An Analysis of Learning Difficulties Experienced by First Time Spreadsheet Users

Tanja Reinhardt
Nelishia Pillay

Abstract

Spreadsheets have become a routine application in most organizations and universities. Consequently, students are required to learn spreadsheet applications such as Microsoft Excel. The learning of spreadsheets is often accompanied by problems related to spreadsheet applications and their mathematical content. The EXITS (Excel Intelligent Tutoring System) research project aims to develop a Microsoft Excel tutor that helps students to overcome their learning difficulties. The paper reports on the first phase of our EXITS research project. We firstly identify barriers that may prevent a student from successfully using a spreadsheet program. We then analyse the different types of errors students make and identify their causes. We also classify the errors. This classification will form the basis for an error library that is required for our student model. At the end of the paper we give an example of how our model will be used for student modelling purposes.

1. Introduction

The main task of a spreadsheet program is to analyse and manipulate numbers. For example, numbers can be entered into cells and mathematical calculations can be performed to solve almost any kind of numeric problem. It also allows the user to represent the numerical data in a more accessible form as charts. Although spreadsheets have been around for about 20 years, novice and expert spreadsheet users still have difficulties when using spreadsheets (Tukiainen 2000: 247; Brown 1987: 258).

*Alter*nation 12.1b (2005) 429-453 *ISSN 1023-1757*

When users start working with spreadsheet programs, they have to cope with two challenges. One is to master the application program and the other is to understand the mathematical concept of spreadsheets. We have found that computer literacy students experience both spreadsheet and mathematical learning difficulties when using a spreadsheet program for the first time. Unfortunately, the majority of available spreadsheet tutorials focus only on spreadsheet concepts, (e.g., how to use a SUM function or how to create a chart), but do not provide mathematical assistance. Thus, these packages will not be able to assist students in performing more complex spreadsheet calculations or overcoming learning difficulties associated with mathematical and logical thinking. Each student has individual needs and problems. Therefore our aim is to develop a software program to provide students with individualised tuition to overcome these problems. We have decided to develop an intelligent tutoring system (ITS), namely EXITS (Excel Intelligent Tutoring System), to provide personal instruction and error-specific feedback for a variety of problems a student may experience.

This paper reports on the first phase of our EXITS research project. The aim of this phase was twofold. For student modelling purposes, barriers that may prevent a student from successfully using a spreadsheet program were identified. Secondly, a survey of the different types of problems experienced by students was conducted. This information will be used to guide the error feedback for EXITS.

2. An Overview of Student Modelling As Part of an Intelligent Tutoring System

Intelligent tutoring systems are software programs that provide individualized tutoring or instructions. They have the ability to present the teaching material in an interactive environment and provide personal instruction and error specific feedback (Mandridou 2003: 157). Compared to other computer-based training techniques ITS systems assess each learner's actions within these interactive environments and develop a model of their knowledge, skills and expertise. Based on the learner model, the ITS system aims to simulate a human tutor who can adapt his or her instructional strategies in terms of both the content and the style, and provide explanations, hints, examples, demonstrations, and practice problem- solving as needed. The widespread use of ITSs has been impeded by the high costs associated with developing these systems (Mizoguchi 2000: 107).

An intelligent tutoring system usually consists of four interconnected core modules: A domain/ expert module, which defines the expert knowledge in an area and the problem-solving characteristics (Serengul 1998). The pedagogy/-tutoring module contains teaching strategies and teaching instructions. The communication module provides an interface, which controls the interaction among student, teacher and knowledge, and the student module captures the student's knowledge, which is used to guide the feedback and information presented (Baffes 1996: 403). In order to provide individualized tuition to students, one of the main tasks of an ITS is to construct a student model, which is used to guide the feedback and information in response to the student's performance.

A number of different student modelling strategies have been developed over the years. One of the approaches is the perturbation student model (Burton 1982: 157). This model assumes that the student's knowledge is a subset of the expert knowledge and that this student knowledge may contain sets of errors or misconceptions, which must also be represented. To build such a model the system requires a library of mistakes (bug library) students are likely to make. The aim of this is to grow the student's subset of the expert knowledge while eliminating the errors and misconceptions in the student's knowledge.

This paper is addressing the student modelling aspects by identifying factors that influence student performance negatively and analyse errors made by students.

3. Course Background and Experimental Methodology

The spreadsheet course is part of a Computer Literacy course that is held every semester at the University of KwaZulu-Natal. The spreadsheet section usually consists of about 12 spreadsheet lectures, one 1½ -hours and two 3-hour practical sessions. The software package used to teach the students is Microsoft Excel 2000.

A questionnaire was administered to Computer Literacy students at the University of KwaZulu-Natal during a Microsoft Excel practical session to collect data for the student module. We used a practical to hand out our questionnaire because it allowed the students to respond to problems experienced, something which cannot be achieved in a test environment.

