Mining learners’ traces from an online collaboration tool

Dilhan Perera, Judy Kay, Kalina Yacef, Irena Koprinska
School of Information Technologies
University of Sydney, Australia
{dper6077, judy, kalina, irena}@it.usyd.edu.au

Abstract: As university courses increasingly require students to use online tools in their studies, the opportunity arises to mine the resulting large amounts of student learning data for hidden useful information. In this paper we study the application of data mining to data collected from third year software development group projects using Trac, an online collaboration tool. We applied two very distinctive techniques, clustering and sequential pattern mining. The results point to the importance of leadership and group interaction, and give promising indications of whether effective leadership is occurring in a group. In addition, patterns were found which appear to indicate good individual practices. The results have considerable promise for advising groups at the start of their work and in early identification of both effective and poor patterns, in time for remediation.

1. Introduction

Educational institutions are increasingly using online computer-based tools to support both face-to-face and distance teaching. Because use of such tools generates a large amount of data about learners, there is the potential that mining that data can disclose useful information that can be used to improve learning and teaching. In this paper, we explore the use of data mining techniques on data collected from students using an online collaboration tool. In particular, our goal was to evaluate the effectiveness of data mining for: 1) gaining knowledge about the behaviour of students (and learners in general) working in groups, and 2) identifying which behavioural patterns are associated with positive and negative outcomes. This knowledge can then be used to inform students during the semester if their current behaviour was associated with positive and negative outcomes in the past, and also to provide them with advice on how to rectify problems. The data mining results could also be regularly presented to students in a comprehensible form, and thus facilitate reflection and self-regulation during the semester. Such ‘mirroring’ of student behaviour has shown promise with the use of suitable visualisation techniques [1]. This information can also play a very important role in discussions between teachers and learners.

Section 2 describes the background of the study: the groups and the online tool used. Section 3 presents the main data exploration, and the application of two data mining techniques: clustering and sequential pattern mining. We discuss the results, problems encountered and possible solutions. Section 4 summarises our conclusions.
2. Learners, learning environment and data

The learners were completing a senior software development project course. Over 12 weeks, and working in groups of 5-7 students, they were required to develop a software solution for a client. These projects varied from creating a computer-based driving ability test to developing an object tracking system for an art installation. The groups were required to use Extreme Programming (XP) [2], including use of user stories, small releases, and collective code ownership. Three semesters of data was collected for cohorts in 2005 and 2006, the last being the focus of this paper.

The learning environment used is an online professional software development tool, Trac [3]. It supports collaboration by integrating three tools: group Wiki for shared web pages, a task management system (also known as a ‘ticketing’ system) and a Subversion (SVN) browsing interface for source code version control. We have enhanced this professional tool with artefacts which extract information from learners’ data (in the form of student models) (1) for students to peruse and reflect on and (2) for teachers to have a bird’s eye view of what students are doing and where to focus their teaching efforts [4].

Data was captured whenever a learner performed an action in the Trac system, e.g. created or modified a Wiki page, ticket, or file in the SVN repository. The addition and reorganisation of directories in the repository was also captured. Information on each of these events was stored, including the time of the event, the author(s) and resources involved. Each event also included information specific to the event type, such as priorities for ticket events and a count of lines added and deleted for Wiki events. More details about the data and Trac can be found in [5].

In addition to these electronic traces, we also had the progressive and final marks, together with a very good understanding of the quality of each group’s processes and product throughout the semester. The groups were ranked based on their performance where Group 1 denotes the strongest and Group 7 the weakest group.

3. Data mining

3.1. Simple statistical analysis

Before any data mining was carried out, the data was examined to see whether any simple statistics could distinguish the stronger from the weaker groups.

Firstly, we checked the total number of ticket events for each group. Intuitively we expect a large number to be associated with strong groups as the tickets allow group members to manage their work (e.g. allocate and accept tasks, post comments about tasks). Indeed, the results show that the top group had the highest number of ticket events. However, the performance of the other groups does not seem to correlate with the number of ticket events. For example, Group 2 had the lowest number. Interviews with this group indicated that they were reluctant to use the system as they felt it to be too cumbersome, and hence preferred to communicate their progress by other means.

Secondly, we looked at the distribution of the individual ticketing events (ticket created, accepted, reopened and closed). As tickets must be accepted by the assignee before they are recorded as being assigned, we expect the better groups to have near equal proportion of created and accepted tickets, which was the case. In contrast, some of the poorer groups had a much lower proportion of accepted than created tickets.
Again, this statistic is not very useful on its own: the poorest group displayed similar patterns to the top groups. Notably, we determined that in Group 4 one person logged in as each team member and entered all contributions for the group: this may explain for their seemingly ideal distribution of ticketing events.

