and filtered stop words, words occurring less than three times as well as texts consisting of less than ten different words. The frequency matrix included 37,773 paragraphs with 83,369 different words (total size of corpus 2,178,432 words). Prior to the SVD, a log-entropy weighting was applied to the frequency matrix. I extracted 400 dimensions (duration of computation: 35min. 17 sec.). Computations in English can be done via http://lsa.colorado.edu/. The attempt to replicate the described results in the English language with the "General reading up to 1st year of college" space failed, however.

multiple sentences, as compared to short answers (e.g., only single sentences; Landauer, Foltz, & Laham, 1998). Another major drawback is the dependency of LSA's performance on predefined settings such as the filter settings, the weighting formula, the number of extracted dimensions and the type, quality and length of texts used. As these parameters are interdependent, there is no optimal value for each factor, but only favourable combinations of parameter settings that are hard to find out. Landauer and Foltz (2007) suggest using corpora with at least 20 000 text units and without filtering or lemmatization. Notably, however, filtering out words which occur only once greatly reduces the resources needed, which is especially important in languages with a high number of inflections and rich compounds.

LSA is only a statistical technique and does not yield real verbal intelligence in the sense of causal understanding of real world phenomena and the productive usage of language. It more or less successfully represents word and text meanings in an abstract form and thus approximates the way in which the human mind reprocess and stores input (Landauer & Dumais, 1997). Despite this fact, LSA nonetheless exhibits an astonishing degree of expertise on tasks that afford verbal intelligence and semantic knowledge, as for example multiple choice knowledge tests (Landauer, & Dumais, 1997; Lenhard, Baier, Hoffmann, & Schneider, 2007), automatic essay grading (Landauer, Laham, Rehder, & Schreiner, 1997; Lenhard, Baier, Hoffmann, & Schneider, 2007), the measurement of textual coherence and prediction of readers' comprehension (Foltz, Kintsch, & Landauer, 1998), the prediction of knowledge gains of readers on the basis of their background knowledge (Wolfe et al., 1998) and last but not least intelligent tutoring systems (e. g. Wade-Stein, & E. Kintsch, 2004). As a result, LSA can be used as an approximation to human semantic knowledge and verbal intelligence in well defined tasks and help to bridge the gap to natural language in ITS. A number of at least partly LSA-based ITS have been developed or are currently under development. The following section gives an overview of these systems. Moreover, their effectiveness is discussed as far as data are available.

INTELLIGENT LSA-BASED TUTORING SYSTEMS

Natural Language Conversation Tutors

AutoTutor (2006) is probably the most complex ITS described in this paper. It mimics the discourse patterns and pedagogical strategies of human tutors (Graesser et al., 1999) and combines a variety of natural language processing techniques, such as syntactic parsers and lexicons, with latent semantic analysis, in order to extract and evaluate the input of the learner (Graesser et al., 2004). It incorporates a curriculum script repository, where the content related to a question or problem is stored. Each entry includes an ideal answer, sets of expectations and misconceptions, hints and prompts, a set of key words and synonyms, and a summary.

The tutoring process is guided by an animated conversational agent, who engages students in natural language dialogs. The agent's feedback is presented via text-to-speech output, facial expressions and gestures of the agent, and of course as text on the screen. The alternatives to the tutoring dialogue are stored within a network, where the nodes refer to knowledge goals and dialogue states, and the arcs represent tutor actions (e.g., prompt, hint, feedback) and dialogue moves. A dialog management module keeps track of the tutoring process (Graesser et al., 2004). In order to accomplish this task, AutoTutor needs to classify speech acts and to identify the next ideal answers. The following excerpt from a real tutoring session with a student shows some of the features of AutoTutor (discourse tags in capitals, Graesser et al., 2004, p. 5):

> *"AutoTutor*: When a car without headrests on the seats is struck from behind, the passengers often suffer neck injuries. Why do passengers get neck injuries in this situation? (MAIN QUESTION)
> *Student*: The passenger experiences a backward acceleration, while the car moves forward.
> *AutoTutor*: And can you add to that? (PUMP)
> *Student*: This is because the passenger experiences a force opposite in direction to the force on the car.
> *AutoTutor*: Um, yes, correct! (POSITIVE FEEDBACK) When the neck provides a large force, what within the neck would get damaged? (HINT)
> *Student*: The spine.
> *AutoTutor*: Umm (NEUTRAL NEGATIVE FEEDBACK), I think you can get this. Damage within the neck will occur in the ligaments and the . . .? [*spoken with rising intonation and accompanied with a prompting hand gesture*] (PROMPT):
> *Student*: Muscles.
> *AutoTutor*: Brilliant! (POSITIVE FEEDBACK) Ligaments and the muscles. (PROMPT RESPONSE) Let's keep going…"

The quality of the tutoring dialogue in AutoTutor has been proved in many respects, for example, conversational smoothness and pedagogical quality of the interaction process (Person, Graesser, Kreuz, Pomeroy, & the Tutoring Research Group, 2001), naturalness of AutoTutor's dialog moves (Person, Graesser, & the Tutoring Research Group, 2002) and learning gains from the students (Graesser, Moreno, et al., 2003; Graesser, Jackson, et al., 2003). While it clearly does not reach the quality of human-human interaction, AutoTutor does engage the student in an interaction and keeps him or her reflecting about a topic (for a funny demonstration see Olney, 2006). Graesser and colleagues (2004) report a mean effect size of .67 compared to a control group without further information, .82 compared to a textbook control group, and .08 compared to human tutoring in computer-mediated conversation. The last figure essentially suggests that AutoTutor is as efficient in teaching students as a human teacher. This effect can of course not be attributed to LSA alone, but rather to the combination of techniques used in AutoTutor.

