American Journal of Educational Research, 2018, Vol. 6, No. 2, 108-116
Available online at http://pubs.sciepub.com/education/6/2/3
©Science and Education Publishing
DOI:10.12691/education-6-2-3

Translating Numbers into Feedback: Providing Students with Automatically Generated Feedback

Sebastian Hedtrich, Nicole Graulich*

Institute of Chemistry Education, Justus-Liebig-University, Gießen, Germany
*Corresponding author: Nicole.Graulich@didaktik.chemie.uni-giessen.de

Abstract Universities nowadays are confronted with the challenge of offering students sufficient formative feedback about their learning progress. Undergraduates in particular struggle in this almost impersonal learning situation, in contrast to their experience at school. Web-based trainings often do not allow a comparable formative feedback from a teacher, as current Learning Management Systems only offer basal feedback mechanisms. The Learning Management System Analyzation Kit and the Easy Snippet Feedback Edit are software tools that have been developed to overcome this shortcoming. The intuitive design of the software allows teachers to develop small feedback generating programs, which use the educational data of EDM solutions, raw data of the LMS, or just teachers' individual records to compose individual formative feedback messages for all students. Initial evaluations show that students appreciate this new type of feedback, and that even inexperienced users can easily operate the software.

Keywords: *blended learning, automated feedback, LMS, distance learning*

Cite This Article: Sebastian Hedtrich, and Nicole Graulich, "Translating Numbers into Feedback: Providing Students with Automatically Generated Feedback." *American Journal of Educational Research*, vol. 6, no. 2 (2018): 108-116. doi: 10.12691/education-6-2-3.

1. Introduction

Feedback is one of the most influential factors supporting a student's learning [1]. Appropriate formative feedback can help students to refine their calibration [2]; therefore, it must consider achieved or unachieved learning goals, besides incorporating proactive suggestions for the future and help for improvements [3]. In contrast, feedback addressing performance issues only in terms of numbers tends to hinder learning [4]. Students who are well calibrated can estimate more precisely what effort is needed to achieve mastery [2]. Otherwise, they are at risk of overestimating their performance, with the result that they might invest inadequate effort in their preparation [5,6].

In light of this, transitioning from school to university is often a challenge for students, as there is considerably less formative feedback than at school [7,8]. Educators are not in a position to satisfy the students' need of formative feedback, because of large-scale classes and learning that occurs predominantly at home. One way to overcome this shortcoming is the extensive support possible through teaching by web-based learning techniques at universities; movies showing experiments support students' laboratory preparation [9], animations improve their conceptual understanding about physics concepts such as the expansion and contraction of gases [10], and flipped teaching helps students learn organic chemistry [11]. Within a Learning Management System (LMS), summative tests are typically used to apply and consolidate the topics of a recent learning session, or to ensure sufficient preparation for the upcoming one. Learning objectives which are part of more than one session or even of the whole class are normally not monitored by the LMS, due to the limited feedback possibilities within a LMS. Feedback in current LMSs can be categorized into two types: direct corrective feedback given for a wrong answer; and summative feedback on a total test performance [12]; the latter seems to induce superficial preparation in learners [13]. Consequently, feedback mechanisms in current LMSs are insufficient to allow for thorough preparation and successful recalibration of learners.

1.1. Learning Progressions Hidden in Numeric Data

It is apparent that students generate a vast amount of educational data while learning through online LMS, such as Moodle or OpenOLAT. Teachers must translate this mostly numeric information into a comprehensible form to further support students' learning processes and to allow them to plan their next learning steps.

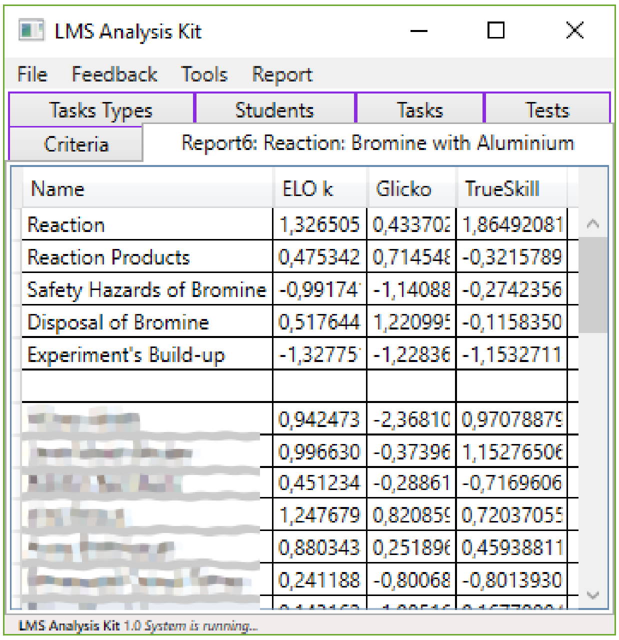
However, current LMSs do not use such collected data to provide teachers with sufficient information about their students' learning [14]. The less information an LMS offers directly to the user, the more information remains hidden in the database behind [15]. Educational Data Mining (EDM) tries to develop tools that unveil such concealed information, but data mining techniques in these tools are often too complex for teachers [16]. Hence, EDM also offers ready-to-use solutions to monitor students' learning progress [17]. However, these monitoring opportunities have some limitations in common. They usually need excessive work within the database for reorganization of the stored data into a more accessible form [18], or the usage of the educational data has to be prepared in advanced during the LMS development. Such considerations lead, for instance, to an activity check tool that allows students to monitor their own activities within the LMS [19] or the development of an "early warning system" [20] to alert students who are at risk of failing.

