# USING STUDENTS' COLLABORATION TO IMPROVE ACTIVE PARTICIPATION TO UNIVERSITY COURSES WITH LARGE NUMBER OF ATTENDEES

**Michael Mogessie Ashenafi, Giuseppe Riccardi, Marco Ronchetti**

*Department of Information Engineering and Computer Science University of Trento (ITALY)*

## Abstract

Students' active participation to large bachelor courses (i.e. courses with hundred or more attendants) is difficult to stimulate and monitor. As a result, many students may defer studying until shortly before the exam, wasting their time and loosing contact with the courses, where they continue to show up but are unable to harvest useful knowledge. To cure the problem, we devised a computer-supported methodology based to an implicit form of collaboration/competition: a sort of serious social game where students are requested to formulate questions for their peers, propose answer and choose the best responses. Evaluation of such approach yields results, which students perceive as excellent.

Keywords: peer evaluation, computer supported learning, formative assessment.

## 1 INTRODUCTION

At least in certain educational systems, bachelor courses are often rather crowded, with more than hundred students sitting in an auditorium. In such conditions, for teachers it is not easy to get an indication of whether students are really understanding and keeping the pace, or if they are simply taking notes for later review and hoping to grab something out of a lecture they mostly don't understand. To make sure students stay aligned with the course, it is possible to assign homework, as usually done e.g. in US colleges, and/or have intermediate summative assessments. Such approach is certainly effective, but it is also expensive. Homework needs to be graded: such time-intensive activity is rarely performed by teachers, but rather assigned to teaching assistants. Not every academic system however has such figures. So what can be done to "force" students to keep the pace, whiteout a massive time investment on the teacher teams side? Can a system be devised, which draws on students' mutual cooperation to solve the mentioned problem?

We came out with a possible solution, which is based on a sort of social gaming strategy. In this paper we present the idea and it implementation We also report the results of an investigation aimed at assessing if the strategy is effective, if it reaches its goals and if students find it useful.

In section II we set the ground, looking at the literature to illustrate related work. In Section III we describe our methodology, which is incarnated in a working prototype. The following section reports various details of our approach. Section V discusses the validation on the field of our work. A general discussion concludes the paper.

## 2 BACKGROUND

Various lines of research have tackled the problem of automatically supervising and assessing students in massive higher education. For instance, an ambitious approach is the one of intelligent tutor systems, active since half a century but yet not widely employed (for a review see [1]).

Assessment always plays a very important role in formal education, and is usually thought as a teacher-managed activity. In its most common form, which is aimed at measuring what students have already achieved, it is called summative assessment. It is based on the idea of levels of achievement, so that passing a checkpoint opens new doors, which allow proceeding to the next level, and/or grants a document stating that the students possesses certain knowledge, ability and/or competences. However, certifying learners achievements is not the only possible assessment goal. Assessment can in fact also be used to improve on-going learning and to provide support and feedback to learner. This form, termed as formative assessment [2], focuses on helping students to achieve progresses, rather than on certifying them. An assessment method that is composed of both formative and summative parts is termed as continuous assessment [3]. In the current educational systems organization, summative assessment is a consolidated, necessary and often standardized procedure, which does

not necessarily impact on the main goals of education, i.e. on developing competencies in the student. Formative assessment instead is not a necessary step, but has a much higher value from a metacognitive point of view, which ultimately can powerfully boost the learning process.

A wide range of commercial and open-source e-assessment solutions are available today. They are mostly intended for summative assessment, and have the advantage of being able to provide the student with instant feedback regarding their performance on a test or quiz they just completed. They decrease the burden put on the instructor, offer high reliability and high levels of impartiality, removing any possible double standards when correcting tests or quizzes. These solutions can also be used for a formative purpose in increasing students' self-awareness about their level of knowledge. However, the level of metacognitive help they provide is limited.

A light form of formative assessment may be achieved in class, asking students to give immediate feedback to questions raised by the teacher. The student becomes aware of his/her shortcomings, and the discussion that may follow has a higher metacognitive value.

Answer gathering in class can be based on Electronic Voting Systems (EVS). For instance, Kennedy & Cutts [4] employed an EVS for a computer science subject at the first year college level and found that there is an association between students' EVS responses in class and their performance at end-of-semester exams. Nowadays EVS are better known as clickers: interactive technological gadgets that enable instructors to pose questions to students and immediately collect and view the responses of the entire class [5]. Clickers instantly collects the responses, and a software tabulates the results, which instructors can view, save, and display anonymously for the entire class to see. Clickers allow eliciting student participation and engagement, monitor students' understanding of course content in real time, and provide students with instant feedback on their comprehension to help them monitor their own understanding. Their effectiveness has been discussed e.g. in [6]. Recently such capability has been incorporated in smartphones apps (e.g. Kahoot!) and scientifically investigated [7].