Knowledge Acquisition and Training of Summarisation Skills

The systems State The Essence, Summary Street® (Steinhardt, 2001), conText (Lenhard, Baier, Hoffmann, & Schneider, 2007) and Apex (Lemaire & Dessus, 2001) aim at guiding students through the process of writing essays and summaries by giving content based feedback on the student's draft. The main goals of these ITS lie in strengthening knowledge acquisition and text comprehension. According to Wade-Stein and E. Kintsch (2004), summarisation is a key method in text comprehension, because it leads to a more attentive, active reading which furthermore can lead to more extensive writing. The reader has to reconstruct the meaning of the text in a generalized form and to link it to prior knowledge. Summarisation has turned out to be the most effective strategy to promote reading comprehension (Souvignier & Antoniou, in press).

State The Essence was implemented and maintained by Stahl, de Paula, Laham, Jones, Schreiner, and Steinhart since 1997 (Steinhardt, 2001, p. 9) and has probably been the first LSA-based ITS (Lemaire & Dessus, 2001)[2]. First experiments were promising but due to design and maintenance problems, a complete redesign was necessary (Steinhardt, 2001, p. 27). The successor, *Summary Street*® (Kintsch, Caccamise, Franzke, Johnson & Dooleay, 2007) has been in development at least since 2000 and is available as a commercial product by Pearson Knowledge Technologies (http://www.pearsonkt.com/) since 2006. It is an easily accessible web-based environment that includes a large selection of mostly expository texts for different class levels. First, before the text is displayed, the student gets an instruction on how to write a good summary. The student then begins to work on his summary and has the opportunity to spell-check his draft. As soon as a specified amount of text has been entered (at least 50 words), the student has access to content feedback. The content results as well as the summary length ratio are displayed as charts, indicating the quality of the summary. The content coverage is calculated by comparing the summary with each paragraph of the source text. The obtained results are weighted against automatically derived thresholds (cosines), which estimate the cosines that would be obtained from an ideal summary. After completing a first draft, the student is advised to work on his essay until it meets the content coverage thresholds. Eventually the system recommends abbreviating the essay in the case that it is too long. Summary Street® provides tools to flag redundant sentences, as well as irrelevant sentences (sentences that do not correlate well with the source text) and plagiarised passages. With Summary Street®, students do not only work longer and harder on their summaries, they also write essays that are more coherent and provide a better coverage of the source text (Steinhart, 2001, 65 ff.). This benefit even increases with text difficulty (E. Kintsch, Steinhart, Stahl et al. 2004). Currently ongoing longitudinal studies indicate an improvement in reading competence when working with Summary Street® measured by standardised texts (Franzke, E. Kintsch, Caccamise, Johnson, & Dooley, 2005): After six sessions, the students showed an improvement on the summarisation subscale of the Colorado Student Assessment Programme (CSAP) of d = .42. Students with lower aptitude showed the largest amount of progress. Research on the effects of long term interventions is currently being done (Kintsch, Caccamise, Franzke, Johnson & Dooley, 2007). Summary Street has been evaluated from 2001 to 2006 in an exemplary way, with 3000 participating students in school year 2005/2006 alone (W. Kintsch, Caccamise & Snyder, 2007). In the main evaluation study (Caccamise et al., in preparation), 2,851 students (184 classrooms) from grade 5 to 9 worked with Summary Street (experimental condition) or received regular instruction (controls). The experimental classrooms worked in an ecologically valid way: They were instructed at the beginning of the study, but could work with Summary Street as the teacher chose. The number of texts studied with Summary Street in the experimental condition ranged between 0 and 12 over the course of a school year. On the post-test, the experimental group outperformed the control group in summary writing. The effect sizes ranged between .20 and .25. While this is a small effect, it shows that the transfer from the laboratory to real school life is possible, albeit with a loss of effectiveness.

[2] *The Intelligent Essay Assessor* (Foltz, Laham & Landauer, 1999), an educational application developed parallel to *State the Essence* is not described in detail here, because its focus lies rather on automatic essay scoring than on tutoring. Please refer to Miller (2003) or Dikli (2006) for a review on this system.

Recently, Summary Street® has been combined with *Intelligent Essay Assessor* (*IEA*), to form a more complete writing and comprehension tutor, called *Write To Learn* (http://www.pearsonkt.com/prodWTL.shtml). Within IEA, students are presented with over 100 different essay prompts in different categories, for example, narrative, persuasive, and expository. An example prompt in the persuasive category may be: "Everyone has a favourite pet. Write an essay about your favourite pet and explain in detail why it makes the best pet." Students can type their essays into an interface very similar to the Summary Street® interface. Upon asking for feedback, they receive a graphic indication of the quality of their essay based on a number of scoring rubrics, such as a holistic, content and mechanics score, as well as the desired length of the essay. They also have access to writing tools similar to the ones included in Summary Street®: a spell-checker, a grammar-checker, and a tool that checks for redundant sentences. For an in-depth discussion of the technical underpinnings and evaluation of this tool please consider Landauer, Foltz and Laham (2002), and Landauer, Laham and Foltz (2003).

A similar system to Summary Street® named *conText* is currently under development for the German language (Lenhard, Baier, Hoffmann, Schneider & Lenhard, 2007, Lenhard, Baier, Hoffmann, & Schneider, 2007; development snapshot available under http://www.summa.psychologie.uni-wuerzburg.de/conText/). It is aimed for the application in school beginning with grade 5 and features equivalent tools and checks. It incorporates an educational text selection module that chooses the adequate degree of text difficulty on the basis of the student's performance in preceding trials and includes basic stylistic checks. *conText* guides the student through the summarization process in sequentially arranged steps (see figure 2). First, the source text is displayed and hints are given. This may be followed by a multiple choice knowledge test to ensure, the student has read the text carefully. After having done the summarisation (online feedback on the length of summary is given during writing), unknown words are underlined, which is comparable with common text processing software. *conText* highlights plagiarised text passages using the Smith-Waterman algorithm (Irving, 2004) and gives basic stylistic feedback. The aim of the sentence analysis is to identify potential redundant sentences and potential irrelevant sentences. To determine redundancies, the maximum sentence-to-sentence cosine of an ideal summary is used as a threshold. Pairs of sentences in the student's summary exceeding this threshold are flagged. To identify potential irrelevant sentences, the cosine between each sentence of the summary and the original source text is computed. Sentences of the student's summary that fall below the minimum value of the ideal summary are marked as potentially irrelevant. And finally, in the content feedback section, the cosine between the summary and each chapter of the source text, as well as the cosine between the whole summary with the source text are computed both for the student's draft and the ideal summary. The content feedback is displayed as bar charts for each chapter of the source text and the whole text. Values of the student's summary that come near to or even surpass the ideal summary get an excellent mark. If the student receives lower the values, the bar chart will reflect this with shorter bars. The student may now choose to revise the draft or to save and exit.

