An overview of LSA-based systems for supporting learning and teaching
Philippe Dessus

To cite this version:

HAL Id: hal-00404731
https://hal.science/hal-00404731
Submitted on 17 Jul 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
An Overview of LSA-Based Systems for Supporting Learning and Teaching

Philippe DESSUS
Laboratoire des Sciences de l’Education, University of Grenoble, France

Abstract. We present an overview of LSA-based systems that have been used in instructional settings. Current research on this subject does not take into account the cognitive aspects of learning and teaching, and describes the systems at a technical level. We propose a cognitive-based classification of these systems that can lead to the design of novel LSA-based applications.

Keywords. Latent Semantic Analysis, Tutoring Systems, Interactive Learning Environments, Natural Language Processing, Feedback, Learning, Teaching.

Introduction

Latent Semantic Analysis (LSA, [1]) is a well-known technique that captures semantic information in texts by uncovering word-usage regularities. Extensive research on LSA has proven its efficiency in the domain of natural language processing, and more specifically for computer-based instruction—tutoring systems, interactive learning environments [2].

The power of LSA lies in its versatility, resulting from its simple procedure: raw texts (representing different kinds of discourse, like teacher-student interactions, course texts, or student productions) are subject to a fast processing, as follows. Words that are shared by similar paragraphs, as well as paragraphs that contain similar words are represented similarly [3]. Without any complex human pre-processing, this mechanism allows semantic-related comparisons of words and texts (e.g., cohesion or meaning); and can lead to the simulation of the high levels of human cognition seen in instructional settings like understanding [4], summarization [5, 6], knowledge building [7], metaphor comprehension [8], tutoring [9], and meta-cognition [10]. Thanks to these main capabilities—text-based, natural language processing, cognitive account, and speed—LSA is a good candidate as a computational technique to use in association with instructional systems, in which discourse in a broad sense (e.g., dialog, written essays, course notes) plays a very important role [11].

There are numerous reviews of LSA-based educational applications, either all-purpose [12] or centered on more specific ends like essay grading [13-17] or tutoring [18]. These reviews carefully compare the performance of LSA-based systems to other techniques [16, 17], but are often dedicated to very technical aspects, without addressing higher levels of description, like learners’ cognitive processes or teachers’ pedagogical intentions. Moreover, they report implemented systems, while some
promising results not applied yet in instructional settings can be relevant as well. An overview that describes both the computational procedures used and the way they interact with instructional settings is lacking. This paper proposes such an overview, which does not focus on systems per se but on the different ways to benefit from LSA’s capabilities to deliver feedback to learners.

1. Ways to Use LSA for Analyzing Instructional Data

Though primarily devised for information retrieval purposes [19], LSA can be viewed as a model of how word meaning is acquired by humans. By the way of a factorial analysis a high dimensional space is built from a raw corpus in which each word or text is represented as a vector. LSA uncovers semantic associations between words by performing a dimension reduction of the initial space.

We can now illustrate the way LSA functions by reviewing the diversity of the texts given as input, the kinds of processing and the output delivered to learners or teachers accordingly. Figure 1 (after [20]) depicts these possibilities. First, the sources as input encompass a wide range of events occurring in instructional settings: (1) teacher written productions (analyses of learning activities); (2) student written productions (from various kinds of writing tasks); (3) raw instructional events from transcribed observations by observers; (4) course texts, from textbooks or encyclopedias. These various inputs can all be processed, as pieces of text, by three main procedures—for clarity’s sake possible additional processing (e.g., clustering, $k$-means) is not described:

- **Word to word comparisons**, for measuring how semantically close a word is to another one. These have been successfully achieved for metaphor comprehension [8], key-word extraction for finding synonyms [21], cross-language retrieval [22], or semantic memory simulation [23].

- **Word to document comparisons**, this functionality is made possible because LSA can represent a set of words (i.e., a paragraph or a text) as the sum vector of the words of which it is composed. This processing has notably been used for improving text search [24], for topic extraction aimed at building ontologies [25], and concept map generation [26].