Studies have shown that it is possible, even in large classrooms, to move towards a more active students participation in class, e.g. with a Delphi-derived methodology, which employs smartphone apps as triggers for students' discussions [8]. Mazur [9,10] pioneered long ago the strategy of having, even in large classes, short (few minutes long) discussions among students gathered in small groups. Discussions and active students' involvement in class may also be achieved by using the "flipped classroom" methodology, e.g. by using video-lectures to support the content-delivery phase out-of class [11]. Didactic value of peer discussion is also evidenced in [12].

While the in-class students' behaviour can be made more productive by using these techniques, the out-of-class study can become more effective when supported in various ways. For instance, recording lectures as videos and making them accessible via web gives students a chance to re-check anytime and anywhere the material presented in class (for a literature review, see [13,14]).

A problem remains: how to make sure that students actually work, and do it in an effective way, when they are out of class? Compulsory, formative assessment may help in this sense, but its cost can be vast. Who grades the hundreds of homework coming on a regular basis from a large class? Typically teaching assistants do, at least in educational systems where such figures are present, which is not always and everywhere true. Moreover, their time could find better uses. Peer assessment has been proposed as a possible alternative [15], which not only is cheaper, but also offers per-se a chance of increasing the effectiveness of students' study. In fact, when a student has to grade their peers' work, s/he needs to assume a different perspective, which might be enlightening. A review of the peer-assessment in higher education is reported in [16]. Some of the most relevant attempts in this direction include PRAISE [17], PeerWise [18], PeerScholar [19].

PRAISE (Peer Review Assignments Increase Student Experience) is a generic peer-assessment tool that has been used in the fields of computer science, accounting and nursing [17]. Before the distribution of assignments, instructors specify criteria, which are stored in the system. Once the instructor publishes the assignments, students start executing them and when ready submit their solutions. The system waits for the number of submissions to reach a specified number and then starts assigning review tasks to students. Students then review the solutions of their peers according to the specified criteria. Once all reviews are complete, the system checks if all reviewers agree and suggests a mark based on the criteria. In the case of disagreement among reviewers, the system submits the solution to the instructor for moderation. The instructor then needs to decide a mark and confirm the release of the result before students can see their overall mark for the assignment.

PeerWise [18] is a peer assessment tool where students create multiple-choice questions. These are stored into an online repository, which is empty at the beginning of the course. Students can answer the questions provided by their peers, rate and comment them and start discussions about them. The author of a question may reply to a comment that has been submitted by a student that has rated the question. All activity remains anonymous to students, however instructors are able to view the identity of questions' and comments' authors and have the ability to delete inappropriate items.

PeerScholar [19] is another peer assessment tool that was initially designed for an undergraduate psychology class. It aims to improve writing and critical thinking skills of students. Students first submit essays. Next, they are required to anonymously assess the works of their peers, assign grades and write a comment for each of their assessments. Students are also allowed to rate the reviews they have received.

Peer assessment techniques may have limitations. For instance, questions regarding the capabilities of the assessors may arise [20]. An issue is plagiarism [21], although such plague is not unique to peer assessment. Other possible issues are favoritism, bias or collusive behavior [22]. If students somehow find a way to give each other positive assessments, for instance by forming factions that are sustained throughout the duration of the course, they could gain unfair positive assessments. Hence peer review may be problematic as it lacks rigor and fairness. Such limitations strongly weaken the effectiveness of peer review when the practice is intended for summative assessment. However, its value as an enabler for metacognitive reflection can nonetheless be high.

## 3 METHODOLOGY AND SYSTEM DESCRIPTION

We started from the assumption that peer review can be valuable for engaging students, with the goal of stimulating them to regularly review the ideas, concepts and material presented in class. Work has hence to be done outside class. As in PeerWise, we request students to pose questions and to respond, but in our case questions are open-ended and not multiple choice. As in PeerScholar and in PRAISE, students are requested to evaluate their peer's work. Apart of these limited similarities, the methodology we propose is quite different. Our goal is to create stimuli that keep students regularly engaged with a course, avoiding that they only limit their involvement to being present in the classroom. Hence, we require them to ask questions about specific topics, respond to them, and evaluate best answers on a reasonably tight schedule. To keep them involved we organize this as a social game, where they can gain points and participate to a ranking. To make it more appealing and worth playing, an award is foreseen at the end. This could in principle be anything: we found that a tiny amount of extra points at the exam is quite appetizing. The award is split in two components: one for encouraging participation, the other to promote serious engagement.

