Design of an Extraction, Transform and Load Process for Calculation of Teamwork Indicators in Moodle

Ángel Hernández-García[1[0000-0002-6549-9549]], Emiliano Acquila-Natale[1[0000-0003-2164-8386]], Santiago Iglesias-Pradas[1[0000-0003-1133-2687]] and Julián Chaparro-Peláez[1[0000-0002-0920-4005]]

[1] Universidad Politécnica de Madrid, Av. Complutense 30, 28040 Madrid, Spain
[angel.hernandez, emiliano.acquila, s.iglesias, julian.chaparro]@upm.es

Abstract. Assessment and measurement of teamwork still remains as one of the main challenges in computer-assisted and digital education. A large majority of virtual learning environments and learning management systems are mostly either student-centred or content-centred. Therefore, most of the database records are stored at an individual level. While this approach for log-based learning analytics of student data at individual level is highly valuable and adequate, it also makes it difficult for students, instructors and researchers to collect, analyze and visualize group data in team-based education methods. Measurement and characterization of the different components of teamwork are essential in order to get insight about whether the activities are being performed by teams effectively. This study refines previous proposals for measurement of teamwork indicators in online education –communication, coordination, cooperation and tracking, at both individual and team levels–, and proposes the design and implementation of extraction, transform and load processes to collect those indicators from Moodle, illustrating the execution of these processes with an example.

Keywords: Learning Analytics, Teamwork, Moodle, Indicators, ETL, Communication, Cooperation, Coordination, Monitoring, Tracking, Online education.

1 Introduction

Information is one of the most important assets in every aspect of the society. Da-ta and information facilitate value creation, understanding the environment and improving decision making processes. Technology-intensive companies first realized about the value of Big Data, leading to the emergence of Business Analytics. The translation of the concepts and technologies applied in Business Analytics to the educational context and learning processes has led to the emergence of Learning Analytics, defined as "the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs" [1].

The application of information technologies to education (in the form of Learning Management Systems and Virtual Learning Environments or Massive Online Open

Courses) has expanded and moved many educational processes to online spaces. In a digital space, the activity of learning agents leaves a trail, stored and logged as records in a database system. So far, data about student activity is stored as low-level information; that is, every single click, or interaction [2], with the platform is associated with a single, unique record in the log database, making it suitable for analysis of individual learning behaviors. Therefore, a vast amount of research in the field of learning analytics involves the study of learning agents –mostly students– at an individual level. While this approach has proven successful, it poses additional challenges when the object of study is not an individual student, but a group of students, a situation that is becoming increasingly common with the application of collaborative learning and team-based methods, based on constructivist approaches, such as project-based learning or problem-based learning.

Teamwork is at the core of collaborative and team-based learning. Teamwork refers to "a behavioral pattern between two or more individuals who interact dynamically, establish a regular and constant negotiation to reach agreements, through knowledge exchanges and problem solving, while keeping a steady pace and coordinating efforts in order to achieve their shared goals" [3]. This definition has two important implications: first, the multidimensional nature of teamwork, as a concept that encompasses different behaviors; and second, the need to assess teamwork at both individual and team or group levels, because it refers to the behavior of different individuals dynamically interacting [3].

Teamwork assessment is gaining relevance in current practice because of its association with competence-based learning. However, teamwork assessment has proven a time-consuming activity for teachers [4][5], especially in online contexts were observation of the group dynamics may pass unnoticed to instructors. The use of information technologies to support collaborative learning processes may also facilitate observation of teamwork behaviors, as all the information pertaining to group dynamics and team member activity is also stored in the log database of the virtual learning environment [6]. However, the volume of that information might be too big for instructors to handle, thus requiring data extraction and preparation in order for the information delivered to be meaningful.