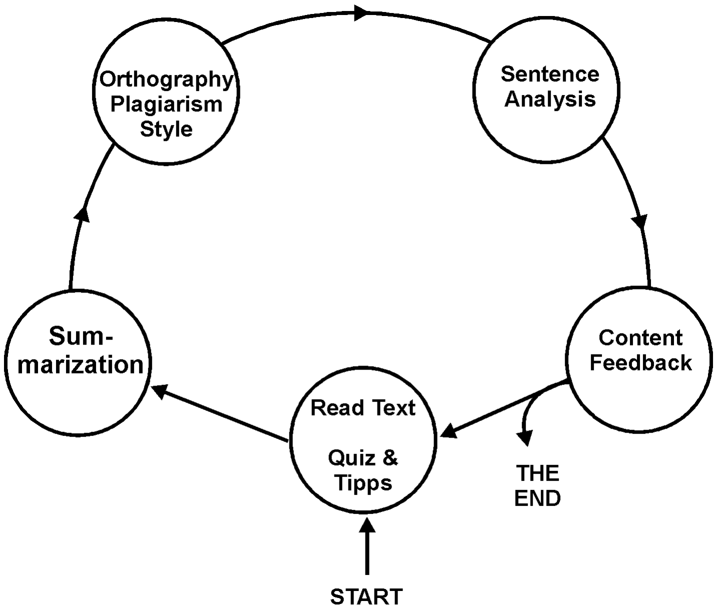

Fig 2. *conText* guides the student through the summarization process in sequentially arranged steps. After reading the source text and receiving additional help, the student engages in the summarisation task. He or she gets information on orthography, plagiarism and style before the content of the sentences is analysed. Potentially irrelevant and redundant sentences are flagged. Afterwards, content feedback is displayed and the student may engage in another program cycle in order to improve the draft.

So far, it has been used as a prototype in laboratory experiments with undergraduate psychology students on expository texts in the areas of biology and geology. Students working with content feedback showed a higher individual progress in content coverage during writing compared to students who summarised without receiving feedback, $t(47) = 2.59$, $p = .006$, and a higher human rated total summary quality, $t(55) = 2.26$, $p = .014$. They worked longer, $t(56) = 2.29$, $p = .013$, did more revisions, $t(56) = 5.53$, $p = .000$ and showed higher knowledge gains compared to read-only and summarize-only groups, $F(2, 34) = 38.86$, $p = .000$. These effects were not due to longer amounts of time spend on the task, but they stayed significant when the working time was statistically controlled. The first experiments in schools during the summer of 2007 with four training sessions during a single month showed comparable effects on time, on task, and number of revisions. The effects generalised and could be found in follow-up tests as well.

Apex (“un système d'*a*ide à la *p*réparation d'*ex*amens” [a system that helps to prepare exams], Dessus, & Lemaire, 1999; Lemaire, & Dessus, 2001) is a French system that features content-based, outline-based, and coherence-based feedback to student essays, as well as a general grade. The teacher enters course material and specifies key concepts within the text, the topic of the text and all necessary thresholds. Thus, in Apex there is a more detailed specification of main ideas compared to Summary Street® or conText, where the source text is simply split into chapters. Analogously to IEA, the student using Apex only knows the topic but does not get a source to summarize. On content level, Apex computes the similarity of the different sentences of the essay with the pre-entered course material, gives feedback about the coverage of the different notions, and assigns an overall grade. On outline level, Apex displays the most similar notion to each sentence in order to provide the student with an overview of the topic. And finally, the student gets information about coherence breakdowns within his or her essay on the basis of sentence to sentence comparison. Lemaire and Dessus (2001) experimented with feedback on demand and online feedback during writing, but while working with Apex, students did not write better essays compared to a control group. Moreover, the overall scores provided by Apex showed correlations with human graders below r = .5. Apparently, the development of Apex discontinued in 2002 (personal correspondence with B. Lemaire, 11/20/06 and G. Dènhiere 11/22/06).

It must be mentioned, however, that Summary Street®, IEA, conText and Apex do not exactly meet the criteria of a classical ITS. For instance, they do not feature an expert module in the traditional sense but rely on the domain knowledge captured by the semantic space and the source texts (Steinhart, 2001, p. 25). Apart from the educational text selection module in conText, these systems do not contain a real student module either, but instead use LSA-based comparisons of semantic similarities between the student's draft and pre-entered or automatically computed ideal solutions. They are, however, definitely tutoring systems, simply due to the fact that they deal with novel input from students and generate individualized feedback. This feedback can be used to improve the drafts and to further interact with the system. They also display intelligence, because they feature content based analysis of the input and generate intelligent feedback for the student. In this sense, these systems may very well be called intelligent tutoring systems (see Steinhart, 2001, p. 26). Furthermore, LSA provides a flexibility that helps to overcome the "implicit commitment [of former ITS] to the existence of a 'correct' representation and a view of the tutor as an agent for effecting the learner's acquisition of this representation" (Koschmann, 1996). Instead, students can present a draft varying from an ideal solution that nonetheless exceeds the predefined thresholds.