In some sense the system may recall certain Q&A web systems, such as Yahoo Answers or Stack Overflow, where participants perform tasks with the sole incentive of gaining a "reputation". However, in such systems the final goal is the product, i.e. the set of answers to questions, while in our case it is the process: the valuable asset is not the produced knowledge, but rather the fact the students created it – and hence that they learned while participating. At the beginning of a new instance of a course that uses this methodology, the product of the old one has to be removed (or at least hidden), so as to allow reproducing the process with the new set of students. Also, it is not very important for us that responses be top quality, nor that the evaluation process be flawless: evaluations do concur in creating the ranking, which in the end contributes to (part of) the final prize, but the prize is so small that errors have an effect, which is no more relevant than the usual component of good and back luck that spice our lives.

The methodology is supported by a system developed as web application. The system manages users profiles and allows performing a certain number of management tasks, such as granting rights, gathering results and data, contacting via e-mail all students. The application core allows defining courses, and within courses sets of contexts. A context is the main logical unit: a knowledge domain, which can e.g. be mapped to one or more lectures, and typically corresponds to a temporal unit (such as e.g. one week). A context is associated with three tasks: Question creation; Answer formulation; Answer evaluation.

All three tasks are assigned to students. First, the teacher sets the stage by defining the semantics of the context: s/he selects a theme for the context, and suggests a set of keywords, which characterize it. Every student is notified via e-mail, and is requested to ask a question about it. Directives specify that every question should be clearly stated, and should contain a single question mark, which

essentially means they should not be articulated in sub-questions. It should be possible to answer the question in a short text – although the system does not impose constraints, the expected answer should be no longer than 150 words. A question could be something a student did not understand, some curiosity s/he might have, something that s/he thinks could be challenging for their peers, or simply something suitable for an examination. Every student is requested to contextually classify his/her question by picking one or two keywords among the ones suggested by the teacher, who in this way sets the stage by defining the semantics of the context.

Every task has to be completed by a given deadline – usually one or two days since the assignment.

After the deadline has expired, the teacher (or someone from her/his staff) selects a subset of questions proposed by the students: 1/N of them, where N is typically around 4 or 5. The selection criteria are up to the teacher. This is the only "manual" operation that is required on teachers' side, apart of defining topic and keywords. In a large class (say of the order of 100 students) it may request a bit more than an hour. To facilitate this operation, questions and keywords can be exported to a spreadsheet, so that they can be easily grouped (e.g. simply by ordering by keyword). Duplicate (or similar) questions are likely to be present in the set generated by the students: we'll discuss later how to deal with such cases.

Once the selection is complete, the system randomly assigns every selected question to N students, so that every pupil gets exactly one question, and none of them gets the question s/he formulated. Students are automatically notified that the new task is ready, and given a new deadline, by which they have to produce an answer. Every question will hence have (approximately) N answers. The approximation comes from rounding effects while assigning, but the most important source of fluctuation stems from the fact that some student might not accomplish the assigned task for whatever reason.

After the deadline, the system automatically elaborates the new task, which is answer evaluation. Every student is assigned a set composed by one question and its (approximately) N answers. No one is ever assigned a set containing either his/her question or answer. The student is then requested to grade the answers. The grading is on a relative (not absolute) scale: the best answer gets more points, the other ones progressively less. The points are credited to the answer author, and summed to a personal score so that a ranking is produced, as in a social game. Everything is done anonymously, although students obviously have to identify themselves before accessing the system via a log-in procedure, and all activities are recorded with associated identities. Initial registration is carried out by the system, and to minimize the administrative load it is based on self-registration. For security reasons however, registration can only be performed by people having an e-mail account on given domain, such as the one of the university: a parameter which can be set into the system.

We keep track of how many tasks each student performed. To keep students engaged, we grant some bonus on the final exam score. In the Italian system, exams are graded on a 1-to-30 scale, with 18 being the minimum vote to pass the exam. Our bonus comprises a point for participation above a minimum threshold, and another point for the best-ranked students. We chose to put the threshold to 2/3 of the tasks, while the extra point is awarded to the best-performing 1/3 of the students (all these parameters are not hard-wired into the code, but can be set for every new course). These up to 2/30 points are added to the (traditional) exam result after passing it, so they do not contribute to reach the minimum level to pass. Of course, these are again parameters, which can be freely be modified and tuned, and their goal is to motivate students to (meaningfully) participate to the "game".