This study proposes the design of such a system for data extraction, transformation and loading (ETL) of educational data from a learning management system (Moodle). The design is based on a critical revision of the proposal of individual-level and team-level indicators of teamwork across four dimensions –communication, cooperation, coordination, and monitoring/tracking– in [3]. This design aims to effectively retrieve all relevant teamwork-related activity course data logged in the Moodle database, and make the necessary operations to transform those data into useful information about teamwork behaviors.

The following sections will describe the design process. Section 2 will cover two different blocks: first, and in order to offer a systematic approach to the design, it is required to provide an overview of the Moodle database and Moodle log system; second, a critical revision of the set of teamwork indicators proposed by [3], coupled with the analysis of the structure of data stored in Moodle logs, will provide the expected target output of the ETL process. As a result of Section 2, Section 3 establishes the link

between the final indicators and the database information, so as to provide a definition and operationalization of the high-level teamwork indicators, and will identify the data sources necessary for the extraction process. Finally, Section 4 briefly introduces the use of RapidMiner as an ETL tool to perform such process and shows one example of implementation of the ETL process.

2 Educational data and teamwork indicators in Moodle

2.1 Moodle's database, logging system

Moodle is an open-source (under General Public License, GPL) learning management system created in 1999. Moodle is currently the most used technology-based educational platform, with systems implementations across many Higher Education institutions around the world. The most recent official version of Moodle is 3.5 (as of May 29th).

Moodle is comprised of the platform *core* and different modules and plug-ins that may extend its functionalities, allowing for customization of the learning experience. The core provides the necessary infrastructure of the learning management system, including all aspects relative to course enrolment, courses and activities, users, groups, roles and permissions, and logs and statistics.

Moodle uses a relational database consisting of more than 250 tables, corresponding to the different modules, with their corresponding fields, records and relation between tables.

From version 2.6 onwards, Moodle uses a new logging system to keep track of the different actions performed by the users and storing the interactions as records in the database. Understanding how the logging system works is essential to identify the actions considered relevant or of interest within the learning process under study and elaborate an effective ETL design. The new logging system is an improvement over the previous system in terms of information collected, performance and scalability, and was designed with from a learning analytics-friendly approach. Nonetheless, three different log stores currently coexist in Moodle:

- Standard Log: the new logging system.
- Legacy Log: the previous version of the logging system.
- External Log: allows connection to an external log database.

For the sake of simplicity, and given that older versions of Moodle (2.5 or below) are currently not in use, this design focuses on the Standard Log. Log generation in the Standard Log uses two different Application Programming Interfaces (APIs): the Event2, the new Events API, and Logging2, the new Logging API. Event 2 facilitates capture, dissemination and notification of event information occurring in the system (an event is considered an atomic piece of information describing something that happened in Moodle), while Logging 2 allows configuration, registering and reporting of the data associated with each event. In order to help log processing and reading, two additional APIs are used (Writing API and Reading API) to define interfaces for log

reading and writing, that are implemented in the Log Manager and Log Storage. In sum, when a user performs an action in Moodle, an event is generated, which is listened by the Log Manager, then, depending on the configuration, the Log Manager decides whether the event or action should be registered and stored in the log database. It storage is necessary, the event information is passed to the plug-ins, which will write the record in the log database using the Writing API.

2.2 A critical revision of teamwork indicators

This study uses the proposal of teamwork indicators in [3]. However, in order to adapt the adequacy of the indicators presented in that study to the design of the ETL system, a critical revision of the indicators is deemed necessary, as the design of the final processes is determined by the structure of the data across the different tables of Moodle's database. An additional consideration and design requirement is the adaptation of the system to courses using the Comprehensive Training Model of the Teamwork Competence (CTMTC) method [7]. Because different collaborative and team-based learning method may result in very heterogeneous course implementations in Moodle, this approach makes it possible to provide a standard configuration of the system.

Putting CTMTC in practice in a virtual workspace requires students to work collaboratively on a project during the course, in teams of between three and four members. The teams must follow a common series of guidelines and have to go through the different stages defined in the method, using message boards for communication and a common wiki where they leave evidence of the work during the whole process, and provide their solution to the project.