In addition of formatting tasks, a multiplication formula, the SUM, AVERAGE, MIN, MAX and COUNT function, and a percentage calculation were included. Functions like SUM, AVERAGE etc. are predefined functions that Microsoft Excel uses to perform calculations by using specific values, called arguments, in a particular order or structure. For example, the SUM function adds together values or ranges of cells, and AVERAGE finds the average of a group of numbers. Both the multiplication formula and the percentage calculation require the use of cell references, which refer to a place in a worksheet where Microsoft Excel picks up the value. Brown (1987: 258) showed that 70% of the reported errors were errors in formulae. From our experience we believed that the students had more problems in dealing with formulae using cell references and functions than with simple formatting aspects.

The aim of the questionnaire was to get information about the students' backgrounds as well as information about problems experienced during the practical. With the analysis of the questionnaire we tried to identify the barriers that may prevent a student from successfully using spreadsheets.

To provide individualized feedback on the errors made by the student, we needed to find a methodology to assess these errors. Our aim was to identify the problems students experience and to look at the kind of errors the student made without the influence of a human tutor, which would obviously affect the result. This study was therefore done during a spreadsheet test with the purpose of using the outcome as a basis to provide necessary error feedback for the students and to create our error library. Although we are mainly concentrating on errors in formulae and functions in this paper, we will also handle other errors in our student model, for example, errors when creating a chart or sorting data.

Not all the students that attended the test attended the practical and vice versa. Therefore the size of our sample groups varied. A total of 151 students attended the practical and returned the questionnaire and a total of 165 students attended the test. 121 students participated in both test groups.

4. Analysis of Learning Problems and Identification of Problem Groups for Student Modelling Purposes

4.1. Student Background

In order to identify barriers and the relationship between student background and performance, we investigated the background of each student to see if the problems experienced by the students might be a lack of computer knowledge (e.g. that students may have difficulties at all to use a computer) or a lack of mathematical knowledge (e.g. that students struggle when dealing with mathematical calculations in spreadsheets). We investigated whether a correlation exists between the understanding of spreadsheets and the mathematical knowledge as well as computer skills. We also included the language aspect in our survey.

A report on problems in Mathematics education at the former University of Durban-Westville discussed the language issue (Paras 2001:66). They stated that students were not sufficiently exposed to the mathematical concepts in English and most of them struggle when they come to university because the instructions are given in English.

When second-language learners do not understand the English instructions they immediately switch to mother tongue. From interviews with Computer Literacy tutors we also found that if a tutor explained something to one student, this student will explain it to the other students in his home language. Research at the University of Port Elizabeth (South Africa) also showed, that apart from study habits and Mathematics pre-knowledge, the English language proficiency can be used to predict academic performance of students in subjects that require the ability of mathematical thinking (Koch 2001: 138). When looking at the language aspect we didn't test the English language proficiency because it would have been too time consuming for the students to fill in the questionnaire and to do an English language proficiency test. Thus we used the student's first language as an indicator.

Figure 1 shows that the background of our survey group was very mixed in terms of their first language, Mathematics level and computer experience. The majority of the students (53%) had English as their first language followed by Zulu (20%) and Setswana (9%) (Figure 1(a)).

We also asked the students about computer experience before they started the Computer Literacy course. The computer skills of the test group (Figure 1(b)) was very mixed and ranged from students with "Computer Science in matric (school-leaving exam), including programming experience" to students with "no computer experience before the course". About 10% of the students stated that they worked with Microsoft Excel before and 48% of the students had no experience with a computer before they started the course. We will refer to the latter group as "novice users" in this survey.

The breakdown of the mathematical experience is shown in Figure 1(c). 43% had higher-grade 12 Mathematics and 29% had standard grade 12 Mathematics. The mathematical education of 17% of the students was below standard-grade 12 Mathematics.

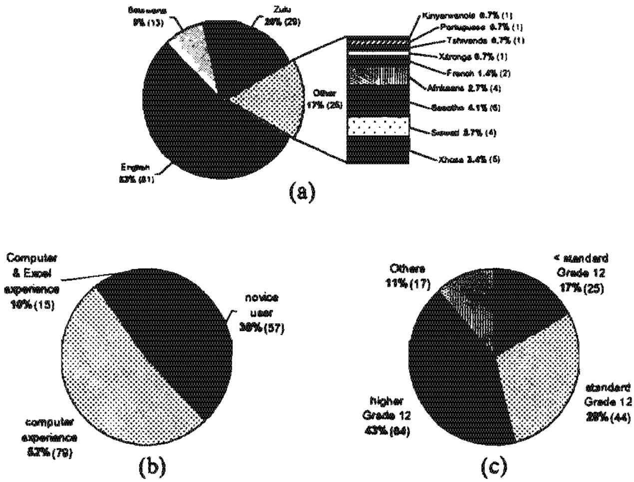


Figure 1: Background of survey group. (a) First language (N=151), (b) Mathematics level (N=150) and (c) Computer experience (N=151).