Identification and Training of Comprehension Strategies and Macro Rule Usage in Text Comprehension

Current literacy research points to strategy training as the most effective approach for fostering reading comprehension (Souvignier & Antoniou, in press). A special technique – self-explanation – was described by Chi, De Leeuw, Chiu, and LaVancher (1994), who found positive learning gains when students were asked to explain to themselves what they just learned from an expository text. This strategy was integrated into a human delivered strategy training called Self-Explanation Reading Training (SERT; McNamara & Scott, 1999) that aims at teaching students to use self-explanation and metacognitive comprehension strategies. More recently, SERT was implemented as a web-based strategy training called iSTART (*Interactive Strategy Training for Active Reading and Thinking;* McNamara, Levinstein, & Boonthum, 2004). The strategy training in SERT and iSTART incorporates five different strategies: *comprehension monitoring* (being aware of comprehension problems during reading), *paraphrasing* (rephrasing the meaning in one's own words), *bridging* (interconnecting the meaning of the current sentence with previously read sentences), *prediction* (anticipating the content of the yet unread text passages) and *elaboration* (linking the current text to prior knowledge).

SERT features three phases (introduction, demonstration and practice), which take two hours time in total and can be applied to small groups of students (McNamara, Levinstein, & Boonthum, 2004). iSTART was implemented to deliver SERT to large numbers of students at reasonable costs. iSTART therefore mimics SERT as closely as possible: It possesses the same order of stages and makes use of animated pedagogical agents and text-to-speech synthesizers in order to intimately model the tutoring process. These agents were inspired by AutoTutor.

In the introduction, the student watches a classroom discussion among a teacher agent and two student agents (McNamara, Levinstein, & Boonthum, 2004). Then he or she follows a demonstration session, where a teacher character (Merlin) interacts with a trainee character (Genie). At several points, the demonstration session is interrupted and the student decides which comprehension strategies were used by Merlin and Genie. This is mainly done by multiple choice questions or by clicking on text passages. In the practice stage, the teacher agent directly interacts with the student and the student articulates self explanations in free text style. In earlier versions of iSTART these free text responses were analysed mainly on surface level, by observing sentence length, lemmatizing the input and counting key words. Later on, LSA-based evaluation of student answers by comparing them to ideal answers has been added (Millis, Kim, Todaro, Magliano, Wiemer-Hastings, & McNamara, 2004). Recent analyses prove that LSA-based systems are superior to keyword-based systems in discriminating different comprehension strategies in self-explanatory statements (McNamara, Boonthum, Levinstein, & Millis, in press). Working with iSTART leads to a greater gain in the quality of self-explanations compared to a teacher-guided practise (d_{korr} = .88[3]), whereas teacher-guided practise turned out to be marginally superior to iSTART in improving reading comprehension (McNamara, & CSEP, 2006). It has to be noted, though, that SERT (and presumably also iStart) has only marginal effects on reading comprehension (McNamara, 2004). Mainly students with low text comprehension profit from a training with SERT and iSTART and the effect is more pronounced on superficial processing levels (McNamara, O'Reilly, Rowe, Boonthum, & Levinstein, in press) as, for example, text-based questions.

A second learning environment for the detection and promotion of strategies is a French system described by Lemaire, Mandin, Dessus and Denhière (2005). It guides the student towards using appropriate macro-rules when summarising a source text. The authors describe six different macro-rules. Three rules were taken from the model of discourse comprehension: deletion, generalization, construction (Kintsch, & van Dijk, 1978). The remaining were inspired by Brown and Day (1983): paraphrase, copy, off-topic. Lemaire et al. (2005) defined LSA-thresholds and patterns to match the sentences of the students' draft to one of the different macro rules. For example, they specified that the copy strategy has been used, when at least one sentence of the source text is very similar to one sentence of the summary, or that generalization has been used when there are several sentences of the source text that are close to one sentence of the summary. The aim of the system is to improve the usage of higher-order macro rules, like generalization and construction, and to favour them over strategies, like copying and off-topic statements. While the theoretical framework already exists and a corresponding learning environment has been implemented, no further empirical data has been reported yet.

Story Writing Tutors

In story writing, there are different constraints a student has to deal with. These constraints include basic mechanics of writing as, for example, spelling, grammar, or punctuation, but also features of the story's content, like coherence, relevance and interestingness (Wiemer-Hastings, & Robertson, 2001). For students it is often difficult to discriminate between what they know, what they have written, and what they intend to say. The story writing tutor *Select-a-Kibitzer* evaluates different aspects of a student's draft and provides feedback via different animated agents, called "kibitzer"

[3] corrected effect size d_{korr} (Klauer, 1993, p. 58) based on the data provided in McNamara and CSEP (2006).

(Wiemer-Hastings, & Graesser, 2000). Each kibitzer represents a different aspect or constraint, for example, coherence, style, grammar, semantics, and tries to express, what it understands from the text, thus making possible problems explicit. Select-a-Kibitzer is seemingly not in use any more and no systematic evaluation is available.

However, the experiences drawn from Select-a-Kibitzer resulted in the development of *StoryStation* (Robertson & Wiemer-Hastings, 2003), a story writing tutor that's user interface was notably partly designed by children aged eleven and twelve. It is dedicated to children with basic writing competency, but who still could benefit from further help. When rewriting a story narrated by a story teller or showed via video, the cognitive load of inventing an entirely new plot is reduced and the student is able to focus on writing techniques. Therefore, StoryStation applies to rewritten stories for which it provides positively phrased, constructive comments. Comparable to Select-a-Kibitzer, it features multiple animated characters, who evaluate a specific aspect of the student's story (Wiemer-Hastings & Robertson, 2001), including spelling, word banks (list of good words), dictionary, thesaurus, vocabulary, characterization techniques (Robertson, 2006), and plot (in progress, Halpin, Moore, & Robertson, 2004a; Robertson, & Cross, 2004). It uses a combination of LSA and naive Bayes for the classification of the plot quality of rewritten stories, which classifies the plot structure almost as reliably as human raters (Halpin, Moore, & Robertson, 2004b).