Data structure. Considering the different dimensions and indicators provided by [3], when implementing CTMTC in a Moodle course, communication interactions occur in Moodle team message boards, cooperative interactions take place in Moodle team wikis, and coordination and monitoring indicators require information about activity in both message boards and team wikis.

The Moodle course used as example to test the system follows the CTMTC, and the database has a total of 388 tables (including the use of additional plug-ins, such as GraphFES [8]) for administration and operation of the LMS. The specific modules of interest for the design of the ETL and associated tables are as follows:

1. Module *groups*: this module presents information about the different teams, and their corresponding team members, in the course. The composition of teams is the same for the whole duration of the course. Information about teams and team membership is stored in the following table:

   - *mdl_groups_members*: contains information about team membership of every student.

2. Module *forum*: this module shows information about the different message boards used in the course, where team members may discuss and exchange ideas. Any team member may open a new discussion in the team message board to discuss a specific topic, and the rest of the team member may post their replies. A single message

board, where each team may only access their corresponding discussions, contains all the course discussions (there are additional message boards for general announcements and/or questions, but team communication only takes place in the single message board for teams). The Moodle database tables that collect information about message boards, discussions and posts are:

- *mdl_forum_discussions*: this table contains information about the different discussions or topic created in every message board.
- *mdl_forum_posts*: this table stores information about posts sent by any user in every discussion of every message board.

3. Module *wiki*: the wiki module manages the information about the shared workspace. In a CTMTC course, all teams have access to a wiki that has a single entry point (i.e. accessed using the same link). However, any content addition (e.g. a new wiki page), deletion or edition is only visible to team members and instructors. In this sense, it is equivalent to have different instances of the wiki created for each team. Information about wiki activity is stored in the following table:

   - *mdl_wiki_versions*, containing all the information about any edition made on any version of the wiki pages.

4. Module *admin*. The main administrative functionality of interest for this study is the one corresponding to logging activities. Logging information resides in the mdl_logstore_standard_log table.

   - *mdl_logstore_standard_log*. This table collects all activity logging and event-related information of the course. Because message board activity and wiki activity are events logged by the LMS, records in this table also include information about the different events included in modules *forum* and *wiki*. The information in this table serves as baseline for the necessary operations involved in indicator calculation, using information from the rest of tables as a support for calculations (e.g. ordering messages posted by a team by discussion, with the help of the tables from the module *forum*). The main fields of interest in this table are *courseid* (course unique identifier), *component* (module or component to which an event refers to; e.g. forum, wiki), *action* (type of action registered; e.g. created, updated, viewed), *target* (submodule involved; e.g. post, discussion, page) and *contextinstanceid* (reference to each unique course module; e.g. team message board, team wiki). All records in the mdl_logstore_standard_log table include information about the specific moment of occurrence (timestamp), which makes it possible to calculate values of time-related indicators.

5. Auxiliary tables. Because some indicators require a specific time as a reference (e.g. earliness, delay), two additional Moodle LMS tables are necessary to perform the ETL process:

   - *mdl_assign*: table including information about the deadline for final team project submission.

- *mdl_course_modules*: it contains information about the different course modules, including starting date of the team project and a reference to any deliverable.

Revision of the set of indicators. The inspection of the information available in the Moodle database, and especially in the mdl_logstore_standard_log, makes it possible to refine the set of indicators proposed by [3]. This revision has four different approaches: indicator retention, indicator deletion, indicator modification and indicator addition. Indicator retention involves keeping the definition of the indicator from the original proposal, which is self-explanatory.

