LOGGING STUDENT ANSWER DATA IN CALL EXERCISES TO GAUGE FEEDBACK EFFICACY

AUTHOR FOWLER, ALISON M. L.

This paper describes the SCLIDE[1] web-based CALL system and its use on ab-initio Spanish modules offered to first year undergraduates at the University of Kent. The format of the data recorded by the system is presented and analysis of the data is used to measure the pedagogical efficacy of the error detection and feedback methods, along with the success of individual exercises.

The system has three principal sections: a question (and answer) generator, a language-independent error-detection module, and software for the overview and moderation of coursework by staff. Learners are presented with questions which elicit whole-phrase input. On submission answers are checked for errors (against a range of acceptable solutions) and immediate feedback is given. If errors are found, users are permitted a second attempt.

It is widely accepted that learners must be aware of linguistic form in order to acquire a second language successfully (Robinson, 1995). Consciousness raising, i.e. making learners aware of the discrepancies between their present state of knowledge and their goal state (James, 1999) is important in the presence of second-language errors and helps learners to notice important linguistic features in the target language (Dodigovic, 2005:86). One method of achieving this is to encourage learners to correct their own errors (Chapelle, 1998). The SCLIDE system's two-attempt method of question presentation (reinforced by basing marks on combined first and second attempt scores) motivates students to attend to their errors and consider how best to rectify them. Given the "whole-phrase" nature of the input, this necessitates a focus on form which has been shown to be important in language learning (Long, 1991).

Trials on the Spanish modules have involved (to date) five cohorts of students, and feature over 66,000 questions and 110,000 processed answers. Users are predominantly native English speakers aged between 18 and 20, largely computer literate, but often with little

previous experience of CALL. The modules are intensive, forming part of a two-year language programme, designed to help students who intend spending their third year in Spain acquire the necessary language skills.

The system was initially offered on a voluntary basis (in 2000-01), using translation-based exercises as the prescribed tasks. Just over half the students in that year's cohort made use of it. Between them they translated 7,451 phrases in just two terms and submitted many requests for additional exercises. The level of use massively exceeded expectations and following the success of the trial, the system was adopted as a compulsory part of the modules.

Table 1. Usage data

	00-01	**02-03**	**03-04**	**04-05**	**05-06**
Average users per lesson	*26*	67	66	21	33
Lessons available	*8*	9	16	12	12
Total questions attempted	*7,451*	17,553	29,797	6,902	11,494
Average min/max qu's required per lesson	*NA*	20-40	19-38	19-38	19-38
Average qu's per lesson per user	*36*	29	27	26	29

Each compulsory exercise features a minimum and maximum number of questions (differing between lessons). Students may complete the minimum number and still gain full marks if all questions are answered correctly. However if they improve as they progress through an exercise they may choose to attempt extra questions (up to the prescribed maximum). In practice students complete considerably more questions than the required minimum (see table 1). This is significant because voluntarily uptake of additional coursework is not the norm.

Error detection and feedback

Given the amount of CALL software available both on and off the web, there ought to be well established patterns of best-practice in relation to feedback generation, but unfortunately this is not the case (Bangs, 2003).

> *The need in CALL for error diagnosis and for both intelligent and real-time feedback is great. Reliable error-diagnosis systems would allow users/authors to overcome limitations of multiple choice questions and fill-in-the-blanks types of exercises and to present more communicative tasks to learners. (L'Haire & Vandeventer Faltin, 2003:482).*

Allowing whole-phrase input to language exercises can provide an excellent test of users' capabilities and a highly effective learning experience, however systems which permit this sort of input are still in the minority. Many exhibit significant disadvantages, with the quality of feedback leaving much to be desired. Manual encoding of feedback for reasonable-sized programs is a very costly activity (Bangs, 2003) and although parser-based courseware can automatically generate specific grammatical feedback, to do this requires over-generating rule systems which incorporate cases for all possible erroneous phenomena (Menzel & Schröder, 1999). This is tremendously difficult to achieve - Schulze (1999) notes that software which attempts to anticipate incorrect answers can only succeed if the answer domain is severely restricted. The number of rules required means that such systems may be particularly inefficient, and evidence from walkthroughs of CALL activities reveal that slow processing speed is problematic for users (Hémard, 2003).

