Clustering and Sequential Pattern Mining of Online Collaborative Learning Data

Dilhan Perera, Judy Kay, Irena Koprinska, *Member, IEEE Computer Society,* Kalina Yacef, and Osmar Zaiane, *Member, IEEE*

Abstract— Group work is widespread in education. The growing use of online tools supporting group work generates huge amounts of data. We aim to exploit this data to support mirroring: presenting useful high-level views of information about the group, together with desired patterns characterizing the behaviour of strong groups. The goal is to enable the groups and their facilitators to see relevant aspects of the group's operation and provide feedback if these are more likely to be associated with positive or negative outcomes and where the problems are. We explore how useful mirror information can be extracted via a theory-driven approach and a range of clustering and sequential pattern mining. The context is a senior software development project where students use the collaboration tool TRAC. We extract patterns distinguishing the better from the weaker groups and get insights in the success factors. The results point to the importance of leadership and group interaction, and give promising indications if they are occurring. Patterns indicating good individual practices were also identified. We found that some key measures can be mined from early data. The results are promising for advising groups at the start and early identification of effective and poor practices, in time for remediation.

Index Terms— Data Mining, Clustering, Sequential Pattern Mining, Learning Group Work Skills, Collaborative Learning, Computer-Assisted Instruction.

1 Introduction

Group work is commonplace in many aspects of life, particularly in the workplace where there are many situations which require small groups of people to work together to achieve a goal. For example, a task that requires a complex combination of skills may only be possible if a group of people, each offering different skills, can work together. To take just one other example, it may be necessary to draw on the combined efforts of a group to achieve a task in the time available. However, it is often difficult to make a group operate effectively, with high productivity and satisfaction within the group about its operation. Reflecting the importance of group work, there has been a huge body of research on how to make groups more effective and how to help group members build relevant skills. In one meta-analysis of this body of work, a set of five key factors and three enablers has been identified [1]. For example, this work points both to the importance of leadership as one of the five key factors and to the effectiveness of training in leadership.

The importance of group work skills is reflected in education systems, where students are given opportunities to develop these valuable skills. Often, and increasingly, such groups are supported by software tools. This may be in the context of distance learning, where the groups are distributed and the members must use software to support their collaboration. In addition, even when student groups work in the same classroom or campus, they may be supported by a range of online tools, such as chat, message boards and wikis. For small groups that need to collaborate on substantial tasks over several weeks, such tools can amass huge amounts of information and generate large electronic traces of their activity. This has the potential to reveal a great deal about the group activity and the effectiveness of the group.

Our goal is to improve the teaching of the group work skills and facilitation of effective team work by small groups, working on substantial projects over several weeks by exploiting the electronic traces of group activity. Our approach is to analyse these traces to create mirroring tools that enable the group members, their teachers or facilitators to see useful indicators of the health and progress of their group. We consider it important that our work should be in the context of standard, state-of-the-art tools for supporting groups. This means that we should be able to exploit the data from a range of tools and media that are valuable for small group management. These include wikis, issues tracking systems and version control software. The key contribution of our work is an improved understanding of how to use data mining to build mirroring tools that can help small long-term teams improve their group work skills.

Our work is situated at the intersection of three main areas: Machine Learning and Data Mining, especially as

- *D. Perera, J. Kay, I. Koprinska and K. Yacef are with the School of Information Technologies, University of Sydney, NSW 2006, Australia. E-mail: {dper6077, judy, irena, kalina}@it.usyd.edu.au.*
- *O. Zaiane is with the Department of Computing Science, University of Alberta, Canada. E-mail: zaiane@cs.ualberta.ca.*

Manuscript received (insert date of submission if desired). Please note that all acknowledgments should be placed at the end of the paper, before the bibliography.

they apply to educational contexts; Computer Supported Collaborative Learning (CSCL); the body of knowledge about small group skills and effectiveness. While our research has been informed by all of these, it is important to distinguish just how our work differs from previous work in them. We now briefly discuss this in terms of each of these areas.

The emerging research community of Educational Data Mining [2] exploits data from learners' interaction with e-learning tools, particularly web-based learning environments. The recognition of the huge potential value of such data has led to a series of ten workshops and a new conference [3]. There have been recent promising results using a range of techniques [4-7]. There is good reason for this new research area, primarily because it needs to deal with issues that differ from those that had previously had most attention in the wider data mining and machine learning research. For example, educational data presents several difficulties for the data mining algorithms as it is temporal, noisy, correlated, incomplete and may lack enough samples for some tasks. In addition, there is a need for understandable and scrutable presentations of the data mining results appropriate for the non-data mining savvy users. This area is establishing the new requirements for effective mining and analysis of learning data. This paper continues this exploration of foundations for this area, taking account of the particular demands of one important class of educational context.

CSCL is an established and active research area. However, much of the focus of that community is based upon the value of collaboration for improved learning across many disciplines. This is rather different from our focus. So, for example, the CSCL community has done considerable work on the use of discussion boards. This is relevant to our work in that it does explore ways to improve participation rates as in the work of Cheng and Vassileva [8]. They created an adaptive rewards system, based on group and individual models of learners. This had elements of mirroring but significantly differs from our goal of supporting small groups for whom learning group work skills is one of the learning objectives and the group work is the key focus.

Some research has brought together CSCL and data mining. Notably, Talavera and Gaudioso [9] applied clustering to student interaction data to build profiles of student behaviours. The context of the study was a course teaching the use of Internet and the data was collected using a learning management system from three main sources: forums, email and chat. Their goal was to support evaluation of collaborative activities and although only preliminary results were presented their work confirmed the potential of data mining to extract useful patterns and get insight into collaboration profiles. Soller [10, 11] analysed conversation data where the goal was knowledge sharing: a student presents and explains new knowledge to peers; peers attempt to understand it. Hidden Markov models and multidimensional scaling were successfully applied to analyse the knowledge sharing conversations. However, Soller required group members to use a special interface using sentence starters, based on Speech Act Theory. The requirement for a special interface, limited to a single collaboration medium, with user classified utterances has characterised other work, such as Barros and Verdejo [12] whose DEGREE system enabled students to submit text proposals, co-edit and refine them, until agreement was reached. By contrast, we wanted to ensure that the learners used collections of conventional collaboration tools in an authentic manner, as they are intended to be used to support group work: we did not want to add interface restrictions or additional activities for learners as a support for the data mining. These goals ensure the potential generality of the tools we want to create. It also means that we can explore use of a range of collaboration tools, not just a single medium such as chat.

