Data mining, some 6 to 12 months after

This is a course retrospective promised for the course Data Mining, which was held during the Spring 2013. For the report, I looked at the feedbacks available on the course pages of the courses held during Spring 2011 and 2012, and the feedbacks from 2013 that I received from Hannu.

To my understanding, the course has now went through three iterations using a problem based learning approach, where a large amount of responsibility is given to the students. The students that are willing to take the course either work on problem sets assigned by the teacher in small groups (4-5 members) during the course and must perform a considerable amount of reporting and other activities, or complete the course by studying the course book on their own and taking a written exam.

In the past, the teaching in the Data Mining course has been organized using a traditional lecture-oriented approach with a set of take-home exercises, that have been went through at a weekly exercise session. I found the course results from Spring 2010, where some 60% of the students that participated in the course passed the course. In the more recent versions, the pass-rates are considerably higher (e.g. in 2011, 90% of the students taking the course data mining passed the course, either by participating in the problem based learning -style of teaching, or simply by participating in an exam).

A few words on learning at a University context

Problem based learning and inquiry based teaching methods have been researched quite a bit in (CS) education literature, where most of the focus has been put into undergraduate education. The major reported benefit has been typically an increase in the course retention rates (which is easy to observe :)), especially when the comparison is a more traditional lecture based teaching approach. However, the benefits do not stop at increasing retention, but are more related to what the students learn. The situative view to learning suggests that knowledge and context are inseparable, also meaning that the whole educational setting affects learning outcomes. This means that the tasks that the students do, the way they do them, the students they collaborate with, the teaching approach, …, and even the way they spend their free time, all contribute to the final outcomes. Typically, parts of the learning is also unintentional and unexpected, and related to other things than the outlined learning objectives of a course.

For example, some students may learn that “lectures are boring”, or that “you can pass this course by simply practicing these three questions, as they always appear in the exam”. Here, the community plays a role; students discuss with other students and teachers on which teachers and courses are good, which are bad, and how they have succeeded or failed in a course. Unfortunately, there are also students that boast with “I read the material of course XXX during the weekend, went to the exam, and passed”, considering passing a course the overall goal. This is actually emphasized in traditional (pr/t)eaching, as the focus of the course is often put on the final exam: “this is what you need to know in the final exam”. Many times, sadly, the only way to see as a student whether you knew what was needed is visible only after the final exam, which is held after the course, and for some, the course grades and corrected exams are available maybe a month after the actual course exam.

Perhaps it is obvious that there is a need for change, especially if you're not at Oxford or Harvard, or any other of the really big players. Having a University facilitate it's teaching using the traditional lecture-based teaching approach is often based on “because this is how I was taught” or “this is the way the big players do
it" -- not a really good excuse. The players with a limited amount of resources should try to apply those resources as effectively as possible. If we consider the overall state of CS education in Finland, there are already enough challenges even without the way teaching is often facilitated. For example, the University of Eastern Finland is struggling to even fulfill their quota of students that are taken into the BSc program. It's good to see some change in the teaching arrangements, but, are the students ready?

**Initial experience**

The first days at a University (or in any facility) matter a lot. If students are brought into a large lecture hall, where someone talks to them, a mass of peons, about high values etc., they learn that that is the way things are done at a University. Especially when students have the freedom (or, the unfortunate freedom?) of choosing pretty much any courses they like, many tend to feel being part of a mass -- part of something and still also nothing, as when the masses are gone the “university identity” is still sought after. One of the reasons for having many first year students fail their courses is simply due to the context change; in the past, for many of them, the structure of their studies and schedule has been set by e.g. the high school. Many of them have also learned that “school is easy”, and that “to pass, you just need to sit in the classes”.

The first day in a course matters also a lot. By simply engaging the students as individuals, as was done in the first class of Data Mining course (everyone told something about themselves, i.e. who they were, how far they were in their studies, why were they taking the course), matters a lot, especially when compared to some courses, where a lecturer may simply walk to the front and start going through a slightly outdated slide show, even if the class has only a handful of students. Another factor that contributes to the success is emphasizing commitment from day one. Simply stating that “If you are not going to participate in this group work, which is going to demand a lot of work, this room is not for you” (or something along those lines), i.e. demanding from the students, adds incentives for commitment. After that, being an anonymous student that seeks to blend into the gray walls is nearly impossible.

