

Analyzing Student Activity in Computer Assisted Language Learning

Petter Karlström, Teresa Cerratto-Pargman

Computer and Systems Sciences, Stockholm University/Royal Institute of Technology

petter,tessy@dsv.su.se

Abstract

We study the use of a computer application, intended for Computer Assisted Language Learning (CALL). We present an analytical framework for CALL, consisting of technology, interaction with technology, a relationship between technology and students, and a context where technology is situated.

The application we study performs several different kinds of state of the art linguistic analyses, and is intended for writing texts while paying attention to linguistic forms.

We have conducted a naturalistic field study of two informants use the tool collaboratively. Our question is in what manners these students put the tool into use.

These students let initiative be taken by the CALL application, despite it being designed with student initiative in mind, and despite students being aware of features and occasions where they could take initiative. We use our framework to point out how this student-system relationship is formed, and provide guidance for future design.

1. Introduction

Using Natural Language Processing (NLP) in Computer Assisted Language Learning (CALL) is a promising area for investigation, especially in the light of advances in NLP that have yielded applications such as spell- and grammar checking for first language writers. Much important work is currently put into perfecting technologies for language analysis, but in introducing systems that serve students' needs it is equally important to consider how technology is realized in design and put to use by second or foreign language learners.

In order to discuss these issues, we frame a tool within four perspectives: *technology*, *interaction*, *student-system relationship* and *context* [1]. These perspectives are influenced by NLP researchers Holland and Kaplan, who used similar categories for discussing problems with faulty output when using NLP-based tutors in CALL [2]. Our view differs from

Holland and Kaplan's in exploring an educational activity rather than solving issues in NLP.

In *technology*, we find the core, "infrastructure", software and hardware used for building CALL applications. Programming languages, multimedia platforms, and the currently popular "Learning Objects" are all examples of technologies. In our case, the core technology consists of different varieties of NLP engines. In *interaction* we are concerned with how that technology is provided to learners, specifically how language information is presented to learners. Closely knit to interaction is the *student-system relationship*, a set of expectations and assumptions that learners form and reform concerning the system, as they use it. A "talking head"-tutor, for example, may instill assumptions regarding its human-like properties, and these assumptions may be challenged through subsequent use. Finally, the *context* highlights where and when a system is put to use.

CALL is an interdisciplinary venture, and each of these perspectives broadly represents different research problems in the area. In actual design and use of CALL tools, the perspectives are intertwined and affect one another. However, it is not enough to merely state that they are intertwined, it is also important to investigate in detail *how* they are intertwined. In our specific case, the area of inquiry lies in the *student-system relationship*, and how the other three perspectives influence that relationship. In other words, we ask in what manners a particular CALL tool is put to use.

In studying this question, our framework is intended to be used within a sociocultural approach, stemming from Vygotsky, and of current interest in Second Language Learning [e.g. 3]. There are important differences between this approach and approaches building on traditional positivist scientific methods. The usual caveats apply, we ask the reader to bear in mind that our study is explorative and descriptive rather than controlled and prescriptive.

In other words, our interest lies not in questions such as the effectiveness of particular technological/pedagogical designs on the majority of students. Rather, we explore other issues, particularly

what kind of activity our informants engage in when using the tool. In order to do that, we must not only investigate what the students do, but also interpret their actions in light of how those actions are meaningful to them. In brief, we will argue that the part of the activity where the tool was used was one concerning error correction, and that their treatment of the tool is meaningful when instead viewing it as a tutor. We will arrive at this conclusion after analyzing what students do with the tool in terms of the categories of our framework.

Thus, we are concerned with how a particular tool is *appropriated* by the user in her activity. The nature of an artifact in use can be understood by identifying the ways people use it, the needs it serves, the history of its development, and the ways in which artifacts shape and change practices facilitating and/or constraining particular actions [4].

This introductory section will conclude with some background concerning pedagogical and theoretical issues, and present the application, "Grim", that we have studied. Then, we will present our case study, our results, and discuss these in terms of our framework.

1.1. Background

Depending on designers' views on language, pedagogy, and technology, systems are conceptualized in different ways. Pedagogically, the tool we study is situated in "Focus on Form" in second language pedagogy. "Focus on Form" advocates communicative exercises that intermittently draw students' attention to linguistic forms [5, 6]. To that end, a writing tool that performs linguistic analysis on learner language was conceived.

Computer applications for natural language analysis, such as grammar and spelling checkers, have been strangely absent from the CALL agenda during the last 20 years. One of the reasons for the absence is that advances in the area of Language Technology coincided with input-only oriented theory in Second Language Acquisition [e.g. 7]. Within this then-popular theory for learning there was no room for teaching grammar, and so computers analyzing language fell out of fashion [8]. Another reason is disappointments in natural language analysis software. The software did not live up to expectations from pedagogues. Precision/recall scores were not (and may admittedly still not be) high enough for second language pedagogues and learners.