The notion of mirroring has been discussed in a similar context to ours [13]. In the current state of research, the goal of mirroring that is effective is a realistic starting point. Moreover, it has the potential to overcome some of the inherent limitations of data mining that does not make use of a deep model of the group task and the complex character of each particular group. So, it offers promise for powerful and useful tools that are more generic, able to be used by many different groups working on different tasks. We have already found that mirroring of simple overall information about a group is valuable [14]. The work on social translucence [15, 16] has also shown the value of mirroring for helping members of groups to realise how they are affecting the group and to alter their behaviour. Our experience with these tools has pointed to their particular power in the context of long-term small groups: the mirrored information serves as valuable starting point for both discussing group work, as part of the facilitation process, and it can serve as an excellent basis for exploring the information within the collaboration environment.

The paper is organised as follows. The next section states our goals of mining group logs, identifies the main stakeholders and how they can benefit from the extracted patterns. Section 3 describes in more detail the context of our study: the learner population, TRAC online system and nature of the data collected. Section 4 presents the initial data exploration performed and discuses its limitations. Then the actual data mining is presented, with Section 5 describing the clustering work and Section 6 presenting the frequent sequential pattern mining. We discuss the results, problems encountered, and how the discovered patterns can be used to improve teaching and learning. Section 6 concludes the paper.

2. Goals of Mining Group Work Logs

We set our primary goal for the data mining as providing mirroring tools that would be useful for helping improve the learning about group work. This goal is realistic in the context of the highly complex and variable nature of long-term, small group activity, especially where the learners undertake a diverse range of tasks, such as creating a software system for an authentic client. Our mirroring goal means that we aim to extract patterns and other information from the group logs and present it together with desired patterns to the people involved, so that they can interpret it, making use of their own knowledge of the group tasks and activities.

To underpin our work, we have used the Big Five theory of group work [1]. It is based on a broad meta-analysis of research on small group interaction, drawing on the large body of literature reporting studies of various aspects of group work and determinants of success. It has established five key factors: *leadership, mutual performance monitoring, backup behaviour, adaptability* and *team orientation*. Backup behaviour involves actions like reallocating work between members as their different loads and progress becomes recognised. Adaptability is a broader form of changing plans as new information about internal group and external issues are identified. Team orientation covers aspects such as commitment to the group as a whole. It also has identified three supporting mechanisms: *shared mental models*, especially shared understanding of how the group should operate; *mutual trust*; and *closed loop communication*, which means that, regardless of the medium, a person communicating a message receives feedback about it and confirms this. This theory provides a language with which to discuss group work and guides our data mining.

Given our goal, it is important to distinguish the key stakeholders because the information relevant to each is somewhat different. We distinguish four classes of stakeholders:

- *individual learner:* each has a good knowledge of their own goals and activities but may be unaware of what others in their group have been doing and how well they have been performing as a team member and what they should be doing to be more effective in their allocated roles;
- *individual group:* the group as a whole is aware of some aspects of their performance but is less aware of how they could improve their performance and how well they are doing on the various dimensions of the Big Five elements;
- *group facilitator*: this person works with the groups, meeting them regularly and helping them see how to improve their performance. This person is more knowledgeable about group processes and has an outsider view of the group. However, they need help in seeing just what the group members have been doing and how they have been interacting;
- *course co-ordinator:* this person needs to teach the group skills and to monitor the progress of all the groups. They have least knowledge of the details of the individual groups and are most in need of support in seeing a big picture overview of the large amounts of log data to understand what the groups are doing.

We were able to refine the goals of mirroring into the following three sub-goals:

- *timely problem identification:* All stakeholders should be keen to know about indicators of problems in the group work, especially if these indicators can be provided in time for remedial action to have a significant effect. In particular, if the group facilitator, can see patterns that are suggestive of potential problems in some key aspects, such as leadership or effective closed loop communication, they can discuss these issues with the group and work with them to find ways to improve the learning about group work and to ensure the success of the group.
- *support for self-monitoring:* This is particularly important for the individual. For example, the leader should have distinctive behaviours and we would like to provide high level mined results reflecting the effectiveness of their interaction, as a leader;
- *improved understanding of how effective groups make use of the online collaboration tools*: this is most important for co-ordinators as it can inform their teaching and organisation of the learning environment.

We will refer to our identified stakeholders and the sub-goals of the data mining in the discussion of the data mining and the value of different results for the different stakeholders.

3 Context of the Study

3.1 Learners

The learners were students 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. The topics 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) [17], including use of user stories, small releases, and collective code ownership.

We have collected data over three semesters, for cohorts in 2005 and 2006. This paper reports the last 2006 cohort because our teaching changed markedly in 2006 and that cohort was given much more support and instruction in group work skills. This means their data is richer and more meaningful, and is also not comparable with the data from 2005.

3.2 Online Learning Environment: TRAC

Student teams were required to use TRAC [18] for online collaboration. TRAC is an open source, professional software development tracking system. It supports collaboration by integrating three tools:

- A group wiki for shared web pages. It is a collaborative authoring tool, allowing the group members to add, remove or edit web pages, linked from the main group page.
- A task management system also known as a ticketing system. A ticket is created for each task that the team has to do. For example, if the team needs to do research on e-learning, one person (often the leader) should create a ticket for this task, which is then allocated to a person for completion. Team members can add comments on a ticket, reassign it to someone else or close it.

- Subversion (SVN) control system. It provides a repository for the software created by the group and manages the changes made over time. It allows recovery of older versions of the software and a view of the history of how the files and directories were changed.

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 [19].

3.3 Data

We collected data from the students' use of TRAC; essentially, all the traces of their actions. This includes capturing data whenever a user: 1) created a wiki page or modified it, e.g. added or removed text, 2) created a new ticket or modified an existing, e.g. by closing it, reassigning it, changing its priority or severity, or adding a comment, 3) committed a file to the SVN repository or modified an existing one, or added and reorganised the directories in the repository. Information about each of these events was stored, including the time of the event and the group members and resources involved.

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 from 1 to 7 where Group 1 denotes the strongest and Group 7 the weakest group.

It should be noted that in addition to TRAC, the student teams collaborated and communicated via other media to which we don't have access, such as instant messaging, telephone conversations, SMS. Most importantly, they had face-to-face meetings, typically at least twice a week. These meetings play a critical role in the group co-ordination.

4 Data Exploration

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, as shown in Fig. 1. Intuitively we expect a large number to be associated with strong groups as the tickets allow group members to keep track of their work, including to allocate and accept 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 one of the lowest numbers. Upon interviewing members from this group (after the completion of the course), we found 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.