There are rules implemented in EDM solutions, which translate a numeric result into comprehensible information. Critical behavior, for example, is symbolized by a red traffic light. More elaborate feedback, such as offering appropriate learning opportunities, is rarely possible. In order to manage the translation of numeric information into comprehensible formative feedback, we have developed these software tools, the Learning Management System Analyzation Kit (LMSA Kit) and the Easy Snippet Feedback Edit (ESF Edit). These tools work together to close the gap between the vast amount of information within the LMS and the students' need for more supportive feedback.

2. Developing New Software Solutions

2.1. The LMSA Kit Offers Insights into Students' Learning

Generating appropriate formative feedback requires a solid, learning-goal oriented data base, which allows to cluster student's performance. This is provided by the Learning Management System Analyzation Kit (LMSA Kit). This analyzation software tool allows teachers to monitor their students' learning progress (cf. Figure 1). It is different from other EDM tools, in that teachers would be able perform the analysis on their own, rather than using predefined algorithms. Predefined and ready-to-use Data Mining algorithms are not flexible enough to support teachers in their daily educational business requirements [21].



Report6: Reaction: Bromine with Aluminium

Name	ELO k	Glicko	TrueSkill
Reaction	1,326505	0,433702	1,86492081
Reaction Products	0,475342	0,714548	-0,3215789
Safety Hazards of Bromine	-0,991741	-1,140888	-0,2742356
Disposal of Bromine	0,517644	1,220995	-0,1158350
Experiment's Build-up	-1,327751	-1,22836	-1,1532711
[redacted]	0,942473	-2,36810	0,97078879
[redacted]	0,996630	-0,373966	1,15276506
[redacted]	0,451234	-0,28861	-0,7169606
[redacted]	1,247679	0,820859	0,72037055
[redacted]	0,880343	0,251896	0,45938811
[redacted]	0,241188	-0,800688	-0,8013930

LMS Analysis Kit 1.0 *System is running...*

Figure 1. Screenshot of a score list in the LMSA Kit

All conventional EDM techniques make use of students' educational data stored within the LMS. Teachers cannot access all data which are required by these techniques, nor are they able to implement additional tools in an existing LMS. The LMSA Kit is a software that teachers can use on their personal computers. The prediction algorithms are already implemented and ready-to-use, but the parameters can be recalibrated for special requirements as well. Additionally, tools are built in, which support teachers, especially during recalibration or validation of their estimations.

The LMSA Kit depends only on data exported via export functions within the LMS. All tasks addressing the same learning objective or competency are combined in one criterion within the LMSA Kit. It then uses Matchmaking algorithms like TrueSkill by Microsoft [22] or the ELO rating algorithms [23] to calculate students' abilities on each specific criterion. In this way, the LMS data becomes more than just information about a test passed or a task solved.

The software extracts information about competency acquisition or attained learning objectives, and presents the results in a comprehensible form to the teacher. In contrast to common techniques like clustering or classification algorithms [21], the LMSA Kit does not necessarily depend on old data material to train the prediction algorithms. In our early trials we could reach average correlations of 0.37 between the prediction of abilities in a criterion and later-earned scores in an exam in this topic [24]. In contrast, the best correlation between a human rater and an automatic scoring system is about r=0.52 [25]. Given the fact that the LMSA Kit uses educational data stored in the LMS to predict results in a future exam, the smaller correlations are not surprising.

The accuracy of the prediction does not render exams unnecessary, nor are the predictions intended to do so. However, it has been shown that high performing students in web-based learning tend to be high performers in the exam, while low performing students are closer to failing [26]. For this reason, the students' scores in the criteria are not only a useful source of information for educators, it is also important to offer this information to the students, as it may have an effect on their final exam performance. Consequently, the students' scores in the criteria are an ideal data basis for automated feedback generation within the ESF Edit.

2.2. Transferring Scores into Automated Feedback

Although the LMSA Kit allows one to monitor the competence acquired in a criterion, the information it offers has a clear grading character like teachers' records, similar to other educational data in the LMS or derived by other EDM techniques. Consequently, the information must be translated into a formative and comprehensible feedback for the students. The simplest and, inevitably, the most time-consuming way is to write the feedback by hand. Other approaches to optimization aimed at simplifying the writing process. The "Electronic Feedback" software, for instance, is such a grading assistant developed by Liverpool University [27], in which teachers define and select different text-templates. Here, the feedback generation process is still carried out by teachers, but the time spent on the writing process is reduced. Students seem to appreciate this out-of-the-box feedback as more fair than handwritten remarks [27].

Unfortunately, the “Electronic Feedback” software has some technical limitations. It is not developed as standalone software; it is, instead, programmed as a macro in Microsoft Word. Therefore, adaptions of the software are necessary with every new Office version. This makes the use of “Electronic Feedback” impractical for non-professionals who are unfamiliar with macro programming.

Another approach is to combine the feedback generation process immediately, while grading students’ work. Such software solutions tend to be quite complex in their implementation, and too specific to facilitate broad application. “Goodle GMS” [28] allows one to grade uploaded students’ protocols about analytical chemistry experiments. The scope of use is, on the one hand, limited to a small number of experiments; on the other hand, the technical requirements for the running server environment are high.

“SuperMarkIt” [29] can be seen as an enhanced version of the “Electronic Feedback” software. It is a standalone software, independent of MS Office products, and composes the feedback message out of previously defined text-templates. The software allows the use of a primitive rubric to assign text-templates automatically to earned scores as a special feature. Consequently, it is possible to generate a large number of feedback texts with “SuperMarkIt” on the basis of a score list. In our early attempts to generate feedback for students, we used a similar rubric scheme to translate different students’ scores in the criteria into feedback messages.