Indicator deletion was performed using two different criteria: operationalizability and feasibility, and significance. According to the first criteria, the lack of a clear definition of synchronicity and pace, and the computation power required to analyze 24 hour intervals after the occurrence of every single event (a single course may have more than 100,000 events), advised against including coordination indicators referring to those concepts. Indicators which mixed active and passive interactions [2], such as individual reading in monitoring/tracking indicators were not considered useful and actionable, as their meaning and role in [3] are not clear. Finally, indicators that referred to other resources that could not be standardized (e.g. course syllabus, guidelines, project instructions, etc.) were also excluded.

Indicator modification mainly involves standardization of indicators and redefinition of unclear concepts in the original. The former includes indicators such as individual message exchanges, individual contributions, etc., where (1/number of team members is subtracted) from the original value to allow for correction of outliers (with the correction, values equal to zero would reflect a level of interaction equal to the expected effort relative to group effort, positive values would mean interactions higher than the rest of the group; conversely, negative values are indicative of lower levels when compared to the rest of the group). Redefinition is applied mainly to regularity-related indicators, involving calculation of standard deviations of temporal distances to account for regularity and a division by the square root of the specific event-related interaction to account for the total number of interactions.

Finally, the indicators added to the original set were derived from the original proposal, after consideration of their usefulness as elements for data visualization or to provide further context for instructors. Regarding the latter, the total number of occurrences of a given event (e.g. total number of posts written in a forum, total number of discussion views, etc.) may help putting in perspective the rest of indicators within each dimension. Regarding the former, the concept of evenness was introduced in this proposal. Evenness refers to the existence of an even or uneven distribution of an indicator of interest among team members, and facilitates identification of unbalances in the distribution of effort among team members in communication, cooperation or monitoring-related tasks. The operationalization of evenness involves the standard deviation of standardized individual indicators, already explained above.

3 Teamwork indicators: operationalization and structure

From the discussion in Section 2, Fig. 1 presents a summary of indicators retained or modified (in bold) and indicators removed (with a minus sign) from the original proposal, as well as the indicators added (with a plus sign) in this study. Fig. 2 shows the final indicators used for the design, and Table 1 shows their operationalization.

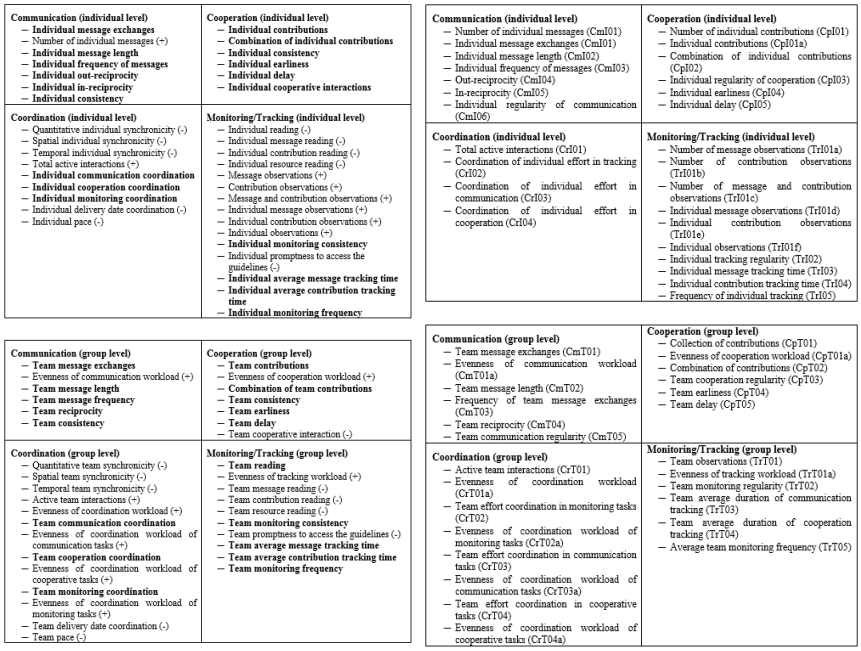

Fig. 1. a. (top-left), **Fig. 2b.** (bottom-left), **Fig. 2a** (top-right) and **Fig. 2b** (bottom-right) Indicators retained or modified (in bold) and removed (minus sign) from [XX], and indicators added (plus sign).