StoryStation tries to teach the following writing strategies (Robertson, & Cross, 2004): (1) use rich, descriptive, infrequent vocabulary, (2) describe characters in the story, and (3) avoid incoherent plots. The first strategy is taught by highlighting good vocabulary in the stories (frequency information of the British National Corpus and from corpora of children's stories is used). The second approach uses a word spotting approach and stresses the parts of the story for which characterization techniques have been used. The third strategy applies LSA and knowledge of narrative schemas to identify, where the student's plot deviates from the expected structure. Robertson and Cross (2004) evaluated the acceptance and usability of StoryStation among 60 ten to twelve year old children. The children enjoyed working with StoryStation and found the feedback useful. There has not yet been an evaluation, whether StoryStation leads to better, more coherent, and more interesting stories. The System is still in use however and there are plans for further studies (personal correspondence with J. Robertson, 08/28/07).

CONCLUSIONS

LSA has shown to be a valuable tool in the development of ITS and especially in attenuating the "language barrier" between humans and tutoring systems. The LSA-based ITS built so far address the goals of modelling human tutoring dialogues (AutoTutor), enhancing text comprehension and summarisation skills (State-The-Essence, Summary Street®, conText, Apex), training toward comprehension strategies (iStart, French system without a name) and improving story and essay writing (IEA, Write To Learn, Select-a-Kibitzer, StoryStation). Though there are serious limitations to LSA as, for example, the lack of syntax information, the presented approaches by no means tap the full potential of LSA in educational applications. Therefore, researchers are currently working on numerous new areas of application. A promising route, for example, could be the usage of LSA in educational text selection on the basis of the student's prior knowledge (Rehder et al., 1998; Wolfe et al., 1998; Landauer, 2002) and to integrate this feature in existing ITS.

Similarly, LSA can help to improve users' performance in tasks requiring the dynamic use of complex information and to speed up learning processes by finding and structuring relevant information in manuals (Foltz & Landauer, 2007). Moreover, LSA can be used for learner positioning in learning networks in the context of life long learning programs (van Bruggen, Rusman, Giesbers, Koper et al., submitted a, & b), in order to select peers for collaborative learning processes (van Rosmalen, Sloep, Brouns, Kester, Koné & Koper, 2007), and to facilitate collaborative learning in distance learning environments (Streeter, Lochbaum, LaVoie & Psotka, 2007). The assessment of personality traits like social competence on the basis of the student's input (Wild & Stahl, 2007) opens a pathway to further individualize tutoring processes. And finally, there are approaches to use LSA for the ergonomic enhancement of user interfaces, as for example word guessing in systems for patients with locked-in syndrome, who possess extremely restricted communicative abilities (Wandmacher & Antoine, 2007). Nonetheless, the lack of syntactic information restricts LSA's performance in evaluating short answers. To overcome this obstacle would considerably widen the potential of LSA in educational applications.

LSA is a complex technology and Kintsch, McNamara, Dennis & Landauer (2007, p. 475) state that "funding (in relatively large amounts) is essential for projects to develop and test educational applications" like those reviewed in this paper. What is more, when trying to place LSA-based ITS in schools, there are several difficulties that are both practical and theoretical in nature and that apply to virtually all ITS. As the resource consumption of LSA does not pose a real problem to modern desktop computers anymore, maybe the biggest part of the work in developing a user friendly and applicable system lies in the enhancement of an already working laboratory prototype to a useable real-world application. As, from a scientific point of view, this is a relatively fruitless and time consuming activity, only few scientists are willing and able to engage in it. In the light of these difficulties, many systems cease to exist with the ending of the corresponding research project (e. g. Apex).

Despite the fact that some LSA systems are among the best evaluated ITS and show remarkable performances (AutoTutor, Summary Street®), most of the others lack a systematic evaluation or even showed to be ineffective. This fact underlines the necessity not only to assess the efficacy of an ITS regarding its ability to model a skilled human tutor, but also to measure the effects it has on the learner (Steinhart, 2001, p. 27f.; Koschmann, 1996; Corbett, Koedinger, & Anderson, 1997). Beside a sound didactic concept and the smooth integration in curricula, this educational assessment is a key precondition in mastering the transition between scientific research and the broad usage of these applications in the educational system.

ACKNOWLEDGEMENTS

This study was financed by the German Scientific Community (DFG, grant no. SCHN 315/29-1). I would like to thank Prof. Dr. Guy Denhière, Dr. Benoît Lemaire and Dr. Judy Robertson for the personal correspondence and Dr. Alexandra Lenhard, Marjorie Kinney, Darius Endlich, Monika Ringleb and Jörg Lenhard for their helpful comments on the paper.

REFERENCES

Autotutor (2006). available online: http://www.autotutor.org/ (last accessed December 2006).

Berry, M. W, Dumais, S. T, and O'Brien, G. W. (1995). Using linear algebra for intelligent information retrieval. *SIAM Review, 37(4),* 573-595.

Brown, A. L., & Day, J. D. (1983). Macrorules for summarizing texts: The development of expertise. *Journal of Verbal Learning and Verbal Behavior, 22,* 1-14.

Caccamise, D., Snyder, L., Johnson, N., Allen, C., DeHart, M., Kintsch, E., et al. (in preparation). Summary Street: Scale-up and evaluation.

Chi, M. T. H., De Leeuw, N., Chiu, M., & LaVancher, C. (1994). Eliciting self-explanations improves understanding. *Cognitive Science*, *18*, 439-477.

Corbett, A. T., Koedinger, K. R., & Anderson, J. R. (1997). Intelligent tutoring systems. In M. G. Helander, T. K. Landauer, & P. V. Prabhu (Eds.) *Handbook of Human-Computer Interaction, (pp. 849-874).* Amsterdam, The Netherlands: Elsevier Science.