However, while writing the different text-templates and setting up the assignment rubric, we found that this way of assigning feedback messages is not flexible enough. Cross-relations, such as learning difficulties in two related topics, could not be depicted appropriately by using such a simple rubric. Therefore, we started to develop the ESF Edit. Instead of assigning text-templates to score ranges rigidly, the user becomes a developer of small programs with the support of the ESF Edit, which later assembles the feedback text. In this way, more flexible feedback becomes possible, which better addresses the specific needs of students.

2.3. Structure of the ESF Edit Software Environment

The ESF Edit is designed as an integrated programming development environment (IDE) and an editor for a visual programming language (VPL). The IDE is a software tool for programmers that provides advanced support in developing programs. Additionally, a VPL is integrated, because teachers are not usually programmers. Visual, in this context, means that the program is not written in text lines. Instead, the program is “programmed” by arranging different graphical parts, the so-called snippets, on a worksheet and linking them to each other in a specific logical order (cf. Figure 2). The complete logic of the feedback generation process is built as a visual tree and, therefore, no programming language with a complex structure has to be learned. A VPL shows many similarities to flow charts, which are used to describe a process in engineering subjects (cf. Figure 3).

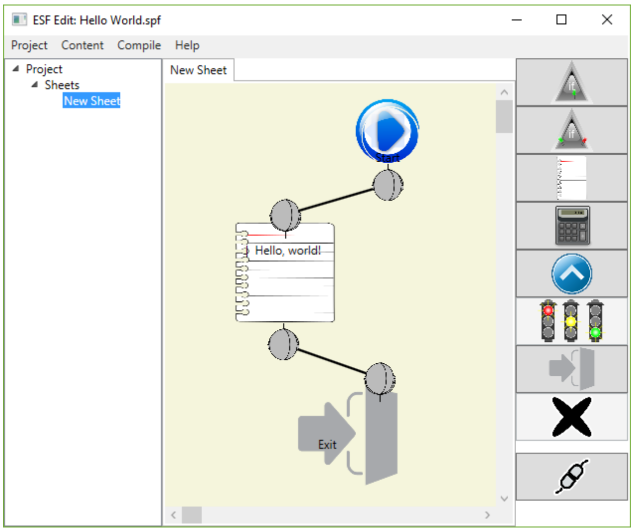


Figure 2. “Hello, World” in ESF Edit

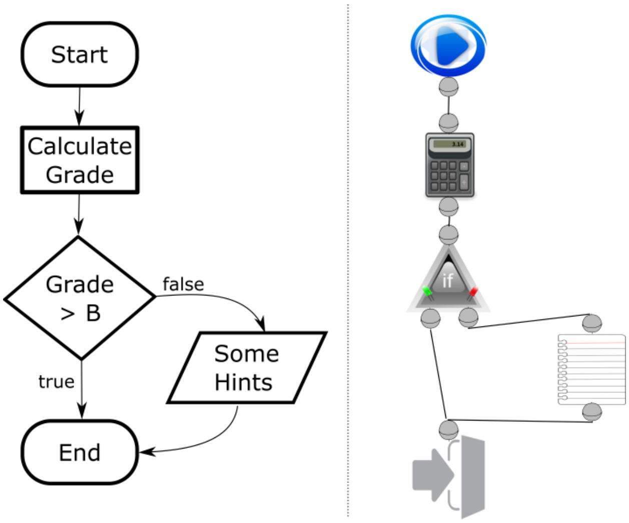


Figure 3. Flow Chart (ISO 5807) vs. Program in ESF Edit

In addition, VPLs are easy to learn and are often used as an early approach to learning programming [30]. Thus, the VPL used in ESF Edit is an easy-to-use feature that enables teachers to build their own feedback generating “programs”. Specific feedback informing students about their strengths and weaknesses as well as further areas of growth can now be sent to large-scale classes.

In contrast to the “Electronic Feedback System” and “SuperMarkIt”, our software offers additional advantages in generating formative feedback. The ESF Edit is written in C# and compiled against Microsoft’s .Net runtime. It can be considered a standalone software and used like normal applications, in contrast to the “Electronic Feedback System”. The process of feedback generation is separated from the ESF Edit into a library, the so-called ESF Engine. The ESF Engine executes the feedback programs developed within the ESF Edit to write the feedback messages. Thus, the ESF Edit is not exclusively linked to our LMSA Kit. Developers of other programs who are interested in feedback generation using the support of the ESF Edit are free to only implement the ESF Engine. By doing this, the new software can execute the feedback generation instructions of ESF Edit programs, in a similar way to the LMSA Kit. This makes the scope of use broader and more flexible compared with “SuperMarkIt”.

The ESF Engine, for instance, is accessible as a COM-Object; therefore, VBA scripts and Microsoft Office macros can also access the functions of the ESF Engine. The ESF Engine can extend Microsoft’s Excel with a feedback generation tool, allowing teachers to translate their score lists directly into electronic feedback. A revised version of the “Electronic Feedback System” as a set of special macros within Word would also be possible. We are convinced that our feedback solution will enrich other EDM applications, as well. Likewise, direct implementation into an LMS is possible, so that feedback can be provided in much greater detail than is possible in existing LMS applications.

2.4. A Feedback Generation Process in the ESF Edit

The feedback generation process in the ESF Edit follows a few, easy-to-learn design rules. There are snippets with different functions that can be placed and arranged by the programmer on a feedback sheet. Snippets possess input and output plugs, which can be connected by wires. The connections describe the logical flow of the feedback generation. Each feedback generation within the ESF Edit can be seen as a line which begins at the *start snippet* and ends at an *exit snippet* with other snippets in between, like beads on a string.