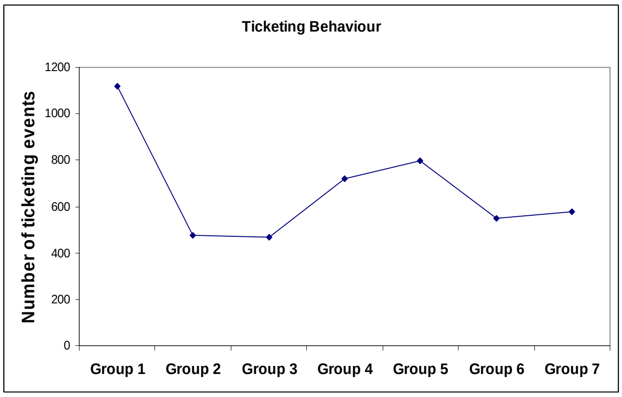

Fig. 1. Total number of ticketing events per group

Secondly, we looked at the distribution of the individual ticketing events (ticket created, accepted, reopened and closed), see Fig.2. 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. It should also be noted that Group 4 admitted at the interviews that one person logged in as the other team members and entered all group contributions, which explains their ideal distribution of the ticketing events.

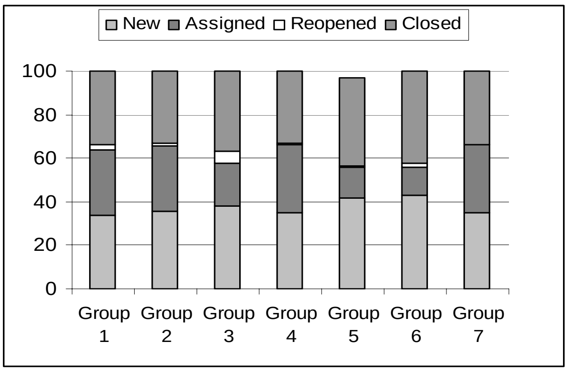

Fig. 2. Distribution of ticket actions per group [%]

Thirdly, we examined the usage span of the wiki pages, i.e. the time between the first and last event on the page, see Fig.3. 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.

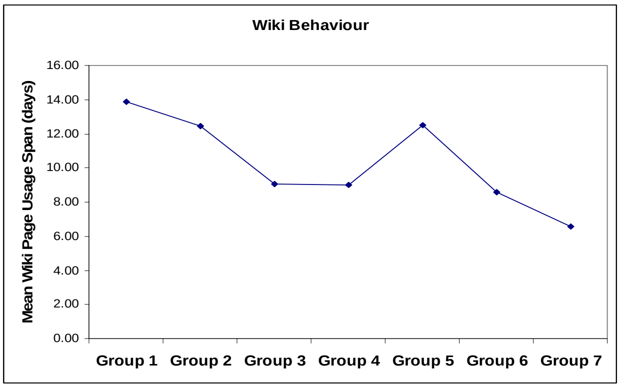

Fig. 3. Usage span of wki pages

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, Fig. 4. The top group again was ranked highest on this measure: however, there was no obvious pattern in this statistic for the other groups.

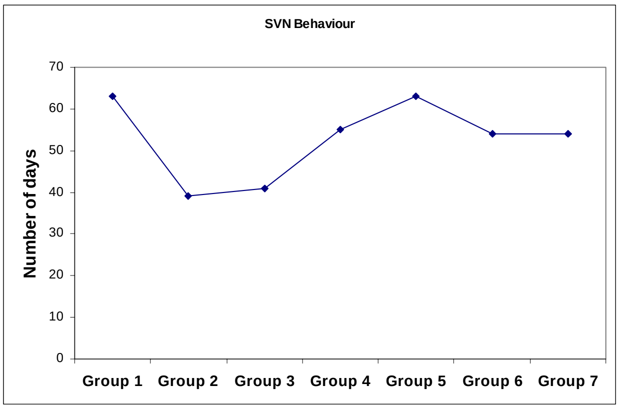

Fig. 4. Number of different days on which SVN event occurred for each group

5 Clustering

As shown in the previous section, simple statistical exploration of the data was quite limited. The results suggested the need to consider multiple data attributes simultaneously. Clustering allows us to use multiple attributes to identify similar groups in an unsupervised fashion. In addition, it provides the opportunity to mine the data at the level of individual learners (i.e. to find groups of similar learners) and then to examine the composition of each group.

An application of clustering in an educational setting is presented in [4], where 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, such analysis 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.

As a main clustering algorithm we selected k-means which is the most popular. It is also simple, effective and relatively efficient [20, 21]. We used the WEKA [22] implementation with Euclidean distance measure.

The data consisted of TRAC usage traces for 43 students working in 7 groups. Its size was 1.6 mega bytes in mySQL format and it contained approximately 15000 events as shown in Table 1.

Table 1. Number of Events for each Group

Group	Number of Events
1	2460
2	1416
3	1499
4	2156
5	3395
6	1639
7	2462
Total	15027

5.1 Clustering Groups

The most important problem was attribute selection. The performance of clustering algorithms is very sensitive to the quality of the 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 firstly ran k-means with k=3 clusters. The number of clusters was set to 3 based on expert knowledge, i.e. consultation with teaching staff and also considering students' final marks in the course. As mentioned before, we had a very good understanding of the quality of the processes followed by each group and their final product. We also experimented with k=2 and 4 but the results were most meaningful for k=3.

Table 2 shows the clustering of the groups, together with the extracted distinguishing characteristics of each cluster. The first cluster consists of Groups 2, 3, 4 and 7 and is characterised by overall low ticketing activity.

While low ticketing activity is typically associated with weaker groups, Group 2, the second best group, also showed this characteristic as it was reluctant to use the ticketing system as discussed in Sec. 3. The separation of Group 1 from Groups 5 and 6 shows that the way the tickets were used, as opposed to just the ticketing activity, was important. More specifically, the results show that tickets are most beneficial when they are actively updated (e.g. through posting comments on progress or adjusting their priority) as opposed to simply being created and closed. However, we found that many of these attributes were correlated (some as high as 0.918, p=0.004). It was also felt that the simultaneous use of the wiki, ticket and SVN behaviours will be more informative than the ticketing activity alone.

TABLE 2. CLUSTERING TICKETING BEHAVIOUR USING K-MEANS (K=3) AND 8 ATTRIBUTES

Clusters	Distinguishing characteristics
Groups 2, 3, 4 & 7	Overall low ticketing activity
Groups 5 & 6	Many tickets
	Fewer ticketing events
	Greater percentage of trivial and minor ticket priorities
	Less accepting events
Group 1	Many tickets
	Many ticketing events
	Lowest percentage of minor ticket priorities
	More events where ticket priorities were changed or comments posted

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. Through this process, the 11 attributes listed in Table 3 were selected. It is interesting to note that 5 of them (the ones marked with *) were automatically ranked favourably when three of the WEKA's [22] supervised attribute selection algorithms were used together (Information Gain, Relief and Support Vector Machines) with two slightly different group performance rankings.