Table 1. Operationalization of indicators

Indicator	Operationalization
CmI01	Number of *post-created+discussion-created* of a student
CmI01a	Number of *post-created+discussion-created* of a student divided by the total number of *post-created+discussion-created* by the team, minus 1/(number of team members)
CmI02	Length (sum of characters) of the total *post-created+discussion-created* of a student, divided by the number of *post-created+discussion-created* of the student
CmI03	Number of *post-created+discussion-created* of a student divided by the temporal distance between the first and last message posted by his/her team
CmI04	Sum of the temporal distance of each *post-created* of a student to the *post-created* or *discussion-created* of another team member he/she is replying to, divided by the number of *post-created* of the student
CmI05	Sum of the temporal distance of each *post-created* or *discussion-created* of a student to a later *post-created* as reply by another team member, divided by the number of *post-created+discussion created* of the student

CmI06	Standard deviation of the temporal distance between each *post-created+discussion-created* of a student, divided by the square root of the number of *post-created+discussion-created* of the student
CmT01	Sum of CmI01 of all team members, divided by the number of team members
CmT01a	Standard deviation of CmI01a of all team members
CmT02	Length (sum of characters) of the *post-created+discussion-created* of all team members, divided by the number of team members
CmT03	Sum of CmI03 of all team members divided by the number of team members
CmT04	Sum CmI04+CmI05 of all team members, divided by the number of team members
CmT05	Sum of CmI06 of all team members, divided by the number of team members
CpI01	Number of *page-created+page-updated* of a student
CpI01a	Number of *page-created+page-updated* of a student divided by the total number of *page-created+page-updated* by the team, minus 1/(number of team members)
CpI02	Length (sum of characters) of the total *page-created+page_updated* of a student, divided by the number of *page-created+page_updated* of the student
CpI03	Standard deviation of the temporal distance between each *page-created+page_updated* of a student, divided by the square root of the number of *page-created+page_updated* of the student
CpI04	Temporal distance between the first *page-created* or *page-updated* of a student and the starting date of the activity, divided by the difference between starting and end date of the task/project
CpI05	Temporal distance between the last *page-created* or *page-updated* of a student and the end date of the activity, divided by the difference between starting and end date of the task/project
CpT01	Sum of CpI01 of all team members, divided by the number of team members
CpT01a	Standard deviation of CpI01a of all team members
CpT02	Length (sum of characters) of the *page-created+page-updated* of all team members, divided by the number of team members
CpT03	Sum of CpI03 of all team members divided by the number of team members
CpT04	Sum of CpI04 of all team members divided by the number of team members
CpT05	Sum of CpI05 of all team members divided by the number of team members
CrI01	Number of *post-created+discussion-created+page-created+page-updated* of a student divided by the total number of *post-created+discussion-created+page-created+page-updated* by the team, minus 1/(number of team members)
CrI02	Sum of temporal distance between every *discussion-viewed* and the following event for a student, and of *page-viewed* and the following event for a student, divided by the sum of the "sums of the same distances" of all team members, minus (1/number of team members)
CrI03	Sum of temporal distance between every *assessable-uploaded* event and the previous event for a student, divided by the sum of "sums of the same distances" of all team members, minus (1/number of team members)
CrI04	Sum of temporal distance between every *page-created* event and the previous event for a student, and of every *page-updated* event and the previous event for a student, divided by the sum of "sums of the same distances" of all team members, minus (1/number of team members)
CrT01	Sum of *post-created+discussion-created+page-created+page-updated* of all team members, divided by the number of team members
CrT01a	Standard deviation of CrI01 of all team members