On its way from the *start snippet* to the *exit snippet*, the logical line usually passes through other snippets. These snippets are the tools to write the feedback and to make it specific. The most important snippets are: the *script snippet*, the *if snippet* and the *message snippet* (cf. Figure 4). The *script snippet* can be used to execute an underlying script which is, for instance, gathering some information about the student. The script can query scores reached from a database or receive the student’s learning progress within a criterion from the LMSA Kit, like that shown in Figure 3. The information received is stored in variables, which are also available in other snippets, like the *if snippet*; the most important snippet for the logical flow is the *if snippet*. There is a condition deposited in the *if snippet*. If the condition is evaluated as true, the program continues on the left-hand side output plug, otherwise it continues on the right-hand side. Consequently, the logical flow of the feedback generation process can be directed in different ways. Hence, the *if snippet* is the key to generating individual feedback for the students. Another useful snippet is the *message snippet*. If the logic flow passes through it, the stored text message is written into the generated feedback text. The *message snippet* can be seen as a pen that writes the feedback.

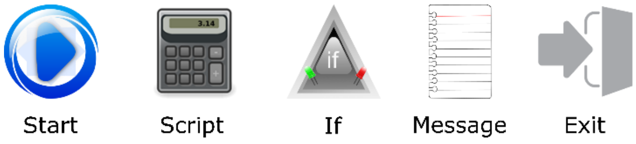


Figure 4. The Snippets most commonly used

The feedback is composed of common feedback blocks for each topic (cf. Figure 5). Firstly, there is a *message snippet* giving an introduction to the topic. The following *if snippets* divide the feedback into two versions. If the performance is adequate, the path continues on the left-hand side and runs to a *message snippet,* which informs about a sufficient amount of preparedness. In the other case, the path continues on the right-hand side and runs to a *message snippet* which informs about insufficient preparation and offers additional hints and learning opportunities. Most of the time, the path on the right-hand side does not run directly to the end, as it does in the sample depicted. It passes more *if snippets* and splits up into a more detailed net of pathways to generate more specific and supportive feedback. Finally, all paths lead to a last *message snippet* giving final information before the feedback generation ends.

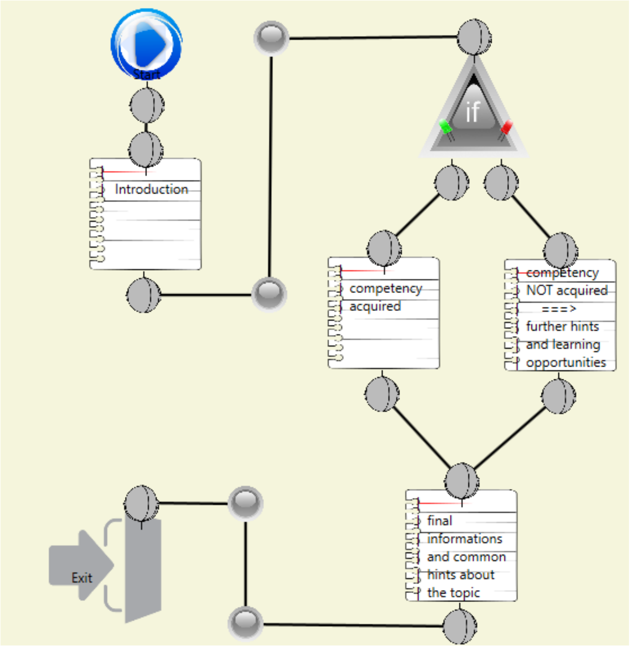


Figure 5. A common feedback block

3. Evaluation and Usability of the Automated Feedback

This type of formative feedback provided by the ESF Edit is completely new in web-based trainings. There are only a few comparable examples of such implementation [4,13]. Consequently, we aimed at addressing the two following research questions:

1. Does such feedback meet students’ needs?
2. Is the software appropriately designed in a way that teachers can use it without problems?

4. Testing the Automated Feedback with Students

In order to answer our first research question, we tested our feedback software in different laboratory classes. We began with classes that are taught in our institute, and, after early encouraging results, we tested it in laboratory classes at a neighboring institute to recruit a bigger population. Evaluation of the automated formative feedback was done from two different perspectives: the students' perceived use (students' perspective) and the impact on performance in class (institutional perspective).

4.1. Methods

To evaluate the students' view of the automated feedback, we presented the students with the feedback in the university's survey system. After students had read their personal feedback text, they were asked to assess different aspects of the feedback received. So, students were asked whether they wish future feedback or not, and how much they thought they could benefit from such a feedback system. To assess the impact of the automated feedback, we compared the final exam statistics in terms of fail and pass rates of classes with and without automated feedback. Except the delivery of the automated feedback, nothing else was changed in the treated classes. Consequently, the differences between the classes presumably resulted only from the feedback. We used Cohen's d to measure the differences and to get a first insight into the impact of the automated feedback.

4.2. Participants and Data Collection

First, we tested our feedback software in laboratory classes for chemistry student teachers who are taught at our institute. Unfortunately, this reduced the size of the population; on the other hand, we had direct access to the students and were involved with the class at this early stage. During the spring semester of 2015 39 students, and one year later 41 students received feedback and were additionally asked to evaluate the feedback received. From these cohorts, 19 and 22 respectively filled out the questionnaire completely. Given the positive evaluation in our laboratory classes, we tested the feedback system in a large-scale laboratory class for chemistry minors at a neighboring institute, afterwards. This class was selected, because the structure of the supporting web-based trainings and the final exam was largely similar to our laboratory class. In the spring semester of 2017, 781 students were enrolled in this class. About 200 were actively participating in the voluntary web-based training. More than 50 %, 134 students, were reading their feedback messages and about 35 were participating in the evaluation.

In parallel to the evaluation of the use perceived in the large laboratory class, we attempted at first to assess the impact of such an automated feedback on students in the same laboratory. For this purpose, we made use of the final exam statistics of two different student groups: The first group were 114 biology students who visited the laboratory in the autumn semester 16/17, the other group were 201 students majoring in human, dental or veterinary medicine in the spring semester 17. These two groups were chosen because the final exams as well as the composition of the students enrolled were quite similar to those of the year before. The fail and pass rates of the aforementioned laboratory classes were compared with those of the year before to provide an insight into the impact of the automated feedback. The year before, 133 biology students and 145 medicine students visited the laboratory.