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*, 391-407.

Dessus, P., & Lemaire, B. (1999a). *Apex*, un système d'aide à la préparation d'examens. *Sciences et Techniques Éducatives*, *6*(2), 409-415.

Dikli, S. (2006). An Overview of Automated Scoring of Essays [available under: http://escholarship.bc.edu/jtla/vol5/1/, last accessed June 2007]. *The Journal of Technology, Learning, and Assessment, 5*.

Dumais, S. T. (1991). Improving the retrieval of information from external resources. *Behavior Research Methods, Instruments, & Computers, 23,* 229-236.

Foltz, P. & Landauer, T. (2007). Helping People Find and Learn From Documents: Exploiting Synergies Between Human and Computer Retrieval With SuperManual®. In T. K. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *The Handbook of Latent Semantic Analysis (pp 323-345).* Mahwah, NJ: Erlbaum.

Foltz, P. W., Kintsch, W.,& Landauer, T. K. (1998). The measurement of textual Coherence with latent Semantic Analysis. *Discourse Processes*, *25*, 285-307.

Foltz, P. W., Laham, D., & Landauer, T. (1999). Automated Essay Scoring: Applications to Educational Technology. *World Conference on Educational Multimedia, Hypermedia and Telecommunications, 1999(1*), 939-944.

Franzke, M., Kintsch, E., Caccamise, D., Johnson, N., & Dooley (2005). Summary Street: Computer Support for Comprehension and Writing. *Journal of Educational Computing Research*, *33*, 53-80.

Graesser, A., Lu, S., Jackson, G.T., Mitchell, H., Ventura, M., Olney, A., & Louwerse, M.M. (2004). AutoTutor: A tutor with dialogue in natural language. Behavioral Research Methods*, Instruments, and Computers, 36*, 180-193.

Graesser, A., Moreno, K., Marineau, J., Adcock, A., Olney, A., & Person, N. (2003). AutoTutor improves deep learning of computer literacy: Is it the dialog or the talking head? In U. Hoppe, F. Verdejo, and J. Kay (Eds.), *Proceedings of Artificial Intelligence in Education. (pp, 47-.54).* Amsterdam: IOS Press.

Graesser, A., Penumatsa, P, Venture, M., Cai, Z., & Hu, X. (2007). Using LSA in AutoTutor: Learning Through Mixed-Initiative Dialogue in Natural Language. In T. K. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *The Handbook of Latent Semantic Analysis (pp 243-262).* Mahwah, NJ: Erlbaum.

Graesser, A., Wiemer-Hastings, K., Wiemer-Hastings, P., Kreuz, R., & the Tutoring Research Group (1999). AutoTutor: A simulation of a human tutor. *Journal of Cognitive Systems Research, 1,* 35-51.

Graesser, A.C., Jackson, G.T., Mathews, E.C., Mitchell, H.H., Olney, A.,Ventura, M., Chipman, P., Franceschetti, D., Hu, X., Louwerse, M.M., Person, N.K., & the Tutoring Research Group (July, 2003). Why/AutoTutor: A test of learning gains from a physics tutor with natural language dialog. *Paper presented at the 25rd Annual Conference of the Cognitive Science Society*. Boston, MA.

Halpin, H., Moore, J. D., & Robertson, J. (2004b). Automatic analysis of plot for story rewriting. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Barcelona, Spain.

Halpin, H., Moore, J., & Robertson, J. (2004a). Towards automated story analysis using participatory design. *International Multimedia Conference, 75-83*. USA: New York.

Irving, R. (2004). Plagiarism and Collusion Detection using the Smith-Waterman Algorithm (available: http://www.dcs.gla.ac.uk/publications/PAPERS/7444/TR-2004-164.pdf). *DCS Technical Report. Dept of Computing Science*, University of Glasgow.

Kintsch, E., Caccamise, D., Franzke, M., Johnson, N. & Dooley, S. (2007). Summary Street®: Computer-Guided Summary Writing. In T. K. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *The Handbook of Latent Semantic Analysis (pp 263-278)*. Mahwah, NJ: Erlbaum.

Kintsch, E., Caccamise, D., Franzke, M., Johnson, N., & Dooley, S. (2007). SummaryStreet®: Computer-Guided Summary Writing. In T. K. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *The Handbook of Latent Semantic Analysis (pp 263-278)*. Mahwah, NJ: Erlbaum.

Kintsch, W., & van Dijk, T. A. (1978). Toward a model of text comprehension and production. *Psychological Review, 85*, 363-394.

Kintsch, W., McNamara, D., Dennis, S. & Landauer, T. (2007). LSA and Meaning: In Theory and Application. In T. K. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *The Handbook of Latent Semantic Analysis (pp 467-479)*. Mahwah, NJ: Erlbaum.

Klauer, K.(1993). *Denktraining für Jugendliche. Ein Programm zur intellektuellen Förderung. Handanweisung*. Hogrefe.Göttingen.

Kontostathis, A., & Pottenger, W.M. (2002). Detecting patterns in the LSI term-term matrix. *Workshop on the Foundation of Data Mining and Discovery, IEEE International Conference on Data Mining*.

Koschmann, T. (1996). Paradigm shifts and instructional technology: An introduction. In T. Koschmann (Ed.), *CSCL: Theory and practice (pp. 1-23)*. Mahwah, NJ: Lawrence Erlbaum Associates.

Lanczos, C. (1950). An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators (available online: http://www.cs.duke.edu/courses/fall06/cps258/references/Krylov-space/Lanczos-original.pdf, date of retrieval: 10.01.2007). *Journal of Research of the National Bureau of Standards, 48*, 255-282.

Landauer, T. K. (2002). On the computational basis of learning and cognition: Arguments from LSA [available online: http://lsa.colorado.edu/papers/Ross-final-submit.pdf, last accessed December 2006]. In B. H. Ross (Ed.), *The psychology of learning and motivation: Advances in research and theory, Vol. 41 (rpp. 43-84)*. San Diego, CA, US: Academic Press.