4.2. Analysis of Problems vs. Student Background

The objective was to determine whether an association exists between the number of problems a student had when doing the practical and their mathematical education, their computer experience and their first language. In order to test these hypothesis we used the Pearson chi-square ($\chi 2$) test, which is often used as a significance test to determine if there is a dependence of two qualitative variables.

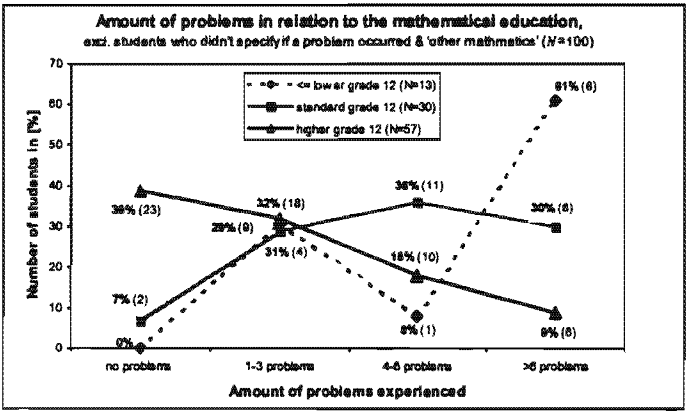
Table 1: Number of problems in relation to mathematical education, first language and computer experience

N	Hypothesis	X^2	df	P	V
100	Mathematical education	30.3	6	P<0.01	0.71
112	First language	16.9	3	P<0.01	0.39
100	Mathematical education/ First language	48.5	15	P<0.01	0.31
112	Computer experience	7.8	3	P<0.055	0.07

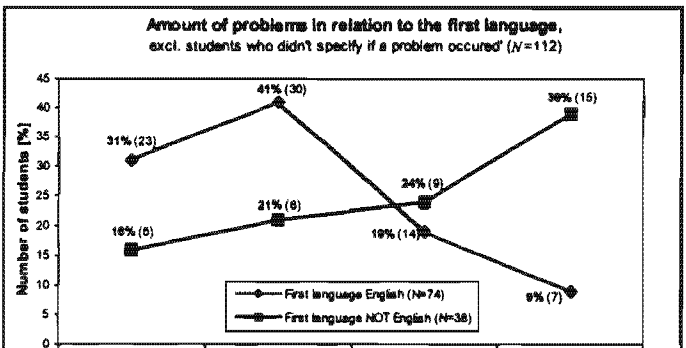
Table 1 shows the preliminary analysis of the demographics with the following results: the mathematical education is strongly associated (V=0.71) with the number of problems experienced with Microsoft Excel. This is also displayed in Figure 2(a), which illustrates the number of problems in relation to the mathematical education. 57 students considered had higher grade 12 Mathematics, 30 students had standard grade 12 maths and 17 had lower grade 12 Mathematics or lower. If we look at the different groups we can see that from the 57 students with higher-grade 12 Mathematics most of them (39%) had no problems with the practical and only 9% of this group had more than 6 problems with the practical. If we compare the group with lower grade Mathematics and below lower grade Mathematics, none of the students in the group had "no problems" and the majority of this group (61%) had more than 6 problems. In general, we observed that students with higher mathematical qualifications had fewer problems doing the Microsoft Excel practical than students with a low level mathematical background. This is what we expected, because spreadsheets in general are mathematically oriented programs and therefore students with a poor mathematics background may find it difficult to form equations and understand the concept of spreadsheet functions. This finding is also reflected in Figure 3 which shows that most of the problems occur when students were requested to write an equation.

Table 1 illustrates that there is a moderate association (V=0.39) between first language English and the number of problems experienced with Microsoft Excel, which is also displayed in Figure 2(b). In this figure, we can see that the number of students with more than 6 problems is larger for students with the first language not being English (39% of the first language not English group) than for student with first language English (9% of the first language English group). Also the number of students with fewer than 4 problems is proportionally higher for students with the first language English (72%) compared to 37% observed for students with the first language not being English.

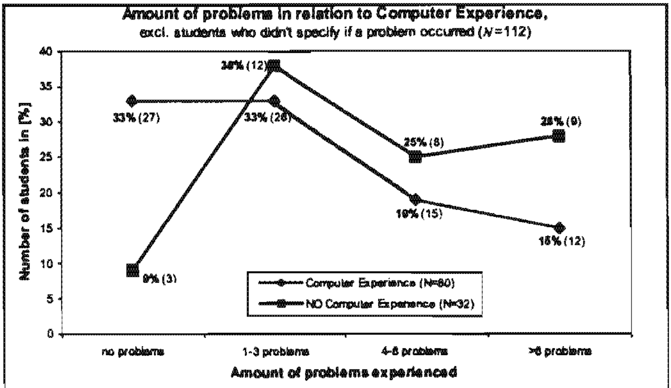
However, there is a correlation between the students' mathematical ability and language proficiency as indicated by a V of 0.31. Thus, there could possibly be an indirect correlation between success in the spreadsheet course and language proficiency due to the methodology employed to teach Mathematics at second language schools. Alternatively, language proficiency could have a direct impact on the students' success in Mathematics.