CrT02	Sum of temporal distance between every *discussion-viewed* and the following event, and every *page-viewed* and the following event, of all team members, divided by the number of team members
CrT02a	Standard deviation of CrI02 of all team members
CrT03	Sum of temporal distance between every *assessable-uploaded* event and the previous event of all team members, divided by the number of team members
CrT03a	Standard deviation of CrI03 of all team members
CrT04	Sum of temporal distance between every *page-created* and the previous event, and every *page-updated* and the previous event, of all team members, divided by the number of team members
CrT04a	Standard deviation of CrI04 of all team members
TrInd01a	Number of *discussion-viewed* of a student
TrInd01b	Number of *page-viewed* of a student
TrInd01c	Number of *discussion-viewed+page-viewed* of a student
TrInd01d	Sum of *discussion-viewed* of a student divided by the sum of *discussion-viewed* of all team members, minus (1/number of team members)
TrInd01e	Sum of *page-viewed* of a student divided by the sum of *page-viewed* all team members, minus (1/number of team members)
TrInd01f	Sum of *discussion-viewed+page-viewed* of a student divided by the sum of *discussion-viewed+page-viewed* of all team members, minus (1/number of team members)
TrInd02	Standard deviation of the temporal distance between each *discussion-viewed+ page-viewed* of a student, divided by the square root of the number of *discussion-viewed+page-viewed* of the student
TrInd03	Sum of the temporal distance between every *discussion-viewed* and the following registered event of a student, divided by the number of *post-created+discussion-created* of all team members
TrInd04	Sum of the temporal distance between every *page-viewed* and the following registered event of a student, divided by the number of *page-created+page-updated* of all team members
TrInd05	Number of *discussion-viewed+page-viewed* of a student divided by the temporal distance between starting and end date of the task/project.
TrEqu01	Sum of TrI01c of all team members divided by the number of team members
TrEqu01a	Standard deviation of TrI01f of all team members
TrEqu02	Sum of TrI02 of all team members divided by the number of team members
TrEqu03	Sum of TrI03 of all team members divided by the number of team members
TrEqu04	Sum of TrI04 of all team members divided by the number of team members
TrEqu05	Sum of TrI05 of all team members divided by the number of team members

Following Fig. 1a, Fig. 1b, Fig. 2a and Fig. 2b, and Table 1, Fig. 3 summarizes the final indicator design layout:

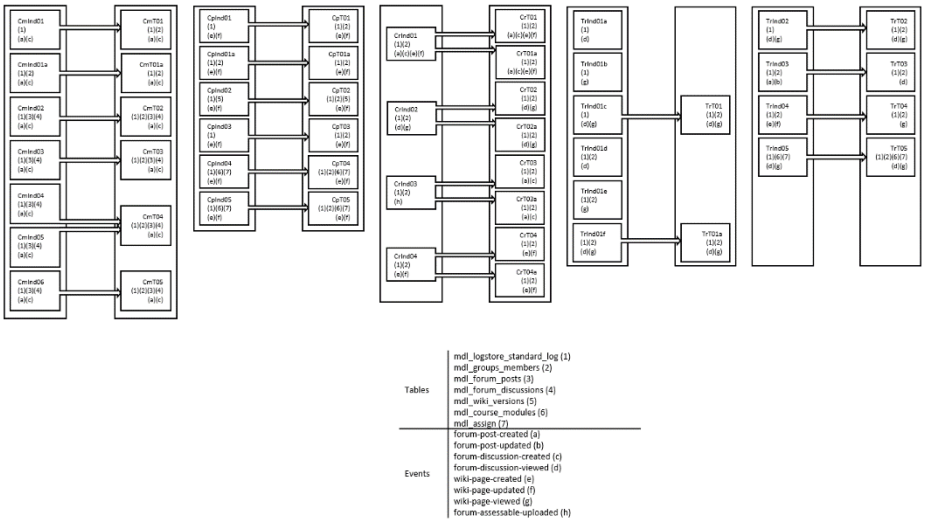

Fig. 3. Summary of the final indicator design structure and layout. The figure describes all the indicators, with their respective database tables and related events, and the correspondence between individual-level and group-level indicators.