Thirdly, we examined the usage span of the Wiki pages, i.e. the time between the first and last event on the page. Group 1 has the lowest number of Wiki pages but they were, on average, active for the longest period of time. This pattern is also evident for the next best group (Group 2), and the opposite pattern is displayed by the two poorest groups. There are several possible interpretations for this result and more work is needed to validate them. It could be that the better groups used the Wiki for more “active” purposes, such as group discussion or a logging of personal progress, while the poorer groups used the Wiki for more “static” purposes such as posting research and guidelines. Considering groups were required to post assessable work (such as reports) on the Wiki, it could also be that the better groups started this work earlier, while the poorer groups worked in a more compressed timeframe. However again, as shown by Group 5, this measure alone was not predictive of the quality of the group.

Lastly, we studied our SVN data and found that it was problematic for two reasons. First, as files were identified by their pathnames, we could not track unique files as they were often moved to different locations within the group repositories. Second, differences between SVN clients meant that data which was recorded on the number of lines added and deleted to committed files was not reliable. Thus, the only reliable SVN data was the time each commit took place. We use it to count the number of days on which SVN activity occurred for a group. The top group again was ranked highest on this measure: however, there was no obvious pattern in this statistic for other groups.

3.2. Clustering

As shown above, simple statistical exploration of the data together with a ranking of the groups was quite limited. The results suggested the need to consider multiple data attributes simultaneously, as well as removing the crudeness of relying on a single ranking system for the groups. Clustering allows us to use multiple attributes to identify similar groups in an unsupervised fashion, without the need to label the groups. In addition, it provides the opportunity to mine the data at the level of individual learners and in this way to examine the composition of each group.

Clustering can have useful applications in educational setting [6, 7]. For instance in [6] students using an intelligent tutoring system were clustered according to the types of mistakes made. The authors suggested that through the use of clustering, teachers could identify different types of learners and apply different remedial methods. A similar goal can be transferred to the current context, with clustering possibly identifying different styles of groups which may benefit from different styles of intervention. However, it must be noted that with a small number of groups, this could be performed by the teachers alone, without the aid of clustering results. Therefore, our primary goal was simply to assess whether our data contained features which could be translated through clustering into meaningful information about groups and individual learners.
3.2.1. Clustering groups

The most important problem was attribute selection. The performance of the algorithms is very sensitive to the quality of attributes. Initially we chose a set of 8 numeric attributes representing ticketing behaviour such as the number of tickets and ticket events; the number of days on which tickets were opened, closed, or a ticket event occurred; and the ticket usage span (number of days between first and last event). We selected the classical K-means clustering algorithm and set the number of clusters to 3. The results are shown in Table 1 and reveal useful characteristics of the groups.

Table 1. Clustering ticketing behaviour using K-means (k=3) and 8 attributes

<table>
<thead>
<tr>
<th>Clustered groups</th>
<th>Distinguishing characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Groups 2, 3, 4 &amp; 7</td>
<td>Overall low ticketing activity</td>
</tr>
<tr>
<td>Groups 5 &amp; 6</td>
<td>Many tickets, Fewer ticketing events, Greater percentage of trivial and minor ticket priorities, Less accepting events</td>
</tr>
<tr>
<td>Group 1</td>
<td>Many tickets and many ticketing events, Lowest percentage of minor ticket priorities, More events where ticket priorities were changed or comments posted</td>
</tr>
</tbody>
</table>

One problem we met was that many attributes were correlated (some as high as 0.918, p=0.004). This motivated the manual creation of composite attributes that seemed to capture essential aspects of team performance. Attributes that measured total activity were excluded in favour of those that gave an indication of how Trac was used when it was used. Also, pairs of attributes were selected as together, they seemed to reveal information that was not obvious in either alone. Through this process, the 11 attributes listed in Table 2 were selected. Notably, 5 of them (marked with *) were ranked favourably when three of the Weka’s [8] supervised attribute selection algorithms were used together (Information Gain, Relief and Support Vector Machines) with two slightly different group performance labelling.

