Comparing Three Approaches to Assess the Quality of Students' Solutions

Niels PINKWART and Frank LOLL
Clausthal University of Technology, Germany

Abstract. Collaborative filtering has potential for usage in Social Semantic Web e-learning applications: the quality of a student provided solution can be heuristically determined by peers who review the solution, thus effectively disburdening the workload of teachers and tutors. This paper compares three collaborative filtering algorithms which are based on different paradigms – one based on the assumption that a student who can classify the quality of given alternative solutions correctly is also able to provide a high-quality solution himself, another following a classical peer review paradigm, and a third being a mixture of both. An evaluation of the algorithms with data collected during a lab study showed that all algorithms can classify peer solutions correctly. Thus, these approaches have high potential as a support for classic academic teaching in larger classes.

Introduction

The term *Social Semantic Web* (SSW) describes an emerging design approach for building and using *Semantic Web* applications which employs *Social Software* and *Web 2.0* approaches. In SSW systems, groups of humans are collaboratively building domain knowledge, aided by socio-semantic systems [1]). The collaboration process of the users in SSW systems can have multiple purposes like group based creation of domain ontologies or the collaborative semantic classification of content (determination of properties of ontology elements). Both are potentially valuable in education. While the former can be a technique for collaborative knowledge building through jointly structuring an unknown knowledge domain, including the discussion of domain concepts and relations, the latter allows for jointly annotating or evaluating learning materials [2] and for heuristically determining the quality of task solutions through collaborative efforts.

This paper presents an example for the latter type of SSW systems in education. We compare three different approaches for solving the problem of automatically determining student solution's quality. Two of these approaches make use of peer reviews: the quality of a student's task solution is determined heuristically by assessments of other students. Typical points of critique concerning a peer review approach in education are related to the students' lack of knowledge and experience in assessing task solutions and to the risks of intentional manipulation [3, 4]. Yet, this form of using SSW approaches for education disburdens tutors and, at the same time, provides the possibility for students to train their critiquing skills. If there are tasks which allow for more than one correct solution, students have a chance to learn different acceptable ways to solve a problem. Also, students may empathize with other learners' problems easier and understand reasons for wrong task solutions sometimes better than experts, which can make their reviews sometimes more valuable than those of experts [5].

In spite of their potential however, peer review mechanisms have only been rarely used and empirically evaluated with respect to their effectiveness in the e-learning sector till now. Some of the few existing systems that make use of a peer-review approach to assess solutions quality are PeerGrader (PG) [6, 7], SWoRD [8], and LARGO [9]. While the systems' approaches are promising, they are limited in several ways: They are either specialized for a particular application area such as legal argumentation (LARGO) or writing skills training (SWoRD), or they involve a rather complicated and long-term review process (SWoRD, PG). In this paper, we present and compare 3 heuristics for estimating the quality of student solutions that are not constrained to a specified task area and that do not require time-consuming re-writing phases but only short quality assessments. While the first algorithm implements a "plain peer review" strategy, the second algorithm additionally relies on the hypothesis that a student who is able to correctly assess other student's solutions (i.e., classify them as poor or good) is likely to have provided a good solution himself – since a good judgment about solution quality can only be made when a task has been understood and solved. The third heuristics relies only on the latter hypothesis and does not include any peer reviews.

1. Peer Review Based Heuristics

The first heuristics consists of two components – an *evaluation rating* and a *quality rating*. The application scenario for this algorithm is then when students work on a task and provide a solution, they are asked to assess some alternative solutions afterwards.

The first component of the heuristics is the *evaluation rating*. Once a student has provided a solution, it is presented to other students to be assessed. All assessments get collected, averaged and weighted (assessments of better students get higher weights). An illustrating example: Assume a solution gets four assessments $w_1=0.9$, $w_2=0.2$, $w_3=0.4$ and $w_4=0.5$ from students whose own solutions have (system-internal) *quality ratings* of $q_1=0.8$, $q_2=0.1$, $q_3=0.3$ and $q_4=0.7$. The first assessment gets a higher weight than the others because the student who provided it has a higher *quality rating* as compared to the others. His opinion is thus considered as more important than the other students' opinions by the system heuristics. Then, the *evaluation rating* for the assessed solution is calculated by:

$$eval = \frac{1}{\sum_{i=1}^{j} q_i}\left(\sum_{i=1}^{j} w_i \cdot q_i\right) \Rightarrow \frac{1}{1.9}(0.9 \cdot 0.8 + 0.2 \cdot 0.1 + 0.4 \cdot 0.3 + 0.5 \cdot 0.7) \approx 0.63$$

The *quality ratings* are calculated based on the *evaluation rating* scores with an additional damping factor to assure that for solutions with very few peer reviews, no extremely low or high scores are possible. The *evaluation rating* gets weighted dependent on the number of received assessments *p* for a solution. Its impact thus increases with an increasing number of assessments. In the formula, *base* denotes a starting value (default: 0.5), and the constant *c* corresponds to the weight of this starting value relative to the weight of the peer reviews. In our study described in the next sections, we have chosen $c=3$. Thus, the *quality rating q* is calculated by:

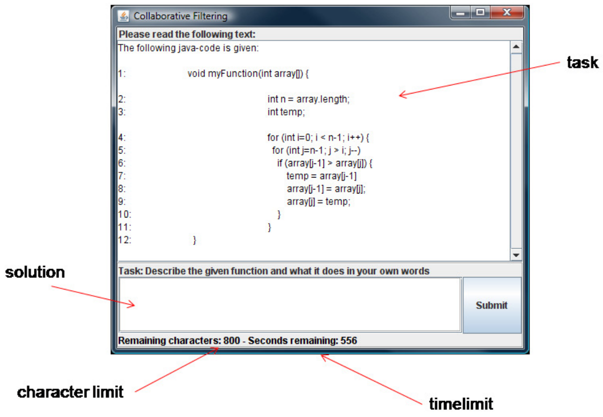
$$q = \frac{c}{p+c} base + \frac{p}{p+c} eval$$

2. Base Rating Heuristics

The algorithm presented in the previous section assumes an equal start rating of *0.5* for all the student solutions (if no peer reviews are available, then $p=0$ in the quality rating formula). Based on the assumption that a student who can classify the quality of given alternative solutions correctly is also able to provide a high-quality solution himself, the heuristics can be improved by replacing the static start value with a dynamically calculated *base rating* for a student's solution. The rationale here is simple: students who classify good solutions as good (and poor ones as poor) are likely to have understood the problem, and thus have probably provided a good solution themselves. Once a student provided n assessments $w_1, \ldots, w_n$ for n other student's solutions (which have *quality ratings* of $q_1, \ldots, q_n$ themselves), the *base rating* is calculated (see below). Independent of the quality of the solution that a student has to assess, this formula allows *base ratings* between *0.0* and *1.0*. To illustrate this: Assume there are solutions with *quality ratings* of q_1=0.35, q_2=0.6 and q_3=1.0. The worst ratings a user might make here, i.e. the ratings with the highest possible difference between q_i and w_i, are $w_1=1.0$, $w_2=0.0$ *and* $w_3=0.0$.

$$base = 1 - \frac{1}{n}\sum_{i=1}^{n} \frac{|\, w_i - q_i \,|}{\max(q_i, 1 - q_i)} \Rightarrow 1 - \frac{1}{3}\left(\frac{|\, 0.35 - 1.0 \,|}{\max(0.35; 0.65)} + \frac{|\, 0.6 - 0.0 \,|}{\max(0.6; 0.4)} + \frac{|\, 1.0 - 0.0 \,|}{\max(1; 0)} \right) = 0.0$$

In summary, we have so far proposed two algorithms to heuristically determine the quality of student solutions. While the first (called PR ONLY) makes use of a classic peer review approach with a fixed "base value" for those solutions that were not assessed yet, the second one (PR + BASE) replaces this base value with a base rating, calculated dynamically. A third variant (BASE ONLY) is to use *only* the base rating formula. We tested the three algorithms with data that we collected for a lab study described in [10]. Originally, the study was conducted to evaluate a slightly different algorithm; however parts of the log data – which essentially contains the student solutions and their peer reviews – can be used to test the algorithms described in this paper as well. We next briefly describe the software and the experimental procedure, and then present the results of our analysis, focusing on the question which (if any) of the algorithms works best for classifying student's solution quality.



Please read the following text:

The following java-code is given:

```
void myFunction(int array[]) {

        int n = array.length;
        int temp;

        for (int i=0; i < n-1; i++) {
          for (int j=n-1; j > i; j--)
            if (array[j-1] > array[j]) {
              temp = array[j-1]
              array[j-1] = array[j];
              array[j] = temp;
            }
        }
}
```

Task: Describe the given function and what it does in your own words

Remaining characters: 800 - Seconds remaining: 556

Figure 1. User interface

3. Implementation

After an initial login to the system, students go through the following phases as they use the system: (1) Work on task (Figure 1 shows a sample task on Java programming), (2) Assess 3 alternative solutions of different students on a scale from 0 to 10 for the just completed task (Figure 2), and (3) Repeat steps 1 & 2 as long as there are tasks to complete. The alternatives to be assessed in step 2 were chosen randomly. Solutions with less received reviews were preferred and a set of alternatives with similar quality was avoided.

4. Study Description

We conducted a controlled lab study in May 2008 at Clausthal University of Technology with 23 students, 11 female and 12 male. The participation was voluntary and paid. The students had to work on 12 tasks from various knowledge areas. The tasks were of the following types: (1) text summaries, (2) text interpretations, (3) knowledge tests without possibility to guess, and (4) knowledge tests with possibility to guess.