- **Document to document comparisons**, two different documents can be compared to each other, this “judgment” being close to a human assessment of their proximity. This processing has notably been used for essay assessment [27].

2. A Taxonomy of the Instructional Applications of LSA

We then focus on the various possible outputs and their use in instructional settings. We categorized the kinds of feedback with regard to the types of data processing, as well as the pedagogical contexts of use. The types of systems are listed by their main pedagogical intention and by growing order of complexity. Systems that combine different pedagogical intentions, thus integrating several of these processing techniques are listed separately. The main categories investigated are as follows and are detailed in Figure 1 and Table 1:
• **Text selection or production** (referred to as TS). Given the text retrieval features of LSA, it is possible to use it for selecting pieces of text or for generating new ones from a set of raw texts.

• **Text production assessment** (TA). This purpose is to assess automatically different textual features of students’ texts in order to provide useful information on their quality (both on form and content).

• **Assessment of knowledge or understanding** (KA). This category is close to the previous one in that it takes students’ production as input. It focuses however on a higher level by uncovering what they have learned or understood when reading a course text and having produced a text about it.

• **Self-regulation assessment and intentions detection and assessment** (SR). This last level focuses on (meta)-cognitive processes analyzed from moves (i.e., behavior) within an environment (either in a real-world or in a computer-based environment). This analysis leads to the identification of the user’s intentions within the environment, and/or to matching the users’ moves with their verbalizations.

![Figure 1. An overview of LSA-based Processing and the Related Input/Output. T for Teacher; S for Student.](image)

3. Conclusion

To illustrate how this taxonomy can lead to new AIED systems being devised, we now describe two possible new ones. Reading both Figure 1 and Table 1 carefully, we observe that “student’s task description” as input is lacking. A LSA-based system for assessing the students’ understanding of a learning task would first let them rephrase the task with their own words and then perform a comparison of the latter with the original task. If the value of the comparison is not between upper and lower threshold values, the students would be prompted to revise the task formulation. Similarly, one can notice that LSA is underused in the context of collaborative learning. It would be possible to guide students’ learning by triangulating their position according to the objectives of the course, their own learning goals and those of their peers. A student could be prompted to complete a particular learning goal with the help of a peer because it matches both the topic of the current course and the goal of this student.
LSA-based research in instructional settings is a decade old. This paper aims to provide an overview of this research and to fuel new research directions. We argue that LSA is a good candidate to analyze instructional interactions associated with learning environments and to deliver feedback accordingly. We emphasize LSA’s versatility and detail ways to discover novel applications in AIED research, by presenting two possible new ones. Moreover, we propose that learner positioning [28], instructional design [29], automated question/answer delivering [30] and dialog acts classification [31] in collaborative learning settings appear to be important areas for future research.

Acknowledgements

This research has been published in a research report [32] and is partly supported by the LTfLL (Language Technologies for LifeLong Learning) 7th PCRD ICT-STREP project of the European Community. We thank Gillian Armitt and Benoît Lemaire for providing thoughtful comments on an earlier version of this paper.

4. References


Table 1. A Taxonomy of LSA-based Instructional Applications. Categories: TS: Text Selection; TA: Text Assessment; KA: Knowledge Assessment; SR: Self-Regulation Processes and Intentions Assessment. Legend: $S$ for Student; $T$ for Teacher; $CT$ for Course Text.