(a)



(b)



(c)

Figure 2: Number of problems in relation to (a) Mathematical education, (b) First language and (c) Computer experience

Students with no computer experience appear to have had more problems than those students that have previously used a computer. It seems that there is a correlation between computer experience and the number of problems experienced. But if we look at the calculated correlation factor in table 1 we can deduce that there is only a very weak association (V=0.07) between computer experience and the number of problems students' experience. This association is not as strong as the language and mathematical correlations.

4.3. Discussion on Problem Groups

The statistical tests indicated that a significant relationship exists between mathematical education and the number of problems a student experienced. We observed that students with a lower level of mathematical education experienced more problems during their practical than students with a more extensive mathematical background. The questionnaire also revealed a significant relationship between the language proficiency and the number of problems experienced. Students whose mother tongue was not English experienced more problems during their practical session. The relationship between a computer experienced user and the number of problems is on the other hand not as significant as the mathematical background and the first language. Based on this survey we can now identify two barriers that may prevent a student from successfully using a spreadsheet program. One is the mathematical literacy, and the other is the first language.

Mathematics is used as an indicator of both the student's mathematical knowledge and problem-solving ability. If the mathematical ability of a student is low and he/ she cannot perform calculations, that student will also experience problems performing calculations in Microsoft Excel. If the problem solving ability of students is low we can predict that those students will experience difficulties with logical and complex problems like sorting, if statements, financial functions, and absolute cell addressing. They can be expected to experience fewer problems with simple functions like the SUM function, or simple calculations.

5. Analysis of Spreadsheet Errors

For our model it was necessary to identify both the errors students make when they solve a problem and the causes of these errors. The analysis of the causes will help us to generate the error library and to guide the feedback in our model. Once we know what the problems are we will be able to provide individual instruction and error-specific feedback.

Most of the error classifications described in the literature only provided a general overview of the errors made and thus lack the details needed for our error library (Panko 1997: 21; Tukiainen 2000: 247).

5.4. Errors Using Formulae and Functions

To reduce complexity and to structure our error library we tried to find out if students frequently made similar errors in different formulae and functions.

We looked at the different types of errors students made, and we can make some generalisations (Figure 3, Table 2 in Appendix A). We classified three error categories:

1. Errors that apply to any functions and formulae. For example, errors where students were not able to write down a solution at all (GE1).
2. Errors that occur in formulae and functions with a similar syntax. A lot of functions (predefined formulae that perform specific calculations), like the SUM or AVERAGE function use the same syntax (=function_name(argument)) to perform a calculation. Errors that occurred in any simple function were errors related to the argument (GE4).
3. Errors that only occur in specific formulae or functions. For example if students added individual cells instead of using the SUM function (SE1).

We defined errors that belong to category 1 and 2 as general errors and those errors that belong to category 3 as specific errors. Table 2 (in Appendix A) gives an overview of the different error types with examples of errors student made. 'GE' indicates a general error and 'SE' a specific error.

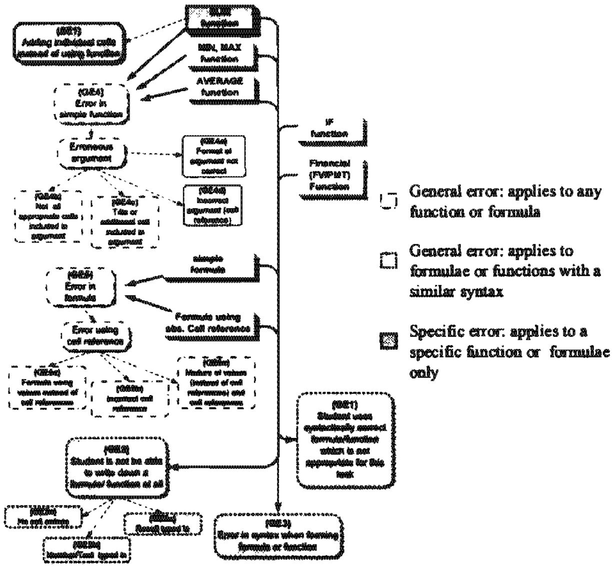


Figure 3: Overview of general errors (GE) and specific (SE) errors using the SUM function

5.5. Classification of Error Categories in Relation to Student Knowledge

To provide error- specific feedback and assistance for our ITS it was necessary to identify the causes of these errors.