Still, Language Technology has delivered some successful and interesting systems: Search engines, grammar checkers, keyword extractors, document summarizers, and voice recognizers to mention a few.

These kinds of systems should be well suited for "Focus on Form". Perhaps it is time for reinstated efforts in language technology for CALL [9]?

One such effort has resulted in the research prototype "Grim" [10]. It is intended to be used for drawing attention to linguistic forms when learning Swedish. Exactly which forms to focus on, and how and when to use the features of the tool, are to be decided by the students themselves and/or their teacher.

The application is intended for writing essays, and outwardly has similarities with text processors. Spell- and grammar checking are implemented by the customary underlining of errors, and by reporting on error types. Apart from the spelling and grammar checkers pervasive in word processing applications, several other means of accessing linguistic information are provided. When the users' cursor is positioned on a word in the text, information about the word is displayed in smaller frames surrounding the text. The information is drawn from a multitude of sources and consists of grammatical class and form, possible inflections, causes of errors in spelling or grammar (if any) and suggested corrections (if any). The frames may be turned on or off at the users' command. Furthermore, linguistic classes may be highlighted in the text, in addition to errors. This is thought to help students reflect on particular forms. Also present are a dictionary (Lexin), and a concordancer that gives examples of use of selected words. These are for technical implementation reasons available through menu choices. At the time for our study, the concordancer was unfortunately off-line, and therefore unavailable for our informants.

Before going into our study, it is important to note that we are critically analyzing use of Grim, but we are not attempting to criticize Grim per se. Extensive research has gone into theorizing, prototyping, and iterative user studies [10]. These studies have yielded a tool that take "Focus on Form" and free writing into account. To our knowledge, no other tool has those features, and we are therefore intrigued by Grim. We hope that our contributions will provide further refinements of Grim and tools like it.

2. Case Study

We observed "Grim" in use by two students working in a pair, during three ca 2 hours long sessions. The sessions were spread apart by ca 1 month in time, and the students made some use of the application individually between sessions. The

students are medical doctors who have migrated to Sweden.

Our informants participated in a course which teaches Swedish at an advanced level, tailor-made for medical practitioners. The course was carefully selected, because it suits "Focus on Form" well. They spoke in different mother-tongues, Persian and Russian, and where therefore obliged to speak Swedish. Before the first session, we gave them an introduction to the language tool, and let them familiarize themselves with it during half an hour.

2.1. Task

A task concerning text-reconstruction was chosen for our study. We developed the educational task together with the class's teacher, in order to find a task at the right level, fitting "Focus on Form", and familiar to the students. The teacher read a short text aloud for the students (twice), while they took notes. Then, they were asked to reconstruct the text, in a pair, capturing the essential content in their own (Swedish) words.

There are two reasons for letting students work in a pair. Pedagogically, learning on an advanced level takes place in discussion and argumentation. Methodologically, we can reach better understanding concerning how users reason about a tool when they talk about the task at hand in a natural manner.

The texts they wrote concerned medical histories of fictitious patients, to simulate writing from the students' professional life.

3. Results

Our findings lie in a gap between design and use. Our informants (we will call them Ali and Natasha) used the grammar and spelling checker more than any other tool, and chose to review their text until Grim displayed it as being correct. Interestingly, they worked in this manner during all three sessions, against their knowing that the application is limited in its linguistic analysis and explicit instructions not to treat it as being authoritative. Instructions and encouragements to make use of other features of Grim were ignored.

To show how Grim was put to use, we present illustrative examples from our data. The dialogue is translated from Swedish, and somewhat abbreviated for space and clarity. Notes are in square brackets: [].

In some senses the application was treated in the way it was intended, namely to discuss and reflect on linguistic forms. In the following excerpt, they are discussing pronouns. The pronoun in question refers to

a name in a previous sentence, a difficult problem for grammar checkers to handle. The application has marked their input as correct, but they are still uncertain:

Excerpt 1: Session 2, 1 hour 13 minutes

Ali: We are not sure about a sentence, but the computer shows it as correct. So we left it like this: "Karin laments that he doesn't recognize her." [The sentence as written was correct.]

Natasha: There are strange suggestions.

Ali: "Karin and 'he'", it should be "she".

Researcher: Who does "she" refer to?

Natasha: There are strange variants, strange sentences. [Concerning Grim's suggestions.]

There are several things to note here. In discussing the tool with us and questioning its output, they criticized the sometimes strange or faulty output of the tool, and they discussed linguistic forms. Still, it was *uncertainty* that made them keep their (correct) text as it was. The computer's linguistic authority was deemed more certain than their own, but not certain enough. Another authority, a teacher or native speaker, was necessary for them to be sure about the correct construct to use. Thus, there is uncertainty about the *technology*, while we simultaneously note a search for authoritative answers in our area of investigation, the *student-system relationship*.