If learners make unanticipated mistakes such systems are often incapable of providing appropriate feedback (Delmonte, 2003). Worse, they may fail to recognise correct but unusually phrased answers, and if user-input is confused such systems often fail to parse the input at all. Tschichold (2003:555) goes as far as to say "no parser at present is able to handle highly erroneous language to a degree that could make it useful for ICALL systems".

Establishing answer-appropriacy in whole-phrase input systems can be problematic, especially where answers are unusual. With parser-based systems it often constitutes a separate stage of processing, and some systems simply look for grammatically correct input, whether it answers the question or not. If a CALL system is to be used summatively then it must be completely consistent in its marking, never failing to parse input and always correctly gauging answer appropriacy.

SCLIDE's error-checking module uses sequence comparison to identify errors. Feedback is at a meta-level (independent of the grammar of the target language) and five types of error are identified:

- Partially incorrect items (erroneous spelling/conjugation)
- Totally incorrect items
- Incorrectly placed items
- Missing items
- Redundant items

Where an answer contains multiple errors, all errors are flagged. It is obviously important that users are not overwhelmed by large volumes of feedback. Concise and precise feedback is far more likely to be of use than several lines of detailed advice (Van der Linden, 1993). It has been suggested that exercises which permit the possibility of multiple errors should be avoided (Schwind, 1990). Nevertheless alerting learners to the presence of multiple errors in an answer does not necessarily require the display of discouraging amounts of text.

On the system's web-interface errors are displayed within the user's answer, using a different typographical notation for each error-type. Judicious use of colour and font draws learners' attention to simultaneous errors without being confusing. Black and white text is not the best medium to demonstrate the mark-up, but the following example gives a taste:

Target: Insistís que el niño vaya a la piscina
Student: Insistamos que niño va ir a la patinaje
Mark-up: Insist**amos** que [_] niño va__ <ir> a la **patinaje**

Students report finding the notation easy to understand – no problems have been encountered where exercises are well-designed and appropriate in their level of difficulty.

If sequence comparison is used for error-detection, feedback is always given – no matter how confused the input or how complex the linguistic structures involved. There is no need to write predictive grammars, and such systems are much less likely to fall over on unanticipated correct answers because question setters need only provide the (limited) range of possible legitimate solutions for individual questions. A further advantage of this method is that answer-appropriacy is automatically established as part of the error-detection process.

Being language-independent the error-detection routines are also extremely versatile; they can be used with many languages and this has important implications in terms of cost. One system will suffice for multiple language courses within a school or university department, and staff need not learn several different interfaces when creating lessons for various language units.

Data logging

The SCLIDE system records student answer-data using separate files for each exercise for each student. These files are not intended for viewing in their raw state by either student or staff users. The format of these internal files is as follows.

At the start of a user session, the date is recorded. Students are not required to complete an exercise in a single session and logging session-start data permits the calculation of the number of separate sessions users take to complete each exercise.

For each new question, its number is recorded, along with its time of display and the question itself (in the example this is an English phrase to be translated).

1 | 16:09:09 | You (plu, familiar) insist that the boy goes to the swimming pool

Following this the first - and if appropriate, second - attempted answer(s) are recorded, with their internal mark-up. Additionally, the submission time of each answer is recorded, plus its raw (un-moderated) mark:

Attempt 1 | 23:48:16 | Insist#a#m#o#s que ~el niño va#_#_ +ir a la *patinaje | 47
Attempt 2 | 23:48:38 | Insist#é#is que el niño vaya a la piscina | 94

The internal error-mark-up is not designed to be user-friendly in terms of legibility – it is translated into the more readable format when displayed on the web.