## 4 SCORING ALGORITHM AND OTHER DETAILS

The algorithm for calculating the score is a delicate ingredient, since it can drive unexpected and vicious users behaviors, which might risk steering the system into unforeseen pitfalls. For instance, if students were free to assign scores without constraint, they could be tempted to give the minimum allowed value to all answers they have to grade, hoping that the others would behave differently. Since nobody evaluates her/his own answer, lowering the competitor's score would ultimately increase one's own probability of ranking high. However, if everybody followed this line of reasoning, all scores would be zero, or the minimum allowed value.

An alternative could be to ask students to simply rank the answers. However, there can be equivalent answers, in which case it would be difficult, if not impossible, to produce a meaningful and fair ranking. Moreover, if e.g. two answers are really bad while two are really good, a ranking does not really reflect the relative gaps.

Another issue comes from the fact that neither the number of evaluations nor the number of answers to be evaluated is constants, as some student might have skipped either the answer or the evaluation task. The scoring algorithm should as much as possible neutralize possible side effects of such situations.

Keeping all these facts into account, we ended up with the following decision. Students are given a number of "tokens" (or "coins"), which is equal to the number of answers multiplied by the mid value of score scale. Hence for instance if the grading scale runs from 1 to 5 and a student has to grade three answers, s/he is given 9 tokens to distribute among the answers, and s/he has to spend all of them to complete the task. Of course s/he could still try to optimize her/his chances by distributing the tokens equally among the answers, but the reasoning that this would be the optimal strategy from his/her point of view is less obvious than the one we discussed first (i.e. to give the minimum to everybody – if no constraints were present). Moreover, such strategy can be contrasted by not allowing all the scores to be equal, or by giving a number of coins, which is slightly different than the number of answers multiplied by the mid value of score scale (e.g. the resulting value plus or minus one), so as to break the symmetry.

Not all the evil can be cured by this recipe: for instance, a question might have a set of all-bad answers and another one a set of all-good answers. The allotted total number of tokens is however the same, which might be considered unfair. However, as we mentioned, in first place the ranking is relative and not absolute. Good or bad luck may hence play a (minor) role, but this happens in all sorts of games – and also in life. After all, we do not aim at building "perfect" rules, but just credible and "fair enough" ones.

We mentioned that another issue is relative to the fact that the number of evaluators could be not constant (either because of rounding factors, of because of students not performing their task): this has to be neutralized in order to have a decent ranking. To do that we evaluate the average number of tokens obtained by each answer: that number is the one contributing to the ranking.

Another compensation comes from the fact that if a student misses one evaluation, s/he would immediately be pushed down the ranking. We want to avoid that, because a students missing one evaluation task early (say due to illness, or to forgetting) would be immediately out of game, and hence demotivated in continuing. Our goal is instead to keep them in the loop as long as possible. We therefore calculate the ranking as based on the average of obtained evaluations, allowing for a maximum number of "absences", which is again a parameter fed in to the system.

All these "fine tuning" strategies have their contraindications, as students who know the details of the algorithm could study a strategy to optimize their performance. For instance, the last compensation mechanism could be exploited: towards the end of the game, a student who achieved the top ranking could avoid performing the last few answer tasks, so as to preserve her/his average. To avoid or minimize such effects, a possibility is not to make fully transparent the algorithm, for instance by not revealing some of the parameters.

We discussed and tested also other variations of the "game" scheme. For instance, in an earlier version of the system we tried to introduce a "Question evaluation". In the end we discarded it for two reasons. The first its that a sensible evaluation turned out to be difficult, as too many parameters were involved, such as originality, interestingness, clarity, difficulty. The second is that it involved one more task, which would add more time to the closing of a context. We aimed at keeping contexts one-week long, which allows for three tasks of two days each, plus one day for the question selection. We feel that for maximum simplicity a task should fit into one week – or multiples of it, since lectures are typically given on a weekly calendar. Of course it would be feasible to have an overlap between two contexts, each of which is in a different phase, but we preferred not to (also because the prototype we used did not foresee that possibility).

Since the "manual" selection of the questions is a time-consuming task, we tried to optimize it. At an intermediate stage, we applied text-similarity and clustering algorithms to attempt creating homogenous groups of questions, and to mark the ones, which are very similar to each other. A similar problem, i.e. investigating semantic similarity in short answers, has been investigated in literature [23]. We found out that our case (classifying questions) was even more difficult, due to the fact that questions are typically very short phrases, and that being on a restricted context, they usually share many of the domain keywords. We were able to achieve some reasonable result, but in the end we preferred to replace the algorithmic grouping with the provision of one or two keywords picked by the students in a (small) set provided by the teacher. This turned out being a reasonable and viable option.