Table 2. The 11 attributes selected for clustering

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average number of events per ticket</td>
<td></td>
</tr>
<tr>
<td>Number of different days ticketing occurred</td>
<td></td>
</tr>
<tr>
<td>Average num. of ticket events per active ticketing day</td>
<td></td>
</tr>
<tr>
<td>Percentage of ticket events not involving an ‘action’ on the ticket (comment added or a priority change)</td>
<td></td>
</tr>
<tr>
<td>Percentage of ticket ‘action’ events where a ticket was accepted</td>
<td></td>
</tr>
<tr>
<td>Average number of events per Wiki page</td>
<td></td>
</tr>
<tr>
<td>Average Wiki page usage span (days between first and last edit)</td>
<td></td>
</tr>
<tr>
<td>Average number of edit days per page</td>
<td></td>
</tr>
<tr>
<td>Average number of lines added per Wiki edit</td>
<td></td>
</tr>
<tr>
<td>Average number of lines deleted per Wiki edit</td>
<td></td>
</tr>
<tr>
<td>Number of different an SVN activity occurred</td>
<td></td>
</tr>
</tbody>
</table>

Table 3 shows the K-means results using the above attributes. The same results were obtained with the EM clustering algorithm. Group 1 was again separated from the others.

Table 3. Clustering Trac activity using K-means (k=3) and 11 attributes

<table>
<thead>
<tr>
<th>Clustered groups</th>
<th>Distinguishing characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Groups 2, 3, 4 &amp; 6</td>
<td>Moderate events per ticket, Infrequent Trac activity (tickets and SVN), Moderate % of ticket update events, Moderate number of lines added/deleted per Wiki edit</td>
</tr>
<tr>
<td>Groups 5 &amp; 7</td>
<td>Moderately frequent Trac activity (tickets and SVN), High edits per Wiki page, Low number of lines added/deleted per Wiki edit, Low number of events per ticket, Low % of ticket update events</td>
</tr>
<tr>
<td>Group 1</td>
<td>Very frequent Trac activity (tickets and SVN), High events per Wiki page and per ticket, High Wiki page usage span, High % of ticket update events, High % of ticket accepting events</td>
</tr>
</tbody>
</table>
3.2.2. Clustering students

We also performed clustering of the individual students, with the hope that the group composition would reveal information that was missed when all individuals in a group were considered together. Table 4 shows the clusters obtained with K-means with the above 11 attributes, along with a label applied to each cluster based on an interpretation of its characteristics. The distribution of students from different clusters is presented in Table 5, with asterisks showing the cluster in which each group’s manager was placed.

Table 4. Student clusters obtained using K-means

<table>
<thead>
<tr>
<th>Cluster size</th>
<th>Distinguishing Characteristics</th>
<th>Cluster label</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 students</td>
<td>High ticketing activity, Involved in many tickets, High Wiki activity, Involved in many Wiki pages, Moderate SVN activity</td>
<td>‘Managers’</td>
</tr>
<tr>
<td>9 students</td>
<td>Moderately high ticketing activity, Ticketing occurring on many different days, Moderate Wiki activity, Very high SVN activity</td>
<td>‘Trac-Oriented Developers’</td>
</tr>
<tr>
<td>11 students</td>
<td>Low ticketing activity, Low Wiki activity, Low SVN activity</td>
<td>‘Loafers’</td>
</tr>
<tr>
<td>15 students</td>
<td>Moderately low ticketing activity, Moderately low Wiki activity, Many Wiki events on days which Wiki events occurred, Many SVN events on days which SVN events occurred</td>
<td>‘Majority’</td>
</tr>
</tbody>
</table>

Table 5. Distribution of students from each cluster shown in Table 3

<table>
<thead>
<tr>
<th>Managers</th>
<th>Trac-Oriented Developers</th>
<th>loafers</th>
<th>Majority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group 1</td>
<td>*1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Group 2</td>
<td>*1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Group 3</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Group 4</td>
<td>*1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Group 5</td>
<td>3</td>
<td>*1</td>
<td>0</td>
</tr>
<tr>
<td>Group 6</td>
<td>*1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Group 7</td>
<td>*1</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

Some differences between previously coupled groups began to emerge. For example, Groups 2 and 3 differ by Group 3’s lack of a ‘manager’. This was consistent with our knowledge of the leadership problems this group encountered, with the original manager leaving the course and another group member taking over. The lack of ‘Trac-oriented developers’ in Group 2 was validated in a group interview where the main developers expressed a reluctance to use Trac. Group 5 is also distinctive in its excess of ‘managers’, perhaps suggesting too many managerial and organisational processes were occurring at the expense of actual work being done. This is further complicated by their designated manager being placed in the cluster which performed more technical than managerial work. One possibility is that this weak leadership resulted in others reacting to fill the manager’s role, with their technical work subsequently being compromised. This may be a pattern to be aware of in future groups.