<table>
<thead>
<tr>
<th>Ref.</th>
<th>Name [Reference]</th>
<th>LSA-based Method of Processing</th>
<th>Instructional Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>TS1</td>
<td>Semantically-based search #1, word to text [24, 33]</td>
<td>Compare the $S$’s query to all the $CT$s. The retrieved ones are the closest.</td>
<td>Deliver Texts closely corresponding to the topic, avoiding synonymy mistakes (Texts with “pupil” not retrieved when typing “student”).</td>
</tr>
<tr>
<td>TS2</td>
<td>Semantically-based search #2, text to text [24]</td>
<td>Compare the $S$’s query (a paragraph from the $CT$) to all other paragraphs of the $CT$. The closest are retrieved.</td>
<td>Deliver like-paragraphs in digital manuals or libraries in order to learn more about a subject.</td>
</tr>
<tr>
<td>TS3</td>
<td>Keyword selection for a summary [24]</td>
<td>Compare a given text (either a $CT$ or a $S$’s Text) to a list of pre-selected keywords. The closest ones are displayed as a “keyword summary”.</td>
<td>Deliver informative keywords about a Text, for a deeper understanding of its content.</td>
</tr>
<tr>
<td>TS4</td>
<td>Plagiarism detection [34]</td>
<td>Compare each of the paragraphs of a $S$’s Text to each of those of a $CT$. The $n$ closest above a given threshold may have been copied by $S$.</td>
<td>Information on $S$’s plagiarism (See TA6 for an extension).</td>
</tr>
<tr>
<td>TS5</td>
<td>Main ideas selection, method #1 [6, 35]</td>
<td>Compare the sentences of a paragraph to each other. The most important has the highest average similarity to the others.</td>
<td>Information on topic coverage: Inform $S$ if (un)important ideas were (not) covered. Test if a paragraph begins/ends by a summary of it.</td>
</tr>
<tr>
<td>TS6</td>
<td>Main ideas selection, method #2 [6]</td>
<td>The most important sentence is the most activated during the simulation of its comprehension (using Kintsch’s Construction-Integration model).</td>
<td>Information on topic coverage: Inform $S$ if (un)important ideas were (not) covered (see also KA7).</td>
</tr>
<tr>
<td>TA1</td>
<td>Measuring text readability [36]</td>
<td>Compare each couple of adjacent paragraph/sentences of a $CT$. The overall mean of the similarities is function of the reading difficulty of the $CT$.</td>
<td>Test whether a $CT$ is difficult or not to read and understand.</td>
</tr>
<tr>
<td>TA2</td>
<td>Grading essays: Gold standard method [27]</td>
<td>Compare each $S$’s Text with preselected expert productions (gold standard). The closer the value, the better the grade.</td>
<td>Essay pre-grading or proofing (intermediate grading as many times as wanted by the $S$, before giving it to the $T$).</td>
</tr>
<tr>
<td>TA3</td>
<td>Grading essays: Holistic Method #1 [35, 37]</td>
<td>Compare each $S$’s Text with a set of pre-graded $S$’s Texts (or sections thereof). Its attributed grade is that of the closest pre-graded.</td>
<td>Essay pre-grading or proofing (intermediate grading as many times as wanted by the $S$, before giving it to the $T$).</td>
</tr>
<tr>
<td>TA4</td>
<td>Grading essays: Holistic Method #2: Most important sentences [38]</td>
<td>Compare each $S$’s Text sentences to 10 most important sentences of a $CT$ (as assessed by a $T$). The grade is the mean similarity between each sentence to the closest of the 10 sentences.</td>
<td>Essay pre-grading or proofing (intermediate grading as many times as wanted by $S$, before giving it to $T$).</td>
</tr>
<tr>
<td>TA5</td>
<td>Grading essays: Percentage covered [37, 39]</td>
<td>Compare each paragraph of the $S$’s Text to each of those of the $CT$. The grade is the mean of each of the similarities (plus adjustment for short paragraphs).</td>
<td>Essay pre-grading or proofing about content (intermediate grading as many times as wanted by $S$, before giving it to $T$).</td>
</tr>
<tr>
<td>TA6</td>
<td>Assessing sentence/paragraph cohesion [35, 40]</td>
<td>Compare each couple of adjacent paragraph/sentences. A cohesion gap is detected between those up to a given threshold.</td>