**Educational system and the students**

One of the major challenges in tertiary education in Finland is that a majority of the students work, and many seem to consider their studies more like a hobby. As some players in the software industry, where a majority of our students end up, care little about degrees, and more about the performance, it is easy to squeeze effort from undergraduate and graduate students, who are supposed to be also putting effort into their studies. Naturally, the monetary compensation offered by “the dark side” is not bad either, and jumping back to full-time studies after working in the industry can be very hard.

There’s been some actions from the state that have tried to remedy this. For example, a recent effort was limiting the “study right” to a set number of years, after which the students’ would lose their study right if they would not proceed in their studies. A good effort, but unfortunately, so far it has only increased the administrative overhead as there is a loophole stating something along the lines “a student will get additional time for studies if s/he can show a valid study plan”. Another action has been to increase the amount of monetary support that Universities receive from the state from students that finish their MSc; something that has been acknowledged partly by increasing the amount of non-finnish MSc students, which on the other hand has also created a few additional challenges in facilitating learning.

As the majority of Finnish students have gone through a specific track, the teacher may assume that a set
of courses have been taken beforehand. For the group work, students may assume that everyone can work with at least one programming language, and are also familiar (and ok!) with collaborative tools such as IRC etc. In addition, all students should be proficient enough with the working language of the course, in our case typically english.

When comparing problem-based learning to traditional lectures, students have more responsibility in the problem-based approach. While it is evident that this is good for first year undergraduate students, based on my experiences in the course, it is about time that more responsibility is given to MSc students as well. Instead of just taking the “rote learning track”, the current approach has students taking more responsibility and forces them to hone their information seeking and problem solving skills, consider their personal learning skills and practices, collaborate with team members, and, at least in theory, helps build intrinsic motivation towards the topic.

Course tasks and what is learned

As the goal of the course is put on problems that need to be solved, instead of learning the content so that it could be repeated in an exam, in hallway discussions it was clear that many received “inner rewards” from solving the problems and understanding the needed topics; being able to understand what was going on was awesome, and it was also seen valuable for the upcoming tasks as the course topics mostly built on top of each other. As the problems are explicit (not explicitly defined, but explicit in a sense that the path to take is relatively clear), there was an existing direction that the students know that they should pursue. In the end, although many still saw the teaching methodology as “the approach used in the course”, the problem solving skills that were honed are lifelong learning skills, which should be emphasized from day one.

The role of the lectures in the course can be questioned, but their value is evident if the students’ do their homework before. When considering problem based learning (or pretty much any inquiry based teaching approach), the main role of the lectures (/ exercise sessions) is helping students fix their existing misconceptions. Even if something is not known, it is typical that there is some sort of “hunch”, which can be either wrong, or right, or somewhere in between. Having an opportunity to scaffold the students in a way that their existing knowledge is molded towards more correct ideas is important, and for this, a bi-directional feedback mechanism is needed.

Many of the students are ready for problem based learning, while some are not. Especially if a student has learned a very specific way of doing things in the past, e.g. completing courses by memorizing facts & algorithms, changing the way of thinking can be very time consuming. As students in the course are working in groups, there may be some that need to be dragged, and on the other hand some, that should be slowed down, i.e. students, that have challenges in sharing responsibilities and working in teams.

Course feedback

Before considering the student feedback, let us think about the foremost observations. The retention rates are higher, and the course organizer(s) have expressed the feeling that the students’ have learned a lot more (than they would have learned in a traditional course). In the hallways, the course is in general talked about in a positive light, and students wondering whether they should take the course are usually encouraged to take it by other students: the course is considered valuable both within “the student community” and in many of the written student feedbacks.
It is evident that many of the students consider the course laborious. This is not surprising, especially when we consider “generic” master level courses (at least in the SE program), which are often quite easy and the learning objectives are well known. On the other hand, at least a part of the perceived workload in the course can be explained by the amount of uncertainty: there were no simple answers that students often are accustomed to, and simply not being sure about something, especially when students have the feeling that their work is assessed, can by itself be stressful.

Some students voice out that their effort is partially misdirected, or put into “unrelated” areas. One of the things that was criticized was the amount of reporting, which was felt unnecessary by some students. On the other hand, having several “levels” of reports enforced recall of the topics -- especially if the work is done with “thought” and not crunched together; group reporting can also be seen as a task designed to help learn and improve collaboration skills.

**Challenges and possible solutions**

Here, I consider some of the challenges and possible improvements that could be tried out;

*Focusing effort*

As some students feel that their time was misdirected due to the amount of reports written (i.e. personal learning journal, group journal etc) and the amount of presentations watched (and prepared), it might be useful to consider which ones to retain and which ones to remove. I think that the group work is a major contributing factor to the increased retention rates, and hence do not see removing it as a good option. The personal learning journal on the other hand can be seen as a way to verify that the students have worked on the tasks themselves, while the group reports act as a way to verify that the students work together.