TABLE 3. THE 11 ATTRIBUTES SELECTED FOR CLUSTERING OF GROUPS

-Average number of events per ticket
-Number of different days ticketing occurred
-Average number of ticket events per active ticketing day *
-Percentage of ticket events not involving an 'action' on the ticket (i.e. the ticket was either updated with a comment or a priority change) *
-Percentage of ticket 'action' events where a ticket was accepted
-Average number of events per wiki page
-Average wiki page usage span (days between first and last edit) *
-Average number of edit days per page *
-Average number of lines added per wiki edit
-Average number of lines deleted per wiki edit *
-Number of different days an SVN activity occurred

The k-means clustering results using the above 11 attributes are shown in Table 4. Comparing Tables 2 and 4, we can see that the results are similar; only Groups 6 and 7 are in different clusters (swapped). Group 1 was again clearly separated from the others.

As k-means is sensitive to the seed initialization and also does not deal well with clusters with non-spherical shape and different size, we also ran the EM clustering algorithm [20] using its WEKA's implementation [22]. As a mixture model clustering, EM is a more general algorithm than k-means and doesn't suffer from the limitations listed above. Using the same settings (k=3 clusters and 11 attributes), we obtained the same results as with k-means (Table 4).

TABLE 4. CLUSTERING TRAC ACTIVITY USING K-MEANS (K=3) AND 11 ATTRIBUTES

Clusters	Distinguishing characteristics
Groups 2, 3, 4 & 6	-Moderate events per ticket
	-Infrequent TRAC activity (tickets and SVN)
	-Moderate % of ticket update events
	-Moderate number of lines added/deleted per wiki edit
Groups 5 & 7	-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
Group 1	-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

To get better insight into the group similarities we also ran hierarchical agglomerative clustering [20] with Euclidean distance using Cluster [23]. The algorithms initially places all examples in a cluster of their own and then iteratively merges the closest two clusters, until all examples form one big cluster. The results are shown in Fig. 5 using TreeView [24]. At level 3 (i.e. 3 clusters considered), the results are the same as using k-means (Table 4). Thus, we obtained the same clustering results for k=3 using k-means, EM and hierarchical agglomerative clustering.

Such results are useful for the course co-ordinator. They highlight that Group 1's behaviour is distinguished from the others. The co-ordinators had some sense that this group was well managed but this cluster analysis pointed to the particular behaviours that distinguished this group. We only noticed these after the results of the data mining prompted us to look at particular parts of the TRAC sight and to see the way they used tickets. It turned out that they made extensive use of the wiki on each ticket for communication about the task associated with it. The course co-ordinators discovered this, more effective way to use TRAC, only because of the results just reported. This new understanding was used in subsequent teaching and was judged by the facilitators to be helpful.

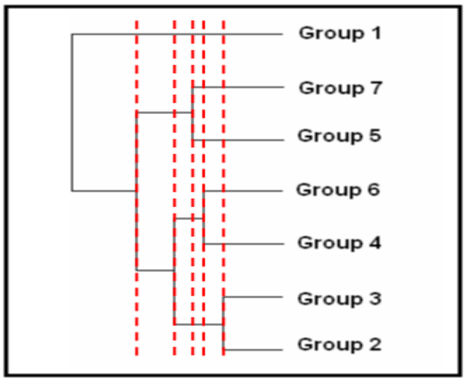

Fig. 5. Hierarchical agglomerative clustering using the 11 attributes

5.2 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. The attributes we selected are listed in Table 5; they are similar to the ones in Table 3 but characterise individual not group activity.

TABLE 5. THE 14 ATTRIBUTES SELECTED FOR CLUSTERING OF INDIVIDUAL STUDENTS

-Number of ticket events
-Number of tickets in which the individual was involved
-Number of different days in which a ticket event occurred
-Average number of ticket events per active (individual) ticketing day *
-Number of wiki events
-Number of wiki pages edited
-Number of different days on which a wiki event occurred
-Average number of wiki events per active (individual) wiki day
-Average lines added per wiki edit
-Average lines deleted per wiki edit
-Number of SVN commits
-Average number of files per SVN commit
-Number of different days in which an SVN commit occurred
-Average number of SVN commits per active (individual) SVN day

Table 6 shows the clusters obtained with k-means for k=4, together with their distinguishing characteristics. Again the number of clusters was set empirically using expert knowledge and by looking for meaningful grouping. Based on our interpretation of the characteristics in Table 6, a cluster label was assigned ("Managers", "TRAC-oriented developers", "Loafers" and "Others"). The distribution of students from each group into these 4 clusters is presented in Table 7, with asterisks showing the cluster in which each group's designated manager (leader) was placed. This role was allocated to one person after the initial start-up period. For example, Group 5 consisted of 7 students; 3 of them were clustered as "Managers", 1 as "TRAC-oriented developers", 0 as "Loafer" and 3 as "Others"; the designated group manager was clustered as "TRAC-oriented developer"

TABLE 6. STUDENT CLUSTERS OBTAINED USING K-MEANS

Cluster size	Distinguishing Characteristics	Cluster label
8 students	High ticketing activity Involved in many tickets High wiki activity Involved in many wiki pages Moderate SVN activity	"Managers"
9 students	Moderately high ticketing activity Ticketing occurring on many different days Moderate wiki activity Very high SVN activity	"TRAC-Oriented Developers"
11 students	Low ticketing activity Low wiki activity Low SVN activity	"Loafers"
15 students	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	"Others"

TABLE 7. DISTRIBUTION OF STUDENTS FROM EACH GROUPS INTO THE CLUSTERS FROM TABLE 6

	Managers	**TRAC-Oriented Developers**	**Loafers**	**Others**
Group 1	*1	3	1	1
Group 2	*1	0	1	3
Group 3	0	1	2	**3
Group 4	*1	3	2	0
Group 5	3	*1	0	3
Group 6	*1	1	3	1
Group 7	*1	0	2	4

Some differences between previous groupings 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 is a pattern to be aware of in future groups.

We also conducted another experiment. We ran the clustering using the data only from the first seven weeks of data and found that, already, some of these key results had already emerged. For example, the Group 5 leader was already showing the developer's behaviours. Had the group facilitator been aware of this, they may have been able to help this group deal with this problem, early enough to have made a difference. The presence of three loafers was also apparent in Group 6. The early data also showed leadership's behaviours by all other leaders at that stage. These results also have great value for the individuals so that they could, as needed, alter their behaviour.