4.3. Results: Perceived use of Automated Feedback

The questionnaire survey of the evaluation of the trials in 2015 and 2016 showed the first encouraging results (cf. Table 1). Students saw the feedback system as beneficial, and requested further feedback. These findings remained constant over all cohorts. Students of the large cohort in 2017 saw a stronger benefit for their test preparation. This could be explained by the larger population, because the situation in class is more anonymous and some students may feel less supervised and, therefore, unsupported. This is the explanation for the slightly stronger wish for further feedback, and shows the great potential residing in such a feedback system. Overall, this first evaluation shows that this type of automated feedback meets the students' needs for more individual support at the university level. Large classes in particular, where there is a high degree of anonymity, benefit through the automated feedback by the ESF Edit.

Table 1. Students' Ratings Regarding Different Aspects of the Feedback Received

	2015 *(n = 19)*	*2016* *(n = 22)*	*2017* *(n ≈ 35)*
Benefit in general	1.9	1.8	2.0
Benefit in test preparation	3.0	2.7	1.9
Request for further feedback	2.0	1.9	1.6

1 means highest accordance, 6 means highest discordance.

4.4. Results: Impact of Automated Feedback Delivery

For a complete evaluation of the developed software tool, it is not only necessary to evaluate the students' perspective in terms of a questionnaire survey, but the impact thereof has also to be verified, thereby obtaining the institutional perspective. The final exam statistics reveal an obvious improvement if an automated feedback is available for students (cf. Table 2). The increased pass rates result in a Cohen's d about .21 in the case of the medicine students and .52 in the laboratory of the biology students. The latter is viewed as a medium effect by Cohen [31] and is assigned to the desired effects by Hattie [32].

Table 2. Statistics of the final exams with and without automated feedback in introductory chemistry

Class	Received Feedback	Semester	Enrolled Students	Average Score %	Passed %	Failed %	Cohen's d
Biology	✓	Fall 16/17	114	60.3	71	29	.52
	×	Fall 15/16	133	50.6	50	50	
Human, Dental, and Veterinary Medicine	✓	Spring 17	201	64.9	79	21	.21
	×	Spring 16	145	60.8	74	26	

All in all, it can be stated that automatically generated formative feedback through the combination of LMSA Kit and ESF Edit is beneficial from different perspectives. Students appreciate this new type of feedback, and institutions such as universities, as well as students, can profit in the form of increased pass rates.

5. Usability of the ESF Edit

In our trials, the feedback tested was designed by experts, namely the developers of the ESF Edit, but the ESF Edit should allow the generation of such a feedback even by inexperienced users. We thus conducted a Delphi study to answer the second research question. A Delphi study is a special iterative interrogation process with the aim of generating a group consensus of an expert group [33]. This type of study is easy to realize, and the results usually outperform other interrogation methods for expert groups, for instance, "Prediction Markets" [34].

5.1. Methods

There are a lot of forms and variants of Delphi studies [35,36]. In our case, we preferred a fast interrogation process, which is why we chose a variant of micro Delphi studies in combination with a shortened expert group. Instead of a consecutive second presence interrogation cycle, the experts received the second survey via e-mail immediately after the first interrogation cycle. The smaller size of the expert group reduces the accuracy of the result of the study, but this was acceptable, as we were only interested in the major trends.

5.2. Participants and Data Collection

The expert group consisted of 11 participants: student teachers in their last semester at the university. None of them had worked with the software before. The participants received a short handbook about the ESF Edit a few days before the study to familiarize them with the principles of the software and the snippets. The software was handed to the participants at the first meeting. Additionally, two saved feedback generation processes were included as sample material: A feedback generation program only writing "Hello, World", as can be seen in Figure 2; and another, more complex feedback generation process, which had been used for tests in 2015 and 2016. Thus, the participants could open a saved feedback generation process. The participants could work with the software for about an hour, before they answered the first survey. The survey consists of different Likert scale items and additional open-ended questions addressing the following topics: User Interface, Usage of Snippets, Scripting, and Compilation Process.

After all participants had submitted their questionnaires, the mean and the standard deviation were calculated for the Likert scale items. Additionally, the open-ended answers were collected. For the next interrogation cycle, the experts received edited copies of their surveys via e-mail with mean and deviation for every Likert scale item and the collected free text answers. The answers given initially by each participant were highlighted, so they could assess their answers in relation to the whole group. They revised the surveys at home and sent the revision back one week later.

5.3. Results of the Delphi Study

The Likert scale in every item ranges from 1 "total accordance" to 8 "total discordance" (cf. Table 3). The Likert scale items showed an average standard deviation of about 1.76 in the first interrogation cycle. The deviation decreased in the resubmitted surveys by about 0.15 to 1.62. This can be considered as evidence of a working consensus evolving within the expert group. The average deviation in the first two topics is smaller than in the last two. The first two topics, the "User Interface" and the "Usage of Snippets", address basic functions which are comparable to activities in some MS Office applications. Therefore, the participants were familiar with it; that is also why there was greater accord with regard to these topics. The more unfamiliar they were with the topic, the more discordant were their ratings. The script language and the compilation of a program are especially challenging for unskilled users. Participants with experience in this area rated completely differently, causing the high average deviation.