If we look at all the errors made when students had to use formulae or functions, we can identify two major causes. Firstly, students made errors because they lack the knowledge of Microsoft Excel concepts. For example, students didn't know how to form a correct argument in a function. Also students often didn't understand some of the concepts like the use of absolute cell references and relative cell references. In our study, we refer to these errors as "spreadsheet errors". For these errors, the assistance will concentrate on the spreadsheet concepts or the syntax of formulae and functions.

Other errors were related to mathematical problems (e.g., the students where not able to set up an equation to calculate a percentage). This also includes logical errors, such as, the use of wrong units when using a financial function. Here we need to assist the student to understand the mathematical concept of that task. For some errors we are not able to distinguish whether the error is due to the lack of spreadsheet knowledge or due to the lack of mathematical/ logical understanding.

If we look, for instance, at errors where students didn't enter anything into the cell (GE2a), we cannot decide whether the student is not able to construct a formula with cell addresses (which would be a "spreadsheet error") or whether the student does not know how to perform the calculation at all (which would be a "mathematics error"). Therefore our feedback needs to cater for both aspects, to assist the student with the mathematical feedback and to assist the student with spreadsheet knowledge.

As mentioned earlier we identified two barriers, a mathematical and a language barrier. So far we have concentrated only on the mathematical/ spreadsheet side and didn't take the language aspect into consideration. From our experience as Computer Literacy tutors we identified some errors that can be related to language problems. For example, if somebody enters a number, which is not the calculated result, as an answer into the cell, there are basically two possibilities as to why the student failed to solve the problem. One possibility is that the student doesn't know how to perform the required calculation, the other is that he/she doesn't know how to apply the required spreadsheet concept. But another possibility is that the student didn't understand or misinterpreted the problem that he was asked to solve due to language problems.

5.6. Error Statistics

An analysis of the distribution of spreadsheet and mathematics errors in formulae and functions revealed that the majority of errors in the SUM function are related to spreadsheet problems (Figure 4a). From the type of errors the student made we can deduce that, although they made errors, they understand the mathematical concept of adding numbers. 64% of the errors students made were due to the fact that they added single cell references (SE1) and 7% of the students typed the result of the sum calculation into the cell (GE2c). Therefore, we can assume that most of the students who made an error using the SUM function didn't understand the concept of the SUM function.

The errors made when using the AVERAGE function were also mostly (67%) spreadsheet errors (Figure 4b). 42% of the students making errors using the AVERAGE function were able to carry out the calculation. The number of spreadsheet errors made when using the MIN or MAX function (36%) is lower than the number of spreadsheet errors in the SUM and AVERAGE function (Figure 4c).

As soon as a function requires more advanced mathematical/ logical knowledge we observed that the proportion of pure mathematical/ logical errors increases, for example, when the students have to deal with the IF-function (27%) or with the PMT/ FV-function (31%). About a third of the errors made when using the IF function were related to spreadsheet errors. It is obvious that some students know the Mathematics and the logic of the task, but they were not able to construct the required function correctly. For the PMT and the FV function we couldn't identify any "pure" spreadsheet errors. The majority of errors that occurred can be related to both categories.

In simple formulae 46% of the errors made were related to spreadsheet errors. Most of the spreadsheet errors were due to cell referencing. This is even more obvious when we look at absolute cell referencing. 58 % of the errors (GE5a, GE2c, SE5) made were due to the fact that students had difficulties with the concept of absolute cell addresses.