3.3. Sequential pattern mining

An important aspect of our data which is ignored by mining techniques such as clustering is the timing of events. We believe that certain sequences of events distinguish the better groups from the weaker ones. These sequences may represent characteristic team interaction on a specific resource, or group members displaying specific work patterns across the three aspects of Trac. A data mining technique which considers this temporal aspect is sequential pattern mining [9, 10]. It finds sequential patterns that occur in a dataset with at least a minimal level of frequency.
To extract patterns, the data first needs to be transformed into a set of sequences. The first problem is how to break down the long traces of events into meaningful sequences. We considered three possibilities:

- A sequence per resource, where a separate sequence is obtained for the events on each ticket, Wiki page, and SVN file.
- A sequence per group ‘session’, where sessions are formed by cutting up the group’s event list where gaps (of no activity) of a minimum length of time occur. A related sequence formation method is a sequence per author ‘session’ – before the event list is broken up into sessions, the event list for each group member is extracted. Sessions are then formed from these event lists of individuals.
- A sequence per ‘task’, defined by a ticket. The task sequence includes all ticket events on that ticket, and SVN and Wiki events referring to it.

The second problem is to define an alphabet to represent the events inside the sequences. In our case this involved removing certain author and resource identification information, as well as collapsing similar consecutive events into single alphabet items. The three alphabets used are summarised in Table 5.

<table>
<thead>
<tr>
<th>Alphabet 1: (i,X,j)</th>
<th>Alphabet 2: (A,i,X)</th>
<th>Alphabet 3: (i,X,A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The number of consecutive events (i) occurring on a particular Trac medium (X) and the number of individuals (j) involved in the events.</td>
<td>The number of consecutive events (i) performed by a single author on a specific resource of Trac medium type X and the role of the author (A).</td>
<td>The number of consecutive events (i) occurring on a particular Trac medium (X), and the role of the author (A).</td>
</tr>
<tr>
<td>A new item is generated when an event appears in the sequence:</td>
<td>from a different Trac medium</td>
<td>by a different author</td>
</tr>
</tbody>
</table>

Alphabet 1 was introduced in [5] and used with the group’s event sequence. Alphabet 2 was developed for use with the resource and Alphabet 3 for the task sequences. Alphabets 2 and 3 were specifically developed to provide a tighter integration of the research with psychological theories of group work. One example is the Big Five framework which emphasises the importance of leadership, mutual performance monitoring, backup behaviour, adaptability, and team orientation [11]. Other theories also emphasise the influence of leadership in determining the success of groups. In addition, the role of ‘tracker’ in XP was thought to correspond to the Big Five function of performance monitoring. For these reasons, author roles in these alphabets were identified as: L (leader), T (tracker), and all other authors identified as a, b, c, etc. in order of appearance on the resource.

Alphabet 2 also allowed us to examine whether earlier authors returned to edit the resource after it was edited by another author, representing a group interaction on the resource. In addition, leaders and trackers, though always displayed with the same symbol, were still placed in the ordered author list, allowing us to identify resources which were created by group members other than the leader or tracker.

3.3.1. Patterns observed in group sessions

As observed in the previous year [5], the better groups had many alternations of SVN and Wiki events, and SVN and Ticket events whereas weaker groups had almost none. We hope these patterns correspond to documentation in the Wiki about the SVN
commits and to tickets being updated following SVN commits. Although we now have
the ability to store the supporting events for each pattern, it is still arduous to trace
events back to actual Trac actions to test such suspicions.

Through an analysis (using sequences per author), we observed that individuals in
the top group displayed a higher than average level of alternating SVN and Wiki events.
The top group also had the highest proportion of author sessions containing many
consecutive ticket events (matching their high use of ticketing) and SVN events
(suggesting they committed their work to the group repository more often).

In contrast, the least successful group displayed a high level of alternating Wiki
and ticketing events, but also had a distinctive lack of sequences containing SVN
events. Although it also showed frequent consecutive Wiki events, which was a
characteristic associated with the most successful 2005 group, their lack of SVN events
seem to suggest that their Wiki and tickets were not being used in support of software
development. This was validated by the course coordinators, who described this group
as technically less proficient.