4 ETL Tool (RapidMiner) and example of implementation

RapidMiner is an open-source (under the Affero General Public License) data mining software application. RapidMiner features an intuitive graphic user interface that allows easy design and implementation of ETL processes. These processes are defined using block sequences, known as operators, which represent different operations. Every operator features one or more inputs, one or more outputs, and operator parameters.

Process creation involves dragging the required operators to the workspace (the workspace represents a process), and sequentially connect them using their inputs and outputs. The last operator's output must necessarily be connected to the *res* (result) endpoint. Once the process has been set up, including connection of the different operators and operator parameter configuration, it is ready for execution. Processed operators show a green tick, making it possible to know the current state of execution of the process. It is also possible to define breakpoints and inspect variable values at any point. Once the process execution is completed, the result is shown in the *Results* screen. This screen presents the raw data of the executed process, statistics about the different fields, basic and advanced charts, and annotations.

4.1 Example of implementation

As an example of implementation of indicator extraction through an ETL process using RapidMiner Studio, this subsection presents and details the whole design and process for one indicator with some degree of complexity, Team Reciprocity (CmT04). The

data used in this example are real data from a mandatory course of the Bachelor in Computer Science degree that follows CTMTC. A total of 115 students were enrolled in the course, of which 53 (46.1 percent) from 23 different groups got a final grade. The calculation of the indicator involves the use of 11 subprocesses (subprocesses refer to processes nested within other processes) across three different levels. The sequence of execution follows the order shown in Fig 4. Description of the different elements and sub-levels will be explained next.

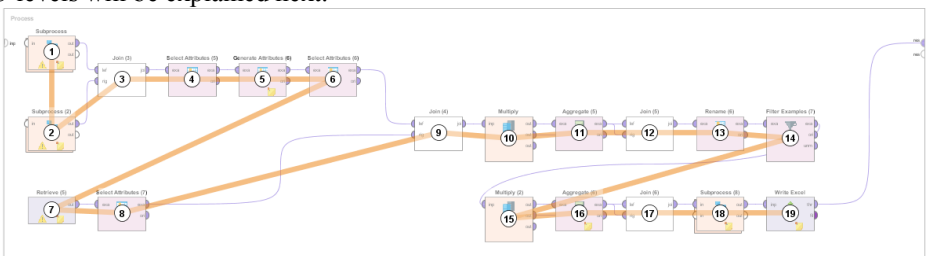

Fig. 4. ETL process for calculation of CmT04

- ① Subprocesses 1 (Fig. 5) and ② 2 calculate out-reciprocity (CmI04) and in-reciprocity (CmI05), respectively. It involves the use of 4 subprocesses that: retrieve the data relative to the selected course (with courseid) and forum (component *mod_forum* and action *created* with contextinstanceid, and target *post*) from mdl_logstore_standard_log ❶; retrieve information about the message board of interest, merging information from mdl_forum_discussions (field *forum*) and mdl_forum_posts ❹; calculates the sums of distances ❺ and returns a structured dataset ❼.

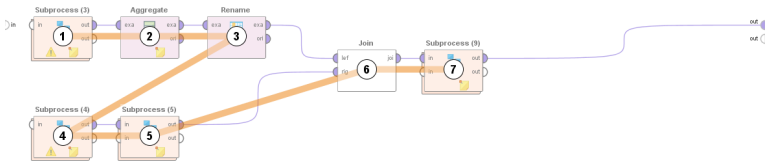

Fig. 5. Subprocess for calculation of out-reciprocity (CmI04)