The topic most unanimously rated was the "User Interface" (cf. Figure 6), albeit the second topic "Usage of Snippets" had only a slightly higher deviation. It is apparent that the users did not seem to be confused by the complexity of the user interface. Perhaps some control elements, such as the menu bar, are familiar from other software. Nevertheless, the intuitive usage and the user friendliness need to be improved in further versions. The control elements were placed at the right spot, the number of control elements seemed to be appropriate for the users, and the toolbox on the right-hand side provided purposeful access for them (cf. Figure 7). Additionally, the participants saw that the snippets allowed the construction of complex solutions in an effective way. Nonetheless, handling and arranging the snippets was perceived as a complex process.

Table 3. Deviations of the Delphi Study

Topic	Average Deviation 1st Interrogation Cycle	Average Deviation 2nd Interrogation Cycle	Change in Deviation
User Interface	1.54	1.40	-0.14
Usage of Snippets	1.55	1.44	-0.11
Script Editors	1.73	1.75	0.02
Compilation Process	2.18	1.87	-0.31
In Total	*1.76*	*1.62*	*-0.15*

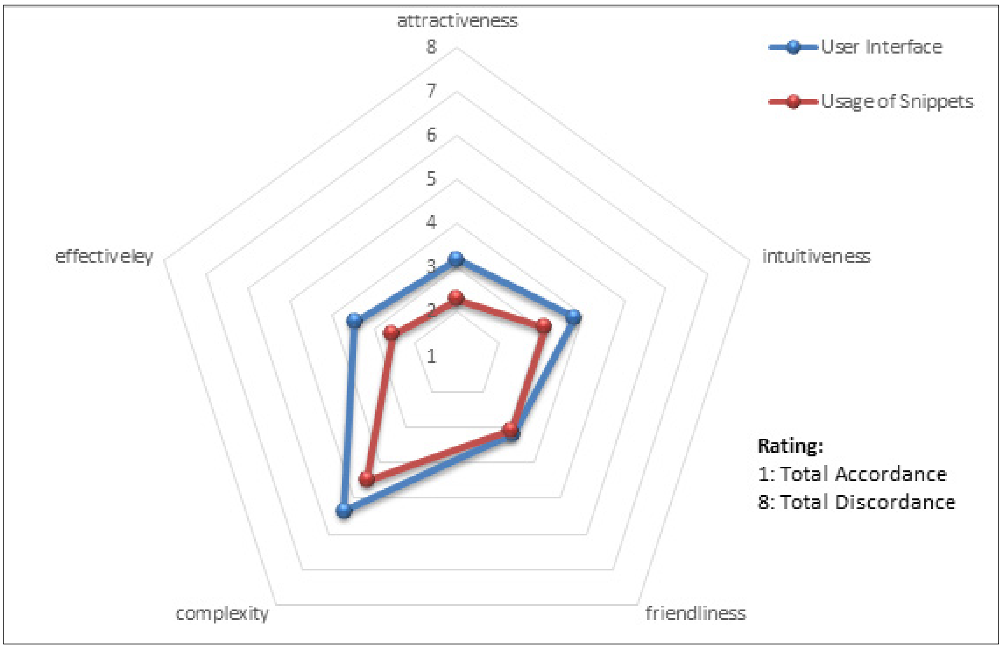


Figure 6. Rating about the "User Interface" and "Usage of Snippets"

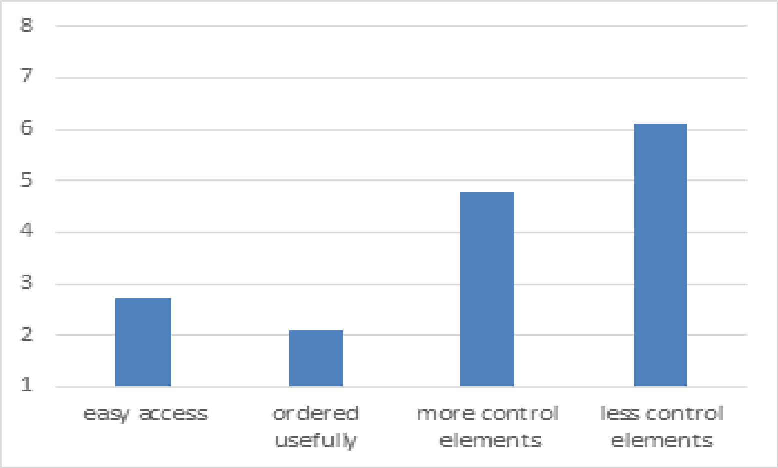


Figure 7. Rating about the control elements within the user interface

In contrast to the first two topics, the "Script Editors" and the "Compilation Process" were rated low (cf. Figure 8 and Figure 9). The low rating is due to the unfamiliar activities encountered during the process. Writing in a script language or compiling programs to prepare the feedback generation is difficult for users without experience of programming. The guidance provided in the ESF Edit, such as error messages or highlighting wrongly connected snippets, is not sufficient for inexperienced users (cf. Table 4).

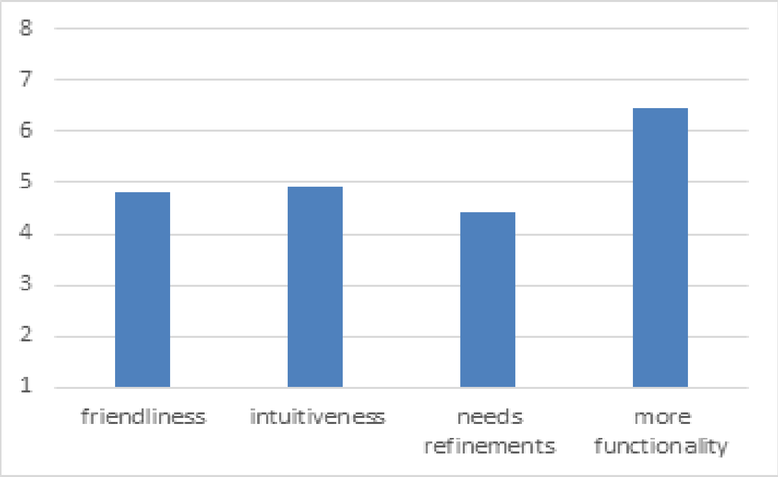


Figure 8. Rating about the "Scripting"