In the first task type (text summaries), the participants got articles dealing with different topics (e.g. a text about Second Life), which differed in their level of complexity and required, at least in parts, domain-specific knowledge to get the main points, which had to be summarized. The second task type (text interpretations) focused a fact-based news article about the take-over of DoubleClick by Google. The students were asked to mention and discuss possible concerns towards privacy of customers based on facts in the text. The third task type (knowledge tests without possibility to guess) consisted of 5 tasks where guessing was not possible (e.g. the calculation of a derivative of a function to calculate the slope at a given coordinate). The last task type (knowledge tests with possibility to guess) consisted of problems which could at least be approximated by logical deduction even without specific knowledge. An example here was the estimation of the population of Austria by means of a text about the size of the country.

The students had an overall time limit of 75 minutes. Furthermore, each task had a character limit as well as a time limit (see Figure 1), which served as orientation what kind of solution was expected. All participants were instructed to assess alternative solutions even if they did not know the correct solution for a task. To solve the cold-start problem [11] and offer alternative solutions also for the first participants who took part in the study, we provided 3 alternative solutions of different quality per task.

5. Results

To evaluate the results of the heuristics, all solutions were manually graded independently by two human experts (a professor of computer science and an advanced graduate student) on a scale from *0* to *10*. To check whether the human graders' assessments were similar (if human graders disagree, then a realistic baseline for the heuristics is hard to define), we first calculated inter-rater reliability based on Cronbach's Alpha [12]. We received excellent agreements with α-values between 0.834 and 0.982 in the different task groups. Therefore we averaged both human graders' scores and used the resulting "human grading" as a baseline for evaluating the algorithm results. Specifically, to analyze the quality of the three different heuristics, we computed the correlation between the algorithm output and the human grading.For the "PR ONLY" heuristics, the result was a correlation of 0.53 – thus, a medium-to-large correlation between the algorithm's quality heuristics and the human grading. Apparently, the peer review paradigm worked quite well for the test scenario. For the "PR + BASE" algorithm, the correlation was not significantly better (0.54). We conclude from this that adding the (relatively complicated) base rating calculations did not change much in terms of the algorithm's predictive power. However, using only the base ratings (algorithm "BASE ONLY") also still produced a considerable correlation to the human grading of 0.46. For calculating the "BASE ONLY" values (cf. section 2: quality scores needed to calculate base ratings!), we used the human provided grades, so that this study confirms the hypothesis that a student who is able to recognize good (or poor) solutions is also likely to have provided a good solution himself.

6. Conclusion

Overall, the data analysis confirmed our expectations and even partially exceeded them. All three tested *SSW* methods for building quality related metadata about learning objects (here: student solutions) produced acceptable results, measured in terms of correlation to data provided by human experts. Particularly, it did not matter if "classical" peer review approaches were used, if a different strategy that relies on recognizing good students based on their correctly classification of others' solutions was followed, or if the two were combined (though the latter produced the best results).

References

[1] Morville, P. (2005). Ambient Findability. O'Reilly Media.

[2] Walker, A., Recker, M. M., Lawless, K., Wiley D. (2004). Collaborative Information Filtering: a review and an educational application. International Journal of AIED, Vol. 14, No. 1/2004 - pp. 3-28

[3] Dancer, W. T., Dancer, J. (1992). Peer Rating in Higher Education. Journal of Education for Business, 67, pp. 306–309.

[4] Mathews, B. (1994). Assessing Individual Contributions: Experience of Peer Evaluation in Major Group Projects. British Journal of Educational Technology, 25, pp. 19–28.

[5] Hinds, P. J. (1999). The Curse of Expertise: The Effects of Expertise and Debiasing Methods on Predictions of Novice Performance. Journal of Experimental Psychology: Applied, 5, 205–221.

[6] Gehringer, E. F. (2001). Electronic Peer-Review and Peer Grading in Computer-Science Courses. In Proceedings of the 32nd SIGCSE Technical Symposium on Computer Science Education, February 2001, Charlotte, North Carolina, United States, pp. 139 – 143.

[7] Lynch, C., Ashley, K., Aleven, V., & Pinkwart, N. (2006). Defining Ill-Defined Domains; A Literature Survey. In Proceedings of the Workshop on Intelligent Tutoring Systems for Ill-Defined Domains at the 8th International Conference on Intelligent Tutoring Systems (pp. 1-10). Jhongli (Taiwan)

[8] Cho, K., Schunn, C. D. (2007). Scaffolded Writing and Rewriting in the Discipline: A Web-Based Reciprocal Peer-Review System. Computers & Education, Vol. 48 (3).

[9] Pinkwart, N., Aleven, V., Ashley, K., Lynch, C. (2007). Evaluating Legal Argument Instruction with Graphical Representations Using LARGO. In Proceedings of the 13th International Conference on Artificial Intelligence in Education (p. 101-108). IOS Press.

[10] Loll, F., Pinkwart, N. (2009). Using Collaborative Filtering Algorithms as eLearning Tools. In Proceedings of the 42nd Hawaii International Conference on System Sciences.

[11] Maltz, D., Ehrlich, E. (1995). Pointing the Way: Active Collaborative Filtering. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.

[12] Cronbach, L. J. (1951). Coefficient Alpha and the Internal Structure of Tests. Psychometrika, 16(3).