In conclusion, we found clustering to be useful, revealing interesting patterns characterising the behaviour of the groups and individual students, when using TRAC. Frequent use of the 3 media, with high number of active events (such as ticket update and ticket accepting, wiki page edits and SVN commits) is associated with positive outcomes. Effective group leadership and monitoring are also linked with positive outcomes. In future work, we would like to make a better use of the SVN data as it is an important data source conveying the "real work" done by the students in producing software.

5.3 Limitations of Clustering

The main limitation was the small data sample, especially in the first task, clustering of groups. Although the data contained more than 15000 events, we had only 7 groups and 43 students. Nevertheless, we think that the collected data and selected attributes allowed for uncovering useful patterns characterising the work of stronger and weaker students as discussed above. The follow-up interviews were very helpful for interpreting and validating the patterns.

How to select the most appropriate clustering algorithm and how to set its parameters is another important issue. There are methods for determining a good number of clusters and evaluating the clustering quality in terms of cohesion and separation of the clusters found [20]. We believe that in this application the expert knowledge of the course co-ordinators and facilitators is essential to find meaningful number of clusters and extract meaningful characteristics, and then use them on new cohorts. For larger datasets, hierarchical clustering may not be applicable due to its high time and memory requirements; k-means may be still a good choice, especially some of its modifications, such as bi-secting k-means [20] which is less sensitive to initialization and is also more efficient.

6 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. In particular, we expected that we should be able to use these to gain indications of closed loop communication, one of the enablers in the Big Five Theory. Such sequence may represent a 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 [25]. It finds sequential patterns that occur in a dataset with at least a minimal level of frequency called support [26]. Sequential pattern mining has been previously used in e-learning although for different goals than others: to support personalised course delivered based on the learner characteristics [7] and to recommend sequences of resources for users to view in order to learn about a given topic [27]. We first present the algorithm we used and then the data pre-processing we applied.

6.1 Algorithm

The goal of sequential pattern mining is to discover all frequent sequences of itemsets in a sequence dataset. An example of a sequence dataset is shown in Table 8 which contains 3 sequences: *S1, S2* and *S3.* A *sequence* is an ordered list of elements. These elements are collections of one or more *events (items),* in our case an element consists of one event. The *length* of a sequence is the number of elements in it; a sequence of a length *k* is called a *k*-sequence. A sequence *a*=<*a1,a2,...,an*> is a subsequence of *b*=<*b1,b2,...,bm*>, if there exist integers $1<=i_1<i_2<\ldots< i_n <=m$ such that $a1=bi_1$, $a2=bi_{2,}$ $an=bi_n$. The support *sup(s)* of a sequence *s* is the number of sequences of which *s* is a subsequence. So, for our example, the sequence <*a,d*> is a subsequence of *S1, S2* and *S3* and its support is 3, while <c,d> is a subsequence of S2 and S3, and thus its support is 2.

TABLE 8. EXAMPLE OF DATA USED BY A SEQUENTIAL PATTERN MINING

SeqID	Sequence
S1	<a,b,b,d,c>
S2	<a,c,d>
S3	<a,c,c,d>

There are two main approaches to sequential pattern mining: apriori-based [25] and pattern-growth methods [28, 29]. Both of them find the same results but the pattern growth approach is much faster. We used an appriori-based algorithm for two reasons. Firstly, as our data is in the order of ten thousands of events, the speed performance was not a critical criterion. Secondly, apriori-based algorithms are easier to understand and modify than pattern-growth based algorithms. More specifically, we used a slightly modified version of the Generalized Sequential Pattern Mining (GSP) algorithm. GSP is based on the so called apriori (or antimonotony) property which states that if a sequence is frequent then all its sub-sequences must be frequent as well. Based on this heuristic, GSP [25] adopts a multiple-pass, candidate generation-and-test approach in sequential pattern mining. We have modified it as we consider sequence of items, not itemsets.

Our ***GSP modified algorithm*** is described below:

- **First pass:** The first pass determines the support of each item, that is, the number of data sequences that include the item. At the end of the first pass, the algorithm knows which items are frequent, that is, have minimum support.
- **Candidate generation pass:** These frequent sequences are used to generate new potential patterns, called *candidate sequences*. Given the set of all the frequent k-1 sequences found in the previous pass, we generate new k-sequence candidates. Candidates are generated in two steps:
  - **Join step**: A sequence s1 joins with s2 if the sub-

sequence obtained by dropping the first item of s1 is the same as the subsequence obtained by dropping the last item of s2. For example <a,b,c,d> is a candidate 4-sequence of the 3- sequences <a,b,c> and <b, c, d>.
- **Pruning step:** We remove all the candidates that have a contiguous (k-1) subsequence whose support is less than the minimum support.

- **Test pass:** All the candidate sequences in a pass have the same *length* (i.e. number of items). The scan of the database in one pass finds the support for each candidate sequence. All the candidates whose support is above the minimum support, form the set of the newly found sequential patterns. This set then becomes the seed set for the next pass.

The algorithm terminates when no new sequential pattern is found in a pass, or no candidate sequence can be generated.

6.2 Data Pre-processing

Data pre-processing is a very important step needed before the sequential data mining can be performed. Firstly, the raw data needs to be represented in a more abstract form, e.g. as a long, unique, chronological sequence of events for a student group. Secondly, this long sequence needs to be split into a dataset of several meaningful sequences. Thirdly, the events need to be suitably encoded to facilitate data mining.

Abstraction of raw data

The raw data for each group is first transformed into a list of events, which are defined as:

Event = {eventType, Resource, Author, Time}, where

EventType is either *T* (for Ticket), *S* (for SVN) or *W* (for wiki), *Resource* is the identifier of the ticket number, source code file or wiki page, *Author* is the name of the user who performed the action and *Time* is the absolute time when the event occurred.

Generation of a Dataset of Sequences

The original sequence obtained for each group was from 1416 to 3395 events long. We then needed to break down this long sequence into several meaningful sequences to form a dataset of sequences of events. We considered the following three ways.

- *A sequence per resource*, where a separate sequence is obtained for the events on each ticket, wiki page, and SVN file. Therefore, the number of sequences in the dataset (for a group) will be equal to the number of resources used.
- *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 (we used 7 hours). A related sequence formation method is *a sequence per author session:* the event list for each group member is extracted and then sessions are formed as above.
- *A sequence per task*, where the task is defined by a ticket. The task sequence includes: 1) all ticket events on that ticket, 2) all SVN and wiki events referring to the ticket and occurring between the ticket open and close dates, and 3) all events on SVN and wiki pages referred to by the ticket and occurring between the ticket close and open dates. Therefore, the number of sequences in the dataset (for a group) will be equal to the number of tickets for the group.