One point we mentioned but did not discuss yet is the presence of duplicate (or very similar) questions. In principle, there is nothing wrong in repeating questions, as every student gets only one o them. Hence in principle even if all students would get the same question, the system would work well the same. The only drawback we saw is that, at present, the system does guarantee that a student never has to respond to his own question, but if there are duplicate ones, one could get a question which is very similar to the one s/he posed. We plan to take care of this issue in the future developments. A simple strategy could e.g. be to add a requirement: every student should be assigned a questions marked with a keyword that is different from that s/he used in the question w/he formulated. This would have the additional advantage of pushing the student to review a topic, which is different from the one s/he chose, so as to guarantee an increased coverage of the lecture set material (which is actually our ultimate goal).

## 5 EVALUATION

We have been running the system for three years in various forms, evolving and consolidating it while we progressively better understood the issues we discussed in the previous section. After the first year, we conducted an evaluation [24], which yielded encouraging results in terms of students' acceptance and perceived utility of the underlying methodology.

Recently (academic year 2015/16) we run a new, deeper evaluation, investigating also aspects other than those touched in the first evaluation. The subset of result on issues that were touched also in the first evaluation has been substantially confirmed or strengthened.

We prepared a form containing 23 closed and three open questions. The closed ones required answering on a 5-points Likert scale. We submitted via web the questionnaire to all the students who enrolled for an introductory course to Object Oriented Programming, taken by approximately 150 first-year Computer Science students and about 30 third year Math students, for a total of 181 students. We assigned a total of 21 tasks (i.e., 7 contexts). 132 students (73% of the total) completed at least 14 tasks, and 47 (26%) obtained the extra point for their top ranking. Out of the 50 students (27%) who did not reach the threshold of 14 tasks, 15 completed less than 5 tasks (i.e. they dropped out almost immediately) and 21 were between 8 and 13 tasks.

All the students were invited to complete an anonymous survey after the course end but before the exam, regardless of the amount of work they performed. We got 117 responses (65% of the students enrolled). Not all classes of students were represented in equal measure: 39 out of 47 of those obtaining 2 points (83%), 59 out of 85 of those obtaining 1 point (69%) and 18 out of 50 (36%) of those who got no points. We do not know how many students actually dropped out during the course: a non-negligible fraction of the first year students end up with no credits after one year, and do not enrol in the second. These data will be available only later this year, so they are not known right now, but it is not unreasonable to assume that at least half of the students who got zero points fall in this category, which would explain why this sector is well under-represented in the our poll.

In the following, the statistical uncertainty of the polls result has been evaluated in the simple random sampling assumption as $\sqrt{(p(1-p)/n)}$ where p is the percentage and n is the sample size.

First, we wanted to examine how students perceived the fact that the question evaluation was not performed by an expert, but rather on a peer base. According to individual interviews, we found that some student expressed some perplexity about peer evaluation, so we asked some questions to dig deeper into the issue.

We explicitly asked if they find this to be an injustice. No strong opinion was expressed: 33±5% agreed and 34±5% disagreed, with the rest having no feeling. Only 5±2% said they felt uneasy about being evaluated by peers, and 25±4% declared they did not like this form of assessment. 55±5% felt this form of evaluation is useful, against a mere 18±4% who thought it is not. A strong signal regards the recognition of the formative value of such operation, with a positive 60±5% against a 13±3% who disagreed. In all these questions, a percentage ranging between 25% and 33% expressed no opinion.

Overall hence the majority of students accepted the peer modality, showing understanding of the formative value of the operation and no negative preconceptions.

We then tried to assess if the extra work induced by the methodology was considered too much, if it was worth the effort, and if so how. These issues were asked with diverse phrasing in more questions, also to verify the overall coherence of the answers.

We asked if the needed effort was excessive: the answer was no for 61±5%, yes for 11±3%. This result is in-line with our previous evaluation [24]. The time dedicated to working with the system was badly spent for 11±3%, and worth spending for 69±5%. Again, time spent on the system was not productive for 6±3%, and productive for 63±5%. We can hence note that repeated answer to similar questions asked in different contexts yield the same results, with percentages that, within the statistical error, are equal, which makes us confident on the obtained results.

Using the system was considered to be a nuisance by 15±3%, and not at all by 51±5%. As with the other questions, the percentage of those expressing no opinions was around 30%.