Another reason for getting rid of application error reports was that the text looked better; it is the way it "should be". The *interaction* encourages correcting errors:

Excerpt 2: Session 2, 1 hour 16 minutes.

Researcher: It doesn't show any errors, no underscores. You know the program is sometimes wrong, but you still want to get rid of underscoring?

Ali: Yes.

Natasha: Yes.

Ali: It's how it looks, it should be without them.

Natasha: Yes.

Not only were they focused on getting rid of error reports, but they were not much interested at all in using features that provide less explicit feedback (e.g. syntax highlighting, language examples, dictionary, etc.). Late during the first session they confronted us:

Excerpt 3: Session 1, 1 hour 48 minutes

Ali: What is the intention of this function?

Researcher: Which one?

Ali: When you use color to highlight verbs, adjectives et cetera. It shows us that it is a verb. Why?

Researcher: It is a way of visualizing different word classes, for example adjectives.

Researcher 2: Do you think that is useful?

Ali: When we write we know which word to use. It just recapitulates that it is a verb, adjective et cetera.

Researcher: I can show you an example...

We then exemplified how the function may be useful when finalizing a text (which, incidentally, is what they were doing at this point), and one is uncertain about prepositions. If one highlights all the prepositions in the text, it is easier to concentrate on pronouns and leave other linguistic forms aside for the moment. The same argument can be used for verb tense and other linguistic issues. Our informants agreed with our explanation, but they still did not make use of the highlighting feature at that time, or in their subsequent use of Grim. For example, during the third session, they discussed linguistic forms, but did not look for support from Grim:

Excerpt 4: Session 3, 37 minutes

Natasha: She wants to step [typing what she is saying]

Ali: She *wanted* [points at the word "wants" on screen]

Natasha: She wants

Ali: (laughs)

Natasha: It is now, it happens now

Ali: But no no, it is about before surgery

Natasha: Surgery is in the future

Ali: yes

Natasha: As I understand

Ali: But we chose to write all verbs on past.

Natasha: In the past

Ali: Yes in the past. But on last sentences it is present

Natasha: Yes. Ok. Wanted. [hesitant]

While this discussion took place, linguistic information concerning verbs was available, but not used. They did not benefit from the opportunities given in the tool. These two last examples show that neither the *interaction* nor the task *context* encouraged highlighting of linguistic categories. Grim does not provide explicit feedback, and their task does not oblige them to use implicit features of Grim. Neither context nor interacting with the tool encourages them to actively seek linguistic information.

Error-checking took place by the end of the exercise, during the revision phase of writing the text. This mode of conduct was explicitly decided upon:

Excerpt 5: Session 3, 12 minutes

Ali: But you can't eh [pointing at screen]

Natasha: How do we?

Ali: But eh we can write the entire text.

Natasha: The entire?

Ali: And then correct it.

This agreement was made in accordance with their history of using the tool, which had followed this pattern. Then, while revising, they made extensive use of the tool, discussing its output and how to treat it:

Excerpt 6: Session 3, 43 minutes

Ali: There she got antidepressants [Reading from screen. The sentence is marked as incorrect]

Ali: We get "suspected erroneous construct" [citing output from Grim]

Ali: We have erroneous construct.

They discussed output from Grim at length, and tried to get rid of everything that was tagged as "wrong". These last examples shows yet again that the *student-system relationship* was one where Grim was expected to take initiative for correcting their text, and that the *context* was indeed seen as one where producing a correct text was important (other options would be e.g. a "coherent", "exhaustive", or just "good" text).

4. Analysis and Discussion

Dialogue about linguistic forms was important during the students' task. Therefore, the activity was in many senses the intended one, writing a text while discussing language ("Focus on Form"), but the tool was used only for correction. They wrote a text and then used the "tool" to "teach" them about what was wrong with it.

The dialogue during writing could probably have taken place without using the tool, which only came into question during the revision part of the writing process. This occurs despite the students knowing about other features, being encouraged to use them, and having opportunities to use them.

Because Grim does not always yield correct analyses, discussion during revision occasionally concerned the sometimes strange output. This is not necessarily problematic. As long as students do focus on linguistic forms, one may say that the design was to some extent successful. Still, it is a problem *technologically*, because it shows that using state-of-the-art NLP in language education still has some limitations, and more investigation is needed in the area. However, solely focusing on technology will not provide guidance concerning actual use of tools.

Even though error correction is one of many available tools, it is made the center piece in *interaction*. This is no accident; learners request some explicit feedback, lest they do not use the tool at all [10]. The underscoring of grammatical errors is on by default and it is the most prominent feature that users see. There is an urge to "tidy up" all the red markings.