Encoding the Events into Items

To compare events and find frequent sequences in the dataset, we need to suitably encode the events using a higher level of abstraction. For instance, when the group members collaborate on a wiki page the actual identification number of this page is not relevant for our analysis; what is important is the fact that the same page was modified several times by different authors. Thus, we need to remove certain author and resource identification information, as well as to combine similar consecutive events into single alphabet items. We propose three alphabets which are summarised in Table 9.

TABLE 9. THREE ALPHABETS USED

Alphabet type	**Captures:**	**A new item is generated when an event appears in the sequence:**
Alphabet 1 items in the form (iXj) Used in *per session* sequencing	the number of consecutive events *i* occurring on a particular TRAC medium type *X* and the number of individuals *j* involved in the events. Example: (2t1) – 2 ticket actions by the same author (5w3) – 5 wiki actions by 3 different authors	from a different TRAC medium
Alphabet 2: items in the form (AiX) Used in *per resource* sequencing	the number of consecutive events *i* performed by a single author with a role A, on a TRAC medium of type X. Note: The author's role A is: L (leader), T (tracker) or a, b, c etc. for other non-leader and non-tracker authors in order of their appearance in the sequence. Example: (L3t) – 3 ticket actions by the leader of the group (b2s) – 2 SVN actions by person *b* who is not a leader or tracker	by a different author
Alphabet 3: items in the form (iXA) Used in *per task* sequencing	As in Alphabet 2	from a different TRAC medium OR by a different author

Alphabet 1 was developed for use with the per session sequencing (group or author). Note that when it is used with the per author sequencing, j is always 1. Alphabet 2 was developed for use with the resource sequencing, 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. These choices of alphabet were also inspired by elements of the Big Five Theory: for example, we expected that leaders behaviour would be very important and patterns that identified the activities and interactions of the leader would be important. In XP the manager has the role of leader and in our context this role was allocated to one person after a start-up period. In addition, the role of tracker in XP was thought to correspond to the Big Five function of performance monitoring. For these reasons, author role (A) in these alphabets were identified as: L (leader), T (tracker), and all other (non-leader and non-tracker) authors identified as a, b, c, etc. in order of appearance in the sequence.

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.

6.3 Pattern extraction process

We ran the GSP modified algorithm on the data, using each of the 3 alphabets described in the previous section. We found large numbers of patterns of various length and support. The patterns found were manually analysed. For each set of results, we sorted the patterns firstly on support, then on length, and compared the results across groups. That way we identified which patterns were most frequent in certain groups and least frequent in others. We ran the GSP algorithm with a very low support threshold because when patterns are found to be frequent in one group, we need to compare this with the exact frequency in other groups where they are not frequent. Hence, the need to also compute patterns with a low frequency (support). Although the overall number of patterns found was very high (up to 40,000), only the ones that are frequent in at least one group and low in at least one other were retained, reducing easily that number down to a humanly manageable size of about 50.

A support of a sequential pattern is the number of dataset sequences in which it occurred. As the groups had different numbers of sequences (when group and author sequencing was used), we calculated percentage support to allow comparison across groups. For example, when sequencing per group was used, percentage support of a sequential pattern is the number of sequences for the group in which the pattern occurred over the total number of sequences for this group.

Following this conversion to percentage support, the average support for each pattern was also computed. The patterns were then sorted by support for each group, with a secondary sort by average support or support for a contrasting group performed in order to identify characteristic and contrasting patterns. Patterns for which a group had noticeably higher or lower than average support were noted, as were patterns which distinguished groups with different levels of success. Table 10 shows some of the results we obtained.

TABLE 10. PERCENTAGE SUPPORT FOR SOME OF THE PATTERNS FOUND ALTERNATING SVN AND WIKI EVENTS, AND SVN AND TICKET EVENTS

	group						
patterns	1	2	3	4	5	6	7
(1s1)(1w1)(1s1)	12.9	5.5	8.0	8.0	17.0	7.1	2.9
(1s1)(1w1)(1s1)(1w1)	6.5	3.7	0	3.2	8.8	4.4	0
(2s1)(1w1)(2s1)	5.2	0	0	2.4	0	0	0
(1s1)(1t1)	27.7	11.0	18.0	16.0	22.6	31.0	15.7
(2s1)(1t1)	9.0	2.8	5.0	8.8	6.9	3.5	4.3
(1s1)(1t1)(1s1)	10.3	7.3	7.0	8.8	12.0	16.8	5.0

These patterns are only a small subset of the ones found. We looked at how often a particular pattern occurred, but also at the frequency of related patterns, such as those involving similar sequences of the three media, and those differing slightly on the number of events, authors, or items involved. For instance whereas Group 1 shows the second highest occurrence of the pattern (1s1)(1t1)(1s1), on the last row of table 10, it had far more related patterns overall showing an alternation of wiki and ticket events.

We will now present some of the patterns found.

6.4 Patterns Observed in Group and Author Sessions

From data such as the one shown in Table 10, we found that the best groups had many alternations of SVN and wiki events, and also 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 difficult to trace events back to the actual TRAC actions to test such suspicions.

Through additional analysis of individual behavior (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). Armed with this information, the course co-ordinators examined some of the detailed actions on the TRAC site for this group to confirm the meaningfulness of results suggested by the data mining. This provided concrete examples of good practice and these were used in teaching the next class.

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 this group had frequent consecutive wiki events, 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 co-ordinators, who described this group as technically less proficient. This, too, served as a basis for the course co-ordinators to identify concrete examples of practices that are associated with poor group work.

A more detailed analysis of these patterns revealed that the top group used the ticketing system 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. Such patterns are useful for facilitators who can use them as a basis for targeted exploration of the TRAC site and as a basis for counselling individuals and groups.

6.5 Patterns Observed in Task Sequence

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 tools of TRAC in completing project tasks.

Note that by using Alphabet 2 we distinguish between all group members. Thus, a sequence (1tL)(1ta)(1tb) is different from (1tL)(2tb): the second sequence does not mean a ticket opened by the leader followed by two actions of 1 or 2 group members but a ticket opened by the leader followed by 2 ticketing actions by group member b. In this way we do not lose information about the sequence in which the group members interacted while collaborating.

We found that the two top groups had by far the greatest percentage support for the pattern (1tL)(1tb), which were most likely tickets initiated by the leader and accepted by another team member. The fact that this occurred more often than (1tL)(2tb), 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 groups to display the patterns (1tb)(1sb) and (1sb)(1tb) – 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 (1tb)(1tb)(1tb), (1tb)(1sb)(1tb) and (1tb)(1wb)(1tb). 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 (2ta)(1sa)(1ta), 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 (1tT), (1tT)(1tb) and (2tT). As it is the tracker's role to follow up tasks, their general involvement in tickets should not be a matter of concern. In the poor groups these patterns may be due to weaker leadership, resulting in trackers performing a share of the leader's role. Conversely, in the better groups, the tracker might have been less involved due to prominent leaders who were also able to perform tracker duties. An alternative explanation may be that group problems lead to greater tracker activity. These patterns are of particular value for the manger and tracker in monitoring their own behaviour. It is valuable for the facilitators in identifying these problematical behaviours early. The patterns also point to concrete examples that the course co-ordinators used in their teaching, linking the abstract theory of group work to concrete examples of good, as well as poor, practices.