Let us come to the utility of the approach. A solid 74±4% considers the experience to have been useful, against a bare 10±3%, who is convinced of the opposite. An analysis of the why shows that 67±4% of the students were pushed to review the material covered by the lectures, while 9±3% were not urged to do so. 55±5% think that in this way they are better prepared for the exam than they would have been if the methodology was not employed (once again, 10±3% do not feel the advantage). 60±4% (against 12±3%) believe they kept the course pace in a better way thanks to the recurring duty of the assigned tasks. Instead, only 22±4% reports that their attention in class was increased due to need of preparing for the question/answering process; 26±4% says this wasn't the case and 52% has no opinion on this issue.

We then tried to understand what resources were used to perform each type of task. Let us start with the questions. Since the topic was about programming, a natural place where to look for inspiration or answer could be Stack Overflow , a well known privately held website which features questions and answers on a wide range of topics in computer programming. 71±4% actually never used it for satisfying the tasks goals, and another 19±4% only rarely resorted to it. We got similar figure of Wikipedia: 73±4% never used it for finding stimuli for question-posing, and 23%±4% used it only rarely. A bit, but not much more popular were other (generic) web resources: they were only rarely used by 28±3% and never by 53±5%.

The most used resources were the lecturers' slides (which were regularly made available before each lecture): 75±4% used them (always 37% and very frequently 38%). Only 13±3% never used them. Less habitually used were the personal notes taken in class: 36±4% always or often reviewed them, 36±5% never. Finally, 53% always or often trusted their own memory, 17±4% never.

We analysed similarly the usage of resources for performing the other two types of tasks: answering and giving an evaluation. To favour a comparison among the different tasks, it is useful to group the results: we do that in Table I.

TABLE I.

| Resource | Questions | Answers | Evaluations |
|---|---|---|---|
| Stack Overflow | 10% - 70% | 32% - 35% | 15% - 28% |
| Wikipedia | 4% - 73% | 28% - 29% | 14% - 27% |
| Other web resources | 18% - 53% | 51% - 37% | 33% - 30% |
| Teacher's slides | 78% - 13% | 81% - 17% | 63% - 21% |
| Students' notes | 37% - 30% | 38% - 24% | 35% - 24% |
| Students' memory | 60% - 24% | 52% - 26% | 69% - 16% |

Each cell contains two numbers: the percentage of positive answer (always/often) and the negative ones (never), The missing percentage responded "rarely". For sake of readability, in the table the statistic uncertainty is not reported (it can be found in the above text for the questions, and it is similar for the other two categories).

In this table, the data refer only to the 98 respondents who achieved at least one point (i.e. those who completed at least 2/3 of the assigned tasks): the data reported earlier for question answering included also the 18 respondents not reaching the threshold of 2/3 of the tasks.

The answer task is (predictably) the one for which students maximized the use of resources other than their own memory. In fact occasionally, by casual inspection of the responses, we could spot some phrases literally taken from Wikipedia. It is also evident from the table that the evaluation is the one for

which the least use of external information is done: the mostly used resource for this task is students' own memory. Curiously, students' notes taken in class do not play a major role in any of the tasks.

Recurring to resources other than their own memory was less for the evaluation task than it was for the other two.

Overall, teachers slides were definitely the preferred source of information, which reinforces the indication that the system pushes students to review the material presented in class, and that this happens in all three phases: questions, answers and evaluation. Hence for every context students review three times (at least part of) the material discussed in class, which fully accomplishes the methodology's raison d'être.

It is interesting to observe the difference in the behaviour of the most successful students (the ones getting 2 points) against the other ones – we compare them with those who completed most of the tasks but only achieved 1 point. Table II reports in each cell the values of positive responses only (i.e. the respondent used the specified resource always or often) for students ranking best (i.e. those achieving 2 points) and for those who got 1 point (i.e. those who completed most of the tasks, but were in the lower two thirds of the ranking. Again, statistical error is not reported for easier reading but it is at 3% for results in the intervals 7%-15% and 85%-93%, 4% in the intervals 15%-30% and 70%-85% an at 5% in the range 30%-70%.

The most remarkable difference is about the usage of one's own notes, where the share of successful students using them in all phases doubles the share of less successful ones. Less striking, but still noticeable, is the larger use of teacher's slides among the successful students.