Finally the target answer(s) are logged:

Target | Insistís que el niño vaya a la piscina
Target | Vosotros insistís que el niño vaya a la piscina

Data analysis

The ability to record learners' answering patterns in detail permits the collection of large data sets, analysis of which can positively influence future software design and usage (Heift, 2004). Fulcher (2000) predicted that the next real focus of research regarding the

use of computers for language testing would be on the inferences that could be drawn from test scores. A major part of this project has been the investigation of marks awarded by the SCLIDE system to gauge whether learning is taking place effectively.

There is an obvious initial need to show that the linguistic forms targeted in exercises are at an appropriate level of difficulty for learners (Chapelle, 2001:80). Given the intensive nature of the modules these CALL exercises support, it is not possible to further burden students with pre- and post-tests to accompany every exercise. However, analysis of the average percentage of questions answered correctly on the first attempt, for the first third of each exercise (per student) gives a good indication of how challenging the material is proving. Figure 1 shows the percentages for the 12 exercises completed by the 2005-06 cohort of 33 students, and encompasses over 3,800 answers.

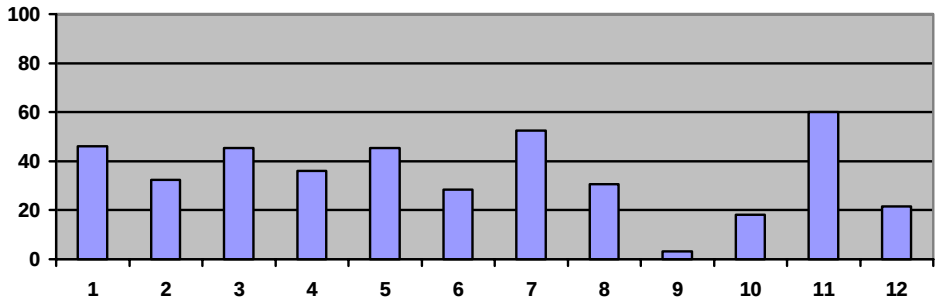

Figure 1. Average number of correct attempt-one answers (first 3rd of each exercise)

The difficulty level of these exercises obviously varies, but it is evident that in no case is an exercise so simple that users are consistently able to provide perfect answers to the initial questions.

It must also be shown that the exercises are not problematically difficult. The material is designed to be testing, so learners obviously will make mistakes in their first attempts, but it is abundantly clear from the data that there is almost always a significant improvement by attempt two. Figure 2 shows the score difference between first and second attempts, averaged over all students, for each of the 12 exercises from 2005-06. Some exercises are more successful than others, but it is obvious that there is an improvement in average score between the first and second answer attempts for every exercise. The chart encompasses all 11,494 answers to the 2005-06 exercises.

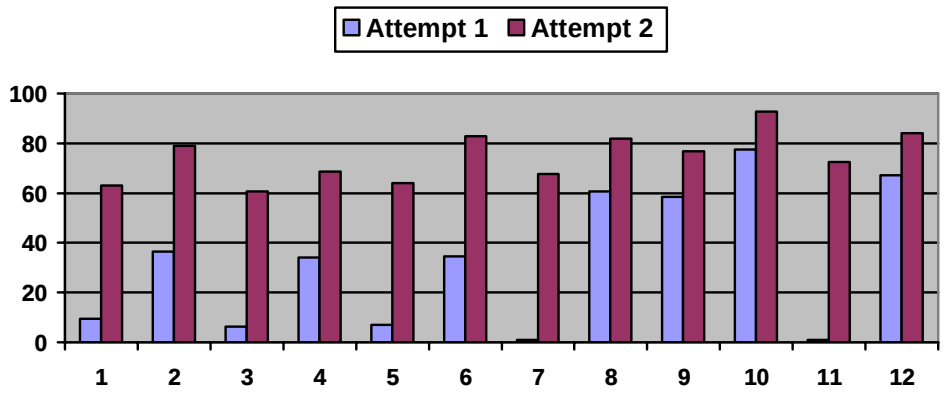

Figure 2. Average scores for 1st and 2nd attempts

As explained, the feedback the system produces is generic rather than specifically grammatical (since the latter approach would compromise the language-independent nature of the error-detection algorithms). There was initial concern that this sort of feedback might not provide enough detail to enable learners to understand and correct errors, but figure 2 shows that this is not the case.