6.6 Patterns Observed in Resource Sessions

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 (3w2) would be indicative of two 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 (L1t)(b1t)(L1t). 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 [1, 30, 31], and the success of our data mining in detecting differences in leadership is especially promising. This is important for leaders and facilitators.

In addition, the best group displayed the highest support for patterns such as (b3t) and (b4t), 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 (b2t), 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 (b1t)(c1t)(L1t). 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 the second best group (Group 2) 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 (a2s) and (a3s). 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 (a1s)(b1s), 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. In either case, this deserves follow up by facilitators.

6.7 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 web logs. A problem with our mining program itself is the lack of gap constraints – as noted in [25], a frequent subsequence of (X)(Y) may not be meaningful if many other events occur between X and Y. Another issue is the need for more automated methods of processing output which go beyond manual sorting techniques. Emergent pattern mining [32] and contrast sets [33] 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.

7 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. Our goal was to support learning group skills in the context of a standard state-of-the art tool. Clustering was applied to find both groups of similar teams and similar individual members, and sequential pattern mining was used to extract sequences of frequent events. The results revealed interesting patterns characterising the work of stronger and weaker students. Key results point to the value of analysis based on each resource and on individuals, rather than just the group level. We also found that some key measures can be mined from early data, in time for these to be used by facilitators as well as individuals in the groups. Some of the patterns 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 the patterns associated with some of the general principles, such as effective leadership and monitoring activity, and illustrating them with actual wikis and tickets. Secondly, we can automate the identification of the most salient patterns described above and present them as a form of formative feedback to students. Students can use these patterns in their own group discussions or in the discussions with facilitators. This may help to rectify ineffective patterns of group operation and consolidate effective ones. Thirdly, we can discover new efficient ways of using the collaboration system to achieve effective group work (as shown in Section 5.1) that can be explicitly taught to students.

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. The patterns are also a starting point of facilitator discussions with the groups. The importance of specific, regular and timely feedback for helping students' learning is widely recognised [34, 35]. Course facilitators can also greatly benefit from such feedback. Although more work is needed before such formative and timely feedback can be provided, students and facilitators could be made aware of the current findings and limitations, to encourage better group practice and teaching and learning experience. We are currently developing a TRAC plug-in which stores the patterns found interesting by the course coordinator and allows facilitators and students to search for them within the current data. Corresponding feedback including links for further exploration and remedial examples will also be displayed to the user.

This work also highlights some of the data mining challenges posed by educational data; more specifically, the data is temporal, noisy, correlated, incomplete and may lack enough samples for some tasks. We addressed some of these challenges by providing specific solutions for our task. The quality of the data is essential, e.g. one of the groups was reluctant to use the ticketing system which resulted in non-representative ticketing data; this can be alleviated by better linking of the course assessment to the use of the system.

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.

Acknowledgment

This work was supported in part by the Australian Research Council, Discovery grant DP0560014 and the University of Sydney bridging support grants U1143 and U1214.

References

[1] E. Salas, D. E. Sims, and C. S. Burke, "Is There a "Big Five" in Teamwork?," *Small Group Research*, vol. 36, pp. 555-599, 2005.

[2] http://www.educationaldatamining.org, Educational Data Mining, 2008.

[3] http://www.educationaldatamining.org/events.html, Educational Data Mining Events, 2008.

[4] A. Merceron and K. Yacef, "Clustering Students to Help Evaluate Learning," in *Technology Enhanced Learning*, vol. 171, J.-P. Courtiat, C. Davarakis, and T. Villemur, Eds.: Springer, 2005, pp. 31-42.

[5] C. Romero, S. Ventura, C. d. Castro, W. Hall, and N. H. Ng, "Using Genetic Algorithms for Data Mining in Web-based Educational Hypermedia Systems," *Proc. Adaptive Systems for Web-based Education*, Malaga, Spain, 2002.

[6] R. Mazza and V. Dimitrova, "CourseVis: Externalising Student Information to Facilitate Instructors in Distance Learning," *Proc. Int'l Conf. Artificial Intelligence in Education*, Sydney, 2003.

[7] W. Wang, J.-F. Weng, J.-M. Su, and S.-S. Tseng, "Learning Portfolio Analysis and Mining in SCORM Complaint Environment," *Proc. 34th ASEE/IEEE Frontiers in Education Conf.*, 2004.

[8] R. Cheng and J. Vassileva, "Design and Evaluation of an Adaptive Incentive Mechanism for Sustained Educational Online Communities," *User Modeling and User-Adapted Interaction*, vol. 16, pp. 321-348, 2006.

[9] L. Talavera and E. Gaudioso, "Mining Student Data to Characterize Similar Behavior Groups in Unstructured Collaboration Spaces," *Proc. European Conf. Artificial Intelligence*, 2004.

[10] A. Soller and A. Lesgold, "A Computational Approach to Analyzing Online Knowledge Sharing Interaction " *Proc. Int'l Conf. Artificial Intelligence in Education,* Sydney, Australia, 2003.

[11] A. Soller, "Computational Modeling and Analysis of Knowledge Sharing in Collaborative Distance Learning," *User Modeling and User-Adapted Interaction*, vol. 14, pp. 351-381, 2004.

[12] B. Barros and M. F. Verdejo, "Analysing Student Interaction Processes in Order to Improve Collaboration. The DEGREE Approach," *Int'l J. Artificial Intelligence in Education*, vol. 11, pp. 221-241, 2000.

[13] P. Jermann, A. Soller, and M. Muehlenbrock, "From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning," *Proc. First European Conf. Computer-Supported Collaborative Learning*, Maastricht, The Netherlands, 2001.

[14] J. Kay, P. Reimann, and K. Yacef, "Mirroring of group activity to support learning as participation," *Proc. Int'l Conf. Artificial Intelligence in Education*, Los Angeles, USA, 2007.

[15] T. Erickson, C. Halverson, W. A. Kellogg, M. Laff, and T. Wolf, " Social Translucence: Designing Social Infrastructures That Make Collective Activity Visible," *Communications of the ACM*, vol. 45, pp. 40-44, 2002

[16] T. Erickson and W. A. Kellogg, "Social Translucence: An Approach to Designing Systems that Support Social Processes," *ACM Transactions on Computer-Human Interaction*, vol. 7, pp. 59-83, 2000.