Landauer, T. K. , Laham, D., & Foltz, P. W., (2003). Automatic essay assessment. *Assessment in Education: Principles, Policy & Practice 10(3*), 295-308.

Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review, 104*, 211-240.

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

Landauer, T. K., Laham, D., Rehder, B., & Schreiner, M. E., (1997). How well can passage meaning be derived without using word order? A comparison of Latent Semantic Analysis and humans. In M. G. Shafto, & P. Langley (Eds.), *Proceedings of the 19th annual meeting of the Cognitive Science Society (pp. 412-417)*. Mawhwah, NJ: Erlbaum.

Landauer, T.K., & Foltz, P. (March 2007). *New LSA Education Applications at University of Colorado and Pearson Knowledge Technologies*. Talk held at the 1st European Workshop on Latent Semantic Analysis in Technology Enhanced Learning, Heerlen, Netherlands.

Landauer, T.K., Foltz, P. W., & Laham, D. (2002) Automated scoring and annotation of essays with the Intelligent Essay Assessor, In M. Shermis, & J. Burstein (Eds.), *Advances in Automated Essay Scoring.* New York: Academic Press.

Landauer, T.K., McNamara, D., Dennis, S., & Kintsch, W. (2007). *Handbook of Latent Semantic Analysis*. Mahwah: Erlbaum.

Lemaire, B, Mandin, S., DessusP, & Denhière, G. (2005). Computational Cognitive Models of Summarization Assessment Skills. In B. G. Bara, L. Barsalou, & M. Bucciarelli (Eds.), Proceedings of the 27th Annual Conference of the Cognitive Science Society (CogSci' 2005) (pp. 1266-1271). Mahwah: Erlbaum.

Lemaire, B., & Denhière, G. (2004) Incremental Construction of an Associative Network from a Corpus. In K. Forbus, D. Gentner, & T. Regier (Eds.), *Proceedings 26th Annual Meeting of the Cognitive Science Society (825-830)*, Chicago.

Lemaire, B., & Dessus, P. (2001). A system to assess the semantic content of student essays. *Journal of Educational Computing Research, 24(3)*, 305-320.

Lenhard, W., Baier, H., Hoffmann, J., & Schneider, W. (2007). Automatische Bewertung offener Antwortformate mittels Latenter Semantischer Analyse [Automatic Scoring of Open Text Responses via Latent Semantic Analysis]. *Diagnostica*.

Lenhard, W., Baier, H., Hoffmann, J., Schneider, W., & Lenhard, A. (2007). Training of summarisation skills via the use of content based feedback [available online: http://www.wolfgang.lenhard.info/Lenhard-Baier_et_al_Heerlen_-final.pdf, last accessed June 2007]. *Proceedings of the First European Workshop on Latent Semantic Analysis in Technology Enhanced Learning*, 26-27.

Martin, D., & Berry, M. (2007). Mathematical Foundations Behind Latent Semantic Analysis. In T. K. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *The Handbook of Latent Semantic Analysis (pp 35-56)*. Mahwah, NJ: Erlbaum.

McNamara, D. S. (2004). SERT: Self-explanation reading training. *Discourse Processes, 38*, 1-30.

McNamara, D. S., & Scott, J. L. (1999). Training reading strategies. In M. Hahn, & S. C. Stoness (Eds.), *Proceedings of the Twenty-first Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum.

McNamara, D. S., Boonthum, C., Levinstein, I. B., & Millis, K. (2007). Evaluating self-explanations in iSTART: comparing word-based and LSA algorithms. In T. Landauer, D.S. McNamara, S. Dennis, & W. Kintsch (Eds*.), Handbook of Latent Semantic Analysis (227-242)*. Mahwah, NJ: Erlbaum.

McNamara, D. S., Levinstein, I. B., & Boonthum, C. (2004). iSTART: Interactive strategy training for active reading and thinking. *Behavioral Research Methods, Instruments, and Computers, 36*, 222 - 233.

McNamara, D. S., O'Reilly, T., Rowe, M., Boonthum, C., & Levinstein, I. B. (in press). iSTART: A web-based tutor that teaches self-explanation and metacognitive reading strategies. In D.S. McNamara (Ed.), *Reading comprehension strategies: Theories, interventions, and technologies*. Mahwah, NJ: Erlbaum.

McNamara, D.S., & the CSEP Lab. (2006). iSTART: Benefits and Effects of Extended Practice. *Technical Report*. University of Memphis.

Miller, T. (2003). Essay Assessment with Latent Semantic Analysis. *Journal of Educational Computing Research, 29(4)*, 495–512.

Millis, K., Kim, H. J., Todaro, S. Magliano, J. P., Wiemer-Hastings, K., & McNamara, D. S. (2004). Identifying reading strategies using latent semantic analysis: Comparing semantic benchmarks. *Behavior Research Methods, Instruments, & Computers, 36*, 213-221.

Nakov, P. (2000). Getting Better Results With Latent Semantic Indexing. *Proceedings of the Students Presentations at the European Summer School in Logic Language and Information (ESSLLI'00),* 156-166.

Nakov, P., Popova, A., & Mateev P (2001). Weight functions impact on LSA performance. *Proceedings of the EuroConference Recent Advances in Natural Language Processing, RANLP'01*, 187-193.

Olney, A. (2006). *The Physhics Tutor* [demonstration available under: https://umdrive.memphis.edu/aolney/public/projects/bass/asr_billybass_divx.avi, last accessed June 2006].

Person, N.K., Graesser, A.C., & the Tutoring Research Group (2002). Human or computer?: AutoTutor in a bystander Turing test. In S. A. Cerri, G. Gouarderes, & F. Paraguacu (Eds.), *Intelligent Tutoring Systems 2002 (pp. 821-830)*. Berlin, Germany: Springer.