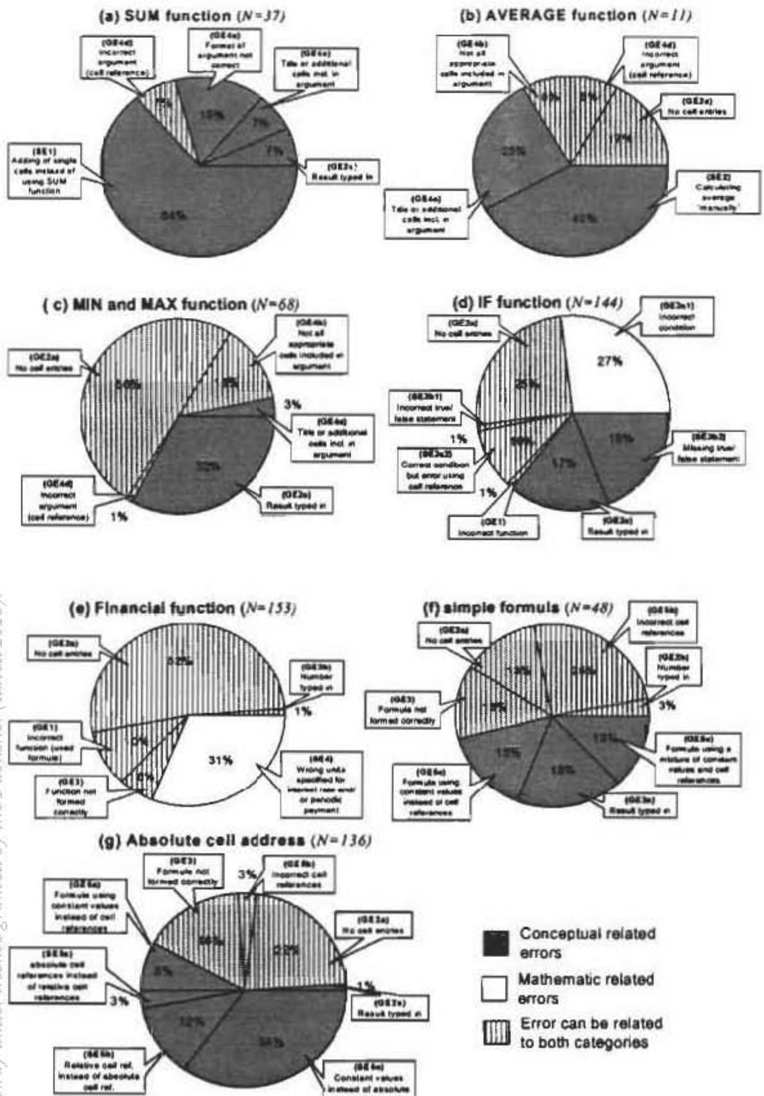


Figure 4: Distribution of spreadsheet and mathematical errors in (a) SUM-, (b), Average- (c) Min/Max-, (d) IF-, (e) PMT/ FV-function, (f) simple formulae and (g) formulae using absolute cell addressing. (Error keys see Table 2 in Appendix A).

6. Feedback Guidance Using the Buggy Model – An Example

Based on the different error categories we are now able to generate a buggy model for our ITS. The classification of general and specific errors allows us to structure the error library and to reduce the complexity of the library. The analysis of spreadsheet errors in terms of mathematical and spreadsheet knowledge will help us to provide the necessary assistance for the students and to guide the feedback in our buggy model. The investigation of student barriers also revealed that it is necessary to provide additional help for students with language problems, which needs to be implemented in our system.

Figure 5 shows the general concept of our buggy model. To demonstrate the operation of our buggy model we use a problem involving the SUM function as an example. The system generates a SUM function problem ("*Calculate the number of tickets sold*") that the student has to solve. Our model will generate problems with different levels of difficulty. These levels will depend on knowledge of the individual student. The student types his solution into the cell ("*1675*") and the system compares this string with the solution (*"1675" ζ "=SUM(C1:E5"*). If the student solves the problem he will get positive feedback. In this example the student made an error. The program now analyses and classifies the error to decide on a response strategy.

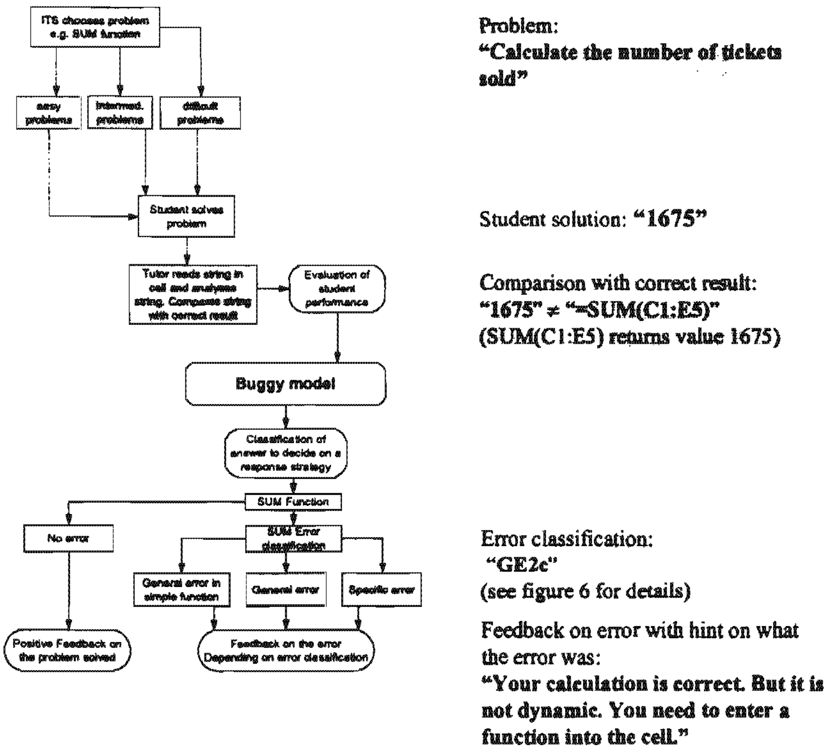


Figure 5: Flow chart of feedback module with SUM function example

Figure 6 illustrates in detail how the buggy model will analyse the error and provide feedback to the specific problems using a decision tree. The student's solution to this problem is a number, but not the required calculation that uses the SUM-function. The decision tree identifies the error as a *GE2c error* (the student manually calculated the result and typed it in). The feedback presented by the system is similar to a response from a human tutor to this problem. The response that the system will provide is: *"Your calculation is correct, but it is not dynamic. You need to use a function for the calculation."* The feedback gives the student an indication of what his error is *"Your calculation is correct, but it is not dynamic"* and also provides the student with additional information that his answer was not correct because he didn't use a function. "*You need to use a function for the calculation.*"

The student has the possibility to correct his error and to submit his solution again. If this solution is correct he will continue with another problem, otherwise the system will provide additional feedback on the problems the student experiences.