The type of analysis performed shows that the feedback is effective in making users attend to immediate errors - however it does not prove the pedagogical efficacy of the means of exercise presentation. Evidence is needed that the learners are acquiring the target forms focussed on during the tasks (Chapelle, 2001:86). For this, every student answer file, for every exercise, has been split into three equal parts and all the part Is, part IIs and part IIIs per exercise have been examined as a whole. Figures 3 to 5 show the improvements averaged over all 12 exercises from 2005-06, under three different analysis categories.

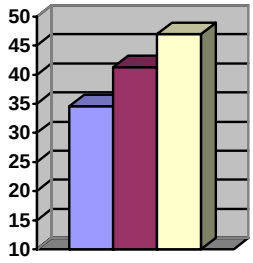

Figure 3. Correct 1st tries (as a %)

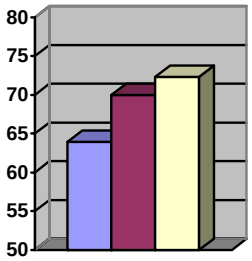

Figure 4. Question scores (as %)

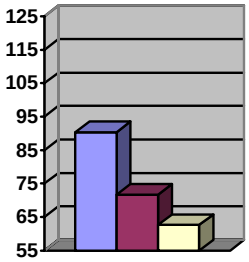

Figure 5. Thinking time (in seconds)

- Figure 3 shows the average percentage of questions answered correctly on the first attempt, in the 1st, 2nd and 3rd sections of all the student data files. This figure clearly rises as students progress through an exercise.

- Figure 4 shows the average question scores (taking into account the marks for both 1st and 2nd attempts) across the three sections. Again an obvious increase is evident.
- Figure 5 shows the average thinking time (in seconds) across the three sections. Here there is a clear decrease.

These figures show that over the course of an exercise students' answers become more accurate whilst requiring less formulation time. Most students take more than one session to complete an exercise, so this is not an effect of short-term memory. Furthermore, questions within exercises are presented randomly so order of presentation cannot influence marks. The results show that effective learning is taking place as students work through exercises, and is typical of the results achieved.

A further significant benefit of the types of analysis that can be performed on the logged data is simple identification of exercises which are not pedagogically effective. In 2003-04 an exercise on conditional clauses was offered. Despite the average question score being an acceptable 54.98%, it was obvious that it was not a successful task. Figures 6 to 8 show the results analyses for this exercise. The scales used in the axes are identical to those in figures 3 to 5 for comparison purposes.

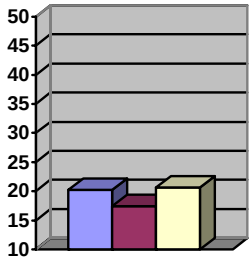

Figure 6. Correct 1st tries (as a %)

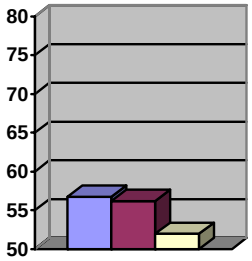

Figure 7. Question scores (as %)

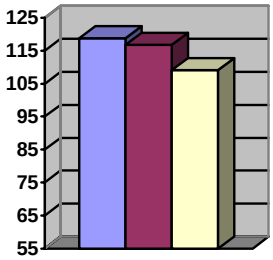

Figure 8. Thinking time (in seconds)

Thinking time for this exercise (figure 8) was considerably slower than the norm, and although it decreased over the course of the exercise, this was not to the extent normally seen. The average percentage of questions answered correctly on the first attempt (figure 6) remained low across the whole exercise, where normally some improvement would be expected, and most seriously, as students progressed their average question score (figure 7) actually dropped. Students later commented that this exercise was far too difficult and they simply lost motivation. The opportunity to perform this type of analysis permits staff to be much more proactive about exercise design and course development. It would be exceedingly difficult and time-consuming to perform the same sort of analysis with tutor-marked exercises.