[17] www.extremeprogramming.org, XP - Extreme Programming, 2007.

[18] http://trac.edgewall.org/, TRAC, 2007.

[19] J. Kay, P. Reimann, and K. Yacef, "Visualisations for Team Learning: Small Teams Working on Long-Term Projects," *Proc. Int'l Conf. Computer-Supported Collaborative Learning*, New Brunswick, USA, 2007.

[20] P.-N. Tan, M. Steinback, and V. Kumar, *Introduction to Data Mining:* Pearson Addison Wesley, 2006.

[21] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*: Morgan Kaufmann, 2005.

[22] WEKA, www.cs.waikato.ac.nz/ml/weka, 2007.

[23] http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/cluster/software.htm, Cluster software, 2006.

[24] http://jtreeview.sourceforge.net/, TreeView software, 2006.

[25] R. Srikant and R. Agrawal, "Mining sequential patterns: Generalizations and performance improvements," *Proc. Fifth Int'l Conf. Extending Database Technology (EDBT),* Avignon, France, 1996.

[26] J. Kay, N. Maisonneuve, K. Yacef, and O. Zaiane, "Mining patterns of events in students' teamwork data.," *Proc. Workshop on Educational Data Mining at the 8th Int'l Conf. Intelligent Tutoring Systems,* Jhongli, Taiwan., C. Heiner, R. Baker, and K. Yacef, Eds.: Springer, 2006, pp. 45-52.

[27] D. Cummins, K. Yacef, and I. Koprinska, "A Sequence Based Recommender System for Learning Resources," *Australian Journal of Intelligent Information Processing Systems,* vol. 9, pp. 49-56, 2006.

[28] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Datal, and M.-C. Hsu, "Freespan: Frequent Pattern-Projected Sequential Pattern

Mining," *Int'l Conf. Knowledge Discovery and Data Mining (KDD),* Boston, USA, 2000.

[29] J. Pei, B. Mortazavi-Asl, H. Punto, Q. Chen, U. Dayal, and M.-C. Hsu, "PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth," *Int'l Conf. Data Engineering (ICDE),* Heidelberg, Germany, 2001.

[30] F. E. Fiedler, *A Theory of Leadership Effectiveness.* New York: McGraw-Hill, 1967.

[31] E. A. Fleishman, "The description of supervisory behavior," *Journal of Applied Psychology*, vol. 37, pp. 1-6, 1953.

[32] G. Dong and J. LI, "Efficient Mining of Emerging Patterns: Discovering Trends and Differences," *Knowledge Discovery and Data Mining*, pp. 43-53, 1999.

[33] S. Bay and M. Pazzani, "Detecting Group Differences: Mining Contrast Sets," *Data Mining and Knowledge Discovery*, vol. 5, pp. 213-246, 2001.

[34] C. Rust, "Impact of Assessment on Student Learning," *Active Learning in Higher Education,* vol. 3, pp. 145-158, 2002.

[35] P. Ramsden, *Learning to Teach in Higher Education*: Routledge-Falmer, 2003.

Dilhan Perera is an undergraduate student enrolled in the combined science and commerce degree at the University of Sydney, Australia. He takes part in the Talented Student Program offered by the Faculty of Science. Dilhan has spent time working with the Computer Human Adapted Interaction (CHAI) research group as part of two vacation scholarships with the School of Information Technologies. In addition to information technologies, he also studies psychology and economics.

Judy Kay is a principal of the Computer Human Adaptive Interaction (CHAI) lab, at the University of Sydney, Australia. Her research aims to exploit the huge amounts of data available about people, from conventional and emerging systems, to create useful mirroring tools and user models to support lifelong learning and personalization of future pervasive computing environments. She has over 200 publications in the areas of personalization and teaching and learning. She has been a keynote speaker at major conferences: UM'94 User Modeling, Boston; IJCAI'95 International Joint Conference on Artificial Intelligence, Montreal; ICCE'97, International Conference on Computers in Education, Kuching; ITS'2000, Intelligent Tutoring Systems, Montreal; AH2006 Adaptive Hypermedia and Adaptive Web-Based Systems, Dublin; ITS'2008, Montreal. She is on the editorial board of UMUAI, User Modeling and User Adapted Interaction, Associate Editor of International Journal of Artificial Intelligence in Education and IEEE Transactions on Learning Technologies.

Irena Koprinska is a Senior Lecturer at the School of Information Technologies, University of Sydney, Australia, and a member of the CHAI research lab. She received a MEng from the Technical University of Sofia, Bulgaria, a PhD from the Institute for Information Technologies in Sofia, both in Computer Science, and also MEd (Higher Education) from the University of Sydney. Irena was a visiting researcher at the Dept. of Medical Informatics, Graz University of Technology, Austria, and a post-doctoral fellow at the Dept. of Electrical Engineering and Computer Science, University of Trieste, Italy and also at the Dept. of Information Sciences, University of Otago, New Zealand. Her research interests are in machine learning, data mining and neural networks, both applications and novel algorithms. She has published more than 50 refereed papers in these areas and regularly serves as a reviewer for funding bodies, conferences and journals, among them IEEE Trans. Neural Networks, IEEE Trans. Knowledge and Data Eng., IEEE Trans. Circuits and Systems for Video Technology.

Kalina Yacef is a Senior Lecturer in the Computer Human Adaptive Interaction (CHAI) research lab, at the University of Sydney, Australia. She received her PhD in Computer Science from University of Paris V in 1999. Her research interests spans across the fields of Artificial Intelligence and Data Mining, Personalisation and Computer Human Adapted Interaction with a strong focus on Education applications. Her work focuses on mining users' data for building smart, personalised solutions as well as on the creation of novel and adaptive interfaces for supporting users' tasks. She regularly serves on the program committees of international conferences in the fields of Artificial Intelligence in Education and Educational Data Mining and she is the editor of the new Journal on Educational Data Mining.

Osmar R. Zaïane received an MSc in electronics from the University of Paris XI, France, in 1989 and an MSc in Computing Science from Laval University, Canada, in 1992. He received his PhD in Computing Science from Simon Fraser University in 1999 specializing in data mining. He is an Associate Professor at the University of Alberta with research interest in novel data mining techniques and currently focuses on e-learning as well as Health Informatics applications. He regularly serves on the program committees of international conferences in the field of knowledge discovery and data mining and was the program co-chair for the IEEE international conference on data mining ICDM'2007. He is the editor-in-chief of ACM SIGKDD Explorations and Associate Editor of Knowledge and Information Systems.