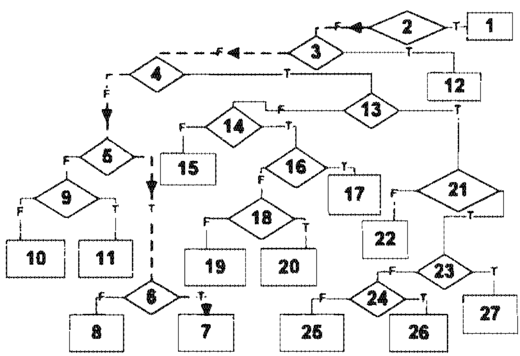


Figure 6: Error decision tree for feedback in SUM function

1 Correct solution
2 Check if "=SUM(cell range)" is correct
3 *GE2a*, Check if cell is empty
4 *GE3*, Check if function starts with "="
5 *GE2*, Check if entry is a number
6 Check if number = "result"
7 *GE2c*, "Your calculation is correct. But it is not dynamic. You need to enter a function into the cell."
8 *GE2b*, "You entered a number into the cell. Did you place your result in another cell? Or do you want me to" 1) "explain the task", 2) "explain the syntax of the SUM function"
9 Check if cell entry starts with "SUM"
10 "You entered a text into the cell. Did you place your result in another cell? Or do you want me to" 1) "explain the task", 2) "explain the syntax of the SUM function"
11 *GE3*, "You are on the right way. Check the syntax of your function if something is

missing."
12 *GE2a*, "You didn't fill anything into the cell. Did you place your result in another cell? Or do you want me to" 1) "explain the task", 2) "explain the syntax of the SUM function"
13 Check if function-name ="SUM"
14 Check if function has syntax "=...+...+"
15 *GE1*, "You used an incorrect function/formula" Do you want me to" 1) "explain the task", 2) "explain the syntax of the SUM function"
16 Check if function adds all cells in cell range
17 *SE1*, "Your calculation is correct. But it is better to use a function instead of a formula."
18 Check if formula adds cell references
19 *GE5a* and *GE5c*, "You should use a function and you should use cell references instead of values"
20 *GE5b*, "You should use a function and you should also check on cell references"
21 Check if cell ref has format "cellref : cellref"
22 *GE4a*, "You used the correct function. But you should check on the format of your argument."
23 Check if stud_cell range = cell range - cells
24 Check if stud_cell range = cell range + cells
25 *GE4d*, "You used the correct function. But your cell references are not correct." Do you want me to" 1) "explain the task", 2) "explain the syntax of the SUM function"
26 *GE4c*, "You used the correct function. But you should check if you included to many cells into your argument."
27 *GE4b*, "You used the correct function. But you should check if you included all the cells into your argument."

7. Conclusions

The main aim of this paper was twofold. For student modelling purposes, barriers that may prevent a student from successfully using a spreadsheet program were identified. Secondly, a survey of the different types of problems experienced by students was conducted. This information forms the basis of our buggy model.

We have demonstrated how the background of the student in terms of English language proficiency and mathematical knowledge can influence the performance in Microsoft Excel spreadsheets.

The study also showed that most errors occurred in areas where students had to use formulae and functions that require logical or mathematical thinking, or an advanced understanding of concepts, such as the use of absolute cell addresses.

The errors that the students made can be classified as spreadsheet errors, or mathematical-/ logical errors. Apart from the two error categories "mathematical/logical" and "spreadsheet" we could also distinguish between general (GE) and specific (SE) errors.

We implemented the result of this study in our buggy model and are now able to provide error- specific feedback and support.

Bibliography

Baffes, PT & RJ Mooney 1996. A Novel Application of Theory Refinement to Student Modeling. Proceedings of the Thirteenth National Conference on Artificial Intelligence, Portland Oregon: 403-408.

Brown, PS & JD Gould 1987. An Experimental Study of People Creating Spreadsheets. *ACM Transaction on Office Information Systems* Volume 5, Issue 3: 258-272.

Burton, RR 1982. Diagnosing Bugs in a Simple Procedural Skill. In DH Sleeman & JS Brown (eds.) *Intelligent Tutoring Systems*. London, UK: Academic Press: 157-184.

Koch, E & M Snyders 2001. The Effects of Video Supplemental Instruction on the Academic Performance in Mathematics of Disadvantaged Students. *South African Journal of Higher Education* Volume 15 Issue 1:138-146.