A more detailed analysis of these patterns revealed that the top group used the
Ticket more than the Wiki, whereas the weakest group displayed the opposite pattern.
The use of the ticketing system may be indicative of actual work being done, as it is
more task-oriented than the Wiki. This trend was even stronger when we exclusively
considered the sessions of the group leaders. This suggests that the work of the group
leaders clearly influences the success of the groups, with the leaders of the top groups
using tickets more than the other leaders. Note that this does not just include leaders
assigning work to other group members (i.e. tickets being created), but also leaders
commenting on tickets and following up assigned work. In addition, the data suggested
these group leaders were much less involved in technical work, suggesting work was
being delegated properly and the leader was leading rather than simply doing all the
work. In contrast, the leaders of the poorer groups either seemed to use the Wiki (a less
focused medium) more than the tickets, or be involved in too much technical work.

3.3.2. Patterns observed in task sequences

A task sequence can be more informative than a session sequence because it only
contains events that are related, as opposed to events that occurred in the same window
of time. Task sequence mining also shows how the different groups used the three
elements of Trac in completing project tasks.

We found that the two top groups had by far the greatest percentage support for the
pattern (1,t,L)(1,t,b), which were most likely tickets initiated by the leader and accepted
by another team member. The fact this occurred more often than (1,t,L)(2,t,b), suggests
that the better groups were distinguished by tasks being performed on the Wiki or SVN
files before the ticket was closed by the second member. Notably, the weakest group
had higher support for this latter pattern than the former. As a validation to this
interpretation, we also found that the best group was one of only two to display the
patterns (1,t,b)(1,s,b) and (1,s,b)(1,t,b) – the first likely being a ticket being accepted by
a team member and then SVN work relating to that task being completed and the
second likely being work being done followed by the ticket being closed. The close
coupling of task-related SVN and Wiki activity and Ticket events for this group was
also shown by relatively high support for the patterns (1,t,b)(1,t,b)(1,t,b),
(1,t,b)(1,s,b)(1,t,b) and (1,t,b)(1,w,b)(1,t,b). Interestingly, the poorest group displayed
the highest support for the last pattern, but no support for the former, again indicating their lack of SVN use in tasks.

Another series of patterns which characterised the best groups were tickets being initiated by non-leader group members. These tickets were evidently not created just for the sake of the course requirements, as this group also showed high support for wiki and SVN patterns by these team members. An example is \((2,t,a)(1,s,a)(1,t,a)\), which may likely be a ticket being created by a team member for him/herself, the ticket being accepted, work being committed related to the ticket, and the ticket finally being closed.

A pattern which characterised the poorest group was the tracker creating and editing many tickets, for example in the patterns \((1,t,T)\), \((1,t,T)(1,t,b)\) and \((2,t,T)\). As it is the tracker’s role to follow up tasks, their general involvement in tickets should not be a matter of concern. However these patterns may have been more common in the poorer groups because of weaker leadership, resulting in trackers performing a share of the leader’s role. Conversely, the better groups may have shown less involvement by the tracker because of prominent leaders who were also able to perform tracker duties. An alternative explanation may be that group problems lead to greater tracker activity.

### 3.3.3. Patterns observed in resource sequences

Apart from good individual practises, such as SVN commits being documented on Wiki pages, another aspect of good group work which we hoped the original sequence generators and alphabets captured was interaction between team members. For example, it was hoped that events such as \((3,w,2)\) would be indicative of 2 group members interacting on the Wiki. However we cannot be certain about this conclusion because the pattern does not tell us that the three events occurred on the same Wiki page. To better capture interactions between group members we decided to examine sequences across specific resources.

By forming new alphabet items when a new author appeared in the resource’s event sequence, and by identifying the managers and assigning within-resource roles to other group members, we were better able to track these group interactions. We found that the top group had very high support for patterns where the leader interacted with group members on tickets, such as \((L,1,t)(b,1,t)(L,1,t)\). The poorest group in contrast lacked these interaction patterns, and had more tickets which were created by the Tracker rather than the Leader, suggestive of weaker leadership. The importance of leadership and leadership style has been emphasised by the Big Five theory and other classic psychological studies, and the success of our data mining in detecting differences in leadership is especially promising.

In addition, the best group displayed the highest support for patterns such as \((b,3,t)\) and \((b,4,t)\), suggestive of group members making at least one update on tickets before closing them. In contrast, the weaker groups showed support mainly for the pattern \((b,2,t)\), most likely indicative of group members accepting and closing tickets with no update events in between. These extra events on tickets may be important in allowing the team to monitor each other (one of the other Big Five aspects) and also indicates the presence of frequent task-focused communication in successful groups.