- Next, ③ creates a joint dataset, ④ selects the values of CmI04, CmI05 and userid from the dataset, ⑤ creates a new attribute that adds CmI04 and CmI05, ⑥ refines the dataset, dropping CmI04 and CmI05 and only selecting userid and the total value of reciprocity for each student.
- ⑦ retrieves the data from mdl_groups_members, in order to establish the relation between students and teams, while ⑧ only selects groupid and userid from the resulting dataset.
- The Join operator used in ⑨ merges the dataset with reciprocity values and the dataset containing information about team membership; the resulting dataset gets duplicated in ⑩ for later operations. In ⑪ an aggregation operation to count the number of records with the same userid is performed over one of the duplicates to calculate the number of team members, and this new dataset is merged with the other duplicate

in ⑫. ⑬ handles a variable renaming –from *count(userid)* to *number of team members*–, and ⑭ discards records with missing values of either userid or groupid. Analogously to ⑩-⑫, in ⑮-⑰, two copies of the dataset are merged: the original and a processed copy that includes calculation of the total sum of reciprocity by team –by adding in- and out- reciprocity of all team members. Finally, ⑱ performs some data presentation operations, including indicator normalization, and ⑲ writes the result to an Excel file for later processing. Fig. 6 summarizes some of the results shown in RapidMiner.

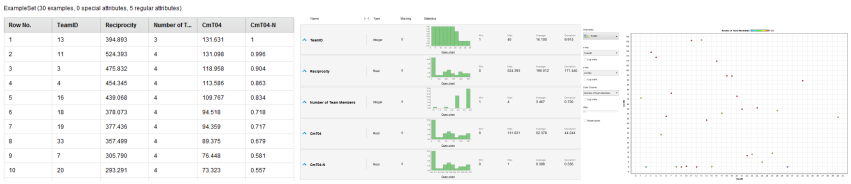

Fig. 6. Results of calculation of CmT04 in RapidMiner after ETL

References

1. Long, P.D., & Siemens, G.: Penetrating the Fog: Analytics in Learning and Education. EDUCAUSE Review 46(5), 31–40 (2011).Author, F.: Article title. Journal 2(5), 99–110 (2016).
2. Agudo-Peregrina, Á.F., Iglesias-Pradas, S., Conde-González, M.Á., Hernández-García, Á.: Can we predict success from log data in VLEs? Classification of interactions for learning analytics and their relation with performance in VLE-supported F2F and online learning. Computers in Human Behavior 31, 542–550 (2014)
3. Ruiz-de-Azcárate, C., Hernández-García, Á., Iglesias-Pradas, S., Acquila-Natale, E.: Proposal of a system of indicators to assess teamwork using log-based learning analytics. In: Manuel Caeiro-Rodríguez, Ángel Hernández-García, Pedro J. Muñoz-Merino and Salvador Ros (eds.). Proceedings of the Learning Analytics Summer Institute Spain 2017: Advances in Learning Analytics. CEUR Workshop Proceedings 1925, 78-92 (2017)
4. Buckingham-Shum, S., Ferguson, R.: Social Learning Analytics. Journal of Educational Technology & Society 15(3), 3–26 (2012).
5. Fidalgo-Blanco, Á., Sein-Echaluce, M.L., García-Peñalvo, F.J., Conde, M.Á.: Using Learning Analytics to improve teamwork assessment. Computers in Human Behavior 47, 149–156 (2015).
6. Davies, A., Fidler, D., Gorbis, M.: Future work skills 2020. Institute for the Future for University of Phoenix Research Institute, 540 (2011).
7. Lerís, D., Fidalgo, Á., Sein-Echaluce, M.L.: A comprehensive training model of the teamwork competence. International Journal of Learning and Intellectual Capital 11(1), 1–19 (2014).
8. Hernández-García, Á., & Suárez-Navas, I. (2017). GraphFES: A web service and application for Moodle message board social graph extraction. In Big Data and Learning Analytics in Higher Education (pp. 167-194). Springer, Cham.