Conclusions and future work

Pedagogically SCLIDE has been a success at the University and student feedback has been immensely positive. Responses to an anonymous survey during the system's first year of compulsory use revealed that the students found the system really straightforward to use (84%), felt the immediacy of the feedback was enormously beneficial (95%), found it very useful to be able to review their work post-submission (75%) and much preferred this method of assessment to the traditional pen and paper exercises (86%). Despite the potentially problematic whole-phrase input style, the marks awarded by the system are accurate and consistent, and are fed directly into University Exam Boards.

Given the language-independent nature of the error-detection algorithms, the system can be used to provide exercises in many languages. The feedback engine that forms the heart of the software works for any exercise-type where there is a definitive set of acceptable answers, thus source material for lessons can be tailored to any course the system is used to support and can be presented in a variety of formats. The system is applicable to many learning situations and from September 2006 will be undergoing new trials with GCSE-level students in a number of Kent schools.

REFERENCES

Bangs, P. (2003). Engaging the learner – how to author for best feedback. In U. Felix (Ed.), Language learning online – Towards best practice (pp. 81-96). Lisse: Swets & Zeitlinger.

Chapelle, C. (1998). Multimedia CALL: Lessons to be learned from research on instructed SLA. *Language Learning and Technology, 2(1)*, 22-34. Available: http://llt.msu.edu/vol2num1/.

Chapelle, C. (2001). Computer applications in second language acquisition: foundations for teaching, testing and research. Cambridge: Cambridge University Press.

Dodigovic, M. (2005). Artificial intelligence in second language learning: raising error awareness. Clevedon: Multilingual Matters Ltd.

Delmonte, R. (2003). Linguistic knowledge and reasoning for diagnosis and feedback. *CALICO Journal, 20(3)*, 513-532.

Fulcher, G. (2000). Computers in language testing. In P. Brett, & G. Motteram (Eds.), A special interest in computers – Learning and teaching with information and communications technologies (pp. 93-107). Whitstable: IATEFL.

James, C. (1999). Language Awareness: Implications for the Language Curriculum. *Language, Culture and Curriculum, 12(1)*, 96-116.

Heift, T. (2004). Corrective feedback and learner uptake in CALL. *ReCALL*, *16(2)*, 416-431.

Hémard, D. (2003). Language learning online: designing towards user acceptability. In U. Felix (Ed.), Language learning online – Towards best practice (pp. 21-42). Lisse: Swets & Zeitlinger.

L'Haire, S., & Vandeventer Faltin, A. (2003). Error diagnosis in the FreeText project. *CALICO Journal*, *20(3)*, 481-495.

Long, M.H. (1991). Focus on form: a design feature in language teaching methodology. In K. de Bot, R. Ginsberg, & C. Kramsch (Eds.), Foreign language research in cross-cultural perspective (pp. 39-52). Amsterdam: John Benjamins.

Menzel, W., & Schröder, I. (1999). Error diagnosis for language learning systems. Language Processing in CALL, *ReCALL (special edition, May 1999)*, 20-30.

Robinson, P. (1995). Attention, memory and the 'noticing' hypothesis. *Language Learning*, *45(2)*, 283-331.

Schwind, C.B. (1990). An intelligent language tutoring system. *International Journal of Man-Machine Studies*, *33*, 557-579

Schulze, M. (1999). From the developer to the learner: describing grammar – learning grammar. *ReCALL, 11(1)*, 117-124.

Tschichold, C. (2003). Lexically driven error detection and correction. *CALICO Journal*, *20(3)*, 549-559.

Van der Linden, E. (1993). Does feedback enhance computer-assisted language learning. *Computers & Education*, *21(1-2)*, 61-65.