Person, N.K., Graesser, A.C., Kreuz, R.J., Pomeroy, V., & the Tutoring Research Group (2001). Simulating human tutor dialog moves in AutoTutor. *International Journal of Artificial Intelligence in Education , 12*, 23-39.

Puppe, F. (1992). Intelligente Tutorsysteme [Intelligent Tutoring Systems]. *Informatik-Spektrum, 15*, 195-207.

Rehder, B., Schreiner, M. E., Wolfe, M. B., Laham, D., Landauer, T. K., & Kintsch, W. (1998). Using Latent Semantic Analysis to assess knowledge: Some technical considerations. *Discourse Processes, 25*, 337-354.

Robertson, J. (2002) *Experiences of Designing with Children and Teachers in the StoryStation Project*. Interaction Design and Children 2002. Eindhoven, Netherlands.
Robertson, J. (2006). StoryStation: intelligent feedback on story writing. *Language Play and Computers Workshop* [available online: http://groups.inf.ed.ac.uk/standup/workshop/slides/StandUp_Judy_Robertson_1.pdf, last accessed December 2006]. University of Dundee: Schotland.
Robertson, J. and Wiemer-Hastings, P. (2003) StoryStation: An intelligent tutoring system for story writing. *Interactive Event at AIED 2003 conference* [available online: http://www.cs.usyd.edu.au/~aied/vol2/vol2_Robertson.pdf, last accessed December 2006]. Sydney: Australia.
Robertson, J., & Cross, B. (2004). Children's perceptions about writing with their teacher and the StoryStation learning environment. *International Journal of Continuing Engineering Education and Lifelong Learning (IJCEELL), 14,* 454 - 471.
Robertson, J., & Cross, B. (2004). Children's perceptions about writing with their teacher and the StoryStation learning environment. *International Journal of Continuing Engineering Education and Lifelong Learning (IJCEELL), 14(6)*, 454-471.
Souvignier, E., & Antoniou, F. (in press). Förderung des Leseverständnisses bei Schülerinnen und Schülern mit Lernschwierigkeiten -- eine Metaanalyse [Fostering Reading Comprehension in Students with Mental Disabilities – a Metaanalysis]. Vierteljahresschrift für Heilpädagogik und ihre Nachbargebiete.
Steinhardt, D. (2001). *Summary Street: An Intelligent Tutoring System for Improving Student Writing through the use of Latent Semantic Analysis* [available online: http://lsa.colorado.edu/papers/daveDissertation.pdf, last accessed December 2006]. PhD Thesis, Department of Psychology, University of Colorado.
Streeter, L., Lochbaum, K. LaVoie, N., & Psotka, J. (2007). Automated Tools for Collaborative Learning Environments. In T. K. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *The Handbook of Latent Semantic Analysis (pp 279-291)*. Mahwah, NJ: Erlbaum.
Van Bruggen, J. M., Rusman, E., Giesbers, B., & Koper, R. (submitted a). Latent semantic analysis of small-scale corpora for positioning in learning networks [available online: http http://hdl.handle.net/1820/561, last accessed December 2006].
Van Bruggen, J. M., Rusman, E., Giesbers, B., & Koper, R. (submitted b). Content-based positioning in learning networks [available online: http://hdl.handle.net/1820/576, last accessed December 2006].
Van Rosmalen, P., Sloep, P., Brouns, F., Kester, L., Koné, M. &Koper, K. (2007). Question-Answering – Connecting and Supporting the Learner [available online: http://hdl.handle.net/1820/576, last accessed August 2007]. *Proceedings of the First European Workshop on Latent Semantic Analysis in Technology Enhanced Learning*, 1-2.
Wade-Stein, D., & Kintsch, E. (2004). Summary Street: Interactive computer support for writing. *Cognition and Instruction, 22*, 333-362.
Wandmacher, T., & Antoine, J. (2007). Using LSA for Word Prediction: A Comparison of Integration Techniques [available online: http://hdl.handle.net/1820/576, last accessed August 2007]. *Proceedings of the First European Workshop on Latent Semantic Analysis in Technology Enhanced Learning*, 9-10.
Wiemer-Hastings, P. (1999). How latent is Latent Semantic Analysis? *Proceedings of the Sixteenth International Joint Congress on Artificial Intelligence (pp. 932- 937*). San Francisco: Morgan Kaufmann.
Wiemer-Hastings, P. and Robertson, J. (2001). Story Station: Children's story writing software. *Poster presented at AIED* [available online: http://www.cogsci.ed.ac.uk/~judyr/aiedposter.pdf, last accessed December 2006].
Wiemer-Hastings, P., & Graesser, A. (2000). Select-a-Kibitzer: A computer tool that gives meaningful feedback on student compositions. *Interactive Learning Environments, 8*, 149-169.
Wiemer-Hastings, P., Wiemer-Hastings, K., & Graesser, A.C., and the Tutoring Research Group. (1999). Improving an intelligent tutor's comprehension of students with Latent Semantic Analysis. In S. Lajoie, & M. Vivet (Eds.), *Artificial Inteligence in Education (pp. 535-542).* Amsterdam: IOS Press.

Wild, F., & Stahl, Ch. (2007). Using Latent Semantic Analysis to Assess Social Competence [available online: http://hdl.handle.net/1820/576, last accessed August 2007]. *Proceedings of the First European Workshop on Latent Semantic Analysis in Technology Enhanced Learning*, 3-4.

Wild, F., Stahl, Ch., Stermsek, G., & Neumann, G. (2005). Parameters Driving Effectiveness of Automated Essay Scoring with LSA. *Proceedings of the 9th International Computer Assisted Assessment Conference*, 485-494.

Wolfe, M. B., Schreiner, M. E., Rehder, B., Laham, D., Foltz, P. W., Kintsch, W., & Landauer, T. K. (1998). Learning from text: Matching readers and text by Latent Semantic Analysis. *Discourse Processes*, *25*, 309-336.