TABLE II.

| Resource | Questions | Answers | Evaluations |
|---|---|---|---|
| Stack Overflow | 10% - 10% | 33% - 31% | 13% - 17% |
| Wikipedia | 3% - 5% | 33% - 24% | 18% - 12% |
| Other web resources | 18% - 19% | 59% - 46% | 41% - 27% |
| Teacher's slides | 85% - 73% | 85% - 78% | 72% - 58% |
| Students' notes | 54% - 25% | 56% - 25% | 50% - 25% |
| Students' memory | 62% - 59% | 51% - 53% | 74% - 66% |

Finally, we tried to extract some other indication about the usage of the system. As we mentioned, the scoring algorithms changed over the years: an (unfortunate) side effect is that the actual score obtained is now revealed to the students only at the end of the "game" and not while they are playing it, because the user-interface portion of the code was never adapted to the new scoring algorithm. Also, at present students only see the questions/answers/evaluations contained in their own tasks, and not the whole set, which could also present a didactic value. We asked students if they considered such lack of feedback a problem. 43±5% manifested no opinion, 33±4% would have liked more feedback and 24±3% seems not to care about this issue.

Occasionally the system, which was an experimental prototype, presented some technical problem (such as e.g. difficulties on logging-in, or early session expiration). These problems disturbed 35±4% of the students, and were not an issue for 38±54% (again, some students expressed no opinion).

As a last point, we asked students if they would like the same system to be employed on other courses, and the answer was yes for 61±5%, and no for 20±3%.

## 6 DISCUSSION AND CONCLUSIONS

We presented a methodology for peer collaboration among students based on a (mildly) competitive social game. The methodology is best applied to courses with large attendance, such as the typical introductory bachelor courses, where the audience can be of at least 100 students, and might even reach two or three hundreds. Although it would be applicable in small classes (e.g. 20 students), the small number would not guarantee anonymity and sufficient randomization. On the contrary, it would be applicable to very large audience, such as that of Massive Open On-line Courses (a.k.a. MOOCs). The only hurdle for this perspective is the question selection, a task presently performed manually and hence non-scaling. We plan to further investigate strategies to at least reduce the need of manual

intervention, as this is the only hurdle for scalability. Possibilities would include a random selection of questions, with an a-posteriori evaluation of bad or unsuitable questions: for instance these could be indicated by students and then examined by the teacher, and they could result in a penalty for their authors.

Our approach can be considered a sort of evolution of other peer-based evaluations present in literature, from which it however differs radically. Our main goal was to find a way for convincing students to study more regularly, reviewing the material presented in class. In fact, they end up revisiting the material three times: when they answer a question, when they respond, and when they evaluate responses. Hence, the approach forces them to timely and regularly recheck the content of the lectures. The small but carefully planned incentive achieves both the goals of keeping them hooked, and to promote a serious engagement.

The methodology has been experimented over three academic years, and evolved until it reached the form described in this paper. In its last incarnation, it has been evaluated yielding the results we reported in section V. The outcome, which was encouraging when we run a first implementation of the idea, has been consolidated and deepened.

At the end of the course, students manifested the clear perception that this process is useful, it allows them to better keep the pace of the course and to achieve a better final preparation. They think that, thanks to the pushes given by the system, their study has been more productive and effective than it would have been without. The extra work needed is not considered excessive.

Although the system interface is still old-fashioned and rather rudimental, they liked the system up to the point of expressing the wish to use it in other courses

No major issues have been reported relative to the fact that peer evaluation is less precise than expert evaluation: in our case, evaluation is essentially formative rather then summative, even though the final “award” slightly influences the final exam score, so it has a mild summative component. Students understood the formative value of the approach.

The additional burden on the teacher is extremely limited, as it is confined to provide the set of keywords that define the context, and to manually filter the subset of questions that are actually given to the students. One might object that the teacher could propose a better set of questions, but this actually would defeat the purpose of the methodology, which is to make the students active and to prompt them to rehash the lectures material over and over. In fact, as we already mentioned, the true educational value of this approach resides, as often happens, more in the process itself than in its product.

As a side effect, we could ascertain that best performing students are characterized by a larger use of their own notes and of the teacher-provided learning material. Although this is an almost obvious consideration, it is nice to see it well supported from the statistical evidence.

Future developments include the rewriting of the software so as give it a better user interface and to fix some minor bugs, after which it will be released as open-source. We also plan to deploy it internally on a larger scale.

## ACKNOWLEDGEMENTS

This work was made possible by the Italian MIUR under the “Città Educante” Cluster project.

## REFERENCES

[1] Desmarais, M. C., & d Baker, R. S. “A review of recent advances in learner and skill modeling in intelligent learning environments.” User Modeling and User-Adapted Interaction, 22(1-2) 2012 , pp. 9-38.

[2] Rowntree, D. Teaching through self-instruction: How to develop open learning materials. London: Kogan Pag, 1990.