<td>Give information for revision purposes (unwanted cohesion gaps). Outline detection, to be compared to that intended by $S$.</td>
</tr>
<tr>
<td>TA7</td>
<td>Outline of the notions/topics composed so far [39]</td>
<td>Compare each paragraph of the S’s Text to each notion of the CT then display the closest one.</td>
<td>Outline detection, to be compared to that intended by S.</td>
</tr>
<tr>
<td>-----</td>
<td>-----------------------------------------------</td>
<td>-----------------------------------------------------------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>TA8</td>
<td>Analyzing macrorule use during summarization [6]</td>
<td>Compare each sentence of the S’s Text (summary) to each of the Texts to be summarized. Sort them against different thresholds to infer the strategies used (text copy, text deletion, off-the-subject...)</td>
<td>Summarize a (Source or S’s) Text. Analyze the usefulness of the macro-rules used while summarizing a Text.</td>
</tr>
<tr>
<td>KA1</td>
<td>Metaphor comprehension [8, 41]</td>
<td>For a metaphor like A is like a B, find the closest terms to both A and B.</td>
<td>Detect inferences from reading synonyms or notions.</td>
</tr>
<tr>
<td>KA2</td>
<td>Matching CT to S’s knowledge. Method # 1 [7]</td>
<td>Let Ss write out about their knowledge of a CT. Compare S’s Texts to each paragraph/section of the CT.</td>
<td>Match Ss to text difficulty (more conceptually-driven than TA6).</td>
</tr>
<tr>
<td>KA3</td>
<td>Matching CT to S’s knowledge. Method # 2 [42]</td>
<td>Compare Texts read so far and all the Texts to be read. The set of the Texts to read are neither very close nor very far from the Text read so far.</td>
<td>Simulate a Proximal Development Zone-based Text selection procedure.</td>
</tr>
<tr>
<td>KA4</td>
<td>Generating [43] or evaluating [44] concept maps</td>
<td>Compare each word (notion) to each other. Spatially organize each concept with respect to the most central one.</td>
<td>Give the S a big picture of the content taught and/or understood so far.</td>
</tr>
<tr>
<td>KA5</td>
<td>Matching questions with topics and students [45]</td>
<td>Compare a question with related topics/questions and possible students who can answer it, according to their competency.</td>
<td>Learning network at a distance. A S asking a given question can find peers to work with.</td>
</tr>
<tr>
<td>KA6</td>
<td>Knowledge pattern matching [46, 47]</td>
<td>Compare the utterances of Ss to a set of pre-defined patterns (e.g., problem setting, question, hypothesis). The utterance gets the category of the nearest pattern.</td>
<td>Used to determine the epistemic orientation of an utterance.</td>
</tr>
<tr>
<td>KA7</td>
<td>Simulating reader/user understanding [23, 48]</td>
<td>Simulates the inferences made during reading a CT. LSA serves as semantic memory.</td>
<td>Used to mimic text understanding or to simulate a cognitive walkthrough for testing web usability.</td>
</tr>
<tr>
<td>SR1</td>
<td>Pattern matching of writing intentions (why? how?) [35]</td>
<td>Compare each sentence/proposition to a set of pattern sentences. The closest sentence belongs to the given pattern.</td>
<td>Determine dialogue moves, pedagogical orientation, etc. (see KA6).</td>
</tr>
<tr>
<td>SR2</td>
<td>S’s assessment of understanding [33]</td>
<td>After TA5 processing, compare this processing to Ss’ own judgment. Prompt in case of discrepancy between both.</td>
<td>Check S’s judgment of understanding; allow self-regulation processes in case of discrepancy to what the machine has assessed.</td>
</tr>
<tr>
<td>SR3</td>
<td>Self-Regulated Learning and explanations analysis [49, 50]</td>
<td>Let a S read a text and say out loud what s/he understands. Compare the text read to the reflective comments.</td>
<td>Check reader’s inferences during reading. Perform cognitive task analysis.</td>
</tr>
<tr>
<td>SR4</td>
<td>Intentions uncovering [51, 52]</td>
<td>Compare the different moves within an environment to each other. The closest ones may share the same intention.</td>
<td>Intention detection within a learning environment.</td>
</tr>
</tbody>
</table>