Mizoguchi, R & J Bourdeau 2000. Using Ontological Engineering to Overcome Common AI-ED Problems. *International Journal of Artificial Intelligence* Volume 11, Issue 2: 107 -121.

Moundridou, M & M Virvou 2003. Analysis and design of a web-based authoring tool generating intelligent tutoring systems." *Computers & Education* Volume 40: 157-181.

Panko, RR & RP Halverson 1997. Are Two heads Better than One? (At Reducing Errors in Spreadsheet Modelling?). *Office Systems Research Journal* Volume 15 Issue 1: 21-32.

Paras, J 2001. Crisis in Mathematics education. Student failure: challenges and possibilities. *South African Journal of Higher Education* Volume 15 Issue 3: 66-73.

Serengul, GV 1998. Intelligent Tutoring Systems Architecture. Accessed on 20 May 2004 http://www.cs.mdx.ac.uk/staffpages/serengul/Intelligent.Tutoring.System.Architectures.htm.

Tukuiainen, M 2000. Uncovering Effects of Programming Paradigms: Errors in Two Spreadsheet Systems. 12th Workshop of the Psychology of Programming Interest Group, Cozenza Italy: 247-266.

Appendix

Table 2: Overview of error types and error examples in formulae and functions (*GE* general-, *SE* function/ formula- specific errors)

	Error type	Error example
GE1	Student uses syntactically correct formula/ function, which is not appropriate for this task.	Student used the future value function although the calculation required the periodic payment function
GE2	Student is not able to write down a formula/ function at all.	
GE2a	No cell entries.	Student didn't enter any formula/ function into the cell at all.
GE2b	Number typed in.	Student entered an arbitrary number, which was not the calculated result.
GE2c	Result typed in.	Student entered the result in form of a number or a text into the cell.
GE3	Error in syntax when forming formula or function.	Student left out brackets in a formula or forgot to place an equal sign at the beginning of a formula or function.
GE4	Error in simple function – Erroneous	

	argument	
GE4a	Format of argument not correct.	Student used a formula as the argument instead of cell references.
GE4b	Not all appropriate cells included in argument.	Student didn't include cell references that were required for the correct calculation.
GE4c	Title or additional cells included in argument.	Student included the title or additional cells at the end or at the beginning of the necessary block of cells.
GE4d	Incorrect cell reference (argument)	Student referred to wrong cell references and therefore calculated something completely different from what was required.
GE5	Error in formula – error using cell references	
GE5a	Formula using values instead of cell references.	Student applied the correct equation but instead of using cell references that allows the calculation to be dynamic they typed in the constant value found in that cell.
GE5b	Incorrect cell reference.	Student referred to a cell that contained text or to a cell, which was not relevant for the calculation.
GE5c	Mixture of values (instead of cell references) and cell references.	Student used a cell reference for variable1 and a constant value for variable2 in the same formula although he could have used a cell reference for variable 2.
SE1	Adding individual cells instead of SUM function	Student added individual cells instead of using the SUM function (e.g. =A2+A3+B2+B3 instead of =SUM(A2:B3))
SE2	Calculating average manually	
SE2a	Using SUM function.	Student used SUM function instead of AVERAGE function for the average calculation (e.g. =SUM(A2:A6)/5)).
SE2b	Using formula.	Student used formula instead of AVERAGE function for the average

		calculation (e.g. =(A2+A3+A4+A5+A6)/5).
SE3a	IF-function - Condition	
SE3a1	Incorrect condition	Student used an incorrect condition.
SE3a2	Correct condition, but error in cell reference	Student was supposed to use absolute cell reference in condition but used relative cell reference. Student used an incorrect cell reference in condition.
SE3b	IF-function – True/ False statement	
SE3b1	Incorrect statement.	Student used the same statement for the true and the false statement.
SE3b2	Missing statement.	Student left out false statement.
SE4	Financial function – Erroneous argument	
SE4a	Wrong unit specified for interest rate.	Student used yearly interest rate whereas a monthly interest rate was required.
SE4b	Wrong unit specified for time period.	Student used yearly periodic payment whereas monthly periodic payment was required.
SE5	Formula – Absolute cell reference	
SE5a	Values instead of absolute cell reference.	Similar to GE5a. Student used a constant value instead of an absolute cell reference.
SE5b	Relative cell reference instead of absolute cell reference.	Student used relative cell reference whereas an absolute cell address was required
SE5c	Made cell reference absolute	Student confused the concept of absolute and relative cell addresses and used an absolute cell address whereas a relative cell address was required.

Authors' Contact Details

Tanja Reinhardt (reinhardtt2@ukzn.ac.za)
Nelishia Pillay (pillayn32@ukzn.ac.za)
School of Computer Science
University of KwaZulu-Natal, Durban, South Africa