The effort aspect has a lot to do with the students’ time management skills as well as with the communication inside the group. As some students suggested in the feedback, perhaps the group sizes could be reduced a bit. In addition, it might make sense to have a small set of small, relatively easy individual assignments as well, which would give the students the feeling of proceeding in the course. It might be a good idea to have these tasks be related to the topics for the upcoming week, which would enforce that the students would familiarize themselves with the required topics already before the lecture; which in turn perhaps could increase the amount of discussion in the class (this is actually typically hard in the Finnish system, it is rare for teachers to have a longer dialogue with students during class). Perhaps handing in the solutions for the easy assignments could work as the personal learning journal.

Some of the overlapping in presentations could also be reduced. Instead of semi-weekly presentations on the results students’ have received from a learning task, the groups could also share responsibilities for creating “introductory lectures” that contain presentations on the topic at hand. Here, with introductory lectures I mean presenting a problem, when or in what kind of data it occurs, and a way to solve it (perhaps creating a visual simulation of an algorithm on a dataset). As an example, group A could do a presentation on mining temporal data, group B on mining sequential data, and as only a few students/researchers/persons) understand formulas when they’re first introduced in a presentation (they’re often the product of work spent on understanding how something works), the introductory presentation would only show how something works. When the groups are working on the tasks, one or two groups could still do a presentation on the results, while others would comment etc; group responsibilities would rotate. Each
group would still work on the assignments, and would create a group report when a task is finished. In addition, each group would have to comment on the reports from other groups (commenting responsibilities could be shared between group members (e.g. person A reviews and comments on work from group B, ...)), and perhaps the groups would also have to respond to the comments?

Another approach that could be considered is using worked examples in the lectures, i.e., having a specific problem case that the instructor would work through during (each?) lecture. This would potentially both help the students to improve their own working habits and correct possible misconceptions, as well as build a mental model on an approach that they could/should try out in the next assignments. However, it might be a good idea to provide alternative actions for more advanced students, as worked examples have been noted to be ineffective if the students already "know what they are doing" (in undergraduate prog education).

*Helping students with e.g. poor programming background*

Some students had challenges with implementing the very basic algorithms in the course, which is quite natural as there are students with varying backgrounds. While I do believe that programming should be a part of the course, it might be a good idea to first provide students again with some worked examples, and have them complete the most important algorithms e.g. in a step-by-step-manner, where there would be subgoals that the students could aim for. This would both help the students to become familiar with the initial structure of the implemented algorithms, as well as possibly help them to structure the code in a way that adding features would be easier.

Although it should not be the goal of this course, a generic set of practice programming exercises could be handed out to the students.

*Avoiding freeloading*

Unfortunately, there are still students that rely on the team handle at least parts of their tasks. I do not see this completely going away, as each student has their own set of obligations as well as motivation levels. However, one possible way to verify that the students have even read through the required material beforehand would be to use a simple set of readings that everyone has to go through (or the aforementioned weekly small exercises that they would have to complete). Verification could be done by enforcing that everyone has to comment on the readings, perhaps using a system, where the students can actually annotate the text that they have read, and add the comments into the annotations.

In addition to forcing the students to prepare better to the lectures, this could also act as a feedback point for figuring out major misconceptions and harder topics, as well as help the students improve their skills for reading scientific text.

*Some additional notes*

So far there is no magic bullet to learning, and having students perform concentrated effort into trying to understand and solve something, also learning it on the side, is pretty much the only thing we can do -- especially when our goal is not to provide an inspirational speech or to combine several topics and help students see how things are related. Having students read articles during their MSc is also important.
Given that we are often fighting for the students’ time, taking action and attempting to change the context to be more meaningful and more educational is by itself already virtuous. When considering the changes that our department has made in e.g. having teachers react to the student feedback is also positive; unfortunately not all see the value, and a part of the personnel considers this an administrative burden, while others see it as a chance to improve their own course content and teaching.

If one thinks about what it takes for a teacher to improve a course, receiving feedback at the end of the course is often way too late, and easily forgotten by the time for the next course. What would be useful is a continuous feedback and improvement cycle (e.g. using design based research?), which would help the teacher to have a more continuous plan on what to improve, and what works already. Naturally, it is not only a task for a single course, but something that the whole dept should consider.