[3] Morgan, C., & O'Reilly, M. Assessing open and distance learners. Psychology Press., 1999

[4] Kennedy, G. E., & Cutts, Q. I. “The association between students' use of an electronic voting system and their learning outcomes.”J.of Computer Assisted Learning, 21(4) 2005., pp.260-268.

[5] D. Duncan, “Clickers in the classroom,” Addison: San Francisco, CA, 2005

[6] Beth, S., Kohanfars, M., Lee, J. Tamayo, K., Cutts, Q. "Experience report: Peer instruction in introductory computing." Proceedings of the 41st ACM technical symposium on Computer science education. ACM, 2010.

[7] Wang A.I., The wear out effect of a game-based student response system, Computers & Education, Volume 82, March 2015, pp 217-227,

[8] Chuang Y.T. “SSCLS: A Smartphone-Supported Collaborative Learning System, Telematics and Informatics, Volume 32, Issue 3, August 2015, pp. 463-474,

[9] Crouch, C. H., and Mazur., E. "Peer instruction: Ten years of experience and results." American Journal of Physics 69 (2001): 970.

[10] Fagen, Adam P., Catherine H. Crouch, and Eric Mazur. "Peer instruction: Results from a range of classrooms." The Physics Teacher 40 (2002): 206.

[11] Ronchetti, M. "Using video lectures to make teaching more interactive" International Journal on Emerging Technologies in Learning, v. vol. 5, n. no. 2 (2010), p. 45-48

[12] Porter L., Lee C.B., Simon B., and Zingaro D.. “Peer instruction: do students really learn from peer discussion in computing?”. In Proceedings of the seventh international workshop on Computing education research (ICER '11). ACM, New York, NY, USA, 2011, pp. 45-52.

[13] Ronchetti, M. “Perspectives of the Application of Video Streaming to Education” in Ce Zhu, Y. Li, Xiamu Niu (eds), Streaming Media Architectures, Techniques, and Applications: Recent Advances, Hershey PA, USA: Information Science Reference, IGI Global, 2011, pp. 411-428.

[14] Ronchetti, M. Video-Lectures over Internet: The Impact on Education in G. Magoulas (ed.) E-Infrastructures and Technologies for Lifelong Learning: Next Generation Environments, New York: IGI Global, 2011, pp. 253-270. - doi: 10.4018/978-1-61520-983-5.ch010

[15] Topping, K. “Peer Assessment Between Students in Colleges and Universities, Review of Educational Research, Fall 1998 vol. 68 no. 3 249-276

[16] Ashenafi M.M. “Peer-assessment in higher education – twenty-first century practices, challenges and the way forward,” Assessment & Evaluation in Higher Education, 2015, pp. 1-26

[17] de Raadt, Michael, David Lai, and Richard Watson. "An evaluation of electronic individual peer assessment in an introductory programming course." In Proc. of the Seventh Baltic Sea Conf. on Computing Education Research-Volume 88. Australian Computer Society, Inc., (2007).

[18] Denny P., Hanks, B. and Simon B.. 2010. “Peerwise: replication study of a student-collaborative self-testing web service in a u.s. setting”. In Proceedings of the 41st ACM technical symposium on Computer science education (SIGCSE '10). ACM, New York, NY, USA, 2010, pp. 421-425

[19] Paré, Dwayne E., and Steve Joordens. "Peering into large lectures: examining peer and expert mark agreement using peerScholar, an online peer assessment tool." Journal of Computer Assisted Learning 24.6 (2008): 526-540.

[20] Kaufman, J. H., & Schunn, C. D. (2011). “Students’ perceptions about peer assessment for writing: their origin and impact on revision work”. Instructional Science, 39(3), 387-406.

[21] Ledwith, Ann, and Angelica Risquez. "Using anti-plagiarism software to promote academic honesty in the context of peer reviewed assignments." Studies in Higher Education 33.4 (2008): 371-384.

[22] Pond, K., Coates, D., & Palermo, O. “Student experiences of peer review marking of team projects.” International Journal of Management Education 6 (2), 2007, pp. 30-43

[23] Mohler, M., & Mihalcea, R. (2009, March). “Text-to-text semantic similarity for automatic short answer grading”. In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (pp. 567-575). Assoc. for Computational Linguistics.

[24] Ashenafi, M., Riccardi, G. & Ronchetti, M. "A Web-Based Peer Interaction Framework for Improved Assessment and Supervision of Students" in World Conference on Educational Media and Technology 2014, Tampere, Finland: (AACE), 2014, p. 1371-1380.