After giving other features at least some thought during the first session, the students quickly got into a procedure where they did not use features other than error-reporting. Regarding error-reporting, they had agreed to keep some constructs reported as erroneous, if they believed them to be correct. However, they did not work in this way, and instead chose to trust the computer when they were uncertain. This approach towards the tool as correcting their text was cemented during later sessions. In working until they are judged

as "correct", they expect Grim to take initiative. They follow its authority rather than their own intuitions, and their version of the task seems to be to satisfy Grim's desire for correctness.

Another feature that *could* have been as prominent as error reporting is syntax highlighting. Visually, these appear within the text rather than in frames surrounding it. However, this feature was not used at all. The students did make use of the informative frames around the writing area, but they did so in conjunction with the more prominent error reporting. The authoritative underlining of errors, together with students being uncertain about their own ability, makes them form a *student-system relationship*, where the application is treated as a tutor that should take initiative in providing explicit feedback on errors rather than informing students about problematic forms after they have taken initiative to ask.

The task *context* dictates the meaning of the task. The indented context was one where students write a text and intermittently focus on linguistic forms. However, from a student perspective, learning situations often call for errors to be corrected. It is no wonder, then, that the natural course of action when confronted with error reports from Grim is to correct those errors. The context becomes one where writing what someone outside of themselves deems to be a *correct* text is more important than reflecting on forms until satisfied with the text. The computer teaches what is correct, and they oblige. Thus, they miss several other options for learning with the tool (e.g. by using linguistic information while writing).

We believe that tools are always transformed and shaped by their users and that there are always more possible uses than those envisaged in the original design [4]. The fact that the users transformed a CALL environment into a tutor might be a result of a particular *student-system relationship*, a particular appropriation process.

The appropriation of a tool entails several issues that have to be further elaborated. On the one hand, the appropriation of the artifact concerns the design of the features of the tool -the models and ideas of the learning activity implemented in technology- and on the other hand the type of activity designed by the teacher and deployed by the learners. It also concerns the relationship between learners and between learners and the knowledge they share.

In this particular case we observed that the users made little use of new linguistic tools in their writing and reviewing. They preferred to use known features and to concentrate on correctness of the texts. One might ask if this really is a consequence of interface design, or just the fact that users found it hard to

engage with a quite demanding activity such as questioning themselves about their language. One might also ask what really was new for the users: the tool supporting writing from different perspectives, or the task of actively questioning knowledge about the target language, and searching a computer application for answers?

We believe that these issues are intertwined, tools in use change practice, and practice changes the way tools are appropriated. With our framework, we highlight the interplay of issues that make the tool a corrective one, in spite of the intentions of both task and tool.

5. Acknowledgement

We are grateful to the Swedish Research Council (VRÅ) for the funding that made this research possible.

6. References

- [1] P. Karlström, T. Cerratto-Pargman and R. Ramberg, "Designing for collaboration in intelligent computer assisted language learning," in *Proceedings of the IEEE International Conference on Advanced Learning Technologies*, Kaoshiung, Taiwan, 2005.
- [2] V. M. Holland and J. D. Kaplan, "Natural language processing techniques in computer-assisted language learning: Status and instructional issues," *Instructional Science*, no. 23, pp. 351–380, 1995.
- [3] J. Lantolf, "(S)econd (L)anguage (A)ctivity theory: Understanding second language learners as people," in *Learner Contributions to Language Learning*, M. Breen, Ed. Longman: Pearson, 2001, pp. 141–159.
- [4] P. Rabardel and Y. Waern, "From artifact to instrument," *Interacting with Computers*, no. 15, pp. 641–645, 2003.
- [5] M. Long, "Focus on form: A design feature in language teaching methodology," in *Foreign Language Research in Cross-Cultural Perspective*, K. de Bot, C. Kramsch, and R. B. Ginsberg, Eds. John Benjamins, 1991, pp. 39–53.
- [6] M. H. Long and P. Robinson, "Focus on form: Theory, research, and practice," in *Focus on Form in Classroom Second Language Acquisition*, C. Doughty and J. Williams, Eds. Cambridge University Press, 1998, ch. 3, pp. 15–41.
- [7] S. D. Krashen, *Principles and Practice in Second Language Acquisition*. Pergamon, 1982.
- [8] C. A. Chapelle, *Computer Applications in Second Language Acquisition*. Cambridge University Press, 2001.
- [9] L. Borin, "What have you done for me lately? The fickle alignment of NLP and CALL." in *NLP in CALL*, 2002, EuroCALL 2002 preconference workshop.
- [10] O. Knutsson, T. C. Pargman, K. S. Eklundh, and S. Westlund, "Designing and developing a language environment for second language writers," *Computers and Education*, 2006, in press.