Patterns indicative of interaction on tickets in the best group were not just limited to the group leader and one other member. Significantly, this group also displayed higher than average support for patterns of interaction involving multiple team members, such as \((b,1,t)(c,1,t)(L,1,t)\). This is especially notable on tickets, which usually only directly involve two individuals (the assigner and the assignee). The
involvement of a third person may be indicative of a number of desirable group characteristics, such as mutual performance monitoring, team orientation (two elements of the Big Five), or collective code ownership (an Extreme Programming practice). It should also be noted that another top group in 2006 and the top group in 2005 displayed similar patterns of long interactions on Wiki pages rather than on tickets. This may suggest that the interactions themselves are more important than the medium on which they take place.

Another pattern with above average support in the best group was consecutive events on SVN files by an individual author, for example \((a,2,s)\) and \((a,3,s)\). These may have been caused by group members committing to files more frequently, or group members requiring less intervention from others in work being completed by them. Regardless of the interpretation, it is interesting to note that the poorest group, despite lacking these patterns, also lacked the pattern \((a,1,s)(b,1,s)\), where a second team member commits to the file. Instead, we found that in this group it was more common for the group leader to be involved in a file after just one commit by the original author. Again this suggests that the leader intervened on technical aspects of the project and may be a sign of group problems. However because of our noted problems with identifying unique files by pathname, it may also simply be that the group leader moved files around in the repository frequently.

### 3.3.4. Some Limitations of Sequential Pattern Mining

A number of issues emerged during the use of this technique, ranging from limitations in the data to how output was interpreted. Currently our data contains only modification and creation events. The common situation where a team member views another’s work but does not feel the need to modify it was thus effectively ignored. This emphasises the need to incorporate data from sources such as weblogs. A problem with our mining program itself is the lack of gap constraints – as noted in [9], a frequent subsequence of \((X)(Y)\) may not be meaningful if many other events occur between \(X\) and \(Y\). Another issue already stated in [5] is the need for more automated methods of processing output which go beyond manual sorting techniques. Emergent pattern mining [12] and contrast sets [13] may be possible solutions. Finally, there still remained the need to assign meaning to the patterns in order to learn about group work in general. The importance of finding the right balance between alphabets that are too abstract (limiting interpretation) and those which are too specific cannot be understated.

### 4. Conclusion

We performed mining of data collected from students working in teams and using an online collaboration tool in a one-semester software development project. Clustering was applied to find both groups of similar teams and individual members, and sequential pattern mining to extract sequences of frequent events. The results revealed interesting patterns characterising the work of stronger and weaker students. Some of them are specific for our context (i.e. the course requirements and tool used). Others are more generic and consistent with psychological theories of group work, e.g. the importance of group interaction and leadership for success.

This knowledge can be used in several valuable ways. Firstly, we already lecture students on various aspects of group work. This work will enable us to give concrete
examples of how of the patterns associated with some of the general principles, such as those for leadership and monitoring activity. Just from the clustering, we can point to the data in Tables 4 and 5. Each of these can be illustrated with actual wikis and tickets. We can explain how these patterns were associated with leadership behaviours that were either more or less effective. Secondly, we can automate the identification of the most salient patterns described above and present these to the students for their own group discussions as well as using them in meetings with teachers. This may help to rectify ineffective patterns of group operation and consolidate effective ones. Essentially, this work will enable us to provide regular feedback to students during the semester if their current work behaviour is more likely to be associated with positive or negative outcomes and where the problems are. Teachers can also greatly benefit from such feedback. Although more work is needed before more formative and timely feedback can be provided, students and teachers could be made aware of the current findings and limitations, to encourage better group practice and teaching and learning experience.

This work also highlights some of the data mining challenges posed by educational data, e.g. it is temporal, noisy, correlated, incomplete, may lack enough samples for some tasks. We addressed some of them by providing specific solutions for our task. There are many avenues for future work. Both the data mining and the data itself could be extended and enriched. For example, the addition of a chat module can increase the student usage of Trac and generate useful data. Clustering can be improved by the collection of data for more groups and individuals. New alphabets could also be developed for the sequential pattern miner to reveal as-yet hidden work behaviours and group interactions. More work is also needed in assessing the generalization ability of the discovered patterns.

References