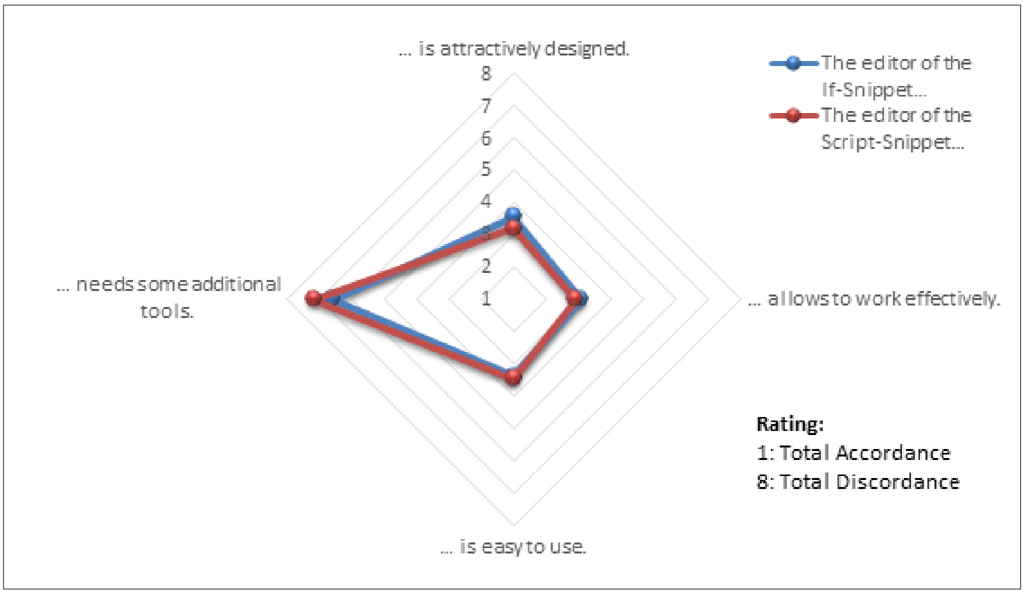


Figure 9. Rating about the script editors

Table 4. Rating about the "Compilation Process"

The compilation was hard to perform	4.9	Compiler error messages are easy to understand.	4.6
I need further assistance while compiling projects.	3.4	Compiler error messages are useful hints.	4.8

1 means total accordance, 8 means total discordance.

Arising out of these results, we propose to include more information and guidance, such as additional video tutorials demonstrating the different steps of feedback generation and the various aspects of handling the ESF Edit. Furthermore, users will receive additional guidance within the user interface through cursor-led tool tips explaining the control elements when the cursor hovers above them. The script editors will be linked more closely to the target software, which is used later as the data source for the feedback generation. In this way, the script editors could offer help in writing the correct condition. We are working on a solution where users can click the whole script out of ready-to-use templates within a script editor. Further tutorials and instructional material should help the users to fulfil the compilation process.

6. Conclusion

Current LMSs cannot provide students with sufficient formative feedback about their learning. Even existing EDM solutions only offer basal information about the learning progress. While teachers can certainly benefit from the insights offered by EDM solutions, such as the LMSA Kit, they are still not able to address all students with specific feedback. Consequently, the full potential for instruction reaching students is not being realized. This gap is closed by the ESF Edit, which can provide feedback based on the LMSA Kit or any other EDM solution, and conveys the feedback to the students. This offers a great extension to existing and future EDM solutions. Generating the feedback programs might be time-consuming for a one-time usage of the program, whereas frequent usage through repeated classes compensates for the time taken in implementation. Combining the LMSA Kit and the ESF Edit allows generation of individual feedback messages for large groups of students. This formative feedback can address specific students' needs, and the results of the trials during 2015 to 2017 have proved its usefulness and practicability. Students appreciate this new type of formative feedback, and consider it useful for final exam preparation. This estimated, perceived use is not only a result in the questionnaire survey, but the impact of the automated feedback can be also identified from the comparison of the final exam statistics. Implementation of the feedback system resulted in a medium, improvement in the pass rates.

We conducted a Delphi study to further investigate the usability of the ESF Edit, especially for inexperienced users to answer the second research question. Users managed to navigate on the user interface and the complexity did not seem to confuse, but the study helped us to identify areas for future improvements. More guidance is needed within the software and outside in terms of further instructional material. Currently, we are including an advanced help system that supports inexperienced users through popup boxes. During the process of revising the instructional material, we are enlarging the material with documented sample projects, how-to guides and video tutorials. We should be able to overcome the existing weaknesses through this package of improvements.

After the refinements are finished, we plan to localize the software tools and the instruction material to address a wider public. Both programs will be available, free of charge, for educational purposes on our institute's homepage.

References

[1] Hattie, J., Timperley, H., "The Power of Feedback, *Review of Educational Research*, 77 (1), 81-112, 2007.

[2] Hattie, J., "Calibration and confidence, *Learning and Instruction*, 24, 62-66, 2013.

[3] Kluger, A. N., DeNisi, A., "Feedback Interventions, *Current Directions in Psychological Science*, 7 (3), 67-72, 1998.

[4] Shute, V. J., "Focus on Formative Feedback, *Review of Educational Research*, 78 (1), 153-189, 2008.

[5] Bol, L., Hacker, D. J., O'Shea, P., Allen, D., "The Influence of Overt Practice, Achievement Level, and Explanatory Style on Calibration Accuracy and Performance, *The Journal of Experimental Education*, 73 (4), 269-290, 2005.

[6] Kruger, J., Dunning, D., "Unskilled and unaware of it, *Journal of personality and social psychology*, 77 (6), 1121-1134, 1999.

[7] Seery, M. K., Donnelly, R., "The implementation of pre-lecture resources to reduce in-class cognitive load, *British Journal of Educational Technology*, 43 (4), 667-677, 2012.

[8] Yorke, M., "Formative assessment in higher education, *Higher Education*, 45 (4), 477-501, 2003.

[9] Fung, F. M., "Using First-Person Perspective Filming Techniques for a Chemistry Laboratory Demonstration To Facilitate a Flipped Pre-Lab, *Journal of Chemical Education*, 92 (9), 1518-1521, 2015.

[10] Sanger, M. J., Phelps, A. J., Fienhold, J., "Using a Computer Animation to Improve Students' Conceptual Understanding of a Can-Crushing Demonstration, *Journal of Chemical Education*, 77 (11), 1517, 2000.

[11] Fautch, J. M., "The flipped classroom for teaching organic chemistry in small classes, *Chem. Educ. Res. Pract.*, 16 (1), 179-186, 2015.

[12] Espasa, A., Meneses, J., "Analysing feedback processes in an online teaching and learning environment, *Higher Education*, 59 (3), 277-292, 2010.

[13] Gikandi, J. W., Morrow, D., Davis, N. E., "Online formative assessment in higher education, *Computers & Education*, 57 (4), 2333-2351, 2011.

[14] Zorrilla, M., Menasalvas, E., Marín, D., Mora, E., Segovia, J., "Web Usage Mining Project for Improving Web-Based Learning Sites in *Computer Aided Systems Theory - EUROCAST 2005: 10th International Conference on Computer Aided Systems Theory, Las Palmas de Gran Canaria, Spain, February 7 - 11, 2005, Revised Selected Papers*, Springer Berlin Heidelberg, 205-210.

[15] Martín-Blas, T., Serrano-Fernández, A., "The role of new technologies in the learning process, *Computers & Education*, 52 (1), 35-44, 2009.

[16] Dutt, A., "Clustering Algorithms Applied in Educational Data Mining, *International Journal of Information and Electronics Engineering*, 2015.

[17] Long, P., Siemens, G., "Penetrating the Fog: Analytics in Learing and Education, *Educause Review*, 46 (5), 31-40, 2011.

[18] Romero, C., Ventura, S., García, E., "Data mining in course management systems, *Computers & Education*, 51 (1), 368-384, 2008.

[19] Fritz, J., "Classroom walls that talk, *The Internet and Higher Education*, 14 (2), 89-97, 2011.

[20] Macfadyen, L. P., Dawson, S., "Mining LMS data to develop an “early warning system” for educators, *Computers & Education*, 54 (2), 588-599, 2010.

[21] Mohamad, S. K., Tasir, Z., "Educational Data Mining, *Procedia - Social and Behavioral Sciences*, 97, 320-324, 2013.

[22] Dangauthier, P., Herbrich, R., Minka, T., Graepel, T., "TrueSkill Through Time in *Advances in Neural Information Processing Systems 20*, Curran Associates, Inc, 337-344.

[23] Pelánek, R., "Applications of the Elo rating system in adaptive educational systems, *Computers & Education*, 98, 169-179, 2016.

[24] Hedtrich, S., Graulich, N., "Crossing Boundaries in Electronic Learning in *Computer-Aided Data Analysis in Chemical Education Research (CADACER)*, Oxford University Press, 21-28.

[25] Liu, O. L., Rios, J. A., Heilman, M., Gerard, L., Linn, M. C., "Validation of automated scoring of science assessments, *Journal of Research in Science Teaching*, 53 (2), 215-233, 2016.

[26] Chambers, K. A., Blake, B., "Enhancing Student Performance in First-Semester General Chemistry Using Active Feedback through the World Wide Web, *Journal of Chemical Education*, 84 (7), 1130, 2007.

[27] Denton, P., Madden, J., Roberts, M., Rowe, P., "Students' response to traditional and computer-assisted formative feedback, *British Journal of Educational Technology*, 39 (3), 486-500, 2008.

[28] La Muñoz de Peña, A., González-Gómez, D., La Muñoz de Peña, D., Gómez-Estern, F., Sánchez Sequedo, M., "Automatic Web-Based Grading System, *Journal of Chemical Education*, 90 (3), 308-314, 2013.

[29] Debuse, J. C. W., Lawley, M., "Benefits and drawbacks of computer-based assessment and feedback systems, *British Journal of Educational Technology*, 47 (2), 294-301, 2016.

[30] Sáez-López, J.-M., Román-González, M., Vázquez-Cano, E., "Visual programming languages integrated across the curriculum in elementary school, *Computers & Education*, 97, 129-141, 2016.

[31] Cohen, J., *Statistical power analysis for the behavioral sciences*, L. Erlbaum Associates, 1988.

[32] Hattie, J. A. C., *Visible learning*, Routledge, 2010.

[33] Rowe, G., Wright, G., "The Delphi technique as a forecasting tool, *International Journal of Forecasting*, 15 (4), 353-375, 1999.

[34] Green, K., Armstrong, J. S., Graefe, A., "Methods to Elicit Forecasts from Groups, *Foresight: The International Journal of Applied Forecasting* (8), 17-20, 2007.

[35] Rowe, G., Wright, G., "The Delphi technique, *Technological Forecasting and Social Change*, 78 (9), 1487-1490, 2011.

[36] Skulmoski, G. J., Hartman, F. T., Krahn, J., "The Delphi Method for Graduate Research, *Journal of Information Technology Education: Research*, 6, 1-21, 2007.